

XXXI CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO

19 a 22 de Julho de 2011
Centro de Convenções de Natal
Natal – RN

COMPUTAÇÃO PARA TODOS: NO CAMINHO DA EVOLUÇÃO SOCIAL

ANAIS



http://www.dimap.ufrn.br/csbc2011

PREFÁCIO

De 19 a 22 de julho de 2011, a cidade do Natal, RN, é palco da trigésima primeira edição do Congresso da Sociedade Brasileira de Computação (CSBC 2011) . O Departamento de Informática e Matemática Aplicada (DIMAp) da Universidade Federal do Rio Grande do Norte (UFRN) é o responsável pela realização do CSBC2011..

Neste ano, o tema do Congresso é ***"Computação para todos: No caminho da evolução social"***. A escolha do tema foi motivada pelo fato de que os impactos da Computação estão levando a uma contínua quebra de paradigmas sociais, econômicos, científicos, educacionais e culturais. A mobilidade, a disseminação e a ubiquidade da informação, assim como os mecanismos de cooperação proporcionados pela Computação estão revolucionando a vida em sociedade de formas inesperadas e surpreendentes. O tema do congresso abre espaço às discussões sobre como as tecnologias da informação e informática podem contribuir ao exercício pleno da cidadania por parte de todos os habitantes da nação.

O tema do evento tem o objetivo principal de guiar as discussões, propiciando a reflexão dos participantes em torno dele; mas o CSBC não limita as discussões apenas a um tema, sendo um evento de caráter geral no qual a comunidade reúne-se para discutir e traçar os planos para o futuro da área no Brasil. Este é um dos motivos pelos quais o Congresso da SBC tem se firmado como o mais importante local de reunião da comunidade brasileira de Computação, sendo palco de discussões dos mais diversos temas relacionados com a área.

Em 2011, o Congresso da SBC conta com 17 sub-eventos, das mais variadas naturezas. O evento conta com atividades de formação, concursos acadêmicos, eventos de natureza científica, política e de planejamento acadêmico, assim como discussões sobre problemas gerais da área, como a inclusão de minorias.

Nesta edição são apresentados mais de 300 trabalhos, os quais foram criteriosamente selecionados pelos comitês de programa de cada sub-evento do Congresso. Estes trabalhos foram incluídos nos anais.

Gostaríamos de registrar o nosso agradecimento a todas as pessoas que fizeram possível a realização deste evento: Agradecemos à Diretoria e Conselho da SBC por ter confiado na nossa equipe e escolhido a nossa proposta de realização do Congresso pela primeira vez em Natal. Todos os organizadores e membros dos comitês de programa dos sub-eventos do Congresso, os quais contribuíram para a qualidade das apresentações e discussões; A UFRN e o DIMAp, por ter apoiado a iniciativa de forma concreta, sem hesitar e desde o primeiro momento; Aos patrocinadores (CGI.br, CAPES, CNPq, SENAC, INES, SERPRO, BNB e Ministério da Previdência Social), pela confiança depositada; Às outras empresas que apoiaram esta iniciativa mediante contribuições menos expressivas mas igualmente importantes (Springer, Google e SOFTEX). Finalmente, mas não menos importante, expressamos aqui o nosso mais profundo agradecimento à equipe de apoio local, formada por professores do

DIMAp/UFRN, IFRN e UERN e alunos de graduação e pós-graduação da UFRN, sem cuja competência, dedicação, paciência e seriedade teria sido impossível organizar este evento.

Aos participantes, desejamos que aproveitem ao máximo das atividades do Congresso. Esperamos lhes fornecer um ambiente fértil no qual possam ser plantadas as sementes de novas parcerias e trabalhos que, certamente, contribuirão ao avanço da Computação no Brasil. Desejamos também que aproveitem os horários livres para usufruir das belezas naturais e diversidade de programação social que a cidade proporciona.

Natal, RN, junho de 2011.

Martin A. Musicante e Thais Vasconcelos Batista, DIMAp/UFRN
Coordenadores Gerais do CSBC2011

COMPUTEC - XIII Computação e Mercado

PREFÁCIO

O XIII COMPUTEC (Computação e Mercado) examina a Computação através de um viés tecnológico e defende a idéia de que o avanço adequado das subáreas da Computação depende fortemente da cooperação entre universidades, setor empresarial e governo, integrados à visão dos usuários atuantes nos diversos segmentos. Assim, o COMPUTEC é organizado sob a perspectiva de que as diversas formas de cooperação entre universidades, empresas e governo nas mais diferente instâncias podem contribuir com o avanço da Computação no Brasil.

Neste ano, o COMPUTEC considera o tema do Congresso da SBC – Computação para todos: no caminho da evolução social – e se organiza em uma série de painéis que tratam de pontos-chave para a viabilização do tema, como usabilidade, design, computação nas nuvens e computação ubíqua. O painel de abertura, intitulado "*Viabilizando a 'Computação para todos'*", discute políticas de incentivos à pesquisa, desenvolvimento e inovação, articulação de cadeias produtivas, financiamento, estado atual da "universalização" da Computação no Brasil e potencial do país em relação ao tema, considerando o crescimento econômico, os eventos esportivos de dimensão global e a consequente modernização, intrinsecamente ligados às tecnologias de informação e comunicação.

O painel "*Usabilidade e Design na 'Computação para todos'" considera o* fato de que para fazer a Computação chegar a todos, academia e indústria têm tratado, com cada vez mais importância, os aspectos de usabilidade e design de sistemas computacionais. O painel discute os avanços de pesquisa e desenvolvimento para proporcionar o mais adequado design e a melhor usabilidade possível para o crescente número de usuários de sistemas computacionais.

Em seguida, o painel "*Cloud Computing e o seu papel na 'Computação para todos'*" discute como o paradigma de Computação nas Nuvens (Cloud Computing) pode, de fato, contribuir para a popularização de acesso a serviços computacionais diversos. Neste sentido, trata de iniciativas de infraestrutura e desenvolvimento de aplicações (corporativas e/ou pessoais) de nuvens no país, ao mesmo tempo em que mostra avanços científicos e tendências tecnológicas e de mercado em Cloud Computing ao redor do mundo.

Por fim, com o advento de dispositivos como tablets e smartphones e de sistemas de software e de comunicação cada vez mais presentes em nossas vidas, o painel "*Computação Ubíqua: a qualquer hora e em qualquer lugar - intrusiva ou não?*" discute como a chamada Computação Ubíqua afeta, positiva e/ou negativamente, a vida dos cidadãos cada vez mais digitais. Em particular, é discutido Mercado Atual e Tendências em dispositivos e aplicações para ubiquidade, além do status atual e perspectivas para o futuro das redes móveis no Brasil.

Em todos os painéis, debatedores dos vários segmentos (governo, indústria e academia) contribuem para discussões enriquecedoras e produtivas, despertando idéias inovadoras entre todos os participantes.

Carlos Ferraz e Gibeon de Aquino Júnior
Coordenadores do COMPUTEC

CTD - XXIV Concurso de Teses e Dissertações

PREFÁCIO

O Concurso de Teses e Dissertações (CTD) do CSBC tem como objetivo divulgar e premiar os melhores trabalhos de doutorado e mestrado na área de Computação realizados no país. Neste ano, o XXIV CTD selecionou 6 teses de doutorado e 10 dissertações de mestrado defendidas e aprovadas ao longo de 2010 para representar os melhores trabalhos desenvolvidos em nossos programas de pós-graduação. Esses trabalhos foram julgados de acordo com seus resultados científicos e tecnológicos já gerados, assim como pelo seu potencial de impacto na sociedade e no estado da arte na área de Computação.

Com base em sugestões de anos anteriores e com o intuito de aprimorar o processo de seleção de trabalhos para o CTD, algumas regras de submissão e análise dos trabalhos foram modificadas este ano. O concurso manteve duas fases de seleção. A primeira fase visa escolher as 6 melhores teses de doutorado e as 10 melhores dissertações de mestrado de 2010, e seu julgamento é baseado no material submetido pelos candidatos. Apenas os trabalhos selecionados na primeira fase participam da segunda fase do concurso, que visa selecionar e premiar os três melhores trabalhos em cada categoria (doutorado e mestrado). Essa segunda fase também leva em consideração a qualidade das apresentações orais dos trabalhos, que serão realizadas durante o CSBC 2011 em Natal, RN.

Este ano recebemos para a primeira fase do concurso 57 submissões, divididas em 25 trabalhos de doutorado e 32 de mestrado. As principais modificações introduzidas este ano foram na primeira fase do concurso. Além do tradicional resumo de seis páginas com a essência do trabalho, os autores submeteram também o texto completo de suas teses e dissertações. O julgamento feito pelos avaliadores nessa fase foi baseado essencialmente nos resumos, que descrevem os principais aspectos do trabalho e procuram realçar a relevância, as contribuições, a originalidade e as possibilidades de impacto do trabalho. Os textos completos dos trabalhos foram usados como fonte de consulta para esclarecer eventuais dúvidas dos avaliadores.

Solicitamos também cartas de recomendação do orientador e de pelo menos um dos membros da banca da defesa da tese ou dissertação. Pedimos que as cartas destacassem as razões pelas quais o trabalho deveria estar entre os melhores de 2010. As cartas de recomendação se mostraram um instrumento muito útil no processo de seleção. Acreditamos que elas serviram como um mecanismo de pré-seleção das submissões. Além disso, muitas vezes elas trouxeram outras informações e perspectivas sobre os trabalhos, ou confirmaram a opinião dos avaliadores com argumentos bem específicos.

Praticamente todos os trabalhos recebidos foram de excelente qualidade. Essa alta qualidade aliada à grande diversidade de áreas tornam a seleção do CTD altamente desafiadora. Optamos por convidar para cada trabalho avaliadores especializados no tema em questão, o que garantiu avaliações de um modo geral muito boas e precisas.

Nestes anais, está publicado o resultado dessa primeira fase do CTD 2011, através de resumos que apresentam a proposta e os resultados de cada trabalho, assim como evidências de sua excepcionalidade. Naturalmente, para um entendimento mais aprofundado dos trabalhos, é necessário consultar os textos originais das teses e dissertações.

Em nome da SBC e da nossa comunidade de pesquisadores, agradecemos aos avaliadores, que fizeram um excelente trabalho e confirmaram o espírito colaborativo de nossa comunidade, e especialmente a todos os recém doutores e mestres, e seus respectivos orientadores, que submeteram trabalhos para o CTD 2011 e que tão bem representam a nossa comunidade com seus trabalhos de excelência inquestionável. Por fim, parabenizamos os autores dos 16 trabalhos selecionados por mais essa conquista.

Renato Cerqueira e Bruno Motta de Carvalho
Coordenadores do CTD 2011

COMISSÕES

Coordenação Geral

Renato Cerqueira, PUC-Rio
Bruno Motta de Carvalho, UFRN

Comissão Julgadora e de Programa

Alberto Raposo, PUC-Rio
Alberto Sardinha, Universidade Técnica de Lisboa, Portugal
Alessandro Garcia, PUC-Rio
Alexandre Falcão, UNICAMP
Alfredo Goldman, IME-USP
Aline Viana, INRIA Saclay-Ile de France, França
Ana Teresa Martins, UFC
Anselmo Montenegro, UFF
Antônio Abelém, UFPA
Antonio Alfredo Ferreira Loureiro, UFMG
Artur Ziviani, LNCC
Avelino Zorzo, PUC-RS
Bianca Zadrozny, IBM Research Brasil
Bruno Motta de Carvalho, UFRN
Carlos Mello, UFPE
Carlos Eduardo Ferreira, IME-USP
Cecilia Rubira, UNICAMP
Celina Figueiredo, COPPE/UFRJ
Clarisse de Souza, PUC-Rio
Claudia Linhares Sales, UFC
Claudio Jung, UFRGS
Claudio Pinhanez, IBM Research Brasil
Clodoveu Davis, UFMG
Edna Barros, UFPE
Eduardo Laber, PUC-Rio
Edward Hermann Haeusler, PUC-Rio
Fabio André Machado Porto, LNCC
Flavio Wagner, UFRGS
Francisco Brasileiro, UFCG
Gisele Pappa, UFMG
Guido Araujo, UNICAMP
Jayme L. Szwarcfiter, UFRJ
Joao Comba, UFRGS
Joaquim Cavalcante-Neto, UFC
Luciano Paschoal Gaspary, UFRGS
Marcelo Carvalho, UFMS
Marcelo Finger, IME-USP
Marcelo Gattass, PUC-Rio
Marco Antonio Casanova, PUC-Rio

Marco Aurélio Spohn, UFCG
Marcus Fontoura, Google
Maria Cristina de Oliveira, ICMC-USP/São Carlos
Mário Benevides, COPPE/UFRJ
Patricia Jaques, UNISINOS
Raimundo Jose de Araújo Macedo, UFBA
Renata Galante, UFRGS
Renato Cerqueira, PUC-Rio
Roberto Ierusalimschy, PUC-Rio
Rogerio de Lemos, University of Kent, Reino Unido
Simone Martins, UFF
Tiago Carneiro, UFOP
Tsang Ing Ren, UFPE
Ulisses Mello, IBM Research Brasil
Vanessa Braganholo, UFF
Vinod Rebello, UFF
Violeta Quental, PUC-Rio
Waldemar Celes, PUC-Rio
Yoshiko Wakabayashi, IME-USP

Lista de Avaliadores

Alberto Raposo, PUC-Rio
Alberto Sardinha, Universidade Técnica de Lisboa, Portugal
Alessandro Garcia, PUC-Rio
Alexandre Falcão, UNICAMP
Alfredo Goldman, IME-USP
Aline Viana, INRIA Saclay-Ile de France, França
Ana Teresa Martins, UFC
Anarosa Brandão, Escola Politécnica - USP
Anselmo Montenegro, UFF
Antônio Abelém, UFPA
Antonio Alfredo Ferreira Loureiro, UFMG
Artur Ziviani, LNCC
Avelino Zorzo, PUC-RS
Bianca Zadrozny, IBM Research Brasil
Bruno Motta de Carvalho, UFRN
Carlos Mello, UFPE
Carlos Eduardo Ferreira, IME-USP
Cecilia Rubira, UNICAMP
Celina Figueiredo, COPPE/UFRJ
Clarisse de Souza, PUC-Rio
Claudia Linhares Sales, UFC
Claudio Jung, UFRGS
Claudio Pinhanez, IBM Research Brasil
Clodoveu Davis, UFMG
Daniel Figueiredo, COPPE/UFRJ
Delano Beder, UFSCar

Edna Barros, UFPE
Eduardo Laber, PUC-Rio
Edward Hermann Haeusler, PUC-Rio
Fabio André Machado Porto, LNCC
Flavio Wagner, UFRGS
Francisco Brasileiro, UFCG
Gisele Pappa, UFMG
Guido Araujo, UNICAMP
Jayme L. Szwarcfiter, UFRJ
Joao Comba, UFRGS
Joaquim Cavalcante-Neto, UFC
Leonardo Santiago, DEP/UFMG
Luciano Paschoal Gaspary, UFRGS
Marcelo Carvalho, UFMS
Marcelo Finger, IME-USP
Marcelo Gattass, PUC-Rio
Marco Antonio Casanova, PUC-Rio
Marco Aurélio Spohn, UFCG
Marcus Fontoura, Google
Maria Cristina de Oliveira, ICMC-USP/São Carlos
Marilia Braga, INMETRO
Mário Benevides, COPPE/UFRJ
Nelio Cacho, UFRN
Nina Hirata, IME-USP
Pablo Viana, UFAL
Patricia Jaques, UNISINOS
Raimundo Jose de Araújo Macedo, UFBA
Renata Galante, UFRGS
Renato Cerqueira, PUC-Rio
Roberto Ierusalimschy, PUC-Rio
Rogerio de Lemos, University of Kent, Reino Unido
Simone Martins, UFF
Tiago Carneiro, UFOP
Toacy Oliveira, COPPE/UFRJ
Tsang Ing Ren, UFPE
Ulisses Mello, IBM Research Brasil
Vanessa Braganholo, UFF
Vinod Rebello, UFF
Violeta Quental, PUC-Rio
Waldemar Celes, PUC-Rio
Yoshiko Wakabayashi, IME-USP

ARTIGOS

A Model for Simulation of Color Vision Deficiency and A Color Contrast Enhancement Technique for Dichromats

Gustavo M. Machado[1], **Manuel M. Oliveira**[1] **(Advisor)**

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)

{gmmachado,oliveira}@inf.ufrgs.br

Abstract. *We present the first model in the literature for simulating human color perception that consistently accounts for normal color vision and for the most prevalent cases (99.96% of all cases) of color vision deficiency. We also present an automatic image-recoloring technique for enhancing color contrast for dichromats that is up to two orders of magnitude faster than previous approaches and, due to its time coherence, is the first technique suitable for real-time video recoloring for dichromats. Finally, we introduce a metric for estimating the loss of color contrast experienced by individuals with color vision deficiency (CVD), or resulting from image recoloring. Together, these results contribute for a better understanding of CVD and for improving the quality of life of the affected individuals.*

Keywords: Models of Color Vision, Color Perception, Simulation of Color Vision Deficiency, Image and Video Recoloring, Visualization.

1. Introduction

Color Vision Deficiency (CVD) affects people's ability to distinguish certain colors. Estimates indicate that approximately 200,000,000 (two hundred million) individuals worldwide are affected by some kind of CVD, and currently there is no clinical or surgical treatment for this kind of condition. Individuals with CVD face difficulties to perform color-related tasks, which interfere with their personal and professional lives. Figure 1 illustrates this situation, where the reference image on the left shows an example of medical visualization in which colors represent tissue density. The image on the center depicts how the reference image is perceived by a class of individuals with CVD. Note the loss of color contrast, which results in the loss of important information. This example illustrates a recurring and challenging situation faced by these individuals in their daily activities.

Human normal color vision (also called normal trichromacy) requires three kinds of retinal photoreceptors with peak sensitivity in the long, medium and short wavelengths portions of the visible spectrum. Such photoreceptors are called L , M , and S *cones*, respectively. The spectral response of each kind of cone is defined by the specific type of photopigment it contains. Natural variations of some proteins that constitute a given photopigment may shift its sensitivity to a different band of the spectrum. In this case, the condition is called *anomalous trichromacy*. *Dichromacy* is caused by the absence of one of the photopigments and can be classified as *protanopia*, *deuteranopia*, or *tritanopia*, if the missing photopigment is the one associated in normal color vision with the L, M, or S cones, respectively. Anomalous trichromacy and dichromacy are the most common types of CVD, covering more than 99.99% of the cases.

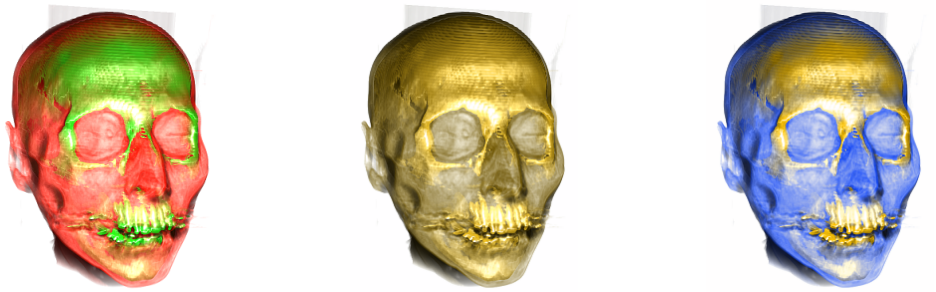


Figure 1. Example illustrating the difficulties faced by individuals with CVD. (left) A medical-visualization image used as reference. (center) Simulation of how deuteranopes perceive the reference image. (right) Recolored image as perceived by deuteranopes, recovering the loss of color contrast. These images were simulated and recolored using the techniques presented in this thesis.

1.1. Contributions

This thesis presents three main contributions related to color vision deficiency: a model for simulating human color perception, an automatic image-recoloring technique to enhance color contrast for dichromats, and a metric for estimating the loss of color contrast experienced by individual with CVD or resulting from image recoloring.

Our physiologically-based model for simulating human color perception is the first model in the literature capable of consistently simulating the perception of normal-color-vision individuals as well as the most prevalent cases of CVD in a unified way. It handles approximately 99.96% of all CVD cases, while previous approaches only handle about 27.46%. Our model was validated by a user experiment, and a paper describing it, entitled *A Physiologically-based Model for Simulation of Color Vision Deficiency* [Machado et al. 2009], was published in the *IEEE Transactions on Visualization and Computer Graphics*, the most prestigious journal in the visualization community.

Our second contribution is an automatic image-recoloring technique for enhancing color contrast for dichromats. Our solution has a linear cost in the number of pixels and, for typical images sizes, is up to two orders of magnitude faster than previous approaches. It is also the first technique capable preserving temporal coherence and performing high-quality video recoloring for dichromats in real time. In this context, we also introduced a third contribution: a metric for estimating the loss of color contrast experienced by individuals with color vision deficiency, or resulting from image recoloring. A paper describing these two contributions, entitled *Real-Time Temporal-Coherent Color Contrast Enhancement for Dichromats* [Machado and Oliveira 2010], was published in the *Computer Graphics Forum* journal, one of the most important in the field.

GPU implementations of both the simulation model and the recoloring technique were successfully integrated with a visualization system, providing practical validation of our results. Figure 1 (right) shows a result of the proposed recoloring technique for enhancing color contrast for dichromats. This image corresponds to the perception of a deuteranope for the image shown on Figure 1 (left). Note the recovery of color contrast by comparing it to Figures 1 (left) and (center).

Due to space restrictions, this article cannot present all the details of the developed simulation model, recoloring technique, and image-contrast met-

ric. The complete thesis work [Machado 2010] and the homepages of both publications [Machado et al. 2009, Machado and Oliveira 2010] can be found in `http://www.inf.ufrgs.br/~gmmachado`. We suggest that the readers watch the videos available in these homepages for more intuitive descriptions and examples of applications of the developed techniques.

2. Simulation

Despite the relevance of understanding how individuals with CVD perceive colors, little had been done in terms of simulating their perception for normal trichromats. In particular, none of the previous approaches is capable of handling both dichromacy and anomalous trichromacy. The first techniques that simulate the perception of individuals with dichromacy [Meyer and Greenberg 1988, Brettel et al. 1997] were developed based on the reports of *unilateral dichromats*, *i.e.*, individuals with one dichromatic eye and one normal trichromatic eye. According to these reports, achromatic colors as well as two other hues are perceived similarly by both eyes. Previous techniques mapped these colors to two semi-planes in some color space, and defined the simulation technique as the projection of the original colors on these semi-planes. This approach, however, cannot be used to simulate anomalous trichromacy.

2.1. The Model for Color Perception Simulation

Our physiologically-based model for color perception simulation treats CVD as changes in the spectral absorption of the cones' photopigments. While CVD is essentially modeled at the retinal photopigment stage, the opponent-color stage is crucial for producing the correct results and cannot be underestimated. For this, our approach uses Ingling and Tsou's model [1977] (Figure 2 left) for representing the opponent-color stage.

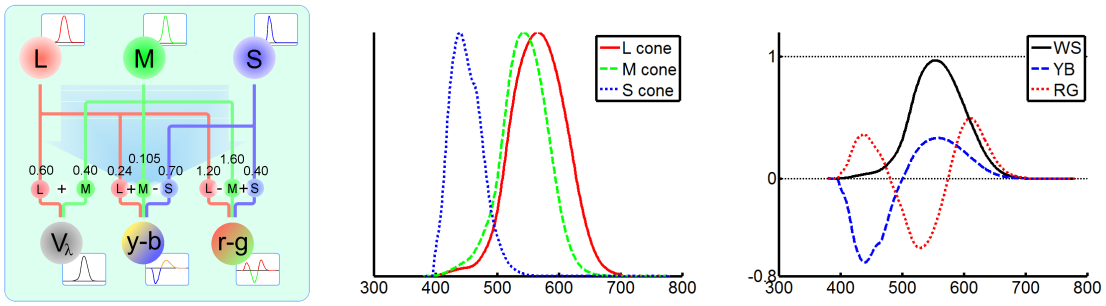


Figure 2. (left) Two-stage model of human color vision. The output of the photoreceptor stage (L, M and S cones) is linearly combined in the opponent stage (V_λ, $y - b$, and $r - g$ nodes). (center) Cone spectral sensitivity functions for an average normal trichromat (after Smith and Pokorny [1975]). (right) Spectral response functions for the opponent channels of the average normal trichromat according to Ingling and Tsou's [1977] model.

A transformation from an RGB color space to an opponent-color space is obtained by projecting the spectral power distribution functions of the RGB primaries onto the appropriate set of opponent functions $WS(\lambda)$, $YB(\lambda)$, and $RG(\lambda)$ (Figure 2, right), which define the opponent-color space for either normal trichromats, for anomalous trichromats,

or for dichromats. These projections provide the elements of a matrix that directly maps RGB to the opponent-color space:

$$\Gamma = \begin{bmatrix} WS_R & WS_G & WS_B \\ YB_R & YB_G & YB_B \\ RG_R & RG_G & RG_B \end{bmatrix}. \quad (1)$$

Let Γ_{normal} and Γ_{CVD} be the matrices that map RGB to the opponent-color space of a normal trichromat and of a given class of CVD, respectively. The simulation, for a normal trichromat, of the color perception of this class of CVD, is obtained with Equation 2. As shown in the thesis, this general solution applies to the simulation of 99.96% of all cases of color vision deficiency.

$$\begin{bmatrix} R_s \\ G_s \\ B_s \end{bmatrix} = \Gamma_{normal}^{-1} \Gamma_{CVD} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2)$$

3. Recoloring

Several researchers have investigated the problem of image-recoloring for dichromats. The most relevant techniques [Rasche et al. 2005, Kuhn et al. 2008] are optimization-based approaches that treat image-recoloring as a dimensionality-reduction problem in an approximately perceptually uniform color space (*e.g.*, CIE $L^*a^*b^*$). They approximate the color gamut of dichromats with a plane and attempt to preserve the distances between all pairs of colors. Kuhn et al. [2008] uses a mass-spring system to optimize color distances on the plane while preserving color naturalness. Despite being about three orders of magnitude faster than previous techniques, Kuhn et al.'s approach still does not achieve real-time performance. A recent study [Lee and dos Santos 2011] published after this thesis dealt with image-recoloring using fuzzy logic.

3.1. The Recoloring Algorithm for Dichromats

Our recoloring solution is based on the key observation that, whenever dichromats experience some significant loss of color contrast, most of this contrast can be recovered by working on a perceptually uniform color space, and orthographically projecting the original colors onto a plane that minimizes contrast loss (in a least-squares sense). The coordinates of these projections then become the new color coordinates on the reduced (2D) color gamut of the dichromat. Figure 5.3 summarizes this process, which consists of the following steps:

1. Estimation of the vector v_{ab} that represents the direction that maximizes contrast loss in the CIE $L^*a^*b^*$ chromaticity plane, and
2. Orthographic projection of the original colors onto the plane defined by v_{ab} and the lightness (L^*) axis. The projection on such a plane minimizes contrast loss. The projected color coordinates are then rotated around L^* to align themselves to the plane of the dichromat, and the resulting colors are used to recolor the image.

In order to guarantee temporal coherence, the algorithm checks and corrects for abrupt changes in the sense of v_{ab}. Figure 4 compares the results of our recoloring technique (*Our Recolor*) against the ones obtained using Kuhn et al.'s regular (*Kuhn's Recolor*) and exaggerated-contrast (*Kuhn's Exag.*) recoloring techniques. Our results are not only superior, but our technique is up to two orders of magnitude faster.

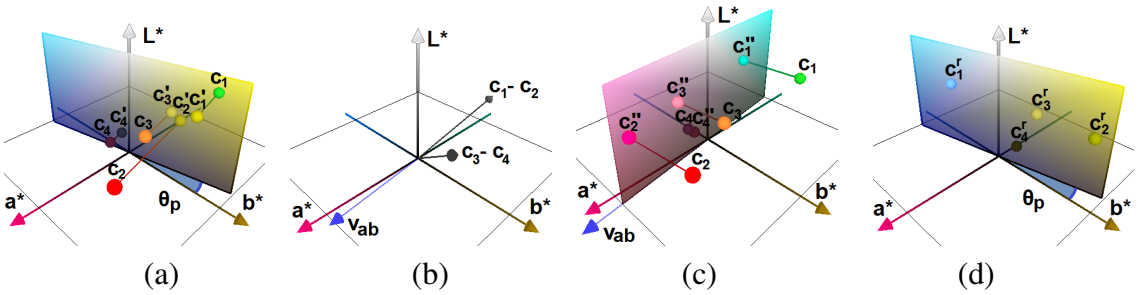


Figure 3. The steps of our recoloring algorithm. (a) Colors c_1 to c_4 are perceived by a dichromat as c'_1 to c'_4, respectively (their projections on the dichromat's gamut plane). The relative loss of contrast experienced by a dichromat for a pair of colors (c_i, c_j) is given by $l_{(c_i,c_j)} = (\|c_i - c_j\| - \|c'_i - c'_j\|)/(\|c_i - c_j\|)$, which happens along the direction $\vartheta_{ij} = c_i - c_j$. (b) Direction v_{ab} (shown in blue) that maximizes the loss of local contrast (in a least-square sense). (c) Projection of the original colors on the plane defined by v_{ab} and L^*. (d) Final colors obtained after rotating the projected colors c''_k in (c) around L^* so that they align with the dichromat's plane.

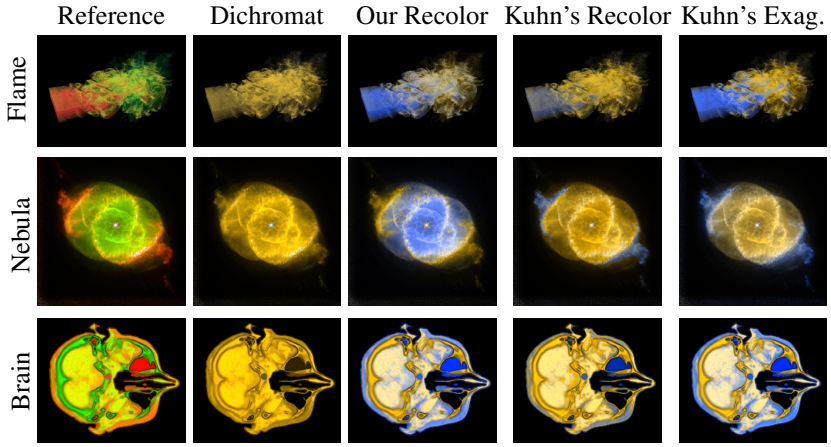


Figure 4. Comparison of the results produced by our recoloring technique and by Kuhn et al.'s [2008] for a set of scientific and medical visualization images. The *Dichromat* column shows the simulated perception of dichromats for the corresponding *Reference* images.

4. Conclusion

Color vision deficiency impacts the professional and personal lives of approximately two hundred million individuals worldwide. It is essential to fully understand this condition and provide solutions that contribute to improve the quality of life of the affected individuals and promote their access to new technologies. This thesis contributes some important steps towards these goals. More specifically, it introduced the first model of human color perception to correctly simulate both normal color vision, anomalous trichromacy, and dichromacy in a unified way, handling about 99.96% of all cases of color vision deficiency. It also presented an automatic image-recoloring technique for dichromats, which is the first to perform high-quality recoloring in real-time and also the first to guarantee temporal coherence. We also developed a metric for estimating the loss of color contrast experienced by individuals with CVD, or resulting from recolorings.

We demonstrated the effectiveness of these techniques by integrating them into a real-time visualization application and showing how they can significantly improve human-computer interface for individuals with CVD. Our simulation model helps normal-color-vision designers to better understand the limitations of individuals with CVD, which results in more effective designs for wider audiences. In turn, our real-time recoloring technique gives individuals with CVD the ability to explore color-related contents autonomously, reducing the ambiguity that often imposes significant challenges to these individuals.

Acknowledgments

We thank Leandro A. F. Fernandes, Giovane Kuhn, and Francisco Pinto for their help in conducting this research. This work was sponsored by CNPq-Brazil (grants 200284/2009-6, 131327/2008-9, 476954/2008-8, 305613/2007-3 and 142627/2007-0).

References

Brettel, H., Viénot, F., and Mollon, J. D. (1997). Computerized simulation of color appearance for dichromats. *J. Opt. Soc. Am.*, 14(10):2647–2655.

Kuhn, G. R., Oliveira, M. M., and Fernandes, L. A. F. (2008). An improved contrast enhancing approach for color-to-grayscale mappings. *Vis. Comput.*, 24(7):505–514.

Lee, J. and dos Santos, W. P. (2011). An adaptive fuzzy-based system to simulate, quantify and compensate color blindness. *Integr. Comput.-Aided Eng.*, 18:29–40.

Machado, G. M. (2010). A physiologically-based model for simulation of color vision deficiency and real-time temporal-coherent color contrast enhancement for dichromats. Master's thesis, UFRGS, Porto Alegre, Brazil.

Machado, G. M. and Oliveira, M. M. (2010). Real-time temporal-coherent color contrast enhancement for dichromats. *Computer Graphics Forum*, 29(3):933–942.

Machado, G. M., Oliveira, M. M., and Fernandes, L. A. F. (2009). A physiologically-based model for simulation of color vision deficiency. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1291–1298.

Meyer, G. W. and Greenberg, D. P. (1988). Color-defective vision and computer graphics displays. *IEEE Comput. Graph. Appl.*, 8(5):28–40.

Rasche, K., Geist, R., and Westall, J. (2005). Re-coloring images for gamuts of lower dimension. *Comput. Graph. Forum*, 24(3):423–432.

Segmentação Supervisionada de Imagens pela Funcional de Mumford-Shah Utilizando Métricas de Distância Não-lineares

Antonio Carlos Sobieranski[1], **Eros Comunello**[1], **Aldo von Wangenheim**[1]

[1]Departamento de Informática e Estatística (INE)
Programa de Pós-Graduação em Ciência da Computação (PPGCC)
Universidade Federal de Santa Catarina (UFSC) – Florianópolis – Brasil

{asobieranski,eros}@cyclops.ufsc.br, awangenh@inf.ufsc.br

Abstract. *In this work a supervised region-merging image segmentation approach is presented. In our approach, a distance metric is learned from some features on the image, and then a function for the Mumford-Shah model is optimized by this metric. The global optimum of the approximation function is inductively achieved under high polynomial terms of the Mahalanobis distance, extracting the nonlinear features of the pattern distributions into topological maps. The results we obtained show a better discrimination of object boundaries and the location of regions when compared with the conventional Mumford-Shah algorithm and other image segmentation methods. A quantitative objective evaluation of the proposed approach was performed in order to compute the quality of the obtained results.*
Image segmentation, Mumford-Shah model, distance metric learning, global optimization.

Resumo. *Neste trabalho é apresentada uma abordagem supervisionada de segmentação de imagens por crescimento de regiões. Nesta abordagem uma métrica de distância é modelada a partir de algumas características da imagem, e então utilizada para otimizar o modelo de Mumford-Shah. A função de aproximação do modelo é indutivamente obtida sobre termos de alta ordem polinomial da distância de Mahalanobis, possibilitando extrair as características não-lineares das distribuições em mapas topológicos. Os resultados obtidos apresentam uma melhor discriminação das fronteiras dos objetos quando comparados com o modelo convencional e em relação a outros métodos de segmentação. Para verificar a qualidade dos resultados obtidos, métricas de avaliação de qualidade foram utilizadas.*
Segmentação de imagens, modelo de Mumford-Shah, aprendizado de métrica de distância, otimização global.

1. Contextualização

A segmentação de imagens é uma das mais importantes ferramentas em Visão Computacional (VC) e Processamento Digital de Imagens (PDI). Seu uso possui aplicação direta nas mais diferentes áreas do conhecimento, tais como a indústria médica, robótica, aeroespacial, dentre outras. A segmentação consiste em codificar uma imagem de entrada I sem qualquer informação geométrica ou semântica em uma imagem de segmentos, onde cada segmento corresponda na totalidade ou parcialidade a uma região homogênea e/ou

contínua na cena. Através destes segmentos, além da imagem tornar-se mais simples de ser analisada (de pixels para regiões), os objetos constituintes tornam-se passíveis de reconhecimento e interpretação, possibilitando a posterior tomada de decisão.

Dentre os conceitos e métodos de segmentação existentes[1], um dos mais interessantes pela sua genialidade é o modelo da energia funcional de *Mumford-Shah* [Mumford and Shah 1989]. Este modelo variacional particiona uma imagem I em segmentos representativos W_i, onde a idéia básica é agrupar regiões onde o sinal de I seja homogêneo e delimitado por um sistema de fronteiras qualitativas K. Esta energia funcional é descrita através de um termo qualificador e dois termos regularizadores sendo:

$$E(u, K) = \beta \int_W (u - I)^2 + \int_{W/K} |\nabla u|^2 dx + \lambda \int_K d\sigma, \quad (1)$$

onde:

- o primeiro termo força a aproximação de um segmento u para a imagem I.
- o segundo termo quantifica e controla a suavidade do segmento u.
- o terceiro termo controla o comprimento, a suavidade e a locação das fronteiras K que dividem os segmentos.

De uma maneira geral, o modelo de *Mumford-Shah* acima descrito alcançou uma completa formulação geral para o problema da segmentação de imagens sob a forma de um problema de otimização [Brox 2005] (pg.102): a minimização da funcional $E(u, K)$ possibilita se obter uma imagem em *cartoon* (de regiões, ou W) em relação a uma determinada imagem de entrada I. A idéia geral do modelo de *Mumford-Shah* é buscar pela minimização desta funcional de energia através da otimização dos 3 termos penalizados acima descritos. Os coeficientes β e λ tornam este modelo multi-escala. β pondera a aproximação de I por u, e λ é um parâmetro de contraste que limita o comprimento total de fronteiras na segmentação: quando λ é pequeno, a minimização força os resultados a apresentarem muitas fronteiras (super-segmentação), enquanto que grandes valores de λ produzem o efeito contrário (sub-segmentação, com poucos segmentos)[2].

No entanto, o processo de otimização dos 3 termos penalizadores; isto é, a forma em como minimizar a energia tem sido deixada em aberto nesta funcional [Brox 2005] (pg.102). Embora o artigo original de *Mumford-Shah* [Mumford and Shah 1989] compreenda quase 100 páginas, este não apresenta uma simples sugestão em como implementar um framework básico para a segmentação [Brox and Cremers 2009]. Só para se ter uma idéia, o modelo chega a ser discutido pela sua genialidade em uma revista de fisiologia [Petitot 2003], apresentando uma visão "pedagógica" do modelo para o leitor não-matemático. Nos modelos analíticos apresentados existe uma grande diferença entre a sua formulação teórica e eficientes meios de se encontrar minimizadores desta funcional na prática [Brox and Cremers 2009]. Na dissertação em anexo são apresentados alguns trabalhos inteiramente dedicados à interpretação do modelo de *Mumford-Shah* (seção 3.2.2, pg.42), e uma estratégia de implementação bem difundida na comunidade científica é o modelo multi-escala proposto por [Koepfler et al. 1994][Morel and Solimini 1995], onde a funcional é acoplada a um modelo de segmentação por crescimento de regiões. De qualquer maneira, um dos maiores desafios identificados nas interpretações e conjecturas do

[1] vide documento de dissertação em anexo, seções 2.2 e 2.3.

[2] conforme demonstrado na imagem em *cartoon* da Figura 3.2 no documento de dissertação (pg.49).

modelo de *Mumford-Shah* é em desenvolver algoritmos eficientes para computar penalizadores de alta qualidade desta funcional [Pock et al. 2009].

2. Método Supervisionado de Segmentação de Imagens

Termos penalizadores podem ser computados através de funções de similaridade, as quais correspondem a um conceito chave para o pleno sucesso ou falha em um algoritmo de processamento de imagens. Por similaridade logo imagina-se proximidade, ou quão duas entidades se assemelham de acordo com suas características. No entanto, a computação de similaridade no contexto de segmentação de imagens está sujeito a muitos fatores, tais como diferentes espaços de cores (RGB, HSV, CIE-Lab, etc) e diferentes métricas de distância (lineares, não-lineares, multivariadas, etc). Em [Sobieranski et al. 2011], é discutido como a grande variabilidade destes fatores pode dificultar na escolha de uma boa métrica de similaridade. Geralmente muitos algoritmos são penalizados pela grande quantidade de parâmetros de execução utilizados para controlar ao mesmo tempo homogeneidade e continuidade dos segmentos.

Nesta dissertação de mestrado optou-se por uma abordagem supervisionada de segmentação de imagens, onde diferentemente das abordagens parametrizadas, a funcional de *Mumford-Shah* é minimizada através de uma função de similaridade não-linear (chamada aqui de função de aproximação), modelada através da inferência do observador externo. Na Figura 4.2 no documento de dissertação é apresentada uma visão geral da metodologia desenvolvida, e esta pode ser sumarizada em 3 etapas:

- Aprendizado da métrica de distância: o observador infere sobre a cena, determinando um conjunto de treinamento (denominado de S), através da extração de características do objeto de interesse. Este conjunto S corresponde a uma distribuição multivariada das cores do objeto selecionado, e o objetivo nesta etapa é modelar estatisticamente o comportamento não-linear que S apresenta no espaço de cores de origem. Através desta modelagem, uma topologia M é obtida para ser utilizada como métrica de similaridade na funcional de *Mumford-Shah*;
- Função de aproximação: a energia funcional de *Mumford-Shah* é adaptada de modo que a topologia de M atue diretamente com os termos penalizadores da equação. Esta alteração nos termos da energia funcional permite que o processo de minimização ocorra em função desta nova topologia não-linear, alterando a ordem dos agrupamentos das regiões;
- A etapa final do processo é a execução do algoritmo através do agrupamento das regiões utilizando os novos termos penalizadores. Nesta etapa o usuário determina o número de regiões esperadas que a funcional deve produzir. O algoritmo inicia em nível de pixel, onde cada pixel é uma região que é verificada em relação aos adjacentes. Regiões similares de acordo a nova $E(u, K)$ são agrupadas sucessivamente, até que o número desejado de regiões seja obtido, conforme pode ser visualizado na Figura 4.2 supracitada, onde a imagem de entrada em questão foi reduzida para 4 segmentos representativos.

Essa não-linearidade é capturada através da distância polinomial de Mahalanobis [Grudic and Mulligan 2006], uma variação da distância estatística onde os termos de S são projetados em alta ordem dimensional. Com isto, as topologias de um determinado conjunto de entrada S podem ser expressas não-linearmente, conforme demon-

strados nos mapas topológicos no documento de dissertação[3], seção 4.2.3, ou em [Sobieranski et al. 2011]. Quanto a função de aproximação, esta teve seus termos penalizados alterados de modo que a topologia de M pudesse trabalhar harmonicamente e com regularidade conforme as regiões fossem agrupadas.



Figure 1. Dois conjuntos de treinamento utilizados (objeto e background), demonstrando 2 minimizações diferentes da energia, ambas de 80 para 2 regiões.

Quanto aos resultados obtidos, o método supervisionado proposto foi testado com base no banco de imagens de *Berkeley*[4], um *dataset* onde cada imagem possui entre 5 a 7 segmentações realizadas manualmente por especialistas humanos. A vantagem da utilização deste *dataset* é a possibilidade de utilizar estas segmentações manuais (denominadas de *ground-truths* - *GT*) para comparar os resultados de diferentes algoritmos de segmentação. Neste contexto, foram utilizadas 4 métricas de avaliação de segmentos, as quais possibilitaram avaliar quão similar é um resultado de segmentação produzido por um algoritmo em relação a uma imagem de GT.

O ambiente de experimento foi o seguinte:

- 60 imagens de *Berkeley* foram selecionadas, onde para cada imagem foram selecionados um conjunto de treinamento com base em alguma característica da cena. Em algumas destas imagens o conjunto de treinamento foi variado, totalizando em 63 experimentações. O algoritmo supervisionado (chamado de SMS) processou 22 segmentações para cada uma destas 63 imagens, decrescendo[5] o número de regiões de 250 para 2, gerando 1.386 resultados. Um exemplo deste decréscimo é apresentado na Figura 1, onde 80 regiões são reduzidas a somente 2;
- cada imagem resultante (das 1.386) foram comparadas com os respectivos GT's (de 5 a 7 por imagem), correspondendo a 7.601 comparações realizadas, conforme demonstrado na Figura 2;
- cada comparação foi regida por 4 índices de avaliação de qualidade: Rand, Fowlkes-Mallows, Jacard, Dongen (referenciados na seção 2.4 da dissertação), totalizando em 30.404 índices de avaliação obtidos.
- várias estatísticas foram geradas a partir destes índices, conforme seção 5 em [Sobieranski et al. 2011].

[3]Figuras 4.5, 4.7 e 4.9.

[4]http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/

[5]250, 225, 180, ..., 4, 3, 2 regiões, em 22 intervalos.

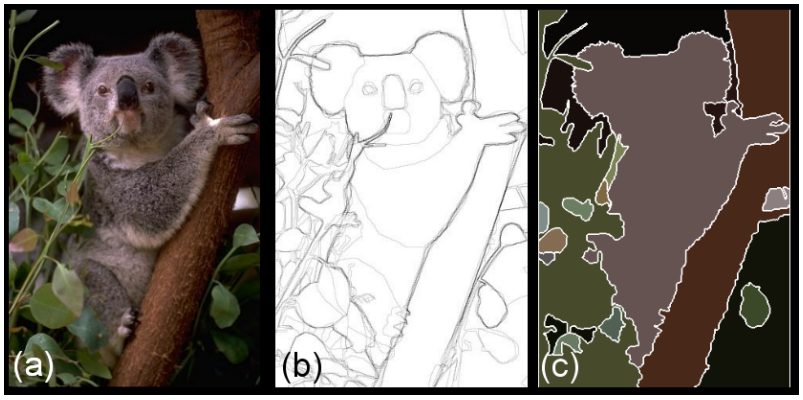


Figure 2. (a) Imagem de entrada, (b) imagem GT, (c) resultado SMS.

O experimento foi subdividido em 3 análises: (a) evolução da qualidade do agrupamento das regiões, com o objetivo de verificar a influência de diferentes conjuntos de treinamento na minimização da funcional (exemplificado na Figura 1); (b) comparação em relação a outros métodos de segmentação estado-da-arte, cujos índices médios de qualidade apresentaram uma vantagem para o método SMS; (c) comparação em relação ao modelo convencional de Mumford-Shah multi-escala.

Todos os experimentos descritos neste resumo estão detalhados no documento de dissertação (capítulo 5), no artigo publicado em [Sobieranski et al. 2011], e adicionalmente no site do método SMS: http://www.lapix.ufsc.br/sms/, onde todos os índices e resultados de segmentação podem ser visualizados.

3. Considerações Finais

O modelo supervisionado de segmentação de imagens proposto apresenta resultados superiores (visuais e quantitativos) em relação aos métodos comparados. Em alguns casos, a flexibilidade do método foi verificada através da utilização de diferentes conjuntos de treinamento sobre a mesma imagem. Quando comparado a outros algoritmos de segmentação, SMS consegue ao mesmo tempo descrever melhor os segmentos em um menor número de regiões, conforme demonstrado pelos índices médios de qualidade apresentados na seção 5.3.2 e 5.4 do documento de dissertação.

A grande quantidade de casos utilizados nos experimentos possibilitaram também avaliar indiretamente não somente a qualidade do *dataset* de *Berkeley*, mas também a representatividade dos índices de avaliação de qualidade de segmentações. Esta é uma tendência que vem sido exigida na comunidade científica, e poucos trabalhos publicados vem realizando este tipo de validação de qualidade. Espera-se com este trabalho ressaltar esta cultura na comunidade científica, além de incentivar o desenvolvimento de novas métricas de qualidade. Quanto às contribuições científicas obtidas ao longo destes aproximadamente 3 anos de pesquisa, destacam-se:

- (preliminar, foco em imagem médica), artigo publicado e apresentado no 22nd IEEE CBMS – International Symposium on Computer-Based Medical Systems [Sobieranski et al. 2009b] – anexo A8 dissertação.
- (preliminar, foco em segmentação), Artigo publicado no periódico Pattern Recognition Letters (qualis A2) [Sobieranski et al. 2009a] – anexo A9 dissertação.

- (outro método de segmentação não relacionado diretamente ao SMS), Artigo publicado no periódico Pattern Recognition Letters (qualis A2) [Wangenheim et al. 2009] – anexo A10 dissertação.
- (resultado final dissertação), Artigo publicado no periódico Computer Vision and Image Understanding (qualis A1) [Sobieranski et al. 2011] – anexo A11 dissertação.
- site do método SMS (http://www.lapix.ufsc.br/sms), onde todas as imagens, tabelas de validação, métodos de avaliação de segmentos e experimentos realizados estão disponíveis para comunidade científica.

References

Brox, T. (2005). *From pixels to regions: partial differential equations in image analysis*. PhD thesis, Faculty of Mathematics and Computer Science, Saarland University, Germany.

Brox, T. and Cremers, D. (2009). On local region models and a statistical interpretation of the piecewise smooth Mumford-Shah functional. *Int. J. Comput. Vision*, 84(2):184–193.

Grudic, G. Z. and Mulligan, J. (2006). Outdoor path labeling using polynomial mahalanobis distance. In *Robotics: Science and Systems*.

Koepfler, G., Lopez, C., and Morel, J. M. (1994). A multiscale algorithm for image segmentation by variational method. *SIAM J. Numer. Anal.*, 31(1):282–299.

Morel, J. M. and Solimini, S. (1995). *Variational methods in image segmentation*. Birkhauser Boston Inc., Cambridge, MA, USA.

Mumford, D. and Shah, J. (1989). Optimal approximations by piecewise smooth functions and associated variational problems. *Pure Appl. Math.*

Petitot, J. (2003). An introduction to the Mumford-Shah segmentation model. *Journal of Physiology-Paris*, 97(2-3):335 – 342. Neurogeometry and visual perception.

Pock, T., Cremers, D., Bischof, H., and Chambolle, A. (2009). An algorithm for minimizing the piecewise smooth Mumford-Shah functional. In *IEEE International Conference on Computer Vision (ICCV)*, Kyoto, Japan.

Sobieranski, A. C., Abdala, D. D., Comunello, E., and von Wangenheim, A. (2009a). Learning a color distance metric for region-based image segmentation. *Pattern Recognition Letters*.

Sobieranski, A. C., Comunello, E., and von Wangenheim, A. (2011). Learning a nonlinear distance metric for supervised region-merging image segmentation. *Comput. Vis. Image Underst.*, 115:127–139.

Sobieranski, A. C., Mantelli Neto, S. L., Coser, L., Comunello, E., and von Wangenheim, A. (2009b). Learning a nonlinear color distance metric for the identification of skin immunohistochemical staining. *IEEE Computer-based Medical Systems - CBMS 2009*.

Wangenheim, A. V., Bertoldi, R. F., Abdala, D. D., Sobieranski, A., Coser, L., Jiang, X., Richter, M. M., Priese, L., and Schmitt, F. (2009). Color image segmentation using an enhanced Gradient Network Method. *Pattern Recogn. Lett.*, 30(15):1404–1412.

An Exact Algorithm for an Art Gallery Problem

Marcelo C. Couto[1] **, Cid C. de Souza**[1] **, Pedro J. de Rezende**[1]

[1]Institute of Computing – State University of Campinas (UNICAMP)

couto.marcelo@gmail.com, {cid,rezende}@ic.unicamp.br

Abstract. *In this master thesis, a geometrical* NP-HARD *problem, the* Art Gallery problem*, is studied. The problem goal is to minimize the number of guards sufficient to cover the interior of an art gallery whose boundary is represented by a simple polygon. Among the many variants, we focus on one where the guards are stationary and restricted to vertices of the polygon, which can be either orthogonal or general simple, without holes. An exact algorithm is proposed in which the original continuous problem is discretized. Proofs of correctness and convergence of the algorithm to an optimal solution are given. Extensive experimentation with the algorithm show that it solves to optimality instances with more than ten times the size of the largest ones reported earlier in the literature.*

Keywords. *combinatorial geometry, combinatorial optimization, art gallery problem, exact algorithm, set covering, visibility.*

1. Introduction

Nowadays, security is one of the main concerns in our society. Stories concerning robbed banks, museums, homes and other buildings are well known all over the world. As the demand for low cost and effective security systems increases, it is important to fully understand the problem in order to improve efficiency. The research on the Art Gallery problems (AGP) plays an important role in the decisions involved.

In 1973, Victor Klee posed the original AGP problem, which consists in determining the minimum number of guards sufficient to cover the interior of an n-wall art gallery. This work focus on the minimization problem of the specific variation where guards have a 360° field of vision and their placement is restricted to the vertices of the simple polygon, either orthogonal or general simple, that represents the outer boundary of a given art gallery. Both minimization problems have been proven NP-HARD and, while being one of the most studied problems in computational geometry, almost no attempts to develop an exact algorithm have been made, and none are complete, efficient and practical. In the literature there is a number of works on approximation algorithms, as well as a few on the placement of a sub-optimal number of guards.

This master thesis successfully addresses this problem and is well aligned within the main context. We propose an exact and efficient algorithm, perform a theoretical analysis on the convergence and exactness of the algorithm, and carry out an extensive experimental study on its practicality.

The proposed approach is iterative and consists of building an initial discretization of the simple polygon that represents the floor plan of the gallery. Once that phase is complete, the problem is modeled as an instance of the classical *Set Cover problem* (SCP), which is solved and then refined until an optimal solution for the original problem is attained.

Results from our work have been published on four international conferences and one international journal. Additionally, a technical video about our work was also published accompanying one of those papers. Lastly, a benchmark comprised of all our instances and results was made publicly available, for future comparative research. Some

of these papers are included as chapters of the thesis, as sanctioned by the university's graduate program, along with detailed comments scattered within those chapters to help the reader to completely understand the details of the work.

In the first three chapters, we introduce the problem and its motivation, explain the theoretical concepts and definitions required for a full understanding of the details, and present the design of the algorithm, the proofs of correctness and convergence, the reduction between the problems, the discretization strategies used in the experimental evaluation and the process of building the instances. Finally, there is an appendix that describes the development and implementation of the algorithm, the issues solved, the design and project, and two interfaces built to perform the experimental and the human evaluation and visualization of the algorithm.

2. Inception of the Algorithm

The geometrical perspective of the Art Gallery problem represents an n-wall art gallery as a planar region whose boundary consists of a simple polygon P with a set V of n vertices. A vertex $v \in V$ is denoted *convex* if the internal angle at v is smaller than $180°$. Whenever no confusion arises, *a point in* P will mean a point either in the interior or on the boundary of P.

A visibility region $\text{Vis}(v)$ of a vertex $v \in V$ is the set of all points in P visible from v. Any point $y \in P$ is visible from any other point $x \in P$ if and only if the closed segment joining x and y does not intersect the exterior of P.

A set of points S is called a *guard set* for P if and only if every point $p \in P$ is visible from at least a point $s \in S$. Thus, a *vertex guard set* G is any subset of vertices such that $\bigcup_{g \in G} \text{Vis}(g) = P$. As a vertex guard set guarantees that the entire gallery is overseen by the guards in the set, it is easy to see that an AGP amounts to finding the smallest subset $G \subset V$ that is a vertex guard set for P.

The problem of finding the smallest vertex guard set for P can be regarded as a specific Set Cover problem (SCP), where we wish to find a smallest cardinality set of visibility regions of the vertices of P whose union cover P, see Figure 1.

It is important to notice that this is a *continuous* SCP since there are infinitely many points in the interior of P to be covered. Notwithstanding, the original problem can be discretized in a finite number of representative points of P, $D(P)$, so that the formulation becomes manageable. This leads to the following IP formulation of the SCP, where A is a matrix and A_{ij} is set to 1 if and only if point $i \in D(P)$ is visible from vertex j, or 0 otherwise; x_j represents whether vertex j is in the solution set or not. We aim to minimize $z = \min \sum_{j \in V} x_j$ subject to $\sum_{j \in V} A_{ij} x_j \geq 1, \forall p_i \in D(P)$.

Given a feasible solution x to the IP above, let $Z(x) = \{j \in V \mid x_j = 1\}$. The constraints on the formulation states that every point in $D(P)$ is visible from at least one selected guard in the solution and the objective function minimizes the cardinality z of $Z(x)$. As the set $D(P)$ is finite, the solution set for the discretized problem $Z(x)$ may not be a viable solution for the original problem, i.e., a vertex guard set for P. In this scenario, the algorithm must iterate, picking a new discretization point inside each uncovered region and solving an IP formulation for the new SCP instance created.

One of the important results we proved is that once the algorithm finds a solution for the discretized instance $I(P, D(P))$ which is viable for the original problem, that solution is also optimal, i.e., a minimum vertex guard cover for P.

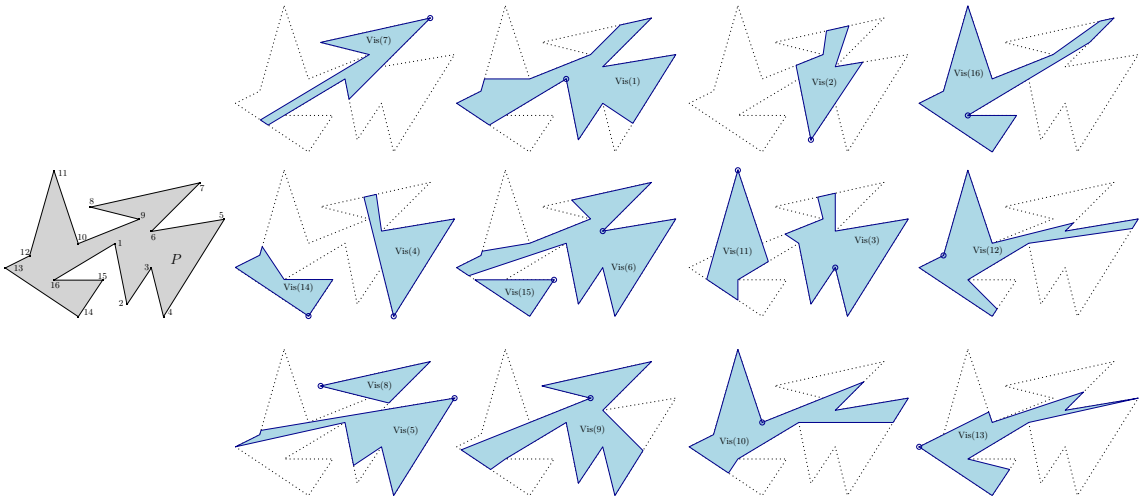


Figure 1. AGP **as a specific** SCP

In order to prove that the algorithm will always find a viable, and thus optimal, solution for the original problem, it suffices to determine the worst case for the number of iterations. Consider the set of all visibility regions of the vertices in V. The edges of these regions induce an arrangement of line segments within P, whose faces are called *atomic visibility polygons* (AVP). It has been shown that there are at most $O(n^3)$ AVPs.

It follows from the definition of AVPs that if any point in the interior of an atomic visibility polygon $\mathcal{V}$ is visible from a vertex guard, the entirety of $\mathcal{V}$ must also be. Since any uncovered region is formed by the union of neighboring AVPs, and since after each iteration one discretization point inside each of those regions is included into the formulation, an upper bound on the maximum number of iterations performed by the algorithm is $O(n^3)$ and this establishes its convergence, see Figure 2.

Moreover, the above algorithm is actually a rough description of a Turing's reduction from the Art Gallery problem to the Set Cover problem. The reader must also consider that if the initial discretization is comprised of the centroids of all atomic visibility polygons, the algorithm needs no more than the first iteration to obtain an optimal solution and this establishes a Karp's reduction.

Notice that each iteration of the algorithm solves an NP-HARD problem, the SCP, and its convergence is closely connected to the number of uncovers regions founded. Furthermore, as the uncovered regions depends on the choice of the initial discretization, there is a trade off between speed and precision that one must take into account when designing a good discretization strategy. It must ideally be light enough to set up instances of SCP that can rapidly be solved while minimizing the number of iterations required to attain an optimal solution. At the same time, it is important to start off with a discretization that represents the polygon well.

3. Evaluation of the Algorithm

In order to evaluate the algorithm and test its practical usability, several discretization strategies were built, each one with its own purpose. Thereafter, an instance building scheme was developed and implemented, with the ability to generate thousands of random orthogonal and general simple polygons, as well as of a few other classes of polygons.

Thus, an extensive experimental evaluation was conducted, and the results, summarized in Figure 3, show that the overall best performance was achieved by the *convex*

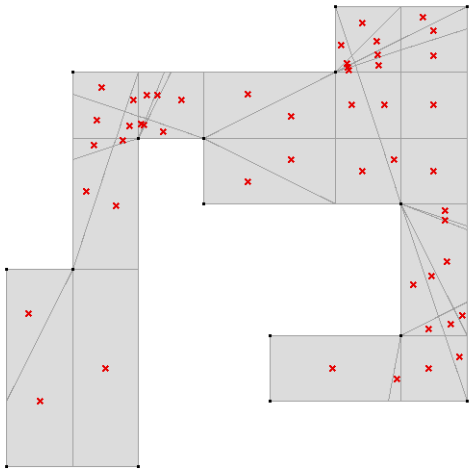


Figure 2. Visibility arrangement and a discretization with the centroids of all AVPs.

vertices strategy, where only the convex vertices of the polygon are included in the initial discretization set.

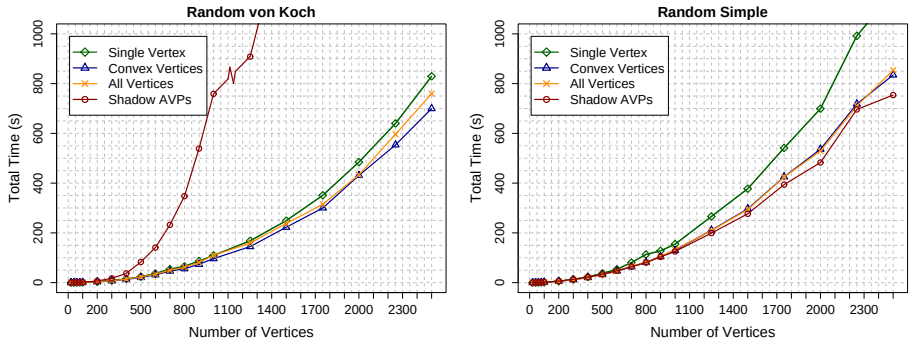


Figure 3. Total processing time for different discretization strategies.

The Convex Vertices strategy yields a sparse discretization and, as a consequence, small SCP instances, and yet dense enough to represent the polygon in a fairly broad sense. As it can be seen in Figure 3, this leads to a very fast implementation for instances of up to 2500 vertices.

4. Conclusion and Key Contributions

In this master thesis, we describe our extensive research of an Art Gallery problem, where the goal is to minimize the number of stationary vertex guards sufficient to cover the interior of an art gallery whose boundary is represented by a simple polygon.

We designed the first known exact algorithm able to solve to optimally (within excellent overall computational times) instances of up to ten times the size of the ones found in prior literature. Moreover, the algorithm proved to be robust, in the sense it was able to tackle instances from a broad range of polygon classes, including very degenerate ones.

A theoretical analysis on the subject is presented, including a full chapter with all basic definitions and concepts one needs to fully understand the problem, and a detailed

explanation of the reductions, both Turing's and Karp's, from the Art Gallery problem to the Set Cover problem, which is the basis for our algorithm, and finally the proof from its correctness and convergence.

In order to validate the algorithm, an implementation was fully coded, as well as a generator of multiple class instances. Besides generating instances from a few well known polygon classes, we developed a new benchmark class of random von Koch polygons – an extremely complex and hard to solve class of instances derived from a modified von Koch curve. In the last part of the thesis, we share insights about implementation issues, degenerate instances, exact arithmetic which allowed our implementation to be robust and exact.

With all that, we carried out a broad experimental evaluation of the algorithm implementation and analyzed the behavior and impact of several discretization strategies on the many classes of instances at our disposal. In the end, the best overall performance strategy was found to be the one built with only the convex vertices of the polygon. It was shown to provide a perfect balance between speed and accuracy, performing only a few iterations in a short computational time.

Besides, we generated and made available for future public reference the first and only known benchmark of instances for the Art Gallery problem, comprised of more than 10,000 instances with up to 2,500 vertices as well as all the detailed results presented in our work.

In conclusion we successfully built, implemented and tested an exact algorithm that is able to solve extremely large instances (of up to 2,500 vertices) in a number of iterations several orders smaller than the theoretical upper bound, within 800 seconds when the best overall discretization strategy we found was employed.

5. Final Remarks and Relevance

This master thesis presents a broad multidisciplinary study on two variants of an important geometrical NP-HARD problem, the Art Gallery problem. This problem has been subject of study by many of the best known researchers in Computational Geometry for the past 40 years, including J. O'Rourke, V. Chvátal, D. T. Lee, A. K. Lin, A. Aggarwal, J. Urrutia, S. K. Ghosh, among others. See [O'Rourke 1987, Lee and Lin 1986, Urrutia 2000, Ghosh 2010].

The fact that our published papers have already begun to be cited attests to the relevance of our work: we have had 3 citations according to Scopus and 8 according to Google Scholar (excluding self-citations). Also, our paper *"An IP solution to the art gallery problem"*, which appeared in the Proceedings of the 25th ACM Annual Symposium on Computational Geometry, has been downloaded from the ACM Digital Library 85 times in the last 12 months and six times in the last six weeks alone.

We attribute the visibility our work is having to the fact that it is the first exact and efficient algorithm to optimally solve the problem in question, as well as for the extensive experimental analysis of the practical viability of the algorithm, and lastly, for having the first known benchmark of instances to the problem, available for public reference and use at *www.ic.unicamp.br/∼cid/Problem-instances/Art-Gallery*. Hopefully this will allow other researchers to compare their results with our exact ones.

Moreover, it is undoubtedly a recognition of excellence the fact that four refereed international conferences and one international journal published the papers that came out of this Master's thesis work. One of said conferences happens to be one of the most pres-

tigious ones in experimental evaluation of algorithms — the Symposium on Experimental Algorithms, SEA, formerly know as Workshop on Experimental Algorithms, WEA.

Here is the full list of publication this work has generated.

1. M. C. Couto, P. J. de Rezende, and C. C. de Souza, *An exact algorithm for minimizing vertex guards on art galleries.* In **International Transactions in Operational Research**, available online since March 1st, 2011, no. doi: 10.1111/j.1475-3995.2011.00804.x., 27 pages.
2. M. C. Couto, P. J. de Rezende, and C. C. de Souza. *An IP solution to the art gallery problem.* In **SCG '09: Proceedings of the 25th annual symposium on Computational geometry**, pages 88–89, New York, NY, USA, 2009. ACM.
3. M. C. Couto, C. C. de Souza, and P. J. de Rezende. *Strategies for optimal placement of surveillance cameras in art galleries.* In **GraphiCon 2008: XI International Conference on Computer Graphics & Vision**, vol. 1, page *http://www.graphicon.ru/2008/proceedings/technical.html.* Lomonosov Moscow State University, 2008.
4. M. C. Couto, C. C. de Souza, and P. J. de Rezende. *Experimental evaluation of an exact algorithm for the orthogonal art gallery problem.* In **WEA**, **Lecture Notes in Computer Science**, volume 5038, pages 101–113. Springer, 2008.
5. M. C. Couto, C. C. de Souza, and P. J. de Rezende. *An exact and efficient algorithm for the orthogonal art gallery problem.* In **Proc. of the XX Brazilian Symp. on Comp. Graphics and Image Processing**, pages 87–94. IEEE Computer Society, 2007.

This master thesis opens plenty opportunities for future research which are detailed both in the publications, and in the thesis itself.

Acknowledgments. This research was partially supported by CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico – Grant #301732/2007-8, #472504/2007-0, #483177/2009-1, #473867/2010-9 and FAPESP – Fundação de Amparo à Pesquisa do Estado de São Paulo – Grant #07/52015-0 and a Grant from FAEPEX/UNICAMP.

References

Ghosh, S. K. (2010). Approximation algorithms for art gallery problems in polygons. *Discrete Applied Mathematics*, 158(6):718–722.

Lee, D. T. and Lin, A. K. (1986). Computational complexity of art gallery problems. *IEEE Trans. Inf. Theor.*, 32(2):276–282.

O'Rourke, J. (1987). *Art Gallery Theorems and Algorithms*. Oxford University Press.

Urrutia, J. (2000). Art gallery and illumination problems. In Sack, J.-R. and Urrutia, J., editors, *Handbook of Computational Geometry*, pages 973–1027. North-Holland.

O problema da subsequência comum máxima sem repetições

Carlos E. Ferreira[1], **Christian Tjandraatmadja**[1]

[1]Instituto de Matemática e Estatística, Universidade de São Paulo, Brasil

Abstract. *We study the following problem: given two sequences X and Y over a finite alphabet, find a longest common subsequence of X and Y without repeated symbols. This NP-hard problem appears in the context of genome rearrangement when gene duplication is considered. We describe an algorithm that solves the problem optimally based on the branch-and-cut technique. The algorithm takes advantage of good heuristics and of a simple but effective technique that eliminates variables during the algorithm. Finally, we describe an algorithm based on dynamic programming that takes linear time and space when the size of the alphabet is constant.*
Keywords: *Repetition-free longest common subsequence, longest common subsequence, integer programming, branch-and-cut, heuristics.*

Resumo. *Estudamos o seguinte problema: dadas duas sequências X e Y sobre um alfabeto finito, encontre uma subsequência comum máxima de X e Y sem símbolos repetidos. Este problema NP-difícil surge no contexto de rearranjo de genomas quando duplicação de genes é considerada. Descrevemos um algoritmo que resolve o problema de forma ótima baseado na técnica branch-and-cut. O algoritmo se aproveita de boas heurísticas e de uma técnica simples mas eficaz que elimina variáveis do problema no decorrer do algoritmo. Finalmente, descrevemos um algoritmo baseado em programação dinâmica que consome tempo e espaço linear quando o tamanho do alfabeto é constante.*
Palavras-chave: *Subsequência comum máxima sem repetições, subsequência comum máxima, programação inteira, branch-and-cut, heurísticas.*

1. Introdução

No contexto de análise de rearranjo de genomas, uma medida de similaridade útil entre dois genomas é o comprimento de uma subsequência comum máxima entre os genomas. Porém, esta medida não considera o evento de duplicação de genes por ser infrequente em genomas simples e tornar o problema computacionalmente mais difícil. Para levá-lo em consideração, Sankoff [Sankoff and Kruskal 1983] propôs o conceito de distância exemplar, que considera um representante para cada família de genes duplicados.

Neste trabalho, consideramos uma medida de similaridade proposta por Adi et al. [Adi et al. 2010] também baseada nesse conceito. A medida é o comprimento de uma subsequência comum máxima com a restrição de que existe no máximo um representante de cada família de genes duplicados na sequência.

Denotamos o comprimento de uma sequência S por $|S|$, o i-ésimo símbolo de S por s_i e a sequência $s_i s_{i+1} \dots s_j$ por $S[i..j]$. Uma *subsequência* de S é uma sequência obtida removendo zero ou mais símbolos de S (não necessariamente contígua em relação a S). Uma *subsequência comum* de S e T é uma subsequência de tanto S como T. Ela é

sem repetições se cada símbolo aparece no máximo uma vez. Uma *subsequência comum máxima*, ou LCS (do inglês *longest common subsequence*), é uma subsequência comum de comprimento máximo. Uma *subsequência comum máxima sem repetições*, ou RFLCS (do inglês *repetition-free longest common subsequence*) é uma subsequência comum sem repetições de comprimento máximo. Por exemplo, *cad* é um RFLCS de *caadbd* e *abacdad*.

Estamos interessados em resolver o problema do RFLCS, que consiste em encontrar uma subsequência comum máxima sem repetições de duas sequências finitas X e Y sobre um alfabeto finito Σ. Foi provado em [Adi et al. 2010] que o problema do RFLCS é APX-difícil mesmo quando cada símbolo ocorre no máximo duas vezes em cada sequência X e Y. Isto também implica que o problema é NP-difícil.

Na seção seguinte, apresentamos uma formulação de programação inteira para o problema. A seguir, discutimos uma abordagem *branch-and-cut* e técnicas usadas. Finalmente, descrevemos um algoritmo de programação dinâmica para casos de alfabeto pequeno. Estes tópicos são abordados de forma bastante superficial. O leitor interessado em detalhes pode consultar a dissertação resumida por este texto [Tjandraatmadja 2010].

2. Formulação

Um par ordenado (i, j) é um *casamento* sobre X e Y (de símbolo σ) se $x_i = y_j$ $(= \sigma)$. Seja $\mathcal{C}$ o conjunto de todos os casamentos sobre X e Y. Dizemos que dois casamentos (i, j) e (k, l) se *cruzam* se $i \leq k$ e $j \geq l$, ou $i \geq k$ e $j \leq l$. Como representado visualmente na Figura 1, eles se cruzam se os segmentos que o representam se intersectam graficamente, incluindo o caso em que eles têm um extremo em comum. Dizemos que dois casamentos (i, j) e (k, l) são *conflitantes* se eles se cruzam ou são do mesmo símbolo.

Seja C um conjunto de casamentos não-conflitantes dois a dois. É possível ordenar os casamentos de C em ordem crescente e tomar seus símbolos nessa ordem para obter S, uma subsequência comum sem repetições de X e Y. Dizemos que C *define* S. Por exemplo, se $X =$ *caadbd* e $Y =$ *abacdad*, o conjunto de casamentos $\{(1, 4), (2, 6), (6, 7)\}$ define *cad*. Ademais, qualquer subsequência comum sem repetições é definida por algum conjunto desse tipo. Portanto, o problema do RFLCS pode ser visto como encontrar um conjunto de casamentos não-conflitantes dois a dois de cardinalidade máxima.

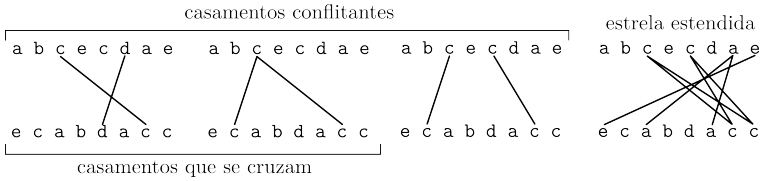


Figura 1. Casamentos que se cruzam, conflitantes e estrela estendida

Uma *estrela estendida* S é um conjunto de casamentos conflitantes dois a dois. Ela é *maximal* se não existe casamento em $\mathcal{C} \setminus S$ conflitante com todo casamento em S.

Construiremos uma formulação de programação linear inteira. Uma *formulação de programação linear* descreve um problema de otimização através de uma função linear que se deseja maximizar ou minimizar (*função objetivo*) e um conjunto de inequações lineares sobre um conjunto de variáveis. Se restringirmos variáveis a serem inteiras, ela é

uma *formulação de programação inteira*. Uma *solução viável* é um conjunto de valores para as variáveis que satisfaz as restrições, e ela é *ótima* se ela otimiza a função objetivo.

A formulação a seguir modela o problema do RFLCS, pois cada uma de suas soluções viáveis corresponde a um conjunto de casamentos não-conflitantes dois a dois de $\mathcal{C}$ e vice-versa. Seja z_{ij} uma variável binária para um casamento (i, j) em $\mathcal{C}$.

$$\begin{aligned} \max \quad & \sum_{(i,j)\in\mathcal{C}} z_{ij} & & \\ \text{s. a} \quad & \sum_{(i,j)\in K} z_{ij} \;\leq\; 1 & & \text{para cada estrela estendida maximal } K \subseteq \mathcal{C}, \\ & z_{ij} \;\in\; \{0,1\} & & \text{para cada } (i,j) \in \mathcal{C}. \end{aligned}$$

Provamos em [Fernandes et al. 2008] que as inequações acima definem facetas, ou seja, são necessárias na descrição do poliedro dado pela formulação.

Neste trabalho, usamos a técnica *branch-and-cut* para resolver o problema. Existem algoritmos eficientes para resolver um problema de programação linear, mas a formulação acima apresenta dois obstáculos: restrições inteiras e um número exponencial de restrições lineares. Para lidar com o primeiro obstáculo, é construída uma árvore de subproblemas menos restritos, com apenas restrições lineares, em que cada ramificação fixa uma variável em 0 e em 1 (para o caso binário), com a ajuda de uma estratégia de poda. Para lidar com o segundo, as restrições são adicionadas por demanda: a cada solução ótima encontrada para um problema relaxado, procuramos uma restrição violada por tal solução, adicionamos-na, e reotimizamos o problema. Encontrar tal restrição é o *problema de separação*, discutido a seguir. Esta técnica funciona bem na prática.

3. Algoritmo de separação

O problema de separação para inequações de estrela estendida é o seguinte. Dadas duas sequências X e Y e uma solução viável z de um problema relaxado, encontre uma estrela estendida maximal K tal que $\sum_{(i,j)\in K} z_{ij} > 1$, ou informe se não houver tal K. Portanto, é suficiente encontrar uma estrela estendida maximal de peso máximo usando o valor z_{ij} como o peso de (i, j) para cada $(i, j) \in \mathcal{C}$ (denotada por *estrela estendida máxima*).

Realizaremos uma pequena transformação por conveniência. Se invertermos uma das sequências, digamos X, toda estrela estendida se torna um conjunto de casamentos no qual quaisquer dois casamentos de símbolos diferentes não são conflitantes. Denotamos este conjunto por *emparelhamento planar estendido* e desejamos encontrar tal conjunto.

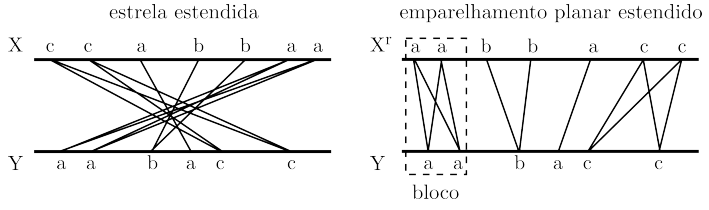


Figura 2. Estrela estendida e emparelhamento planar estendido correspondente.

Sejam (i', j') e (i, j) dois casamentos de um mesmo símbolo σ. Denote por *bloco* de ínicio (i', j') e fim (i, j) o conjunto de casamentos (k, l) de símbolo σ tal que $i' \leq k \leq i$

e $j' \leq l \leq j$ (veja Figura 2). Dizemos que dois blocos se cruzam se existem dois casamentos, um em cada bloco, que se cruzam. Podemos interpretar um emparelhamento planar estendido como um conjunto de blocos que não se cruzam dois a dois.

Seja $L_e(i,j)$ o peso de um emparelhamento planar estendido máximo de $X[1..i]$ e $Y[1..j]$. A recorrência a seguir é inspirada em uma recorrência para o problema do LCS.

$$L_e(i,j) = \begin{cases} 0 & \text{se } i = 0 \text{ ou } j = 0, \\ \max\{L_e(i,j-1), L_e(i-1,j), B_{max}(i,j)\} & \text{se } x_i^r = y_j, \\ \max\{L_e(i,j-1), L_e(i-1,j)\} & \text{se } x_i^r \neq y_j; \end{cases}$$

onde $B_{max}(i,j)$ é o peso de um emparelhamento planar estendido máximo de $X[1..i]$ e $Y[1..j]$ que contém o casamento (i,j).

A solução $L_e(|X|,|Y|)$ e o emparelhamento planar estendido associado podem ser computados em tempo $O(|X|^2|Y|^2)$ usando programação dinâmica, isto é, preenchemos uma tabela $|X| \times |Y|$ linha por linha de acordo com a recorrência acima. Uma modificação não-trivial para obtermos uma melhoria de tempo pode ser feita adaptando a ideia do algoritmo para o problema do LCS de Hunt e Szymanski [Hunt and Szymanski 1977].

4. Eliminação de variáveis

Dado um casamento (i,j), denote por $u_{pref}(i,j)$ e $u_{suf}(i,j)$ um limitante superior para o comprimento de um RFLCS de $X[1..i-1]$ e $Y[1..j-1]$, e de $X[i+1..|X|]$ e $Y[j+1..|Y|]$ respectivamente. Dizemos que a função $u(i,j) = u_{pref}(i,j) + u_{suf}(i,j) + 1$ é um *limitante de casamento* para (i,j).

Pode-se provar que, dado um casamento (i,j), se existe uma subsequência comum sem repetições de X e Y de comprimento maior que $u(i,j)$, então (i,j) não pertence a um conjunto que define um RFLCS de X e Y. Este fato pode ser usado para fixar variáveis em zero conforme encontramos soluções viáveis. Como limitante superior, usamos o menor entre o comprimento de um LCS de X e Y e o número de símbolos distintos em X e Y.

5. Heurísticas

Heurísticas são usadas para obter limitantes primais para o algoritmo *branch-and-cut*, o que ajuda a obter uma solução ótima mais cedo e a eliminar variáveis através da técnica anterior. Descrevemos brevemente as heurísticas que mais contribuem ao algoritmo.

As heurísticas seguintes provêm ao algoritmo *branch-and-cut* um limitante inicial, mas também podem ser executadas independentemente dele.

LCS e remove repetições. Compute uma subsequência comum máxima das sequências e remova ocorrências repetidas de símbolos. Ele é eficiente e tem bons resultados.

Limitante de casamento. Esta heurística procura escolher recursivamente casamentos de alto limitante de casamento, que têm o potencial de pertencer a longas subsequências comuns sem repetições. Ela geralmente devolve a melhor solução entre as implementadas, embora tenha maior tempo de execução.

As heurísticas seguintes usam como parâmetro uma solução viável da relaxação e geralmente devolvem bons resultados quando ela está perto da ótima. Assim, elas podem ser executadas várias vezes durante o processo. Seja z uma solução viável da relaxação.

LCS com peso e remove repetições. Compute um conjunto de casamentos que não se cruzam dois a dois de peso máximo usando z como peso e, da sequência definida por este conjunto, remova ocorrências repetidas de símbolos.

Escolha simples. Escolha de forma gulosa casamentos (i, j) com z_{ij} o mais perto de 1, ignorando aqueles que conflitam com algum casamento já escolhido.

6. Resultados computacionais

O gráfico a seguir (Figura 3) compara o algoritmo *branch-and-cut* com e sem as técnicas desenvolvidas[1]. As sequências X e Y são sequências uniformemente aleatórias de comprimento 1024 e o tamanho do alfabeto é variado. A resolução de programas lineares e gerenciamento da árvore *branch-and-cut* é realizada pelo pacote GLPK.

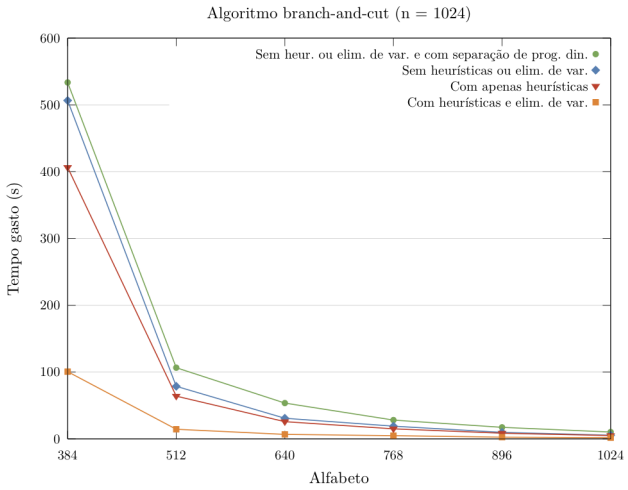


Figura 3. Gráfico ilustrando tempos de execução do algoritmo *branch-and-cut*.

O formato da curva é devido ao fato de que é mais difícil resolver instâncias com alfabeto pequeno usando *branch-and-cut*, pois há um número alto de repetições e variáveis. Os resultados mostram que usar a técnica de eliminação de variáveis e heurísticas primais em conjunto melhoram significativamente o tempo de execução do algoritmo. Para os resultados numéricos em si e outros testes, pode-se consultar a dissertação [Tjandraatmadja 2010].

7. Algoritmo de programação dinâmica

Considere a seguinte recorrência. Seja $R(i, j, S) = 1$ se existe uma subsequência comum sem repetições cujos símbolos são os de S, ou 0 caso contrário.

$$R(i,j,S) = \begin{cases} 0 & \text{se } i = 0 \text{ ou } j = 0 \text{ (e } S \neq \varnothing), \\ 1 & \text{se } S = \varnothing, \\ \max\{R(i,j-1,S), R(i-1,j,S)\} & \text{se } x_i \neq y_j \text{ ou } x_i = y_j \notin S, \\ R(i-1,j-1,S \setminus \{x_i\}) & \text{se } x_i = y_j \in S. \end{cases}$$

[1] A separação de programação dinâmica a que a primeira legenda se refere é a separação sem a melhoria.

Seja $R'(i, S)$ o menor j tal que $R(i, j, S) = 1$. Baseado em $R'(i, S)$, podemos desenvolver um algoritmo de programação dinâmica usando um vetor de tamanho $|X|$ para cada conjunto S. O algoritmo consome tempo $O(|\Sigma|n+m2^{|\Sigma|}|\Sigma|)$ e espaço $O(|\Sigma|n+(m+|\Sigma|)2^{|\Sigma|})$, supondo consumo de tempo e espaço $O(|S|)$ para acessar e armazenar os conjuntos S. Logo, se $|\Sigma|$ for constante, temos um algoritmo de tempo e espaço $O(n+m)$.

8. Conclusão

Neste projeto, exploramos o problema do RFLCS particularmente do ponto de vista de programação inteira, resultando em um algoritmo *branch-and-cut* e técnicas e heurísticas para melhorá-lo significativamente. Além disso, desenvolvemos um algoritmo de programação dinâmica eficiente na prática para alfabetos de tamanho baixo (menos de 20).

Notas sobre o trabalho

A parte central deste trabalho foi publicada e apresentada no ISCO 2010 (International Symposium on Combinatorial Optimization) [Ferreira and Tjandraatmadja 2010][2]. Uma versão do código-fonte do algoritmo *branch-and-cut* que usa CPLEX foi publicada em `http://www.ime.usp.br/~cef/rflcs/`.

Referências

Adi, S., Braga, M., Fernandes, C., Ferreira, C., Martinez, F., Sagot, M.-F., Stefanes, M., Tjandraatmadja, C., and Wakabayashi, Y. (2010). Repetition-free longest common subsequence. *Discrete Applied Mathematics*, 158(12):1315–1324. Traces from LAGOS'07 IV Latin American Algorithms, Graphs, and Optimization Symposium Puerto Varas - 2007.

Fernandes, C., Ferreira, C., Tjandraatmadja, C., and Wakabayashi, Y. (2008). A polyhedral investigation of the LCS problem and a repetition-free variant. In *Proc. of the 8th Latin American Symposium on Theoretical Informatics (LATIN'08)*, volume 4957 of *Lecture Notes in Computer Science*, pages 329–338. Springer.

Ferreira, C. and Tjandraatmadja, C. (2010). A branch-and-cut approach to the repetition-free longest common subsequence problem. *Electronic Notes in Discrete Mathematics*, 36:527–534. ISCO 2010 - International Symposium on Combinatorial Optimization.

Hunt, J. and Szymanski, T. (1977). A fast algorithm for computing longest common subsequences. *Communications of the ACM*, 20(5):350–353.

Sankoff, D. and Kruskal, J. (1983). *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley.

Tjandraatmadja, C. (2010). O problema da subsequência comum máxima. Dissertação de mestrado.

[2]As duas outras referências citadas neste texto sobre o problema do RFLCS, para as quais o autor também contribuiu, foram desenvolvidas antes do projeto de mestrado. Elas têm foco em dificuldade e aproximabilidade do problema [Adi et al. 2010] e em características poliédricas de formulações para os problemas do LCS e RFLCS [Fernandes et al. 2008].

Alocação de Dados e Código em Memórias Embarcadas: Uma Abordagem Pós-Compilação

Alexandre Keunecke Ignácio de Mendonça[1]
José Luís Almada Güntzel [1], **Luiz Cláudio Villar dos Santos** [1]

[1]Programa de Pós-Graduação em Ciência da Computação
Universidade Federal de Santa Catarina – Florianópolis – SC – Brasil
{mendonca,guntzel,santos}@inf.ufsc.br

Resumo. *Este trabalho reporta as principais contribuições e resultados de uma dissertação de mestrado que aborda um problema contemporâneo de Computação Embarcada: o gerenciamento explícito de memória. Ao ser modelado como um problema clássico de Computação (Problema Binário da Mochila), ele pode ser eficientemente resolvido de forma exata (com um algoritmo de programação dinâmica) para minimizar o consumo energético do subsistema de memória, sem a necessidade de recompilação da aplicação, através do uso eficiente de uma combinação de memória cache com memória não-associativa do tipo scractchpad.*

Palavras-Chave: Computação Embarcada, Eficiência Energética, Problema da Mochila, Ligadores.

1. Introdução

Devido à sua maior latência e significativo consumo de energia, memórias *off-chip* limitam a eficiência energética em sistemas embarcados. Por este motivo, memórias cache embarcadas tem sido utilizadas. Porém, para algumas aplicações o mero uso de memórias cache não garante suficiente eficiência energética, pois os processadores embarcados contemporâneos podem gastar até 70% de seu consumo de energia suprindo dados e instruções via caches [Dally et al. 2008] (enquanto o consumo das operações aritméticas corresponde a 7%). Parte da energia gasta em caches é consumida na lógica que implementa sua estrutura associativa (e.g.: armazenamento e comparação de *tags*). Tendo essencialmente a mesma latência, porém consumindo menos energia que uma cache de mesma capacidade, uma memória do tipo *scratchpad* (SPM) pode substituir ou auxiliar a(s) cache(s). Ao contrário das caches (onde a alocação é implicitamente escolhida pelo hardware), elementos alocados em SPM (dados e instruções) são gerenciados pelo software embarcado: deve-se mapear explicitamente elementos de programa para o espaço de endereçamento da SPM, que é disjunto do da memória principal. Este mapeamento pode permanecer inalterado durante a execução da aplicação (alocação *non-overlaying*) ou pode ser alterado dinamicamente para atender às mudanças nos padrões de acesso da aplicação (alocação *overlaying*).

2. Principais contribuições deste trabalho

A maioria dos trabalhos reportados na literatura faz a alocação em SPM a partir do código-fonte ou a partir de arquivo executável. Esta dissertação propõe uma nova técnica que viabiliza a alocação de código e dados em SPM a partir de **arquivos-objeto relocáveis**. Esta **abordagem pós-compilação** permite que, além de elementos de programas aplicativos, itens encapsulados em bibliotecas sejam passíveis de alocação em SPM, ao contrário das técnicas que fazem a alocação em tempo de compilação. Mostra-se que a adoção de arquivos-objeto relocáveis como ponto de partida melhora a eficiência da relocação em SPM, quando comparada com técnicas que manipulam diretamente arquivos executáveis. Mostra-se também que a mera inclusão de dados estáticos como candidatos para a alocação em SPM resulta em economia extra de 21%, em média, para os programas do *benchmark* Mibench [Guthaus et al. 2001].

As principais contribuições dessa dissertação foram publicadas nos anais do *IEEE Computer Society Annual Symposium on VLSI* (ISVLSI 2009) [Mendonca et al. 2009]. As pesquisas nela iniciadas foram continuadas pelo autor em trabalho colaborativo posterior, dando origem a outro trabalho já publicado no ISVLSI 2010 [Volpato et al. 2010] e outro aceito para publicação no SBCCI 2011 [Volpato et al. 2011]. Além disso, um artigo foi submetido a um periódico da IEEE [Santos et al.]

3. Trabalhos Relacionados

a) Persistência do conteúdo em SPM: Algumas técnicas adotam a abordagem *non-overlaying* [Panda et al. 2000] e [Angiolini et al. 2004]. Com esta abordagem não há *overhead*

relacionado ao gerenciamento do conteúdo da SPM. Porém, quando a razão entre o tamanho do código e dados e o tamanho da SPM aumenta, este tipo de técnica pode não ser muito eficaz. Outras técnicas adotam a abordagem *overlaying* [Kandemir et al. 2001], [Steinke et al. 2002b], [Steinke et al. 2002a], [Dominguez et al. 2005], [Udayakumaran et al. 2006], [Cho et al. 2007] e [McIlroy et al. 2008], que tenta manter em SPM os dados e instruções que são mais frequentemente acessados durante a execução, efetuando cópias da memória principal para a SPM e vice-versa. A decisão sobre quais dados serão copiados de ou para a SPM ocorre em tempo de compilação através de *profiling*.

b) Tratamento de estruturas de dados: Algumas técnicas mapeiam apenas código para a SPM ([Angiolini et al. 2004] e [Steinke et al. 2002b]), enquanto outras mapeiam somente dados ([Panda et al. 2000], [Kandemir et al. 2001], [Cho et al. 2007] e [Udayakumaran et al. 2006]). Finalmente, algumas mapeiam ambos, código e dados ([Steinke et al. 2002a] e [Verma et al. 2004]).

c) Dependência do código-fonte e tratamento de bibliotecas: Grande parte dos trabalhos correlatos consiste de técnicas que operam em tempo de compilação ([Kandemir et al. 2001], [Steinke et al. 2002b], [Steinke et al. 2002a], [Dominguez et al. 2005], [Udayakumaran et al. 2006], [McIlroy et al. 2008], [Panda et al. 2000]). Nessas técnicas, o impacto reportado assume implicitamente o acesso a todo o código-fonte da aplicação. Porém, nos casos mais realistas em que bibliotecas de terceiros, código legado ou código protegido por propriedade intelectual devem ser utilizados no projeto de sistemas embarcados, a melhoria esperada tende a ser menor do que a reportada, pois elementos relevantes são excluídos do espaço de otimização se estiverem encapsulados em arquivos binários cujo código-fonte não está disponível. Por outro lado, poucas técnicas otimizam a partir de arquivos binários ([Angiolini et al. 2004] e [Cho et al. 2007]). Uma alternativa é identificar elementos de programa canditados à otimização diretamente do arquivo executável, como feito em [Angiolini et al. 2004]. Porém, o gerenciamento das relocações em SPM a partir de arquivos executáveis é ineficiente, pois uma dada relocação para a SPM tende a gerar muitos ajustes no código e nos elementos de dados.

d) A proposta deste trabalho frente aos trabalhos correlatos: É difícil (se não for impossível) conceber um método unificado que possa harmonizar todos os aspectos citados anteriormente. Uma divisão de tarefas entre abortagens complementares é provalvelmente mais pragmática. Por um lado, a maioria das técnicas do tipo *overlaying* apresentam um tratamento mais natural de dados dinâmicos (pilha e/ou *heap*). Porém, o tratamento de bibliotecas é geralmente limitado. Por outro lado, o tratamento de bibliotecas requer a manipulação de arquivos binários, os quais internamente particionam o código em elementos de dados estáticos e procedimentos. Esta dissertação propõe um método de alocação do tipo *non-overlaying* de código e dados em SPM para uma dada aplicação-alvo. O método proposto não requer a disponibilidade do código-fonte nem de hardware dedicado, o que amplia sua aplicação a bibliotecas pré-compiladas.

4. O problema-alvo e a solução proposta

Considerando-se um sistema embarcado que possua SPM e caches, quando uma aplicação embarcada é compilada, seus dados podem ser armazenados tanto na SPM quanto em memória principal externa. Neste último caso, o acesso aos dados e instruções é feito via cache. O objetivo é encontrar quais elementos de programa devem ser alocados para a SPM de maneira a maximizar a economia de energia. Antes de formular o problema-alvo, é preciso formalizar algumas noções importantes.

A **energia consumida** para acessar uma posição na memória M, denotada por E_M, é definida como a energia média consumida ao ler ou escrever naquela posição. Uma memória M pode ser a memória principal (MP), uma memória SPM, uma cache de instruções ou de dados (I-cache ou D-cache), ou uma cache unificada (U-cache). A **capacidade** de uma memória M, denotada por C_M, é seu tamanho expresso em bytes. Um ***trace*** de memória `T` é uma tupla (α_1, α_2, ..., α_i, ... α_n) que representa a sequência de sucessivos endereços a serem acessados no subsistema de memória, onde α_i denota o i-ésimo endereço.

Sejam $D_1, ..., D_i, ..., D_n$ os elementos de programa (variáveis, instruções ou estruturas) acessados por um código embarcado. Sua **caracterização** é denotada por uma matriz-linha $W = [w_1, ..., w_i, ..., w_n]$, onde w_i é o espaço ocupado em SPM, expresso em bytes, para alocar o elemento D_i. Uma **alocação** para uma SPM é representada por uma matriz-coluna $X = [x_1, ..., x_i, ..., x_n]^{-1}$, onde $x_i = 1$ denota a alocação do elemento D_i na SPM e $x_i = 0$ denota sua não-alocação.

O problema-alvo pode ser formulado para recair no clássico Problema Binário da Mochila:

Problema-alvo: Dados um conjunto de elementos D_i caracterizados pelo vetor W e um

padrão de acesso caracterizado por um *trace* T, encontrar a alocação X que maximize $P * X$ e seja tal que $W * X \leq C_{SPM}$, onde $P = [p_1, ..., p_i, ..., p_n]$, é uma matriz-linha onde p_i representa o lucro da alocação x_i (como será definido na Seção 5).

Como o problema-alvo é NP-Completo [Garey and Johnson 1979], sua resolução usando técnicas exatas pode resultar em tempos de execução proibitivos para casos de uso reais. Embora muitos métodos utilizem Programação Linear Inteira (ILP: *integer linear programming*), ele pode também ser resolvido de forma exata via Programação Dinâmica em tempo pseudo-polinomial.

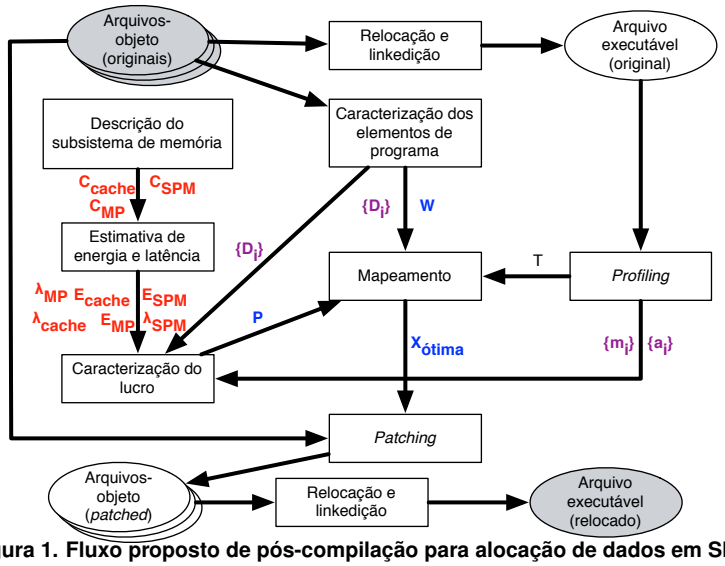


Figura 1. Fluxo proposto de pós-compilação para alocação de dados em SPM.

O fluxo de alocação ótima em SPM é ilustrado na Figura 1, o qual assume que, como resultado de pré-compilação, o código-objeto relocável para o software embarcado esteja disponível. A partir do código-objeto, são extraídos os tamanhos dos elementos de programa D_i, os quais são anotados na matriz-linha W. Também a partir do código-objeto, gera-se um arquivo executável através de relocação e linkedição. A partir do código executável gerado, obtém-se o *trace* T correspondente ao padrão de acessos induzidos pela execução do programa (aplicação), a taxa de faltas (m_i) induzida na memória cache para cada elemento de programa D_i e o número de acessos a_i de cada elemento de programa. A partir da descrição do subsistema de memória, extraem-se a energia média gasta no acesso a cada um deles (E_{MP}, E_{CACHE}, E_{SPM}) e a capacidade da SPM (C_{SPM}). Os parâmetros dependentes de tecnologia foram obtidos através do modelo físico de memórias CACTI [Thoziyoor et al. 2008]. As propriedades extraídas permitem a criação da matriz-linha P associada à função lucro (definida na Seção 5).

Com base nas propriedades do programa (aplicação) e do subsistema de memória onde será executado, a etapa de mapeamento consiste na resolução do Problema-alvo, instrumentado com a função lucro pré-caracterizada. Uma vez obtida a alocação ótima ($X_{ótima}$), os elementos de programas dos arquivos-objeto são relocados para obedecê-la. Finalmente, as referências aos elementos alocados em SPM sofrem linkedição, dando origem ao código executável relocado para a SPM.

5. Aspectos mais relevantes da implementação

a) Formulação da função-lucro: Como o método aqui proposto não gera expansão de código, o cômputo do lucro em se alocar um elemento D_i na SPM é a energia economizada por acesso vezes o número de acessos a D_i, ou seja:

$$p_i = a_i \times (E_i - E_{SPM}), \quad onde: E_i = E_{cache} + m_i \times E_{MM} \tag{1}$$

b) Relocação e linkedição: Para suportar a otimização de bibliotecas mantendo seus elementos no espaço de otimização, sem se deixar limitar por um *patching* computacionalmente ineficiente esta dissertação propõe o uso de arquivos-objeto relocáveis, ao invés de se utilizar somente o arquivo executável. O que diferencia um arquivo-objeto relocável é basicamente a seção que contém as informações de relocação. Nessa seção, cada instrução que necessita de relocação é identificada pelo seu endereço no segmento de código e sua depêndencia a uma referência simbólica (um elemento de programa). A operação de *patching* em arquivos-objeto é muito mais fácil e mais eficiente que em

arquivos executáveis, pois em arquivos-objeto as referências a serem relocadas (sendo simbólicas) não estão ainda codificadas como endereços.

6. Validação experimental e resultados

Para se fazer a experimentação do protótipo foram utilizados treze programas extraídos do *benchmark* MiBench [Guthaus et al. 2001]. Neste resumo apenas duas configurações serão consideradas para o subsistema de memória, ambas com $C_{I-cache} = C_{D-cache} = 1KB$, mas com SPMs de diferentes capacidades: 1KB e 4KB.

6.1. Impacto da inclusão de dados e de código no espaço de otimização

A Figura 2 mostra a contribuição de cada tipo de elemento de programa (código, dados estáticos e dados dinâmicos) ao consumo de energia de uma configuração-base com caches de capacidade $C_{I-cache} = C_{D-cache} = 1KB$, mas sem SPM. Como estes valores foram obtidos antes da introdução de SPM, eles nos permitem quantificar o impacto potencial em se alocar código e dados estáticos para SPM. Os valores foram medidos para todos os elementos de programa acessados, independentemente de estarem encapsulados em bibliotecas ou não.

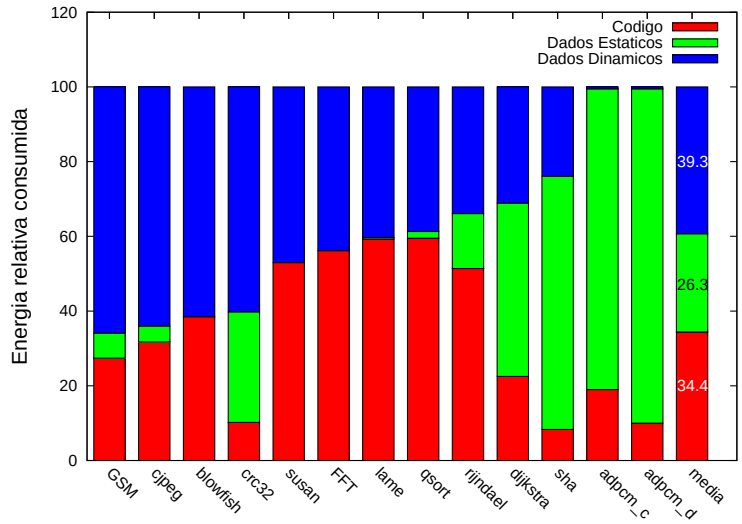


Figura 2. Contribuição ao consumo de energia

Em média, 61% da energia é consumida quando se acessam dados estáticos ou código. Essas porcentagens dão clara evidência de que, mesmo que uma técnica não aborde a alocação em SPM de dados dinâmicos, seu uso prático se justifica em face do significativo potencial de otimização. Assim, podemos definir a **margem disponível para otimização** do método proposto como a soma das contribuições ao consumo de energia quando se acessam código e dados estáticos. Espera-se que a técnica aqui proposta tenha maior impacto para programas com maior margem de otimização, tais como: `adpcm_code` (99%), `adpcm_decode` (99%) e `sha` (76%). Para programas com menor margem de otimização, tais como `blowfish` (38%) e `GSM` (34%) espera-se que a técnica aqui proposta tenha um desempenho inferior.

Vamos comparar o impacto da técnica para configurações com a mesma quantidade de cache, mas com diferentes tamanhos de SPM. Isso permite avaliar o impacto da inclusão de SPM no consumo de energia da hierarquia de memória.

A Tabela 1 resume o melhor e o pior resultados obtidos, bem como a média obtida no conjunto de programas para as duas configurações de SPM, em ambos os cenários, comparando-os com as respectivas margens de otimização. Note que, para `blowfish`, apenas 23% da margem de otimização disponível foi aproveitada para $C_{SPM} = 1KB$ e quase totalmente aproveitada para $C_{SPM} = 4KB$ em ambos os cenários. O fato de se ter obtido a mesma economia em ambos os cenários deve-se ao fato de que a energia relativa consumida em dados estáticos é praticamente nula (ver Figura 2). Por outro lado, observe que, para `adpcm_decode`, toda a margem de otimização foi aproveitada no Cenário 1 em ambas as configurações. Entretanto, no Cenário 2, o aproveitamento completo da margem de otimização só ocorreu para $C_{SPM} = 4KB$. A diferença de impacto observada no Cenário 2 explica-se pelo fato de aquele programa utilizar uma estrutura de dados frequentemente

Tabela 1. Impacto da minimização de energia

Caso	Programa	Cenário 1: somente código			Cenário 2: código+dados		
		Margem	C_{SPM}		Margem	C_{SPM}	
			1KB	4KB		1KB	4KB
Pior	blowfish	39%	9%	38%	39%	9%	38%
Melhor	adpcm_decode	10%	10%	10%	99%	13%	99%
Média	–	34%	17%	31%	61%	29%	52%

acessada para a decodificação de áudio, a qual é maior que 1KB e menor que 4KB. Por isso, para $C_{SPM} = 1KB$ apenas 13% da margem de otimização foi aproveitada.

Em média, cerca de metade da margem de otimização disponível foi aproveitada para $C_{SPM} = 1KB$ e quase totalmente aproveitada para $C_{SPM} = 4KB$ (85%), não havendo diferença de impacto sensível entre os cenários. Assim, pode-se concluir que, para o conjunto de programas utilizado, embora o impacto de se incluir dados no espaço de otimização seja pequeno em média, podem haver casos importantes em que esse impacto é substancial (como em `adpcm_decode` e `adpcm_code`), pois a margem de otimização pode aumentar consideravelmente diante de estruturas de dados estaticamente alocadas e frequentemente acessadas. Assim, apesar de seu relativamente baixo impacto potencial frente ao código (contribuições médias de 26%), dados estáticos não devem ser descartados do espaço de otimização.

6.2. Impacto da inclusão de bibliotecas no espaço de otimização

Para avaliar o impacto de bibliotecas, dois cenários são considerados na Figura 3: exclusão (Cenário 1) e inclusão (Cenário 2) de elementos de bibliotecas no espaço de otimização.

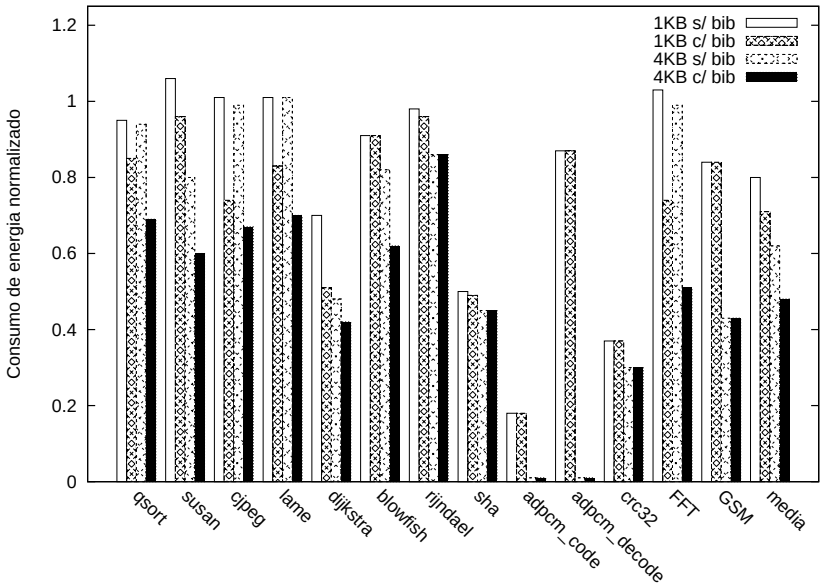


Figura 3. Impacto da alocação de bibliotecas na economia de energia

O menor impacto foi observado para `GSM`. A razão de o impacto ter sido nulo em ambas as configurações deve-se ao fato de que a energia relativa consumida por elementos de programa encapsulados em bibliotecas é muito baixa (conforme verificado através de *profiling*), embora o tamanho relativo de código encapsulado em bibliotecas seja de 90% e o tamanho relativo de dados encapsulados em bibliotecas seja de 61%. Os maiores impactos foram observados para `cjpeg` e `FFT`. Para `cjpeg`, economias de energia de 27% e 32% puderam ser obtidos para $C_{SPM} = 1KB$ e $C_{SPM} = 4KB$, respectivamente. Isto correlata com a maior contribuição de energia proveniente de bibliotecas, especialmente de bibliotecas de terceiros. Para `FFT`, economias de energia de 28% e 48% puderam ser obtidos, respectivamente.

Em média, economias extra de 11% ($C_{SPM} = 1KB$) e 23% ($C_{SPM} = 4KB$) puderam ser obtidas ao se incluir bibliotecas.

7. Conclusões

a) Impacto das limitações impostas à alocação: Os resultados mostram que o impacto aparentemente marginal de se limitar o espaço de otimização somente a dados [Cho et al. 2007] ou

somente a código [Angiolini et al. 2004] é resultado de experimentação com um conjunto muito limitado de casos de uso reais. A mera inclusão de dados estáticos pode resultar em economia extra de 21% no consumo médio de energia. Foram encontradas instâncias de casos de uso reais com economias extra de 67% e 91% ao incluir dados estáticos.

b) Impacto da restrição adotada na técnica proposta: A restrição da técnica proposta em não tratar dados dinâmicos resulta em se limitar a margem disponível para otimização em 61% da margem total (em média). Entretanto, a técnica mostrou-se eficaz em explorar a margem disponível, cuja metade foi aproveitada, em média, e cuja quarta parte foi aproveitada no pior caso.

c) Eficiência da técnica proposta: Ficou comprovada a maior eficiência da técnica devido ao uso de arquivos-objeto relocáveis, pois os tempos de execução medidos são pelo menos uma ordem de magnitude inferiores aos observados em outras abordagens que operam sobre arquivos binários [Cho et al. 2007] [Angiolini et al. 2004].

Referências

Angiolini, F. et al. (2004). A Post-compiler Approach to Scratchpad Mapping of Code. *CASES (2004)*, pages 259–267.

Cho, H. et al. (2007). Dynamic Data Scratchpad Memory Management for a Memory Subsystem With an MMU. *LCTES (2007)*, pages 195–206.

Dally, W. et al. (2008). Efficient Embedded Computing. *IEEE Computer*, 41(7):27–32.

Dominguez, A., Udayakumaran, S., and Barua, R. (2005). Heap Data Allocation to Scratch-pad Memory in Embedded Systems. *Journal Embedded Computing*, 1(4):521–540.

Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA.

Guthaus, M. R. et al. (2001). MiBench: A Free, Commercially Representative Embedded Benchmark Suite. *Proceedings of the International Workshop on Workload*, pages 3–14.

Kandemir, M. et al. (2001). Dynamic Management of Scratch-pad Memory Space. *DAC (2001)*, pages 690–695.

McIlroy, R., Dickman, P., and Sventek, J. (2008). Efficient Dynamic Heap Allocation of Scratch-pad Memory. *ISMM (2008)*, pages 31–40.

Mendonca, A. K. I. et al. (2009). Mapping Data and Code into Scratchpads from Relocatable Binaries. *ISVLSI (2009)*, pages 157–162.

Panda, P., Dutt, N., and Nicolau, A. (2000). On-chip vs. Off-chip Memory: The Data Partitioning Problem in Embedded Processor-based Systems. *TODAES (2000)*, 5(3):682–704.

Santos, L. C. V., Volpato, D. P., Mendonca, A. K. I., and Güntzel, J. L. A Fresh Perspective on Scratchpad Usage. *Submetido ao IEEE Transactions on VLSI Systems (2011)*.

Steinke, S. et al. (2002a). Assigning Program and Data Objects to Scratchpad for Energy Reduction. *DATE (2002)*, 0:0409.

Steinke, S. et al. (2002b). Reducing Energy Consumption by Dynamic Copying of Instructions Onto Onchip Memory. *ISSS (2002)*, pages 213–218.

Thoziyoor, S. et al. (2008). CACTI 5.1. Technical report, Hewlett-Packard Laboratories.

Udayakumaran, S., Dominguez, A., and Barua, R. (2006). Dynamic Allocation for Scratch-pad Memory Using Compile-time Decisions. *TECS (2006)*, 5(2):472–511.

Verma, M., Wehmeyer, L., and Marwedel, P. (2004). Dynamic Overlay of Scratchpad Memory for Energy Minimization. *CODES+ISSS (2004)*, pages 104–109.

Volpato, D. P., Mendonca, A. K. I., Güntzel, J. L., and Santos, L. C. V. (2011). Cache-tuning-aware allocation from binaries: a fresh perspective on scratchpad usage. In *SBCCI*.

Volpato, D. P., Mendonca, A. K. I., Santos, L. C. V., and Güntzel, J. L. (2010). A Post-compiling Approach that Exploits Code Granularity in Scratchpads to Improve Energy Efficiency. In *ISVLSI (2010)*, pages 127–132.

Busca Exaustiva em Redes P2P

Péricles C. M. Lopes[1], **Ronaldo A. Ferreira**[1]

[1]Faculdade de Computação – Universidade Federal de Mato Grosso do Sul (UFMS)
Caixa Postal 549 – 79.070-900 – Campo Grande – MS – Brasil

pericles@facom.ufms.br,raf@facom.ufms.br

Abstract. *Despite numerous efforts in the past few years, efficient complex queries in large-scale P2P networks remain an open and challenging problem. This work presents a hybrid architecture for complex queries in P2P networks that relies on a lightweight network structure, in a predominantly non-structured network, to avoid unnecessary query replication and to speed up query propagation in unstructured P2P networks. In addition to this architecture, this work presents a controlled and exhaustive search protocol that combines structured and non-structured P2P networks features. Extensive simulation results, considering static and dynamic scenarios, show that the proposed architecture, in conjunction with a search protocol, outperforms the best known solution in number of messages, response time, number of hops, and success rate for query resolution. Furthermore, this work shows, analytically, using random trees results, an upper bound on query routing.*
Palavras-Chave: *P2P Systems, Distributed Algorithms, Analysis, Simulation*

Resumo. *Apesar de inúmeros esforços nos últimos anos, buscas complexas em redes P2P de grande escala permanecem um problema em aberto e desafiador. Este trabalho propõe uma arquitetura híbrida para buscas complexas em redes P2P que utiliza uma leve estrutura, em uma rede predominantemente não estruturada, para evitar replicações desnecessárias e acelerar a propagação de mensagens de busca. Além da arquitetura, também é proposto um protocolo eficiente de busca que combina características de redes P2P estruturadas e não-estruturadas. Resultados extensivos de simulação, considerando cenários estáticos e dinâmicos, mostram que a arquitetura proposta, em conjunto com o protocolo de busca, possui desempenho superior à melhor solução até então conhecida em termos de número de mensagens, tempo de resposta, número de saltos em respostas positivas e taxa de sucesso. Além disso, o trabalho mostra analiticamente, utilizando resultados de árvores aleatórias, o limite superior teórico para o número máximo de pares que uma mensagem de busca deve percorrer para resolver uma consulta.*
Palavras-Chave: *Sistemas P2P, Algoritmos Distribuídos, Análise, Simulação*

1. Introdução

Nos últimos anos, sistemas par-a-par (P2P - *peer-to-peer*) se tornaram um poderoso paradigma de redes que permite o compartilhamento de vários recursos da Internet de maneira totalmente distribuída. Mais importante, este novo paradigma deu vida a várias outras novas aplicacões, tais como mensagens instantâneas, voz sobre IP (Skype), distribuição de vídeos, computação distribuída (SETI@Home) e muitas outras. Além disso, usuários finais se transformaram em grandes produtores de informações que necessitam ser compartilhadas com o resto do mundo. Para manter essas fontes de informacões, os usuários ainda necessitam de infraestruturas centralizadas, normalmente mantidas por empresas privadas. Uma solução P2P descentralizada para esse problema ainda não é possível porque, apesar de inúmeros esforços, buscas complexas em redes P2P ainda podem ser consideradas um problema em aberto e desafiador.

O problema de busca em redes P2P estruturadas foi resolvido de forma elegante e eficiente utilizando-se tabelas de dispersão distribuídas (DHT - *Distributed Hash Table*) [Stoica et al. 2001]. Entretanto, a semântica da busca é muito limitada, já que um usuário deve saber o nome exato do objeto para determinar o par responsável por manter a réplica do objeto ou sua referência.

Buscas exaustivas foram introduzidas recentemente para redes P2P [Ferreira et al. 2005, Terpstra et al. 2007] como um método de busca efetivo e confiável. Nesse tipo de busca a semântica é bem mais rica, em comparação com as buscas baseadas em *hashing*, já que algoritmos sofisticados podem ser executados localmente em cada par. Entretanto, as soluções mais eficientes se baseiam em caminhadas aleatórias [Ferreira et al. 2005], ou suas variantes com vários fatores de ramificação [Terpstra et al. 2007], para prover garantias probabilísticas. Apesar da eficiência em algumas aplicações, sabe-se que caminhada aleatória é um método de busca "cego" que pode visitar o mesmo par mais de uma vez, gerando sobrecarga desnecessária, e que não garante a cobertura da rede.

Neste trabalho, com o objetivo de se explorar as melhores características tanto do modelo estruturado como do modelo não estruturado, é proposto e avaliado SplitQuest. Em SplitQuest, os pares possuem identificadores únicos e são posicionados uniformemente em um anel virtual. Além dessa leve estrutura, os pares estabelecem conexões aleatórias entre si, como em uma rede P2P não-estruturada. Essa heterogeneidade das conexões dos pares é explorada de forma inteligente por SplitQuest, que propaga de forma agressiva as mensagens de busca pelos pares com grande número de conexões. Além disso, a estrutura simples e a identificação dos pares permitem que SplitQuest direcione as mensagens de busca para partes específicas da rede, evitando a sobrecarga gerada pela duplicação de mensagens.

SplitQuest foi simulado em diferentes topologias geradas por dados reais e sintéticos, com número variável de pares, em cenários estáticos e dinâmicos e apresentou desempenho superior à melhor solução conhecida para busca exaustiva em relação a tempo de resposta, taxa de sucesso e número de mensagens. O trabalho ainda apresenta uma análise matemática, baseada em resultados de árvores aleatórias, que mostra o limite superior para o número máximo de saltos que uma mensagem de busca é propagada pela rede.

As principais contribuições deste trabalho são: (a) um protocolo de busca exaustiva eficiente que explora a heterogeneidade dos pares para distribuir rapidamente mensagens de busca em redes P2P, evita duplicações desnecessárias de mensagens na rede e atinge altas taxas de recuperação de informação (*recall*); (b) uma análise matemática que prova um limite superior no número máximo de saltos que uma mensagem de busca pode propagar; (c) um novo algoritmo de entrada de pares em uma rede P2P que distribui uniformemente os pares em um anel virtual; (d) resultados extensivos de simulação que comparam SplitQuest favoravelmente com BubbleStorm [Terpstra et al. 2007]–a melhor solução conhecida para buscas exaustivas em redes P2P–em importantes métricas, como número de mensagens, taxa de sucesso e tempo de resposta.

Como resultado deste trabalho, foram publicados os artigos [Lopes and Ferreira 2010a] e [Lopes and Ferreira 2010b] e submetido o artigo [Lopes and Ferreira 2011]. Cabe salientar que o IPTPS é considerado um dos eventos de maior impacto na área, com Comitê de Programa de altíssimo nível e classificação A2 no Qualis da Capes. O SBRC também é um evento bastante seletivo e é considerado o fórum mais importante das áreas de redes e sistemas distribuídos do Brasil.

2. O Protocolo SplitQuest

SplitQuest evita a sobrecarga desnecessária de métodos de busca baseados em caminhadas aleatórias (e suas variações) com uma estrutura híbrida que permite que as mensagens de busca

sejam direcionadas para partes específicas e distintas da rede. Em adição às conexões aleatórias, normalmente presentes em redes não estruturadas, SplitQuest dispõe os pares em um anel, ou seja, cada par possui um predecessor e um sucessor identificados com valores no intervalo $[0, 1]$.

2.1. Replicação de Índices

A replicação de índices em SplitQuest é realizada em um conjunto de pares localizados em um subintervalo contíguo do intervalo $[0, 1]$. Quando pares são distribuídos de forma uniforme por todo intervalo, o número esperado de pares em um subintervalo contíguo é proporcional ao tamanho do subintervalo. Consequentemente, o problema básico neste caso é determinar o tamanho do subintervalo, ou, mais especificamente, o tamanho do grupo de replicação que minimiza a replicação de índices e mensagens de busca. Quando o valor de n é conhecido, o tamanho do subintervalo com número esperado de pares igual a d é dado por $\frac{d}{n}$. Seja então d o tamanho do grupo e q o número de grupos na rede. O número total de mensagens de replicação e busca de um objeto, $M = q + d$, é dado por: $M = \frac{n}{d} + d$. É fácil verificar que d igual a $\sqrt{n}$ minimiza M. A Figura 1 ilustra como os pares são organizados em grupos de replicação no intervalo $[0, 1]$.

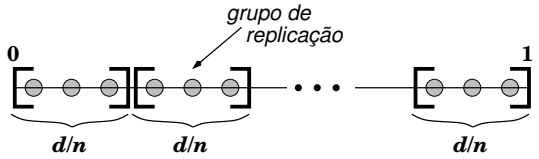


Figura 1. Grupos de replicação em SplitQuest.

2.2. Algoritmo de Busca

SplitQuest utiliza dois artifícios para garantir cobertura de rede e evitar duplicação de mensagens: busca por intervalo e atalhos para grupos. Busca por intervalo significa que cada mensagem de busca contém, além da consulta em si, um intervalo $[X, Y]$ que a busca deve cobrir. Quando um par i recebe uma mensagem com o intervalo $[X, Y]$, ele verifica suas conexões e determina quais vizinhos estão contidos no intervalo recebido. Esses vizinhos são possíveis candidatos para o repasse dessa mensagem de busca. Em seguida, o par i calcula os identificadores de grupos desses possíveis candidatos e mantém apenas um vizinho para cada grupo. Dependendo dos grupos selecionados, o par i decompõe o intervalo inicial em subintervalos e envia mensagens de busca contendo esses subintervalos para cada um dos vizinhos selecionados.

A Figura 2 mostra um exemplo simples de propagação de uma mensagem de busca. Apenas um pequeno conjunto de pares e conexões é mostrado. A busca é iniciada em um par **a** que está conectado aos pares **e**, **b** e **g**. O par **a** é inicialmente responsável por cobrir todo o intervalo de $[0, 1]$. Assim, este par cria três mensagens com intervalos $[0, x_4]$, $[x_5, x_8]$ e $[x_8, 1]$ e as envia para os pares **b**, **e** e **g** respectivamente. O par **b** envia uma mensagem para **c** com intervalo $[0, x_2]$ e o par **e** envia uma mensagem para **f** com intervalo $[x_6, x_8]$. Finalmente, o par **c** envia uma mensagem para **d** com intervalo $[0, x_1]$ e os pares **b** e **f** são responsáveis por realizar a cobertura dos subintervalos restantes ($[x_2, x_3]$ e $[x_6, x_7]$).

Outro artifício utilizado por SplitQuest para melhorar o desempenho da busca são os atalhos para grupos. Os pares constroem atalhos (conexões diretas para outros pares) para os dois grupos adjacentes aos grupos a que eles pertencem. Esses atalhos garantem que sempre existe um caminho para um grupo, ou seja, um caminho que a cada novo salto levará a mensagem de busca a cobrir um novo grupo, evitando que uma consulta seja realizada várias vezes em um mesmo grupo.

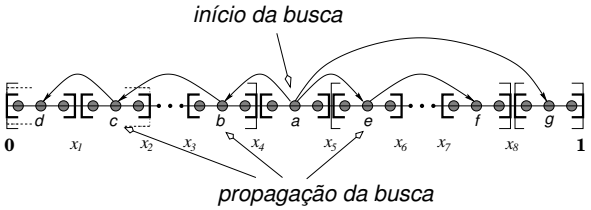


Figura 2. Propagação de uma mensagem de busca em SplitQuest.

2.3. Atribuição de Identificadores

Uma questão-chave em SplitQuest é como os identificadores dos pares são gerados. Os identificadores devem ser distribuídos uniformemente no espaço de identificadores para se garantir tamanhos uniformes de grupo. Além disso, os pares não devem depender de um protocolo complexo de roteamento para entrar na rede, como é o caso de sistemas DHT que utilizam funções de dispersão para geração dos identificadores e um protocolo de entrada custoso.

O algoritmo de atribuição de identificadores de SplitQuest se baseia em resultados obtidos por [Naor and Wieder 2007] e também pela extrapolação do "poder de duas escolhas" [Mitzenmacher 2001] para k-escolhas. Nesse novo algoritmo, o par que deseja se conectar à rede requisita a um *proxy* (um par que já está na rede) que sejam selecionados k intervalos da rede de maneira aleatória e uniforme. Mais especificamente, o *proxy* seleciona k pares e os maiores intervalos adjacentes a esses. O identificador do par entrante é o ponto médio do maior segmento. Para se ter uma boa distribuição uniforme, o valor de k deve ser proporcional a $\log n$. A Figura 3(a) mostra a eficiência do algoritmo proposto em distribuir uniformemente os identificadores no intervalo $[0, 1]$ em comparação com a atribuição de identificadores por funções de dispersão.

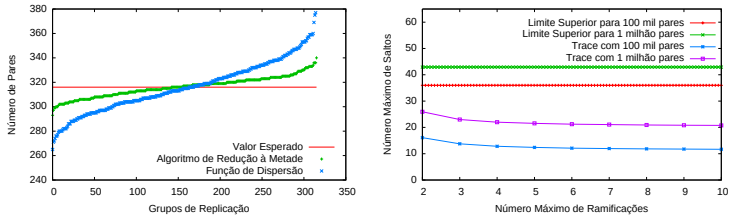
2.4. Análise do Protocolo

A estrutura formada pelos pares e suas conexões em SplitQuest pode ser vista como um grafo aleatório. Se forem consideradas as conexões de grupos, montadas como conexões aleatórias durante a busca, pode-se modelar a propagação de mensagens de busca para os diferentes grupos como uma difusão (*broadcast*) em uma árvore aleatória. Em cada passo do algoritmo, os grupos são particionados em função das conexões do par que recebeu a mensagem de busca e do intervalo que este deve cobrir.

Quando uma busca se inicia em um par i, as mensagens de busca devem visitar um conjunto G de grupos disjuntos, inicialmente G é igual ao número total de grupos. O conjunto é dividido em R subgrupos, que dependem das conexões de i, com distribuição dada por $P(R = r) = p_r$, em que p_r é a distribuição de probabilidade em $\{1, 2, \ldots\}$. Condicionado ao evento $\{R = r\}$, para $1 \leq k \leq r$, um grupo está no k-ésimo subconjunto com probabilidade $V_{k,r}$, em que $V_r = (V_{k,r}; 1 \leq k \leq r)$ é um vetor de probabilidade em $\{1, \ldots, r\}$. Se N_k é a cardinalidade do k-ésimo subconjunto, então, condicionado ao evento $\{R = r\}$ e às variáveis aleatórias $V_{1,r}, \ldots, V_{r,r}$, a distribuição do vetor $(N_1, \ldots, N_r)$ é multinomial com parâmetros $G-1$ e $(V_{1,r}, \ldots, V_{r,r})$ [Mohamed and Robert 2005].

A análise da altura (H_G) da árvore aleatória para um distribuição geral de R é difícil e é geralmente obtida por técnicas de análise complexa. Um limite superior mais conservador pode ser obtido quando r pertence ao conjunto $\{1, 2\}$, o que significa que temos no máximo duas ramificações em cada passo da difusão. Devroye [Devroye 1998] mostra que, quando r pertence a $\{1, 2\}$, a altura da árvore é limitada por $O(\log G)$, mais especificamente Devroye mostra que $\lim_{G\to\infty} P\{H_G > c \log G\} = 0$, em que c é uma constante. Um resultado mais forte é derivado em [Devroye 1986], em que é mostrado que $H_G / \log G \to 4.31106\ldots$ em probabilidade.

É fácil verificar que o limite superior derivado acima é também um limite superior para o maior número de saltos que uma mensagem de busca é propagada em SplitQuest. Como SplitQuest pode enviar uma mensagem de busca para mais de dois grupos, o limite superior é, de fato, bastante conservador. A Figura 3(b) mostra o número máximo de saltos que uma mensagem de busca percorre em redes com 100 mil e 1 milhão de pares e quando o número máximo de ramificações é incrementado de 2 a 10. A demonstração completa do limite superior pode ser encontrada na Seção 3.5 da dissertação.



(a) Atribuição de IDs (b) Limite Superior

Figura 3. a) Número de pares em grupos de replicação para tamanho de rede de 100 mil pares. b) Distância máxima de propagação de uma mensagem de busca com diferentes ramificações. As linhas superiores são limites teóricos para 1 milhão e 100 mil pares respectivamente.

3. Resultados

Para avaliar SplitQuest, foi desenvolvido um simulador em C++ e realizados vários experimentos para comparar seu desempenho com a melhor solução conhecida até então [Terpstra et al. 2007]. Foram avaliadas três métricas: número de mensagens, taxa de sucesso e latência, em três topologias de rede distintas: regular, power-law e *trace* real (Gnutella) em cenários estáticos e dinâmicos. Em todas as figuras, a coordenada x representa diferentes tamanhos de rede e a coordenada y uma medida específica computada como sendo a média de 11 simulações independentes. Para todos os resultados, foram calculados os intervalos de confiança com grau de confiança de 95%.

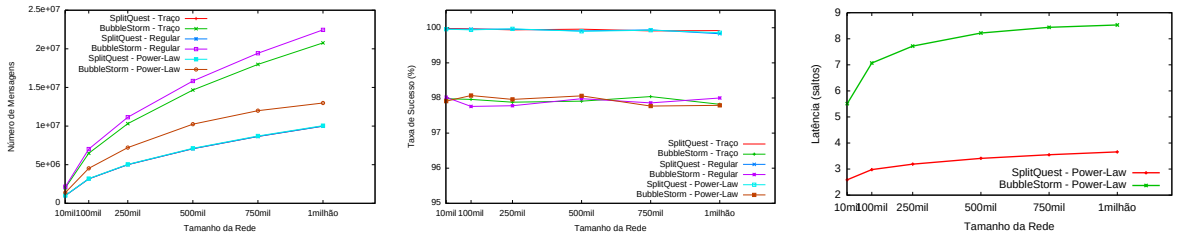
A Figura 4(a) mostra os resultados obtidos para o número de mensagens quando 5000 documentos são inseridos na rede. O número total de mensagens presentes corresponde à soma dos dados e de mensagens de busca durante toda simulação. É possível observar que, em todas as três topologias e tamanhos de rede, o número de mensagens em SplitQuest é o mesmo (linha única) já que SplitQuest utiliza apenas o tamanho da rede como parâmetro para determinar o número de mensagens inseridas na rede. Dessa forma, esse número de mensagens é independente de topologia e distribuição do grau dos nós dos pares. Além disso seu desempenho com relação ao número de mensagens inseridas na rede superou o BubbleStorm em todos os cenários.

A taxa de sucesso em um cenário dinâmico pode ser vista na Figura 4(b). Pode-se observar que Splitquest atinge altas taxas de sucesso mesmo em cenários dinâmicos. Como SplitQuest direciona a busca para pares não visitados, ele apresenta resultado bem superior a BubbleStorm que utiliza caminhada aleatória.

Outro resultado importante é a baixa latência na resolução de consultas como mostrado na Figura 4(c). Esse resultado se deve à exploração agressiva da heterogeneidade dos pares que faz com que as mensagens sejam distribuídas pela rede de maneira muito mais rápida.

4. Conclusão

Este trabalho apresentou SplitQuest, um novo protocolo de busca para redes P2P que se baseia em uma estrutura híbrida para se evitar replicações desnecessárias de mensagens de busca e para



(a) Número de Mensagens (b) Taxa de Sucesso (c) Latência

Figura 4. Resultados.

garantir cobertura da rede. Resultados numéricos e análise matemática mostram que SplitQuest é capaz de buscar informações em uma rede P2P de maneira eficiente e que, dessa forma, possui potencial para estimular o desenvolvimento de novas aplicações distribuídas, em particular aplicações que publicam baixas quantidades de dados, tais como microblogs e wikis.

Referências

Devroye, L. (1986). A Note on the Height of Binary Search Trees. *Journal of ACM*, 33(3):489–498.

Devroye, L. (1998). Universal Limit Laws for Depths in Random Trees. *SIAM Journal on Computing*, 28(2):409–432.

Ferreira, R. A., Ramanathan, M. K., Awan, A., Grama, A., and Jagannathan, S. (2005). Search with Probabilistic Guarantees in Unstructured Peer-to-Peer Networks. In *Proceedings of the 2005 IEEE P2P (P2P'05)*, pages 165–172, Konstanz, Alemanha.

Lopes, P. and Ferreira, R. A. (2010a). SplitQuest: Controlled and Exhaustive Search in Peer-to-Peer Networks. In *Proceedings of the 2010 Usenix IPTPS (IPTPS'10)*, San Jose, CA.

Lopes, P. and Ferreira, R. A. (2010b). Uma Arquitetura Híbrida para Buscas Complexas em Redes P2P. In *Anais do SBRC 2010*, Gramado, RS.

Lopes, P. and Ferreira, R. A. (2011). Exhaustive Search in Peer-to-Peer Networks. *Journal of Parallel and Distributed Computing (JPDC) - submetido, em avaliação*.

Mitzenmacher, M. (2001). The Power of Two Choices in Randomized Load Balancing. *IEEE Trans. Parallel Distrib. Syst.*, 12(10):1094–1104.

Mohamed, H. and Robert, P. (2005). A Probabilistic Analysis of Some Tree Algorithms. *The Annals of Applied Probability*, 15(4):2445–2471.

Naor, M. and Wieder, U. (2007). Novel Architectures for P2P Applications: The Continuous-Discrete Approach. *ACM Transactions on Algorithms (*Eletronic Edition*)*, 3(3).

Stoica, I., Morris, R., Karger, D., Kaashoek, F., and Balakrishnan, H. (2001). Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications. In *Proceedings of the 2001 ACM SIGCOMM (SIGCOMM'01)*, pages 149–160, San Diego, CA.

Terpstra, W. W., Kangasharju, J., Leng, C., and Buchmann, A. P. (2007). BubbleStorm: Resilient, Probabilistic, and Exhaustive Peer-to-Peer Search. In *Proceedings of the 2007 ACM SIGCOMM (SIGCOMM'07)*, pages 49–60, Tóquio, Japão.

Paralelização do Algoritmo de Migração Sísmica em Plataformas Heterogêneas

Thiago S. F. X. Teixeira[1,3], **Wagner Meira Jr**[1], **Jairo Panetta**[2,3]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte – MG – Brasil

[2]Divisão de Ciência da Computação – Instituto Tecnológico de Aeronáutica (ITA)
São José dos Campos – SP – Brasil

[3]Tecnologia Geofísica – Petróleo Brasileiro S.A. – PETROBRAS

{thiago, meira}@dcc.ufmg.br, jairo.panetta@gmail.com

Palavras-chave: Processamento paralelo e distribuído, processamento sísmico, GPGPU, escalonamento.

Abstract. *Petrobras daily uses Kirchhoff seismic migration to search for new oil and gas reserves. Kirchhoff is a CPU bound algorithm—a single run may use up to 1000 dedicated x86 cores during a month. The emergence of novel parallel computing architectures, such as GPUs, poses an oportunity and a challenge to reduce such execution times. In this thesis, we discuss the parallelization of the Kirchhoff seismic migration for an heterogeneous environment with CPUs and GPUs. We search and evaluate parallelism strategies that eficiently use the available hardware. We explore parallelism opportunities, by designing, implementing and evaluating various possible configurations. We also devise and implement a finer grain dynamic scheduling for the devices, achieving highly efficient executions. Experiments show an acceleration of up to 87 times over a single x86 core execution.*

Resumo. *A Petrobras usa diariamente a migração sísmica de Kirchhoff para procurar novas reservas de petróleo. Kirchhoff é um algoritmo limitado pela CPU—uma única execução pode utilizar 1000 núcleos x86 dedicados durante um mês. O surgimento de arquiteturas paralelas inovadoras, como as GPUs, apresenta oportunidades e desafios para reduzir tais tempos de execução. Nesta dissertação, discutimos a paralelização do algoritmo da migração sísmica de Kirchhoff para um ambiente heterogêneo com CPUs e GPUs. Pesquisamos e avaliamos estratégias de paralelismo que utilizam eficientemente o maquinário disponível. Exploramos as oportunidades de paralelismo projetando, implementando e avaliando várias configurações possíveis. Nós também projetamos e implementamos um escalonamento dinâmico de grão mais fino para os dispositivos, possibilitando execuções mais eficientes. Os experimentos apresentaram uma aceleração de até 87 vezes sobre a execução em um único núcleo x86.*

1. Introdução

O processamento sísmico é uma das tarefas de maior demanda computacional na indústria do petróleo. A Petrobras executa, em produção, a migração sísmica de Kirchhoff em aglomerados dedicados de até 1000 núcleos x86. Uma execução pode demandar um mês.

A indústria constantemente avalia a utilização de novas arquiteturas para acelerar esse processamento e reduzir os custos de resfriamento, armazenamento e consumo de energia dos aglomerados. Atualmente, surgem novas arquiteturas paralelas que transformam os nós de processamento em ambientes hierárquicos e heterogêneos, onde cada nó pode ter múltiplos processadores diferentes, como as CPUs multi-core e as GPUs many-core.

Esses ambientes heterogêneos têm sido bastante pesquisados pela comunidade científica nos últimos anos, principalmente, por causa de seu melhor preço/watt e preço/flops. No entanto, é um desafio utilizar de maneira eficiente esse conjunto assimétrico de processadores quando o número de núcleos das CPUs e GPUs varia. Por exemplo, em aglomerados adquiridos em anos distintos, contendo nós distintos. Para se beneficiar de toda essa capacidade de processamento, as aplicações precisam escalonar as tarefas de maneira balanceada, considerando o melhor momento e a melhor unidade de processamento para cada caso. Isso leva a novas abordagens e estratégias para escalonar tarefas e alcançar o melhor desempenho, sem perder a flexibilidade requerida pelas diferentes configurações dos aglomerados.

A tendência atual das novas arquiteturas é aumentar o desempenho acrescentando mais núcleos. Isso significa que para se beneficiar de todo esse poder computacional adicional será necessário implementar aplicações escaláveis, pois é inviável reimplementar a aplicação para cada novo aglomerado.

A paralelização eficiente e escalável da migração sísmica é um desafio pela própria complexidade da aplicação e suas várias oportunidades de paralelização possíveis, o que demanda o desenvolvimento de técnicas que explorem essas oportunidades em ambientes de processamento heterogêneos. Mais ainda, ela tem grande importância para a indústria do petróleo, uma vez que, quanto mais rápido os resultados forem obtidos, mais regiões podem ser exploradas na busca por óleo e gás. Assim, se torna evidente o interesse e o desafio do ponto de vista acadêmico de estudar e fornecer soluções para indústria sobre como explorar de maneira eficiente essas novas arquiteturas.

Resumidamente, as principais motivações para esse trabalho são: (i) disponibilidade de novas arquiteturas paralelas, como as CPUs multi-core e as GPUs programáveis, atualmente comums nos ambientes de pesquisa e desenvolvimento em computação científica; (ii) utilização eficiente dessas novas arquiteturas, em que os desenvolvedores precisarão se preocupar com a escalabilidade e levar em conta as características de cada arquitetura. Além de lidar com o escalonamento de tarefas flexível para plataformas multiaceleradas, multiprocessadas, multi-core e heterogêneas; e (iii) relevância da migração sísmica no processamento dos dados sísmicos, pois, além de ser muito utilizada, é um dos algoritmos de maior demanda computacional.

Portanto, o objetivo é o de estudar, implementar e avaliar oportunidades de paralelismo na migração sísmica em ambientes heterogêneos e variáveis para obter o máximo de eficiência e desempenho na utilização dos recursos computacionais disponíveis.

A principal contribuição desse trabalho foi a paralelização do algoritmo de migração sísmica de Kirchhoff para um ambiente heterogêneo distribuído composto de CPUs multi-core e GPUs. Apesar de as CPUs e as GPUs terem sido o alvo da implementação, esta pode ser facilmente generalizada para outros dispositivos e ambi-

entes com características semelhantes. A implementação foi feita de maneira escalável e flexível para permitir a utilização concorrente de vários processadores e múltiplos aceleradores em múltiplas configurações.

Três dimensões de paralelismo foram desenvolvidas para atingir os objetivos mencionados anteriormente. Essas três dimensões utilizaram assincronia, paralelismo de dados e paralelismo de tarefas para alcançar uma significativa melhora no desempenho.

Outra contribuição importante foi a implementação de um escalonamento dinâmico de tarefas de grão mais fino entre as unidades de processamento. Essa contribuição permitiu que não houvesse desbalanceamento de carga entre as múltiplas unidades heterogêneas e que todas operassem na máxima capacidade de processamento.

Parte dos resultados do trabalho de pesquisa e desenvolvimento da migração sísmica em ambientes heterogêneos foram publicados no artigo [Panetta et al. 2009]. Esse mesmo artigo foi convidado para ser estendido para a revista *International Journal of Parallel Programming* e o texto submetido foi aceito para publicação. Esse texto contém um resumo das informações consideradas mais importantes. Para mais informações veja [Teixeira 2010].

2. Migração Sísmica

O método sísmico pode ser dividido em três etapas: aquisição, processamento e interpretação. A primeira etapa, a aquisição dos dados, utiliza uma fonte de ondas sísmicas na superfície, como uma bomba ou um canhão de ar comprimido, para periodicamente gerar ondas que são refletidas nas interfaces das camadas que formam o subsolo e depois são coletadas por receptores na superfície. Um traço é o conjunto de sinais originados por uma única emissão da fonte e coletados por um único receptor. Um traço é composto por valores de amplitude denominados amostras, coletados em intervalos de tempo constantes.

O processamento dos dados sísmicos, que corresponde à segunda etapa, tem como objetivo extrair informações do subsolo a partir dos dados adquiridos. A indústria do petróleo utiliza, internacionalmente, pacotes de software proprietários que contém centenas de módulos sísmicos. Profissionais de processamento sísmico selecionam os módulos mais apropriados para a aquisição e para a área alvo. Esse processamento demanda meses de trabalho de uma equipe especializada e seu tempo de execução é totalmente dominado pelo módulo de migração sísmica. A terceira etapa corresponde à interpretação dos dados por outros especialistas, que indicam onde perfurar.

A migração é o processo de produzir uma imagem do subsolo consistente com os dados adquiridos, através do posicionamento correto das superfícies refletoras. É um problema inverso, pois produz os parâmetros do modelo a partir de dados observados. Como em muitos outros problemas inversos, soluções consistentes podem requerer múltiplas execuções. A migração sísmica de Kirchhoff [Yilmaz 1987] é um dos métodos de migração mais utilizados no processamento dos dados.

O algoritmo da migração sísmica calcula as contribuições de cada traço de entrada previamente filtrado para os traços de saída contidos em uma abertura definida pelo usuário. A migração possui duas etapas que dominam o seu tempo de execução: **filtragem** dos traços de entrada e o **laço de migração**. Essas duas etapas podem ser sobrepostas,

pois enquanto um traço de entrada está sendo migrado no laço de migração, um outro traço de entrada pode ser filtrado.

3. Paralelização da Migração Sísmica

Para aumentar a cooperação entre os dispositivos e utilizar todos os recursos computacionais de um ambiente heterogêneo, implementamos três dimensões de paralelismo, que podem se adaptar a diferentes plataformas pela simples alteração de parâmetros de execução. Nós utilizamos o Anthill [Ferreira et al. 2005] para implementar a migração sísmica em um ambiente distribuído. As dimensões de paralelismo foram:

1. *Paralelismo pelo dado de saída:* ocorre pela execução simultânea do laço de migração para um único traço de entrada e múltiplos traços de saída. Como a computação de um traço de saída é independente de qualquer outro traço de saída, este processamento é embaraçosamente paralelo e pode ter diferentes abordagens de acordo com o dispositivo usado (i.e., CPU ou GPU);
2. *Paralelismo de tarefas:* as duas tarefas que dominam o tempo de execução (filtragem e laço de migração) podem ser sobrepostas, utilizando a GPU para o laço de migração e a CPU para a filtragem. No Anthill, cada uma dessas etapas é um filtro. Como cada filtro pode ter múltiplas cópias independentes, o paralelismo de tarefas ocorre pela execução simultânea de cópias que atuam em dados distintos;
3. *Paralelismo pelo dado de entrada:* corresponde ao processamento simultâneo de vários traços de entrada para um mesmo conjunto de traços de saída. Isto acarreta uma condição de corrida pois cada traço de saída somente pode receber a contribuição de um traço de entrada por vez. Para superar esse problema, múltiplos traços de entrada são processados simultâneamente, mas em diferentes cópias dos traços de saída e, no final, essas cópias são agregadas para gerar o resultado final.
   Foi utilizada uma abordagem de escalonamento de grão mais fino, por traços de entrada, entre as unidades de processamento. Assim, o mesmo conjunto de traços de saída pode receber as contribuições de vários traços de entrada em dispositivos diferentes, que podem operar na máxima capacidade, sem deixar recursos ociosos. Essa abordagem não garante um escalonamento de traços ótimo entre os dispositivos, mas como o grão é fino, a penalidade do dispositivo mais lento processar o último traço é pequena.

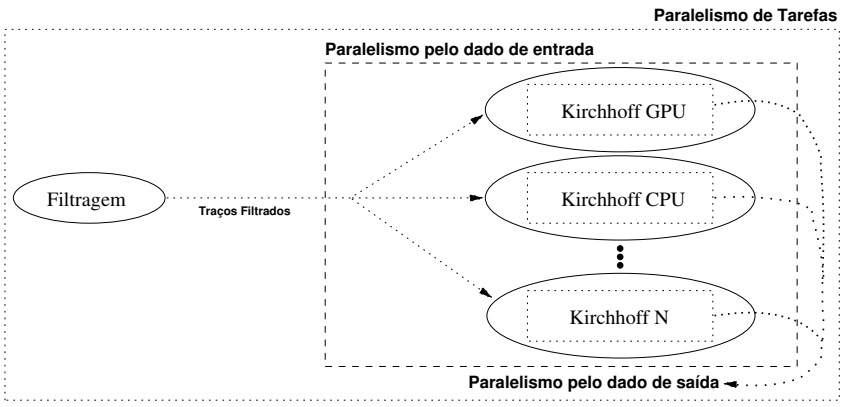


Figura 1. Três dimensões de paralelismo.

A figura 1 apresenta as três dimensões de paralelismo e como elas se relacionam. A etapa *Kirchhoff* na figura corresponde ao laço de migração.

4. Experimentos

Os experimentos foram feitos com o objetivo de avaliar as três dimensões de paralelismo apresentadas anteriormente. Eles foram executados utilizando um aglomerado onde cada nó contem dois processadores Intel Xeon L5420 com 4 núcleos cada, 32 GB de memória principal e uma GPU Nvidia Tesla C1060.

Utilizamos somente um nó do aglomerado pois o processamento dos blocos de traços de saída é embaraçosamente paralelo e o speedup para múltiplos nós, cada um processando um bloco por vez, é linear [Panetta et al. 2009]. A configuração utilizada para um nó pode ser replicada para todos os outros do aglomerado e facilmente alterável (por parâmetros de execução) para outros aglomerados. Utilizamos como métrica de avaliação a taxa de amostras contribuídas por segundo. As configurações analisadas foram: sequencial na CPU (*CPU*), laço de migração paralelizado na CPU (*8 CPUs*), execução síncrona do laço de migração na GPU (*GPU sínc*), execução assíncrona do laço de migração na GPU (*GPU assínc*), execução assíncrona do laço de migração na GPU com duas instâncias para a filtragem no ambiente Anthill (*AH-GPU*), execução assíncrona do laço de migração na GPU e na CPU com duas instâncias para a filtragem no ambiente Anthill (*AH-GPU+CPU*). Em termos de oportunidades de paralelismo exploradas, 8 CPUs e GPU sínc exploram paralelismo pelo dado de saída. GPU sínc, GPU assínc e AH-GPU exploram paralelismo de tarefas. Finalmente, AH-GPU e AH-GPU+CPU exploram paralelismo pelo dado de entrada.

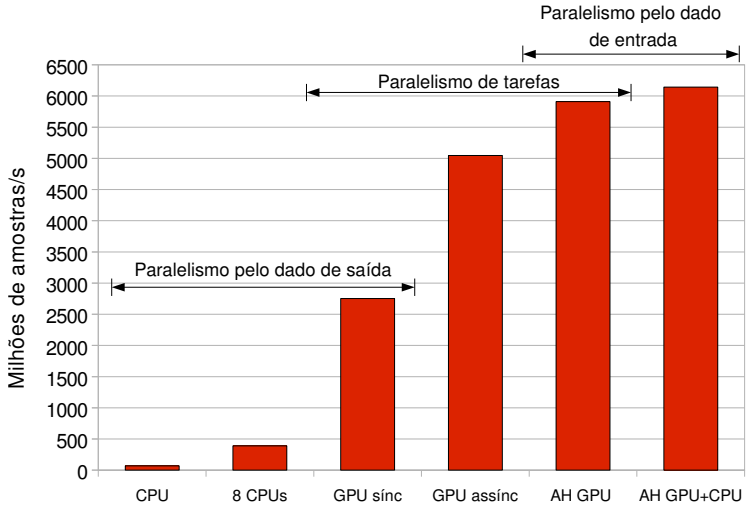


Figura 2. Análise comparativa do desempenho alcançado pelas três dimensões de paralelismo.

A figura 2 resume o desempenho alcançado em cada dimensão. A contribuição do paralelismo pelo dado de saída foi a aceleração de 5,55 vezes para 8 CPUs e de 39 vezes para a GPU (*GPU sínc*) com relação à execução sequencial na CPU. O paralelismo de tarefas ocorre tanto para o laço de migração na CPU quanto na GPU, no entanto, a contribuição foi significativa somente para o caso da GPU. A assincronia de execução entre as tarefas aumentou de 2.753 milhões (*GPU sínc*) para 5.045 milhões (*GPU assínc*) a taxa de amostras contribuídas por segundo. O aumento no paralelismo da etapa de filtra-

gem (*AH-GPU*), através do aumento do número de instâncias no ambiente Anthill, elevou esse ganho para 5.910 milhões. Por último, a utilização conjunta entre CPU e GPU (*AH-GPU+CPU*) no filtro Kirchhoff alcançou 6.143 milhões de amostras por segundo, através da utilização do paralelismo pelo dado de entrada. É importante notar que, tendo em vista as características do problema, a estratégia de paralelização pode ser replicada de forma trivial para várias máquinas que conjuguem CPUs e GPUs, utilizando uma estratégia de paralelismo de tarefas.

5. Conclusões e Trabalhos Futuros

Nesse trabalho discutimos a paralelização da migração sísmica de Kirchhoff para ambientes heterogêneos distribuídos. Para atingir o objetivo de aproveitar todos os recursos presentes, foram implementadas três dimensões de paralelismo: pelo dado de saída, de tarefas e pelo dado de entrada. A cooperação entre CPU e GPU foi priorizada ao invés de acelerar o processamento com toda a execução na GPU.

Os resultados mostraram uma redução enorme no tempo de execução com a utilização da GPU para a tarefa de maior demanda computacional. Além disso, a cooperação entre GPU e CPU, tanto na dimensão de tarefas quanto na dimensão dos dados de entrada, contribuíram ainda mais para melhorar o desempenho da aplicação. A assincronia utilizada na implementação em conjunto com a proposta pelo Anthill também contribuiu muito para o desempenho alcançado. A implementação utilizando as três dimensões propostas alcançou 6.143 milhões de amostras contribuídas por segundo, uma aceleração de até 87 vezes com relação à execução sequencial na CPU.

Como trabalhos futuros vislumbramos a aplicação dessas dimensões de paralelismo e da cooperação entre as múltiplas unidades de execução em outras aplicações. Várias aplicações de processamento sísmico se encaixam nesse perfil, como: RTM, Kirchhoff em profundidade e migração Omega-xy. Também seria interessante avaliar essa cooperação para casos em que a aceleração da GPU com relação à CPU seja menor. Assim, a CPU seria mais eficaz e novas dimensões de paralelismo podem ser exploradas.

Referências

Ferreira, R. A., Meira Jr., W., Guedes, D., Drummond, L. M. A., Coutinho, B., Teodoro, G., Tavares, T., Araujo, R., and Ferreira, G. T. (2005). Anthill: A scalable run-time environment for data mining applications. In *SBAC-PAD '05: Proceedings of the 17th International Symposium on Computer Architecture on High Performance Computing*, pages 159–167, Washington, DC, USA. IEEE Computer Society.

Panetta, J., Teixeira, T., de Souza, P., da Cunha, C., and et al. (2009). Accelerating kirchhoff migration by cpu and gpu cooperation. In *SBAC-PAD '09: Proceedings of the 21st International Symposium on Computer Architecture and High Performance Computing*, pages 81–90, Washington, DC, USA. IEEE Computer Society.

Teixeira, T. S. F. X. (2010). Paralelização do algoritmo de migração sísmica em plataformas heterogêneas. Dissertação de Mestrado, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil. `http://www.dcc.ufmg.br/~thiago/ctd`.

Yilmaz, O. (1987). *Seismic Data Processing*, volume 2 of *Investigations in Geophysics*. Society of Exploration Geophysicists, Tulsa.

Making Program Refactoring Safer

Gustavo Soares[1], **Rohit Gheyi**[1], **Dalton Serey**[1]

[1]Department of Computing and Systems – UFCG
58429-900 – Campina Grande – PB – Brazil

{gsoares,rohit,dalton}@dsc.ufcg.edu.br

Abstract. *Developers rely on compilation, test suite and tools to preserve observable behavior during refactorings. However, most of the refactoring tools do not implement all preconditions that guarantee behavioral preservation, since formally identifying them is cost-prohibitive. Therefore, these tools may change the behavior of the user's code. We propose two approaches for making program refactoring safer. We present an approach and its implementation (*SAFEREFACTOR*) useful for improving the developer's confidence that the refactoring was correctly applied. It analyzes a transformation and generates tests specific for detecting behavioral changes. Moreover, we propose an approach to help refactoring tool developers to test their implementations. This approach evaluates an implementation by automatically generating programs (test inputs), and checking the results with* SAFEREFACTOR. *We evaluated* SAFEREFACTOR *in transformations applied to seven real programs ranging from 3 to 100 KLOC. In one of them (JHotDraw), we identified a behavioral change that was not revealed so far. On the other hand, we evaluated the second approach by testing 22 refactoring implementations in two Java refactoring tools: the Eclipse Refactoring Module and the JastAdd Refactoring Tool. We analyzed more than 80K transformations, and identified 67 bugs.*

Keywords: Refactoring, Behavior-preservation, Testing, Program generation

1. Introduction

Refactoring is defined as the process of changing a software system in such a way that it does not alter the external behavior of the code and improves its internal structure [Fowler 1999, Mens and Tourwé 2004]. In practice, developers perform refactorings either manually – error-prone and time consuming – or with the help of IDEs that support refactoring, such as Eclipse, Netbeans, JBuilder and IntelliJ.

In general, each refactoring may contain a number of preconditions to preserve the observable behavior. For instance, to rename an attribute, name conflicts cannot be present. However, mostly refactoring tools do not implement all preconditions, because formally establishing all of them is not trivial. Therefore, often refactoring tools allow wrong transformations to be applied with no warnings whatsoever. For instance, the program shown in Listing 1 containing the `A` class and its subclass `B`. The `m` method yields 10. When we apply the pull up refactoring to the `k(int)` method using Eclipse 3.6, the IDE moves it to class `A`. After the transformation, the `m` method yields `20`, instead of `10`. Therefore, the transformation does not preserve behavior using the Eclipse 3.6 IDE. Notice that, originally, the expression inside `m` calls `A.k(long)`. However, after pulling up `k(int)`, this method is called instead.

Listing 1. Pulling up B.k(int) using Eclipse 3.6 changes program behavior.

```
public class A {
  public int k(long i) { return 10; }
}
public class B extends A {
  public int k(int i) { return 20; }
  public int m() { return new A().k(2); }
}
```

The current practice to avoid behavioral changes in refactorings relies on solid tests [Fowler 1999]. However, often test suites do not catch behavioral changes during transformations. They may also be refactored (for instance, rename method) by the tools since they may rely on the program structure that is modified by the refactoring [Mens and Tourwé 2004]. In this case, the tool changes the method invocations on the test suite, and the original and the refactored programs are checked against different test suites. This scenario is undesirable since the refactoring tool may change the test suite meaning [Schäfer et al. 2008].

Moreover, testing refactoring tools is not simple. It requires complex test inputs (Java programs). In practice, refactoring tool developers manually write these inputs. As a result, the test suite may leave hidden bugs that the testers are unaware of. In this work, we aim at making program refactoring safer by helping not only refactoring practitioners but also refactoring tool developers.

2. SAFEREFACTOR

We propose an approach (SAFEREFACTOR) [Soares et al. 2010, Soares et al. 2009b, Soares 2010a, Soares 2010b, Soares et al. 2009a] for checking refactoring safety in sequential Java programs. It analyzes a transformation, and generates a test suite useful for detecting behavioral changes.

We implemented SAFEREFACTOR as an Eclipse plugin. Suppose that we use it to evaluate the previous transformation (see Listing 1). Next we explain the whole process (Figure 1). First the developer selects the refactoring to be applied on the source program (Step 1.1) and uses SAFEREFACTOR (Step 1.2). The plugin starts checking the refactoring safety (Steps 2-7). It generates a target program based on the desired transformation using Eclipse refactoring API (Step 2). In Step 3, a static analysis automatically identifies methods in common in both source and target programs. Step 4 aims at generating unit tests for methods identified in Step 3. In Step 5, the plugin runs the generated test suite on the source program. Next, it runs the same test suite on the target program (Step 6). If a test passes in one of the programs and fails in the other one, the plugin detects a behavioral change and reports to the user (Step 7). Otherwise, the developer can have more confidence that the transformation does not introduce behavioral changes.

3. A technique for testing refactoring tools

We also propose an approach to help refactoring tool developers to test their implementations [Soares 2010a, Soares 2010b]. First, it automatically generates a large number of small programs as test inputs. This step aims at generating inputs that testers may be unaware of. Second, it applies the refactoring implementation to each one of them. It

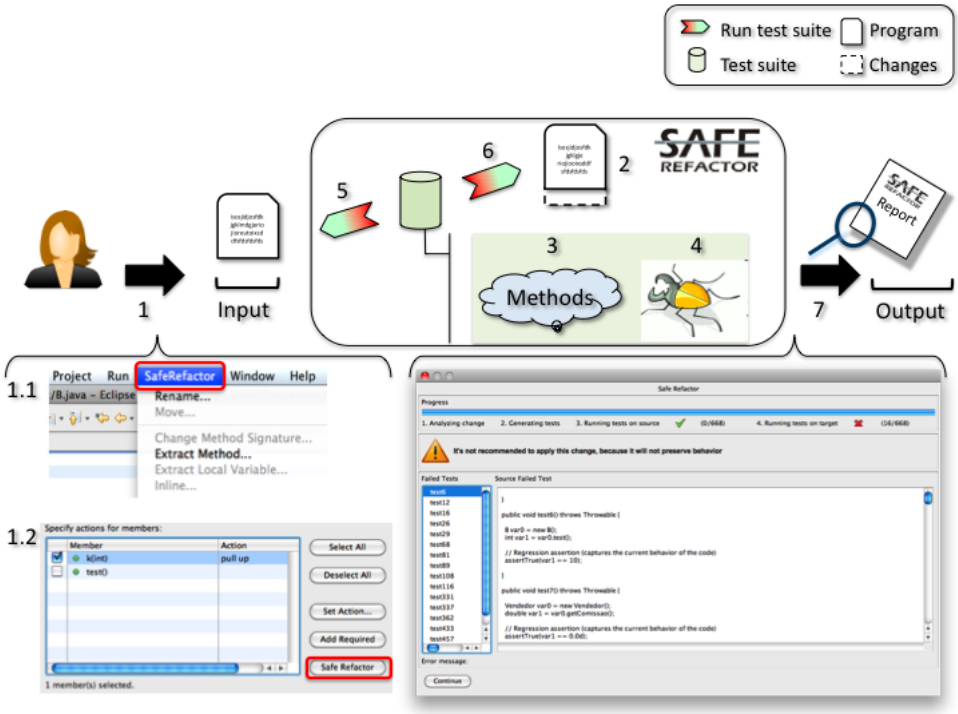


Figure 1. SAFEREFACTOR. 1. The user selects a refactoring in the menu to apply (1.1) and click on the SAFEREFACTOR button (1.2), 2. The tool generates the target program using the Eclipse refactoring API, 3. It identifies common methods in the source and target programs, 4. The tool generates unit tests, 5. It runs the test suite on the source program, 6. The tool runs the test suite on the target program, 7. The tool shows the report to developer. If it finds a behavior change, the developer can see some unit tests that fail.

then uses SAFEREFACTOR to evaluate the correctness of the transformations. To perform the first step, we developed JDolly, a Java program generator. It contains a subset of the Java metamodel specified in Alloy, a formal specification language [Jackson 2006], and uses the Alloy Analyzer, a tool for analysis of Alloy models, for generating solutions for this metamodel. Each solution is translated to a Java program. JDolly receives as input the scope of the generation, that is, the maximum number of elements (packages, classes, fields, and methods) that generated programs must have, and additional constraints for generating test inputs specific for each implementation.

4. Evaluation

We evaluated SAFEREFACTOR against 24 transformations applied to Java programs[1]. They were divided in two categories: *Refactorings in real Java applications*, and *Catalog of Defective Refactorings*. The former (Table 1) consists of refactorings applied to seven real Java applications (3-100 KLOC). Developers used tools or manual steps for applying these transformations, and the test suites of the programs for guaranteeing behavioral preservation. All of the transformations were considered behavior preserving by them. The latter is a catalog of non-behavior-preserving transformations applied by tools

[1]All subjects are available at: http://www.dsc.ufcg.edu.br/~spg/saferefactor/experiments.htm

Subject	Program	KLOC	Refactoring	Result
1	JHotDraw	23	Extract Exception Handler	Behavior Change
2	CheckStylePlugin	20	Extract Exception Handler	-
3	JUnit	3	Infer Generic Type	-
4	Vpoker	4	Infer Generic Type	-
5	ANTLR	32	Infer Generic Type	Compilation Error
6	Xtc	100	Infer Generic Type	Compilation Error
7	TextEditor	15	Replace Deprecated Code	-

Table 1. Refactorings in real Java applications.

Refactoring	Programs	Time (mm:ss)		Failures				Bugs			
				CE		BC		CE		BC	
		Eclipse	JRRT	Eclipse	JRRT	Eclipse	JRRT	Eclipse	JRRT	Eclipse	JRRT
Rename Class	7200	03:12	03:48	194	0	145	0	1	0	1	0
Rename Method	13464	07:54	11:30	549	0	0	0	1	0	0	0
Rename Field	6080	05:18	05:58	6	76	0	16	1	3	0	1
Push Down Method	5880	04:48	04:36	595	618	186	105	2	2	6	2
Push Down Field	7488	04:42	04:26	340	0	92	0	1	0	1	0
Pull Up Method	8760	06:24	06:06	132	296	203	74	2	2	7	2
Pull Up Field	6624	05:36	06:30	222	72	546	0	3	2	3	0
Encapsulate Field	8832	05:26	05:34	2000	0	0	108	1	0	0	1
Move Method	8938	08:31	05:39	889	922	3586	1422	2	3	4	2
Add Parameter	15808	13:41	14:18	706	0	1116	137	1	0	3	2
Remove Parameter	15808	14:18	-	706	-	190	-	1	-	2	-
Change Modifier	7224	11:42	-	602	-	703	-	1	-	3	-
Total of Bugs								**17**	**12**	**28**	**10**

Table 2. Test of refactoring implementations of Eclipse and JRRT; CE = compilation error; BC = behavioral change.

manually catalogued in the literature [Schäfer et al. 2008, Steimann and Thies 2009]. In the first category, SAFEREFACTOR detected a behavioral change in one transformation and two compilation errors (Table 1). Regarding the *Catalog of Defective Refactorings*, it detected all behavioral changes but one.

Moreover, we evaluated our technique for testing refactoring tools by testing 22 refactoring implementations in two Java refactoring tools: the Eclipse Refactoring Module and the JastAdd Refactoring Tool (JRRT) [Schäfer et al. 2010]. Table 2 summarizes the results. It shows the number of programs generated by JDolly, the testing time, and the number of failures (compilation errors and behavioral changes) of each implementation. Many of these failures are related to the same bug. We analyzed them and identified 29 bugs related to compilation errors, and 38 bugs related to behavioral changes (Column Bugs). JDolly and SAFEREFACTOR were useful for detecting bugs that have not been revealed so far. Additionally, many of the bugs that we found are also bugs in Netbeans 6.7 (we manually checked them).

5. Related work

Schäfer et al. [Schäfer et al. 2008] propose a Rename refactoring implementation. It is based on the name binding invariant: each name should refer to the same entity before and after the transformation. Furthermore, Schäfer et al. [Schäfer et al. 2009b, Schäfer et al. 2010] propose a number of Java refactoring implementations using an enriched language. Despite their technique turns easier to implement the transformation itself, it is necessary an additional effort to translate the program back. As correctness criterion, they proposed other invariants such as control flow and data flow preservation. We evaluated ten of these implementations using our technique. In the sample used in our evaluation, the JRRT outperformed Eclipse with respect to refactoring correctness: we found 22 bugs in JRRT and 45 bugs in Eclipse.

Daniel et al. [Daniel et al. 2007] proposed an approach for automated testing refactoring tools. They propose a program generator called ASTGen. It allows developers to guide the program generation by extending Java classes. Our generator allows this by specifying Alloy constraints. To evaluate the refactoring correctness, they implemented six oracles that evaluate the output of each transformation. While their oracles could only syntactically compare the programs to detect behavioral changes, SAFEREFACTOR generates tests that do compare program behavior. They found one bug related to behavioral change (we found 28 bugs). Moreover, both techniques found the same number of bugs related to compilation errors (17).

Other approaches have proposed refactoring specifications [Borba et al. 2004, Tip et al. 2003, Silva et al. 2008, Steimann and Thies 2009]. They analyze various aspects of Java, as accessibility, types, name binding, data flow, and control flow. However, proving refactoring correctness for the whole Java language is considered a grand challenge [Schäfer et al. 2009a]. We propose a more practical approach for detecting behavioral changes using a tool support, regardless the kind of refactoring.

6. Conclusions

We presented an approach and its implementation (SAFEREFACTOR) for improving safety of the refactoring activity. By using SAFEREFACTOR, developers can have more confidence that a refactoring was correctly applied. We used it in transformations with up to 100KLOC, and it was useful for reporting behavioral changes that were not detected by IDEs and existing test suites. We also proposed an approach that helps refactoring tool developers to test their implementations. This approach evaluates an implementation by automatically generating programs (test inputs), and checking the results with SAFEREFACTOR. It was useful for identifying 67 bugs in 22 refactoring implementations of state-of-the-art industrial and academic refactoring tools.

Acknowledgment

We would like to thank Tiago Massoni, Paulo Borba, Augusto Sampaio, and David Naumann. This work was partially supported by the National Institute of Science and Technology for Software Engineering (INES)[2], funded by CNPq grants 573964/2008-4, 477336/2009-4 and 304470/2010-4.

[2] http://www.ines.org.br

References

[Borba et al. 2004] Borba, P., Sampaio, A., Cavalcanti, A., and Cornélio, M. (2004). Algebraic Reasoning for Object-Oriented Programming. *Science of Computer Programming*, 52:53–100.

[Daniel et al. 2007] Daniel, B., Dig, D., Garcia, K., and Marinov, D. (2007). Automated testing of refactoring engines. In *FSE '07*, pages 185–194.

[Fowler 1999] Fowler, M. (1999). *Refactoring: Improving the Design of Existing Code*. Addison-Wesley.

[Jackson 2006] Jackson, D. (2006). *Software Abstractions: Logic, Language and Analysis*. MIT press.

[Mens and Tourwé 2004] Mens, T. and Tourwé, T. (2004). A survey of software refactoring. *IEEE Transactions on Software Engineering*, 30(2):126–139.

[Schäfer et al. 2010] Schäfer, M., , and de Moor, O. (2010). Specifying and implementing refactorings. In *OOPSLA '10*, pages 286–301.

[Schäfer et al. 2008] Schäfer, M., Ekman, T., and de Moor, O. (2008). Sound and extensible renaming for Java. In *OOPSLA '08*, pages 277–294.

[Schäfer et al. 2009a] Schäfer, M., Ekman, T., and de Moor, O. (2009a). Challenge proposal: Verification of refactorings. In *PLPV '09*, pages 67–72.

[Schäfer et al. 2009b] Schäfer, M., Verbaere, M., Ekman, T., and Moor, O. (2009b). Stepping stones over the refactoring rubicon. In *ECOOP '09*, pages 369–393.

[Silva et al. 2008] Silva, L., Sampaio, A., and Liu, Z. (2008). Laws of object-orientation with reference semantics. In *SEFM '08*, pages 217–226.

[Soares 2010a] Soares, G. (2010a). Making program refactoring safer. In *Student Research Competition at ICSE '10*, pages 521–522.

[Soares 2010b] Soares, G. (2010b). Making program refactoring safer. In *XVII Latin-American Master Thesis Contest CLTM 2010 at CLEI '10*.

[Soares et al. 2009a] Soares, G., Cavalcanti, D., Gheyi, R., Massoni, T., Serey, D., and Cornélio, M. (2009a). SafeRefactor - tool for checking refactoring safety. In *Tools Session at SBES*, pages 49–54.

[Soares et al. 2009b] Soares, G., Gheyi, R., Massoni, T., Cornélio, M., and Cavalcanti, D. (2009b). Generating unit tests for checking refactoring safety. In *SBLP*, pages 159–172.

[Soares et al. 2010] Soares, G., Gheyi, R., Serey, D., and Massoni, T. (2010). Making program refactoring safer. *IEEE Software*, 27:52–57.

[Steimann and Thies 2009] Steimann, F. and Thies, A. (2009). From public to private to absent: Refactoring Java programs under constrained accessibility. In *ECOOP '09*, pages 419–443.

[Tip et al. 2003] Tip, F., Kiezun, A., and Baumer, D. (2003). Refactoring for Generalization Using Type Constraints. In *OOPSLA '03*, pages 13–26.

Sumarização Automática Multidocumento: Seleção de Conteúdo com Base no Modelo CST (*Cross-document Structure Theory*)

Maria Lucia del Rosario Castro Jorge, **Thiago Alexandre Salgueiro Pardo**

Núcleo Interinstitucional de Linguística Computacional (NILC)
Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo
Av. Trabalhador São-carlense, 400 - Centro
Caixa Postal: 668 - CEP: 13560-970 - São Carlos - SP

{mluciacj,taspardo}@icmc.usp.br

Mestrado defendido em Abril de 2010 (com apoio do CNPq e da FAPESP)

Resumo. *Este artigo apresenta a definição, a formalização e a avaliação de estratégias de seleção de conteúdo para sumarização automática multidocumento com base na teoria discursiva CST (Cross-document Structure Theory). A tarefa de seleção de conteúdo foi modelada por meio de operadores que representam possíveis preferências do usuário para a sumarização. Nossos experimentos foram feitos usando um córpus jornalístico de textos escritos em português brasileiro e mostram que o uso da CST melhora a informatividade dos sumários. A abordagem mostra-se nova para a sumarização multidocumento em língua portuguesa por ser a primeira abordagem que explora conhecimento linguístico para esta tarefa, e, ao mesmo tempo, avança o estado da arte ao modelar e explorar de maneira diferenciada o conhecimento fornecido pela CST.*

1. Introdução

O uso e a disponibilidade cada vez maior de tecnologias de comunicação têm provocado um aumento considerável no volume de informação, principalmente on-line. Há muita informação redundante, complementar e contraditória, proveniente de diversas fontes e narradas de diferentes formas e perspectivas. Com o surgimento de meios de interação mais sofisticados e atualmente quase onipresentes, como a web 2.0, por exemplo, esse cenário torna-se ainda mais complexo. Consequentemente, o processamento da informação tem se tornado uma tarefa de difícil execução para o humano e para a máquina. Neste contexto, a sumarização multidocumento pode ser uma tarefa útil.

A sumarização automática multidocumento (SAM) consiste na produção automática de um único sumário (também chamado resumo) a partir de um grupo de textos sobre um mesmo tópico ou sobre tópicos relacionados (Mani, 2001). Na SAM, além de se recuperar a informação mais relevante, deseja-se ser capaz de se priorizar informação especificada pelo usuário (por exemplo, informação contextual sobre um determinado fato/evento), visualizar a evolução de alguns fatos/eventos no tempo, lidar com diferentes estilos de escrita, ordenar eventos e fatos cronologicamente, e manter a coerência e coesão do sumário, dentre vários outros desafios.

Mani e Maybury (1999) sugerem que a sumarização envolve idealmente três tarefas: a análise dos textos-fonte, produzindo-se uma representação completa de seu conteúdo; a transformação desse conteúdo completo em um conteúdo condensado; e, finalmente, a síntese desse conteúdo na forma de sumário, expresso em uma língua natural. Sistemas de SAM que adotam tal abordagem, privilegiando a manipulação e o uso de conhecimento lingüístico sofisticado, são ditos pertencerem à abordagem profunda. Neste trabalho, foca-se especificamente no processo mais importante da etapa de transformação: a seleção de conteúdo dos textos de origem para compor o sumário correspondente. Assume-se que a etapa de análise é realizada previamente e corresponde unicamente à representação dos textos-fonte segundo a teoria/modelo lingüístico-computacional CST (*Cross-document Structure Theory*) (Radev, 2000), de natureza semântico-discursiva. Com base na CST, neste trabalho são exploradas estratégias de seleção de conteúdo relevante em função de preferências de sumarização do usuário. Por fim, a etapa de síntese realiza simplesmente a justaposição do conteúdo selecionado, produzindo o sumário final. A hipótese deste trabalho é que o uso de conhecimento semântico-discursivo fornecido pela CST pode produzir sumários multidocumento melhores.

As estratégias propostas são formalizadas na forma de operadores de seleção de conteúdo, os quais contêm regras de manipulação da informação textual para satisfação dos interesses dos usuários. Nossos experimentos foram feitos usando um córpus jornalístico de textos em português brasileiro, e mostram que o uso da CST melhora a informatividade dos sumários, confirmando assim nossa hipótese.

Entre as principais contribuições deste trabalho tem-se: a primeira investigação de SAM profunda para o português do Brasil; os melhores resultados obtidos até o momento para o português, superando, inclusive, um dos melhores sistemas de sumarização disponíveis na área; formalização de estratégias de seleção de conteúdo, mapeando-se preferências de usuários em características do modelo CST; validação e rico refinamento do modelo CST, tornando-o mais consistente; construção de recursos e ferramentas inéditos para o português do Brasil, como córpus e editores especializados.

A seguir, na Seção 2, introduz-se o modelo CST e apresentam-se os principais trabalhos relacionados. Na Seção 3, definimos e formalizamos nossos operadores de seleção de conteúdo. A avaliação e discussão dos resultados obtidos são apresentadas na Seção 4. Por fim, a Seção 5 faz algumas considerações finais.

2. Trabalhos Relacionados

A CST foi originalmente proposta com um conjunto de 24 relações para explicitar o relacionamento entre as partes de vários textos, por exemplo, relações de equivalência, contradição, subsunção, elaboração, etc. Essas relações representam os fenômenos multidocumento usuais de redundância, complementariedade e contradição entre informações, explorados com detalhes por Castro Jorge (2010). Como exemplos de aplicação destas relações a um grupo de textos, mostram-se, na Figura 1, alguns trechos de textos (de fontes diferentes) relacionados.

Sentença 1	Sentença 2	Relação CST
Ao menos 17 pessoas morreram após a queda de um avião de passageiros na República Democrática do Congo.	Um acidente aéreo na localidade de Bukavu, no leste da República Democrática do Congo (RDC), matou 17 pessoas na quinta-feira à tarde, informou nesta sexta-feira um porta-voz das Nações Unidas.	*Subsumtion* (←) Sentença 2 engloba Sentença 1
A aeronave se chocou com uma montanha e caiu, em chamas, sobre uma floresta a **10 quilômetros** de distância da pista do aeroporto.	Todos morreram quando o avião, prejudicado pelo mau tempo, não conseguiu chegar à pista de aterrissagem e caiu numa floresta a **15 quilômetros** do aeroporto de Bukavu.	*Contradiction* (—) Sentenças 1 e 2 apresentam uma contradição (em negrito)

Figura 1. Exemplos de relações CST

É importante notar que algumas relações têm direcionalidade, enquanto outras não. Nos exemplos, a relação *subsumption* é da Sentença 2 para a Sentença 1 (portanto, da sentença da direita para a da esquerda, representada pela seta ←), pois a 2 engloba a 1; por outro lado, a relação *contradiction* não tem direção (representada por —), pois as sentenças contradizem uma a outra igualmente.

Como parte deste trabalho, o conjunto de relações CST foi refinado para se obter uma melhor formalização e reduzir a ambiguidade de muitas relações do modelo, resultando em um conjunto de 14 relações representativas para a língua portuguesa.

Algumas pesquisas têm utilizado CST para fins de SAM, incluindo o próprio autor do modelo, que propôs uma metodologia de sumarização de 4 etapas, a saber: agrupamento de textos de conteúdo similar, estruturação interna dos textos (via análise sintática, por exemplo), estabelecimento de relações CST (gerando, assim, um grafo em que os nós representam as sentenças e as arestas as relações CST), e seleção de sentenças para compor o sumário. Essa seleção, em particular, constrói um ranque de sentenças em função da relevância de cada uma e, em seguida, coleta as primeiras colocadas do ranque para formar o sumário. O número de sentenças coletadas depende da taxa de compressão informada pelo usuário, i.e., o tamanho do sumário em função do tamanho dos textos (com medição em número de palavras).

Dentre os principais trabalhos na área, destaca-se o de McKeown e Radev (1995), que, utilizando um sistema de extração de informação textual, modelaram o conteúdo textual em *templates*

(estruturas pré-definidas com campos preenchidos com informações textuais) e, em função das relações (no estilo da CST) entre *templates*, produziram os *templates* finais para subsidiar a construção do sumário correspondente. Zhang et al. (2002), por sua vez, propuseram a alteração, por meio do uso de relações CST, de ranques produzidos por sumarizadores superficiais (ou seja, que utilizam pouco ou nenhum conhecimento linguístico), mostrando a melhora significativa da qualidade dos sumários. Otterbacher et al. (2002) investigaram como o uso de relações CST ajuda a melhorar a coesão em sumários multidocumento, fazendo com que sentenças relacionadas pela CST apareçam próximas no sumário. Afantenos et al. (2004), com base na CST, propuseram uma nova classificação de relações entre textos. Os autores dividiram as relações em duas categorias: sincrônicas (um fato/evento sendo narrado por diferentes fontes em um determinado momento) e diacrônicas (fatos/eventos em evolução no tempo). Em seguida, propuseram uma metodologia de sumarização com base em *templates* pré-definidos e ontologias. Apesar de muito interessante, esse trabalho foi apenas teórico.

A seguir, delineamos nossa proposta de seleção de conteúdo com base na CST.

3. Definição e Formalização de Operadores de Seleção de Conteúdo

Formalmente, definimos um operador de seleção de conteúdo como um artefato computacional que processa uma representação de conteúdo previamente fornecida e produz uma versão mais curta contendo as informações mais relevantes segundo os critérios especificados. Em particular, neste trabalho, a representação de conteúdo consiste no conjunto de textos representados segundo o modelo CST. Portanto, os operadores são aplicados após os textos-fonte terem sido analisados segundo a CST (na etapa de análise). Atualmente, tal análise deve ser feita manualmente para a língua portuguesa, já que o primeiro analisador automático ainda se encontra em desenvolvimento (Maziero et al., 2010).

O dado de entrada dos operadores é um ranque inicial das unidades informativas dos textos (que, neste trabalho, são sentenças). Esse ranque é obtido com base no grafo construído a partir do relacionamento CST entre as unidades do texto (daqui em diante, esse grafo será referenciado por grafo CST). Esse ranque inicial deve conter as unidades informativas do texto na ordem de preferência em que devem ser inseridas no sumário final. Quanto mais relevante for a unidade informativa, mais acima no ranque ela deve estar. A função dos operadores é, a partir do ranque inicial e da preferência do usuário, produzir um ranque refinado, de tal forma que as unidades informativas mais relevantes segundo o critério especificado pelo usuário melhorem de posição no ranque e, portanto, ganhem preferência para estar o sumário. Por fim, dada uma taxa de compressão, são selecionadas tantas sentenças do ranque refinado quanto possível para que a taxa seja respeitada.

No ranque inicial, a relevância das unidades informativas depende do número de relações CST que elas apresentam, isto é, unidades com mais relações CST são consideradas mais relevantes. Na Figura 2, mostra-se um exemplo hipotético de um grafo CST e o ranque inicial formado a partir deste.

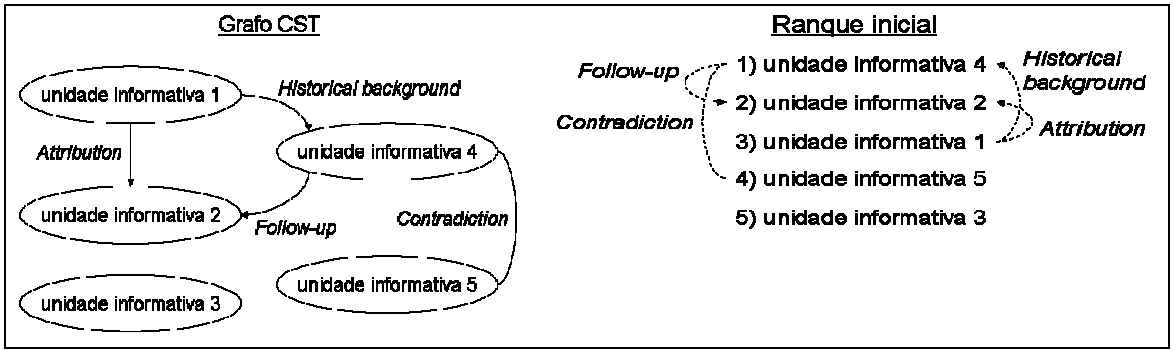


Figura 2. Exemplo de ranque inicial a partir de um grafo CST hipotético

Os operadores de seleção de conteúdo com base na CST estão definidos na forma de *templates*, contendo um conjunto de regras. As regras são especificadas por meio de condições e restrições, as quais, caso sejam satisfeitas, dispararão funções primitivas de manipulação da informação no ranque. Cada regra é definida da seguinte forma: CONDIÇÕES, RESTRIÇÕES $\Rightarrow$ AÇÕES. Cada condição tem o formato CONDIÇÃO(S_i, S_j, Direcionalidade, Relação) e uma dada condição é

satisfeita se existem a relação e a direcionalidade (de S_i até S_j: $\rightarrow$; o caso oposto: $\leftarrow$; ou nenhuma direcionalidade: —) especificadas entre duas sentenças S_i e S_j. As restrições são opcionais, pois representam possíveis requisitos extras para que o operador seja aplicado.

Se todas as condições e restrições forem satisfeitas, então as ações serão aplicadas ao ranque inicial, produzindo assim uma versão refinada do ranque. As ações são definidas em termos de pelo menos uma das três funções primitivas definidas a seguir:

- SOBE(S_i,S_j): a sentença j é colocada em uma posição imediatamente após a sentença i no ranque; é importante notar que a sentença i sempre estará em uma posição superior a sentença j no ranque;
- TROCA(S_i,S_j): trocam-se as posições das sentenças i e j no ranque;
- ELIMINA(S_j): elimina-se a sentença j do ranque.

Para o presente trabalho, definimos e formalizamos operadores que representam possíveis estratégias de seleção de conteúdo. São elas: apresentação de informação de contexto, exibição de informação contraditória, identificação de autoria, tratamento de redundância, e apresentação de eventos que evoluem com o tempo. O processo de construir o ranque inicial também pode ser representado como um operador, no qual a preferência é pela informação principal. Chamamos este último operador de "operador genérico" ou "operador de informação principal".

Cada operador é definido por três campos: um nome de referência, uma breve descrição e um conjunto de regras. Na Figura 3, mostra-se como exemplo o operador para apresentação de informação contextual. Nesse operador, procuram-se por pares de sentenças (ao longo do ranque) que apresentem relações CST do tipo *historical background* e *elaboration*, já que essas relações são as que fornecem informação contextual.

Nome	Operador para apresentação de informação contextual
Descrição	Preferência por informações históricas e complementares
Regras	CONDIÇÃO(S_i, S_j, $\leftarrow$, *Elaboration*) $\Rightarrow$ SOBE(S_i, S_j) CONDIÇÃO(S_i, S_i, $\leftarrow$, *Historical background*) $\Rightarrow$ SOBE(S_i, S_i)

Figura 3. Operador de apresentação de informação de contexto

A aplicação deste operador no ranque inicial da Figura 2 irá produzir o ranque refinado em que a unidade 1 sobe de posição no ranque, sendo realocada logo após a unidade 4 e acima da unidade 2, pois apresenta a relação de *historical background* com a unidade 4.

Após o refinamento do ranque, podem existir sentenças redundantes, por isso, para resolver o problema, faz-se necessário aplicar o operador de tratamento de redundância, o qual elimina sentenças com conteúdo repetido do ranque (indicado por relações de equivalência e subsunção, por exemplo). O ranque refinado, com base na taxa de compressão especificada, indicará as sentenças para compor o sumário final. No texto do exemplo, supondo que 2 sentenças sejam permitidas no sumário, as unidades informativas 4 e 1 seriam escolhidas.

A seguir, apresentamos a avaliação das estratégias de seleção de conteúdo propostas.

4. Experimentos e Resultados

Para avaliar nossos operadores de seleção de conteúdo, construímos um protótipo de um sumarizador multidocumento, ao qual chamamos CSTSumm (*CST SUMMarizer*). Esse protótipo aplica o procedimento explicado na seção anterior.

Para nossos experimentos, usamos um córpus composto de 50 coleções de textos jornalísticos escritos em Português Brasileiro (Aleixo e Pardo, 2008), sendo que cada coleção tem 2 ou 3 textos sobre o mesmo tópico. Esse córpus contém a análise CST (realizada manualmente) de cada coleção de textos e o resumo humano (genérico, sem preferências) correspondente, cujo tamanho corresponde a 30% do tamanho do maior texto da coleção (em número de palavras).

Os sumários automáticos foram produzidos considerando a mesma taxa de compressão dos sumários humanos. Neste trabalho, consideramos dois métodos de avaliação: o automático, que é usado para medir a informatividade dos sumários genéricos**,** e o humano, que é usado para avaliar fatores como a informatividade, coerência, coesão, gramaticalidade e redundância dos sumários com preferências do usuário.

Para a avaliação automática, foi usada a medida ROUGE (Lin e Hovy, 2003). Esta medida produz valores de precisão, cobertura e medida-f (que combina precisão e cobertura), tradicionais na área de pesquisa em questão. Os resultados da ROUGE foram comparados com os resultados obtidos pelo único sumarizador multidocumento superficial conhecido para a língua portuguesa, o GistSumm (Pardo, 2005), e o sumarizador MEAD (Radev et al., 2001), que é um sumarizador multilíngue superficial bastante utilizado na área. Além disso, também estendemos os experimentos aplicando a metodologia proposta por Zhang et al. (2002) aos sumarizadores superficiais anteriores. Assim, as sentenças ranqueadas por esses sumarizadores foram re-ranqueados por meio das relações CST.

Na Tabela 1, são mostrados os resultados da avaliação. Em geral podemos observar que todos os sumários produzidos pelos operadores têm melhores resultados do que o GistSumm e o MEAD em termos da medida-f. Também podemos observar que os desempenhos dos sumarizadores GistSumm e MEAD melhoram com o uso das relações CST, conforme predito por Zhang et al. (2002).

Tabela 1. Resultados das avaliações

	PRECISÃO	COBERTURA	MEDIDA-F
Informação Principal	0.57218	0.52359	0.54384
Tratamento de Redundância	0.55137	0.54539	0.54299
Exibição de Informações Contraditórias	0.57108	0.51974	0.54114
Identificação de Autoria	0.56518	0.52368	0.53994
Apresentação de Eventos que Evoluem no Tempo	0.55136	0.49869	0.52110
Apresentação de Informação de Contexto	0.52079	0.48962	0.50171
GistSumm	0.66435	0.35997	0.45998
GistSumm enriquecido com CST	0.49450	0.50890	0.49940
MEAD	0.52420	0.46020	0.48690
MEAD enriquecido com CST	0.55990	0.49890	0.52300

Já os resultados da avaliação humana mostraram um bom desempenho nos critérios avaliados, sendo que a informatividade é um dos critérios em que melhor desempenho se obteve. Também foi observada uma baixa presença de sentenças redundantes nos sumários finais. A seguir, na Figura 4, mostra-se um exemplo de sumário multidocumento genérico e sem redundância produzido automaticamente pelo CSTSumm a partir de 3 textos.

O Brasil arrasou a Finlândia no primeiro confronto entre as seleções, nesta sexta-feira, na cidade de Tampere, pela Liga Mundial de vôlei 2006, por 3 sets a 0, com parciais de 25/17, 25/22 e 25/21. O resultado de hoje deixou o Brasil perto de conquistar a única vaga do Grupo B da Liga Mundial, que classifica o melhor de cada uma das quatro chaves, a Rússia (país-sede) e mais um time convidado pela Federação Internacional de Vôlei, para a fase final, de 23 a 27 de agosto, em Moscou (Rússia). Brasil e Finlândia se enfrentarão novamente neste sábado, às 12h30 (horário de Brasília), com transmissão ao vivo do canal de TV a cabo SporTV.

Figura 4. Exemplo de sumário multidocumento genérico produzido automaticamente

5. Considerações Finais

Neste trabalho foram definidos, formalizados e avaliados um conjunto de operadores de seleção de conteúdo para SAM com base na CST. Mostramos que o uso da CST permite explorar o conhecimento entre vários textos que versam sobre um mesmo assunto, o que ajuda na seleção de conteúdo, melhorando a informatividade e coerência nos sumários finais.

Por enquanto, além do operador de tratamento de redundância, só permitimos a aplicação de um operador de seleção de conteúdo em um dado momento. Como trabalho futuro, pretende-se estudar a possibilidade do uso de mais operadores por vez. Outros trabalhos futuros incluem a elaboração de novas estratégias de seleção de conteúdo com base na CST.

Na medida do possível, todos os resultados deste trabalho (tanto teóricos como práticos) estão públicos e disponíveis na página do projeto maior do qual este trabalho fez parte: o projeto "sucinto" (www.icmc.usp.br/~taspardo/sucinto).

Ao longo do mestrado foram produzidos 5 artigos. É importante destacar que o artigo publicado no evento STIL 2009 (principal evento de Processamento de Linguagem Natural no Brasil, promovido oficialmente pela SBC) ganhou o prêmio de segundo melhor trabalho do evento. Outro

artigo importante foi o publicado na revista Linguamática, uma das principais revistas da área na comunidade ibero-americana. Destacam-se também os artigos publicados no PROPOR e no workshop da ACL, conferências internacionais importantes na área. A seguir são listados os artigos publicados (também disponíveis na página do projeto citado anteriormente):

- Castro Jorge, M.L.R. e Pardo, T.A.S. (2010). Experiments with CST-based Multidocument Summarization. In the *Proceedings of the ACL Workshop TextGraphs-5: Graph-based Methods for Natural Language Processing*, pp. 74-82. Uppsala/Suécia.
- Maziero, E.G.; Castro Jorge, M.L.R.; Pardo, T.A.S. (2010). Identifying Multidocument Relations. In the *Proceedings of the 7th International Workshop on Natural Language Processing and Cognitive Science* - NLPCS, pp.60-69. Funchal/Madeira, Portugal.
- Jorge, M.L.C. e Pardo, T.A.S. (2010). Formalizing CST-based Content Selection Operations. In the *Proceedings of the 9th International Conference on Computational Processing of Portuguese Language* - PROPOR (Lecture Notes in Artificial Intelligence 6001), pp. 25-29. Porto Alegre/RS, Brasil.
- Castro Jorge, M.L.R. e Pardo, T.A.S. (2010). Estratégias de Seleção de Conteúdo com Base na CST (Cross-document Structure Theory) para Sumarização Automática Multidocumento. *LinguaMÁTICA*, Vol. 2, N. 1, pp. 95-109.
- Castro Jorge, M.L.R. e Pardo, T.A.S. (2009). Content Selection Operators for Multidocument Summarization based on Cross-document Structure Theory. In the *Proceedings of the 7th Brazilian Symposium in Information and Human Language Technology* - STIL, pp. 1-8. São Carlos/SP, Brasil.

Referências

Afantenos, S.D.; Doura, I.; Kapellou, E.; Karkaletsis, V. (2004). Exploiting Cross-Document Relations for Multi-document Evolving Summarization. In the *Proceedings of SETN*, pp. 410-419.

Aleixo, P. and Pardo, T.A.S. (2008). *CSTNews: Um Córpus de Textos Jornalísticos Anotados segundo a Teoria Discursiva Multidocumento CST (Cross-document Structure Theory)*. Série de Relatórios Técnicos do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, N. 326.

Castro Jorge, M.L.R. (2010). *Sumarização automática multidocumento: seleção de conteúdo com base no Modelo CST (Cross-document Structure Theory)*. Dissertação de Mestrado. Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo. São Carlos-SP, Abril, 86p.

Lin, C.Y. and Hovy, E. (2003). Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In the *Proceedings of 2003 Language Technology Conference*. Edmonton, Canada.

Mani, I. (2001). *Automatic Summarization*. John Benjamins Publishing Co. Amsterdam.

Mani, I. and Maybury, M. T. (1999). *Advances in automatic text summarization*. MIT Press, Cambridge, MA.

Maziero, E.G.; Jorge, M.L.C.; Pardo, T.A.S. (2010). Identifying Multidocument Relations. In the *Proceedings of the 7th International Workshop on Natural Language Processing and Cognitive Science* - NLPCS, pp.60-69. Funchal/Madeira, Portugal.

McKeown, K. and Radev, D.R. (1995). Generating summaries of multiple news articles. In the *Proceedings of the 18th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 74-82, Seattle/WA.

Otterbacher, J.C.; Radev, D.R.; Luo, A. (2002). Revisions that improve cohesion in multi-document summaries: a preliminary study. In the *Proceedings of the Workshop on Automatic Summarization*, pp 27-36.

Pardo, T.A.S. (2005). *GistSumm - GIST SUMMarizer: Extensões e Novas Funcionalidades*. Série de Relatórios do NILC. NILC-TR-05-05. São Carlos-SP/Brasil.

Radev, D.R. (2000). A common theory of information fusion from multiple text sources, step one: Cross-document structure. In the *Proceedings of the 1st ACL SIGDIAL Workshop on Discourse and Dialogue*.

Radev, D.R.; Blair-Goldensohn, S.; Zhang, Z. (2001). Experiments in single and multi-document summarization using MEAD. In the *Proceedings of the First Document Understanding Conference*. New Orleans/LA.

Zhang, Z.; Goldenshon, S.B.; Radev, D.R. (2002). Towards CST-Enhanced Sumarization. In the *Proceedings of the 18th National Conference on Artificial Intelligence*.

Um Método Probabilístico para o Preenchimento Automático de Formulários Web a partir de Textos Ricos em Dados

Guilherme Alves Toda[1], **Altigran Soares da Silva (Orientador)**[1]

[1]Departamento de Ciência da Computação – Universidade Federal do Amazonas (UFAM)
Manaus-AM, Brasil
{gat,alti}@dcc.ufam.edu.br

Resumo. *Apresentamos neste artigo um novo método para preencher automaticamente formulários Web utilizando como entrada textos ricos em dados (por exemplo, um anúncio). A partir de tal entrada, nosso método identifica e extrai automaticamente dados de interesse nela contidos e os utiliza para preencher os campos apropriados do formulário. Para essa tarefa, utilizamos o conhecimento obtido a partir de valores utilizados previamente pelos usuários para preencher os formulários. Nosso método, chamado de* iForm*, utiliza características relacionadas ao conteúdo e ao estilo desses valores, que são combinadas através de uma Rede Bayesiana. Mostramos através de experimentos que o iForm atinge resultados superiores na comparação com o método que representa o estado da arte para o problema.*

Palavras-chave: *Formulários Web, Extração de Dados, Recuperação de Informação, Aprendizagem de Máquina.*

1. Introdução

Existem na Web várias aplicações onde usuários provêm dados para serem adicionadas a um banco de dados visando um processamento futuro. Este é o caso de bancos de currículos (p.ex., *CATHO*), bibliotecas digitais (p.ex., *DBLP*, *NDLDT*), sites de leilão (p.ex., *eBay*) e de eventos (p.ex., *DBWORLD*). A solução mais comum nesses casos é disponibilizar formulários compostos por vários campos de entrada de dados, como caixas de texto, caixas de seleção, listas de seleção, caixas de marcação, etc. Diferente dos formulários de busca, os formulários Web de entrada de dados normalmente apresentam um grande número de campos. Isso torna a tarefa de preenchimento por parte dos usuários trabalhosa e sujeita a erros. Este é um problema frequente em sites de comércio eletrônico como *eBay*, *amazon.com* e *TodaOferta.com* que utilizam formulários para usuários registrarem ofertas. Nesses sites podem existir diferentes formulários dependendo do produto oferecido, tornando ainda mais complicado o problema.

Neste artigo, propusemos, implementamos e avaliamos um novo método para resolver o problema de preencher automaticamente formulários de entrada de dados na Web. Nosso método, chamado de *iForm*, consiste em selecionar automaticamente segmentos de um texto rico em dados fornecido por um usuário, e preencher o formulário associando estes segmentos aos campos apropriados. Estes segmentos são identificados a partir do aprendizado de características relacionadas ao conteúdo e ao estilo dos valores previamente submetidos para cada campo, utilizando-se para isso um arcabouço probabilístico. Os usuários podem então verificar o preenchimento, realizar correções e então prosseguir com a sua submissão. Feito isso, os novos valores atribuídos a cada campo

são armazenados e usados como uma evidência extra quando novos textos de entrada são fornecidos.

Por se basear nas características dos campos do formulário e não dos textos de entrada, nosso método é bastante flexível, podendo lidar com textos de diversos estilos e estruturas. Para desenvolvedores, utilizar nosso método não requer nenhum esforço extra além de construir o formulário. Através da experimentação em um conjunto de dados reais, mostramos que nosso método é viável e efetivo, superando o estado da arte [T. Kristjansson et. al. 2004] encontrado na literatura para o problema.

O trabalho de mestrado aqui apresentado deu origem a um pôster apresentado em 2009 na principal conferência mundial da área de Web, a *International World Wide Web Conference* [G. A. Toda et. al. 2009] e a um artigo completo a ser apresentado em setembro de 2011 na principal conferência mundial da área de Bancos da Dados, a *Very Large Data Bases (VLDB) Conference*. O artigo já foi publicado em 2010 no periódico associado à conferência [G. A. Toda et. al. 2010].

2. Trabalhos Relacionados

Várias abordagens propostas na literatura recente têm tratado o problema de prover alternativas mais intuitivas que às interfaces baseadas em formulário para usuários interagirem com bancos de dados na Web. Essas soluções vão desde a estruturação de consultas baseadas em palavras-chave [F. Mesquita et. al. 2007] até o processamento de consultas em linguagem natural [Al-Muhammed and Embley 2007].

Especificamente para o problema de preenchimento automático de formulários Web, Kristjansson et. al. [T. Kristjansson et. al. 2004] propuseram um método baseado no modelo CRF (Condition Random Fields) [J. D. Lafferty et. al. 2001], atualmente o estado da arte em extração de dados de fontes textuais. O CRF se baseia em características do contexto em que ocorrem os valores a serem usados no preenchimento. Entre estas características estão o posicionamento e o sequenciamento dos valores, as quais são aprendidas por meio de um treino prévio sobre um conjunto representativo de textos de entrada manualmente rotulados. Na dissertação, este método foi experimentalmente comparado com o iForm.

3. O Método iForm

O método iForm consiste em estimar a probabilidade de um campo do formulário para cada segmento extraído do texto. Considere um texto de entrada I, composto por $N > 0$ símbolos (palavras). Seja S_{ab} um segmento, ou seja, uma sequência de símbolos em I que inclui os símbolos $t_a, t_{a+1}, \ldots, t_{b-1}, t_b$ ($0 < a \leq b \leq N$). Consideramos S_{ab} como um valor adequado para o campo F se a probabilidade do campo dado esse segmento é maior que um limiar ϵ. Para isso, consideramos que os segmentos tem um tamanho máximo L[1].

O princípio por trás de nosso método é utilizar as características dos valores usados anteriormente para preencher cada campo do formulário quando um novo texto é dado como entrada. Consideramos dois tipos de características: (1) os valores em si e os símbolos que compõem esses valores, as quais chamamos de *características de conteúdo*; e (2) o estilo, ou seja, o uso de letras maiúsculas ou minúsculas, pontuação, etc., que cha-

[1]Em nossos experimentos L não é maior que 10

mamos de *característica de estilo*. Vale ressaltar que nenhuma característica derivada do texto de entrada foi considerada em nosso método.

Com relação às características de conteúdo, consideramos que um segmento de texto pode ser usado para preencher um campo se seu conteúdo é similar ao conteúdo usado previamente para preencher esse campo. Para isso, consideramos tanto os símbolos que compõem o segmento quanto o segmento como um todo. Essas características são capturadas através de funções de probabilidade como descrito a seguir.

Calculamos a probabilidade de um campo F_j do formulário $\mathcal{F}$ dado um segmento S_{ab} com base nos símbolos que compõem o segmento utilizando a Eq.1,

$$TAF(F_j, S_{ab}) = \eta \sum_{\tau \in tokens(S_{ab})} \frac{\texttt{freq}(\tau, F_j)}{\sum_{F_i \in \mathcal{F}} \texttt{freq}(\tau, F_i)} \quad (1) \qquad \eta = \frac{1}{k + |avg(F_j) - k|} \quad (2)$$

onde a função $tokens(S_{ab})$ retorna o conjunto de símbolos em S_{ab}, F_j é um campo em $\mathcal{F}$ e $\texttt{freq}(\tau, F_i)$ retorna a frequência do símbolo τ em um campo F_i, considerando os valores submetidos anteriormente pelo usuário do formulário $\mathcal{F}$. Assim, calculamos a probabilidade entre cada símbolo do segmento e um campo F_j dividindo a frequência de τ nos valores submetidos pelos usuários a F_j pela frequência de τ em todos os campos do formulário. A intuição por trás dessa modelagem é que quanto mais concentradas forem as ocorrências anteriores de um símbolo em um campo, maior é a probabilidade do campo ser relacionado a tal símbolo. A constante η é um fator de normalização cujo valor é dado pela Eq.2 onde k é o número de símbolos em S_{ab} e $avg(F_j)$ é o número médio de símbolos nos valores submetidos como entrada para o campo F_j em interações anteriores do usuário com o formulário. Assim, segmentos com tamanho menor que o número médio de símbolos do campo são penalizados. Isso faz com que valores extraídos apresentem um tamanho compatível com os valores típicos de seus respectivos campos.

Calculamos também a probabilidade do segmento S_{ab} representar um valor de F_j com base na frequência do segmento inteiro nos campos, ao invés de comparar a frequência dos símbolos, dividindo a frequência de S_{ab} nos valores submetidos pelos usuários a F_j pela frequência de S_{ab} em todos os campos do formulário, conforme a equação abaixo.

$$VAF(F_j, S_{ab}) = \frac{\texttt{freq}(S_{ab}, F_j)}{\sum_{F_i \in \mathcal{F}} \texttt{freq}(S_{ab}, F_i)} \quad (3)$$

Utilizamos $TAF(F_j, S_{ab})$ e $VAF(F_j, S_{ab})$ para estimar a probabilidade de F_j dado S_{ab}, com base no *conteúdo* dos valores conhecidos de F. Esta probabilidade é combinada com a probabilidade de F_j dado S_{ab} com base nas características de estilo, descritas a seguir.

Com relação às características de conteúdo, consideramos como relevante para determinar quando um segmento pode ser usado para preencher um campo, a similaridade entre a forma como são escritos os símbolos do segmento e a forma como são escritos os valores submetidos ao campo. Seja SV_j o conjunto de valores submetidos anteriormente ao campo F_j. Para tratar as características relacionadas ao estilo, utilizamos um *Hidden Markov Model (HMM)*[V. Borkar et. al. 2001] $SM(F_j)$, chamado de *Modelo de Estilo de Valores* (MEV), que captura o estilo de escrita dos valores em SV_j. Para gerar um MEV, primeiro separamos cada valor de SV_j em símbolos (palavras) com

base nos espaços entre eles. Em seguida, utilizando uma taxonomia similar à proposta em [V. Borkar et. al. 2001], codificamos esses valores como sequências de máscaras, representadas por expressões regulares. Chamaremos de *Sequências de Máscaras de Símbolo* (SMS).

Um grafo representando o MEV $SM(F_j)$ é gerado com todas as SMS encontradas em valores submetidos anteriormente ao campo F_j. Em $SM(F_j)$, cada nó é representado por uma *máscara de símbolo* que ocorre em SV_j. As arestas são formadas por pares ordenados $\langle n_x, n_y \rangle$ tais que o nó n_x é seguido por n_y em alguma sequência de máscara de símbolos codificada nos valores SV_j. Assim, identificamos cada sequência de máscara de símbolos como um *caminho* em $SM(F_j)$. Além disso, o nó inicial do grafo contém arestas para os primeiros valores representados no grafo e o nó final é ligado por arestas que denotam a probabilidade de serem as últimas máscaras de seus respectivos valores.

Usando a abordagem de *probabilidade máxima* [V. Borkar et. al. 2001], o *peso* de cada aresta em $SM(F_j)$ é calculado como:

$$w(SM(F_j), n_x, n_y) = \frac{\text{\# de pares } \langle n_x, n_y \rangle \text{ em } SM(F_j)}{\text{\# de pares } \langle n_x, n_z \rangle, \forall n_z \in SM(F_j)}. \quad (4)$$

Portanto, o *peso* de uma aresta ligada aos nós n_x e n_y indica a probabilidade de uma *máscara de símbolo* n_x ocorrer seguida de uma máscara n_y.

Para realizar o casamento de um segmento S_{ab} com os valores de SV_j de acordo com o seu estilo de escrita, primeiro codificamos S_{ab} em uma SMS utilizando a taxonomia apresentada acima. Então, computamos a probabilidade da SMS de S_{ab} representar um *caminho* no grafo $SM(F_j)$, como:

$$style(SM(F_j), S_{ab}) = \frac{\sum_{\langle n_x, n_y \rangle \in caminho(S_{ab})} w(SM(F_j), n_x, n_y)}{\text{\# de arestas em } caminho(S_{ab})} \quad (5)$$

onde $caminho(S_{ab})$ retorna o caminho percorrido no MEV $SM(F_j)$ associado ao segmento S_{ab} correspondente à SMS de S_{ab}.

Modelamos a computação da probabilidade do campo F_j dado o segmento S_{ab}, $P(F_j|S_{ab})$, por meio de um modelo de Rede de Crença Bayesiana que é usada para combinar $TAF(F_j, S_{ab})$, $VAF(F_j, S_{ab})$ e $style(SM(F_j), S_{ab})$. Por questões de espaço, omitimos aqui os detalhes sobre esta rede, os quais são discutidos em profundidade no texto da dissertação.

Seja C_j o conjunto de segmentos S_{ab} tal que $P(F_j|S_{ab})$ é maior que o limiar ϵ. Dizemos que C_j é um conjunto de *valores candidatos* para o campo F_j. O objetivo é encontrar um mapeamento $\mathcal{M}$ entre os valores candidatos e os campos no formulário tal que a probabilidade agregada tenha o valor máximo, satisfazendo as seguintes condições: (1) apenas um segmento é associado a cada campo e (2) os segmentos selecionados não sofrem sobreposição, isto é, nenhum símbolo t_i pode ser utilizado para preencher dois ou mais campos. Isto pode ser feito seguindo o procedimento de duas fases descrito a seguir.

Na primeira fase, iniciamos pelo cálculo dos valores candidatos para cada campo F_j, baseado apenas nas características relacionadas ao conteúdo. Seja $\mathcal{I}$ um conjunto composto pela união dos conjuntos dos valores candidatos C_j para todos os campos F_j.

Referimos a $\mathcal{I}$ como o *mapeamento inicial*, o qual contém os pares segmento-campo $\langle S_{ab}, F_j \rangle$. Assumimos que dois pares em $\mathcal{I}$ são *conflitantes* se eles violam alguma das condições acima.

Para encontrar a solução ótima precisamos encontrar todos os possíveis subconjuntos – um número exponencial. Na prática, utilizamos uma heurística simples para encontrar um solução aproximada. Primeiro, extraímos o par com a maior probabilidade de $\mathcal{I}$ e verificamos se ela apresenta conflito com algum par em $\mathcal{I}$. Se o par não for conflitante, o adicionamos ao mapeamento final. Repetimos esse processo até que todos os pares em $\mathcal{I}$ forem extraídos. Isso marca o fim da primeira fase.

Na segunda fase, se algum campo continua não mapeado por um segmento, utilizamos as probabilidades derivadas das características relacionadas ao estilo para tentar encontrar outras associações e assim calcular a probabilidade de cada campo dado um segmento. Então repetimos o processo de mapeamento, agora considerando apenas os pares dos segmentos e os campos que não foram mapeados na primeira fase.

4. Experimentos

Para verificar a eficácia do iForm, foram executados diversos experimentos com o método utilizando dados reais de vários domínios diferentes. Neste artigo nos limitamos a apresentar os resultados de um destes experimentos, onde fazemos uma avaliação comparativa com o método proposto em [T. Kristjansson et. al. 2004]. Como já mencionado, este método é baseado no modelo CRF [T. Kristjansson et. al. 2004], estado da arte em extração dados. Implementamos esse método a partir de uma publicação genérica do CRF [2] que utiliza as mesmas características descritas em [J. D. Lafferty et. al. 2001]: características de dicionário, função de cálculo da média das palavras e características de transição. Utilizamos para o experimentos um conjunto de 150 anúncios de emprego disponíveis na coleção RISE[3], onde os segmentos a serem extraídos se encontram manualmente rotulados. Separamos aleatoriamente um subconjunto de 100 destes anúncios para treinar o CRF e simular submissões anteriores no iForm. Um outro subconjunto disjunto de 50 anúncios foi usado para executar os testes. Esse processo foi repetido 5 vezes. Esses resultados estão na Tabela 1 mostrando os valores da medida-F por campo, considerando a média das 5 execuções.

Campo	iForm	CRF	Campo	iForm	CRF
Estado	**0,85**	0,81	*Formação Desejada*	**0,57**	0,37
Cidade	**0,70**	0,65	*Aplicação*	**0,82**	0,37
Linguagem	**0,84**	0,69	*Área*	0,18	**0,23**
País	0,77	**0,87**	*Empregador*	**0,44**	0,22
Formação Necessária	0,31	**0,59**	*Empresa*	**0,41**	0,17
Plataforma	**0,47**	0,38	*Salário*	0,22	**0,25**

Tabela 1. Comparação de resultados por campo

O iForm obteve valores de medida-F superiores em nove campos, enquanto que o método baseado em CRF superou o iForm em apenas quatro campos. Estes valores estão indicados na tabela pelos números em negrito. A baixa qualidade dos resultados

[2] `http://crf.sourceforge.net/`
[3] `http://www.isi.edu/info-agents/RISE/index.html`

obtidos com CRF pode ser explicada pelo fato dos segmentos a serem extraídos de textos de entrada típicos, como anúncios de emprego, podem não aparecer em um contexto regular, o que é um requisito importante para o CRF. Para o caso do iForm, esse contexto é menos importante, pois nosso método depende apenas das características relacionados aos *campos* ao invés de depender das características dos textos de entrada. Além disso, o iForm foi desenvolvido para explorar essas características relacionadas ao campo das submissões anteriores. Note-se que para aplicar o CRF a esse problema, um treinamento intensivo dos dados de exemplos representativos de texto de entrada é necessário.

5. Conclusões

Neste artigo apresentamos um método chamado iForm para o problema de preenchimento automático de formulários Web a partir de segmentos extraídos automaticamente de textos ricos em dados. Realizamos uma avaliação experimental que mostram a eficácia do iForm. Mostramos que o método obtêm melhores resultados que um método baseado no modelo CRF, o estado da arte em extração de dados, sendo mais flexível e exigindo menos intervenção por parte do usuário.

Como trabalhos futuros, pretendemos investigar o problema de ajudar usuários a encontrar entre um grande conjuntos de formulários, quais deles são mais apropriados para serem preenchidos dado um texto de entrada. Esse problema necessita de uma solução eficiente e escalável para o cenário da Web. Nesse cenário, pretendemos estender nosso método para lidar também com formulários de consulta. Atualmente, o iForm assume que os valores do texto de entrada são indicações positivas para preencher o formulário. Assim, pretendemos estender nosso método para também lidar com eventuais indicações negativas nos textos de entradas.

Referências

Al-Muhammed, M. and Embley, D. W. (2007). Ontology-based constraint recognition for free-form service requests. In *Proc. of the 23rd Intl. Conf. on Data Engineering*, pages 366–375.

F. Mesquita et. al. (2007). LABRADOR: Efficiently publishing relational databases on the web by using keyword-based query interfaces. *Inform. Proc. and Management*, 43(4):983–1004.

G. A. Toda et. al. (2009). Automatically filling form-based web interfaces with free text inputs. In *Proc. of the 18th Intl. Conf. on World wide web*, pages 1163–1164.

G. A. Toda et. al. (2010). A probabilistic approach for automatically filling form-based web interfaces. *Proc. VLDB Endow.*, 4(3):151–160.

J. D. Lafferty et. al. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proc. 8th Intl. Conf. on Mach. Learning*, pages 282–289.

T. Kristjansson et. al. (2004). Interactive information extraction with constrained conditional random fields. In *Proc. of the 19th Nat. Conf. on Artifical intelligence*, pages 412–418.

V. Borkar et. al. (2001). Automatic Segmentation of Text into Structured Records. In *Proc. ACM SIGMOD Intl. Conf. on Management of Data*, pages 175–186.

Agregação de Tráfego em Redes Ópticas com Multiplexação por Comprimentos de Onda

André Costa Drummond[1], **Nelson Luís Saldanha da Fonseca**[1]

[1]Instituto de Computação – Universidade de Campinas (UNICAMP)

Resumo. *Em redes ópticas com multiplexação por comprimento de onda (WDM), a alocação de banda passante ocorre em múltiplos da capacidade de um comprimento de onda. Para a utilização eficiente da grande capacidade disponível em redes WDM, agrega-se diversos fluxos com pequena demandas de banda em comprimentos de onda. Por outro lado, aplicações emergente em e-Ciência e novas aplicações multimídia demandam banda passante superior à capacidade de um comprimento de onda, requerendo que o fluxo seja transmitido em vários caminhos ópticos, possivelmente por rotas distintas, o que traz novos desafios, inclusive para os mecanismos de agregação de tráfego.*

Esta tese investiga diversos problemas em agregação de tráfego e propõe soluções originais para os mesmos. Propõe-se solução para o problema de coloração de grafo auxiliar para a resolução do problema RWA, através da recente Teoria de Complexidade Parametrizada. Introduzem-se, também, algoritmos para a resolução do problema de agregação dinâmica de tráfego que consideram, parcialmente, a topologia da rede e promovem bloqueio balanceado entre os diversos pares comunicantes na rede. Propõem-se, por último, algoritmos para agregar tráfego quando os fluxos demandam maior quantidade de banda passante do que a capacidade de um canal, através do roteamento por múltiplos caminhos, tanto para cenários envolvendo um único domínio como para cenários envolvendo múltiplos domínios.

Os algoritmos propostos permitem a efetivação de mecanismos de agregação de tráfego que possibilitam o uso eficiente da capacidade da Internet baseada em rede ópticas pelas aplicações emergentes bem como pelas aplicações legadas.

Palavras-chave: Redes ópticas, Sistemas WDM, Agregação de tráfego, RWA.

1. Motivação

Surgiram nos anos 80 as primeiras redes de comunicação óptica que transmitiam informação através de sinais de luz trafegando em fibras ópticas. A capacidade de banda passante dos enlaces ópticos foi expandida, posteriormente, com a introdução da tecnologia de multiplexação por comprimento de onda (WDM), que permitiu a multiplexação de diversos sinais ópticos em uma mesma fibra através da utilização de diferentes comprimentos de onda. Além disso, a introdução de novos elementos comutadores, chamados OXCs, permitiu a comutação de comprimentos de onda entre fibras ópticas; Surgiram, assim, as redes ópticas transparentes, nas quais um fluxo atravessa toda a rede sem que haja comutação eletrônica nos nós intermediários, o que permitiu o barateamento das redes, levando a tecnologia WDM a ser adotada amplamente no núcleo da Internet.

Um caminho óptico é composto por uma rota (sequência de enlaces de fibra) e um comprimento de onda. Para estabelece-lo, é preciso, portanto, resolver o problema

de roteamento e alocação de comprimento de onda (RWA). Atualmente, o estado da arte da tecnologia WDM suporta transmissões com largura de banda de 40 Gbps por comprimento de onda. Em laboratórios, já se atinge taxas de transmissão de 160 Gbps. Em contraste, a grande maioria dos fluxos de pacotes existente na Internet, não demandam mais do que centenas de megabits por segundo. Para reduzir o custo da rede, é necessária a agregação de tráfego de baixa velocidade em caminhos ópticos de forma que o custo com equipamentos de conversão opto-eletrônica seja minimizado. Essa técnica é denominada agregação de tráfego (*traffic grooming*).

Em muitos trabalhos na literatura, assumem-se que as demandas de tráfego da rede são estáticas, sendo modeladas através de uma matriz de tráfego. No entanto, a medida em que as redes ópticas WDM vêm sendo implementadas em redes geograficamente dispersas, em redes metropolitanas e em redes locais, as características do tráfego tendem a se tornar mais dinâmicas. Em tais redes, as demandas de tráfego são apropriadamente modeladas como funções no tempo. Neste contexto, o problema de agregação de tráfego deve ser generalizado, definindo um novo problema de agregação dinâmica de tráfego (*Dynamic Traffic Grooming*).

Com o surgimento dos novos paradigmas de computação tais como grades e nuvens, possibilitou-se uma nova gama de aplicações que não eram possíveis anteriormente, tais como, aplicações da chamada e-Ciência. Essas aplicações demandam processamento e transmissão de dados em grande escala e somente o uso eficiente dos recursos da rede permitirá sua utilização. Os novos desafios impostos por essas aplicações demandam soluções de engenharia de tráfego para o problema de agregação dinâmica de tráfego e RWA que sejam rápidas, robustas e escaláveis.

A técnica de agregação de tráfego tem evoluído como descrito no resumo a seguir. Os artigos [Huang et al. 2006, Dutta et al. 2008] revisam a maioria dos trabalhos de pesquisa recentes em agregação dinâmica de tráfego e descrevem os esforços de pesquisa realizados, ao longo dos últimos anos, em busca de soluções para diversos cenários de redes ópticas WDM. Em [Huang et al. 2006], destaca-se a necessidade da realização de estudos que considerem soluções de baixa complexidade. A literatura de agregação de tráfego tem evoluído do estudo do caso estático de agregação para o caso dinâmico [Dutta and Rouskas 2002], bem como do estudo de redes com topologia em anel ou restritas [Liu and Tobagi 2005, Huang et al. 2006] para topologias em malha [Xin 2007a, Palmieri et al. 2007]. A realização de cenários de roteamento de salto único (*single-hop*) [Xin 2007a, Palmieri et al. 2007] e múltiplos saltos (*multi-hop*) [Xin and Qiao 2003], bem como a mudança dos estudos com cenários de tráfego homogêneo para cenários heterogêneos e mais realistas [Doumith and Gagnaire 2007, Tornatore et al. 2008]. Recentemente, tem surgido trabalhos que consideram soluções de baixa complexidade [Xin 2007b, Jaekel et al. 2007, Ho and Lee 2007].

Com o intuito de avançar o estado da arte descrito anteriormente, esta Tese tem como objetivo a criação de algoritmos eficientes para diferentes aspectos de agregação de tráfego em redes WDM, a fim de que se possa desenvolver mecanismos eficientes que permitam prover apoio ao surgimento em larga escala de novas aplicações com grande demanda de banda passante e que estas possam conviver, de forma eficiente, com as aplicações legadas.

2. Contribuições

A Tese apresenta um estudo abrangente sobre questões em aberto do problema de agregação de tráfego em redes ópticas WDM. Primeiramente (Cap. 4), realizou-se um estudo para obtenção de solução *exata* de baixa complexidade computacional para o problema de alocação de comprimentos de onda que visa tornar o problema RWA tratável para cenários realistas de rede e de tráfego. A busca por tal solução utiliza uma teoria moderna sobre a complexidade de algoritmos (Teoria da Complexidade Parametrizada) bem como resultados recentes de sua aplicação na construção de algoritmos de coloração para a família de grafos cordais. A solução proposta diferencia-se por ser o primeiro estudo na literatura a utilizar essa nova teoria para a resolução prática de problemas RWA.

Um levantamento bibliográfico (Cap. 3) evidenciou as limitações dos algoritmos que representam o estado da arte em agregação de tráfego na literatura, mostrando-se que tais algoritmos são incapazes de balancear o tráfego gerando gargalos na rede. Apresentou-se uma nova solução de baixa complexidade com o intuito de sanar as deficiências apresentadas. O Algoritmo proposto (Cap. 5) é capaz de balancear o bloqueio de forma justa sem onerar o seu desempenho, bem como a complexidade da solução. A investigação foi estendida, buscando preparar os algoritmos apresentados para implantação em cenários reais, de forma a se obter uma solução ainda mais rápida com o menor impacto na qualidade da mesma. O algoritmo proposto destaca-se por ser capaz de reduzir a taxa média de bloqueio na rede em até 3 ordens de grandeza em relação a melhor solução existente na literatura, além de balancear de forma justa as taxas de bloqueio entre os pares de comunicação na rede, o que lhe confere as condições necessárias para seu uso em cenários reais.

Em seguida, em uma nova investigação (Cap. 6), o problema de agregação de tráfego foi analisado, adicionalmente, para cenários nos quais as demandas por conexões possuem requisições de banda passante superiores à capacidade de um comprimento de onda, como ocorre com várias aplicações da e-Ciência. Foram propostos dois algoritmos RWA com diferentes mecanismos de roteamento: um tradicional que tenta alocar múltiplos comprimentos de onda em uma mesma rota; e outra que aplica uma nova abordagem que utiliza múltiplos caminhos. Este último é capaz de melhor balancear o tráfego na rede, o que aumenta a capacidade de alocação de recursos além de tornar as conexões estabelecidas menos vulneráveis à falhas na rede. A abordagem proposta é original e não se tem conhecimento de trabalhos anteriores com essa proposta na literatura. Apesar de não constar na tese, derivou-se a prova de NP-completude para esse problema, atestando-se formalmente sua alta complexidade.

A última investigação realizada (Cap. 7) considera redes ópticas multidomínio com conexões com alta demanda por banda passante e requisitos de QoS, que reflete também os ambientes de redes para aplicações da e-Ciência. Foram propostas soluções de agregação de tráfego específicas para um ambiente multidomínio com alocação antecipada de recursos, e para um cenário no qual o tráfego interdomínio disputa recursos de cada domínio com o tráfego local do mesmo. Os algoritmos propostos implementam um mecanismo de alocação de banda passante ajustável, o que permite um melhor balanceamento do tráfego e também a minimização do impacto do tráfego interdomínio sobre o tráfego local de cada domínio. No cenário multidomínio, os custos tornam-se ainda mais relevantes e portanto buscaram-se soluções capazes de minimizar a necessidade de

realocações de recursos, o que permitiu que o tráfego interdomínio de alta capacidade pudesse compartilhar os recursos de caminhos ópticos previamente reservados com o tráfego local de cada domínio atravessado pelo caminho óptico.

A seguir encontram-se especificadas as contribuições originais da Tese:

- Avaliação crítica do estado da arte de algoritmos de agregação dinâmica de tráfego em redes ópticas WDM;
- Utilização da Teoria da Complexidade Parametrizada na solução do problema de alocação de comprimentos de onda para prover solução exata com complexidade computacional reduzida;
- Um algoritmo baseado em zona para a solução do problema de agregação dinâmica de tráfego capaz de balancear o tráfego na rede de forma justa e com redução de complexidade computacional;
- Avaliação do compromisso entre o desempenho e a complexidade computacional de algoritmos de agregação baseados em zona;
- Um algoritmo para a alocação de multicaminhos ópticos para aplicações com altíssima demanda de banda passante;
- Um algoritmo para a alocação de multicaminhos ópticos em redes multidomínio para aplicações com alta demanda de banda e requisitos de QoS;
- Implementação e disponibilização de um simulador para avaliação de algoritmos de agregação/RWA em redes ópticas WDM.

3. Publicações

As publicações científicas decorrentes da pesquisa realizada durante o doutorado foram amplamente aceitas pela comunidade acadêmica especializada, o que é evidenciado pelo grande número de publicações geradas, totalizando 4 artigos em revistas e 11 artigos em conferências, listados a seguir por capítulo da tese. Além dessas publicações, André Drummond foi co-autor de mais 9 publicações realizadas pelo grupo de pesquisa em redes ópticas do Instituto de Computação.

Cap. 4: Alocação de comprimentos de onda através da teoria da complexidade parametrizada

- Drummond, A. C.; Fonseca N. L. S. da; Gyurek, R.; *A Fixed-Parameter Tractable Algorithm for the Wavelength Assignment in WDM Mesh Networks*. In: IEEE ICC - International Conference on Communications, 2008.
- Drummond, A. C.; Fonseca N. L. S. da; *Um Algoritmo Tratável por Parâmetro-Fixo para a Alocação de Comprimentos de Onda em Redes WDM*. In: SBRC - Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2008.
- **[Revista]** Drummond, A. C.; Fonseca N. L. S. da; *A Fixed-Parameter Tractable Approach for the Wavelength Assignment Problem in Transparent Networks*. In: IEEE Communications Letters, 2008.

Cap. 5: Algoritmos justos e de baixa complexidade para agregação de tráfego

- Drummond, A. C.; Fonseca, N. L. S. da; *On-Line Dynamic Traffic Grooming Algorithms for WDM Mesh Networks. In: IEEE ICC - International Conference on Communications*, 2009.

- Drummond, A. C.; Fonseca, N. L. S. da; *Um Algoritmo Eficiente e Justo para Agregação Dinâmica de Tráfego em Redes WDM*. In: SBRC - Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2009.
- Drummond, A. C.; Fonseca, N. L. S. da; *Fair and Efficient Dynamic Traffic Grooming Algorithm for WDM Mesh Networks*. In: IEEE GLOBECOM - Global Communications Conference, 2009.
- Drummond, A. C.; Fonseca, N. L. S. da; *On the Effectiveness of Expansion Mechanisms in Zone-Based Dynamic Traffic Grooming Algorithms*. In: IEEE ICC - International Conference on Communications, 2010.
- **[Revista]** Drummond, A. C.; Fonseca N. L. S. da; *Um Algoritmo Rápido, Eficiente e Justo para Agregação Dinâmica de Tráfego em Redes WDM*. In: Revista Brasileira de Redes de Computadores e Sistemas Distribuídos, 2009.
- **[Revista]** Drummond, A. C.; Fonseca N. L. S. da; *Fairness in Zone-Based Algorithms for Dynamic Traffic Grooming in WDM Mesh Networks*. In: IEEE/OSA Journal of Optical Communications and Networking, 2010.

Cap. 6: Roteamento multicaminho em redes com requisições de alta capacidade

- Chen, X.; Drummond, A. C.; Jukan, A.; Fonseca, N. L. S. da; *A Multipath Routing Mechanism in Optical Networks with Extremely High Bandwidth Requests*. In: IEEE GLOBECOM - Global Communications Conference, 2009.

Cap. 7: Roteamento multicaminho para provisão eficiente de recursos interdomínio com qualidade de serviço

- Chen, X.; Chamania, M.; Jukan, A.; Drummond, A. C.; Fonseca, N. L. S. da; *On the Benefits of Multipath Routing for Distributed Data-intensive Applications with High Bandwidth Requirements and Multidomain Reach*. In: IEEE Conference on Communication Networks and Services Research, 2009.
- Chen, X.; Chamania, M.; Jukan, A.; Drummond, A. C.; Fonseca, N. L. S. da; *QoS-Constrained Multi-path Routing for High-End Network Applications*. In: IEEE High-Speed Networks Workshop, 2009.
- Drummond, A. C. ; Fonseca, N. L. S. da ; Chen, X. ; Jukan, A.; *Roteamento multicaminho para provisão eficiente de recursos interdomínio com qualidade de serviço*. In: Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2011.
- **[Revista]** Chen, X.; Drummond, A. C.; Jukan, A.; Fonseca, N. L. S. da; *Multipath Routing with Advance Reservation for Scalable Inter-domain Service Provisioning in Optical Networks*. In: Optical Switching and Networking, 2011 (Submetido).

Apêndice A: WDMSim

- Drummond, A. C.; Silva, R. T. R.; Rosa, S. R. A. S.; Fonseca N. L. S. da; *IP over WDM module for the NS-2 Simulator*. In: CAMAD - IEEE ICC - International Conference on Communications, 2008.

4. Software

A necessidade do desenvolvimento de um ambiente de simulação no qual os algoritmos de agregação e RWA propostos pudessem ser implementados e avaliados em diferentes cenários de redes ópticas WDM motivou o desenvolvimento do simulador WDMSim.

O WDMSim é um simulador de eventos discretos para redes ópticas desenvolvido em Java, capaz de simular tráfego com chegadas e partidas de requisições de conexão em uma rede WDM com capacidade de agregação. Para cada chegada, o simulador executa o algoritmo de agregação/RWA avaliado, que é responsável pela aceitação ou bloqueio da chamada. Todas as operações necessárias para a obtenção de informações sobre o estado da rede, a alocação de caminhos ópticos e aceitação ou bloqueio das chamadas são acessíveis através de uma interface simples, o que facilita a adição de novos algoritmos ao simulador. O simulador WDMSim foi disponibilizado para a comunidade através do endereço eletrônico:

`http://www.lrc.ic.unicamp.br/wdmsim`

Referências

Doumith, E. A. and Gagnaire, M. (2007). Impact of traffic predictability on wdm exc/oxc network performance. *IEEE Journal on Selected Areas in Communications*, 25(5):895–904.

Dutta, R., Kamal, A. E., and Rouskas, G. N. (2008). *Traffic Grooming for Optical Networks: Foundations, Techniques and Frontiers*. Springer.

Dutta, R. and Rouskas, G. N. (2002). Traffic grooming in wdm networks: Past and future. *IEEE Network*, pages 46–56.

Ho, Q.-D. and Lee, M.-S. (2007). A zone-based approach for scalable dynamic traffic grooming in large wdm mesh networks. *IEEE Journal of Lightwave Technology*, 25(1):261–270.

Huang, S., Dutta, R., and Rouskas, G. N. (2006). Traffic grooming in path, star, and tree networks: Complexity, bounds, and algorithms. *IEEE Journal on Selected Areas in Communications*, 24(4):66–82.

Jaekel, A., Bari, A., Chen, Y., and Bandyopadhyay, S. (2007). New techniques for efficient traffic grooming in wdm mesh networks. In *Proceedings of IEEE International Conference on Computer Communications and Networks*, pages 303–308.

Liu, H. and Tobagi, F. (2005). Traffic grooming in wdm sonet upsr rings with multiple line speeds. In *Proceedings of IEEE INFOCOM*, pages 718–729.

Palmieri, F., Fiore, U., and Ricciardi, S. (2007). Spark: A smart parametric online rwa algorithm. *Journal of Communications and Networks*, 9(4):368–376.

Tornatore, M., Baruffaldi, A., Zhu, H., Mukherjee, B., and Pattavina, A. (2008). Holding-time-aware dynamic traffic grooming. *IEEE Journal on Selected Areas in Communications*, 26(3):28–35.

Xin, C. (2007a). Blocking analysis of dynamic traffic grooming in mesh wdm optical networks. *IEEE/ACM Transactions on Networking*, 15(3):721–733.

Xin, C. (2007b). Dynamic traffic grooming in optical networks with wavelength conversion. *IEEE Journal on Selected Areas in Communications*, 25(9):50–57.

Xin, C. and Qiao, C. (2003). Performance analysis of multi-hop traffic grooming in mesh wdm optical networks. In *Proceedings of International Conference on Computer Communications and Networks*, pages 237–242.

Computação em Fluxos de Dados para Ambientes Paralelos e Heterogêneos

George Teodoro Renato Ferreira

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte – MG – Brasil
{george,renato}@dcc.ufmg.br

Abstract. *As recentes mudanças no projeto de computadores tranformaram as plataformas de computação de alto desempenho em ambientes complexos equipados com diversos processadores paralelos e heterogêneos – como CPUs e GPUs. A utilização eficaz desses recursos continua sendo um grande desafio, pois essas plataformas adicionam novos compromissos e detalhes no projeto de aplicações. Nesse trabalho, apresentamos um conjunto de técnicas para maximizar a utilização de ambientes heterogêneos, que são desenvolvidas em nível de sistemas de execução e facilitam a exploração eficiente desses ambientes. Os resultados apresentam ganhos de aproximadamente 2× sobre as técnicas de escalonamento existentes na maioria dos experimentos.*

1. Introdução

Os avanços em arquitetura de computadores transformaram as plataformas distribuídas tradicionais de computação de alto desempenho em ambientes hierárquicos e heterogêneos, onde cada nó pode conter vários processadores heterogêneos. Essas mudanças resultaram em enormes benefícios devido aos ganhos na capacidade de computação, mas por outro lado também adicionaram complexidade no desenvolvimento de aplicações. A evolução nas soluções para desenvolvimento de software que execute em ambientes heterogêneos continua assumindo, na maioria dos casos, que o uso apenas de aceleradores é suficiente. Poucos trabalhos dedicaram-se a utilização colaborativa desses processadores [Luk et al. 2009], sendo que os mesmos assumem que as tarefas internas que compõem a execução de uma aplicação têm o mesmo desempenho relativo (*speedup*) entre os dispositivos.

Entretanto, argumentamos que o desempenho relativo na execução das tarefas que compõem a aplicação pode variar conforme as características de cada uma. Ou seja, uma mesma operação executada sobre dados distintos, utilizando CPU e GPU, pode apresentar desempenhos relativos diferentes. Essa observação reabre o problema de escalonamento eficiente em ambientes modernos heterogêneos, pois particionar tarefas sem considerar a adequação de cada uma delas aos processadores disponíveis como feito até então [Luk et al. 2009] pode resultar em subutilização dos recursos. Logo, nesse trabalho, propomos um conjunto de técnicas para execução eficiente em ambientes heterogêneos que consideram essas variações para maximizar a utilização desses ambientes.

As otimizações aqui propostas são validadas no contexto do modelo de programação filtro-fluxo e obtêm ganhos de aproximadamente 2× sobre as estratégias existentes. Foram implementadas com parte integrante de um sistema de suporte à execução, permitindo que diversas

aplicações se beneficiem. É importante ressaltar, no entanto, que as soluções propostas são genéricas e podem ser utilizadas em outros modelos de programação.

A seguir listamos algumas das contribuições desse trabalho: (i) extensão do modelo filtro-fluxo para plataformas paralelas e heterogêneas [Teodoro et al. 2008a, Teodoro et al. 2009a]; (ii) escalonamento de tarefas sensível a variação de desempenho [Catalyurek et al. 2010, Teodoro et al. 2009b]; (iii) política para controle de fluxos de dados em ambientes heterogêneos [Catalyurek et al. 2010, Teodoro et al. 2011, Teodoro et al. 2010]; (iv) metodologia para estimar desempenho relativo de tarefas em processadores heterogêneos [Teodoro et al. 2011, Teodoro et al. 2010]. Dentre as publicações destacamos o artigo completo [Teodoro et al. 2010] que recebeu o prêmio de **Best Student Paper** patrocinado pela Google Inc. na conferência HPDC 2010.

2. Anthill

Anthill [Ferreira et al. 2005] é um ambiente de execução que implementa o modelo de programação filtro-fluxo. Aplicações nesse modelo são desenvolvidas como um conjunto de unidades de processamento, chamadas de filtros, que comunicam-se através de fluxos de dados unidirecionais. Em tempo de execução, diversas instâncias de cada filtro podem ser criadas através do mecanismo de cópias transparentes. O desenvolvimento de aplicações nesse modelo conduz naturalmente a dois tipos de paralelismo: dados e tarefas. O paralelismo de dados se dá pela criação de cópias transparentes de um filtro e divisão dos dados de entrada entre elas. O paralelismo de tarefas, por sua vez, ocorre a partir da execução concorrente dos diversos filtros que compõem o fluxo da aplicação. Essa exploração de paralelismo simultaneamente em múltiplas dimensões favorece a criação de um grande número de tarefas potencialmente independentes, o que permite utilização concorrente de diversas unidades de processamento.

3. Anthill orientado a eventos

Na interface de programação original do Anthill os filtros que compõem aplicação são implementados por meio de três funções: *initFilter*, *processFilter* e *finalizeFilter*, sendo cada uma delas invocada uma vez durante a execução. Assim, o programador tradicionalmente escreve a função *processFilter* como um laço sobre os fluxos de entrada, recebendo e processando dados, até que não exista mais trabalho a ser feito. A estrutura da função de processamento deixa a cargo do programador toda a responsabilidade de lidar com: (i) sincronização no recebimento e envio de dados; e (ii) dependências de dados entre fluxos.

Anthill orientado a eventos [Teodoro et al. 2008a] foi proposto como alternativa à interface de programação tradicional, com intuito de prover uma nova abstração para desenvolvimento de aplicações que solucione os problemas destacados. Nessa interface, o controle de fluxos de dados é movido para o ambiente de execução. Assim, o programador passa a prover apenas as funções de processamento relativas ao domínio da aplicação – tratador de eventos, invocadas quando existirem dados disponíveis para processamento. O sistema, por sua vez, controla recebimento/envio de dados e escalona eventos para execução quando as dependências de dados são satisfeitas. Esse esquema favorece a exploração de ambientes *multi-core* através da execução concorrente de eventos internos dos filtros nos dispositivos disponíveis.

4. Anthill orientado a eventos em ambientes heterogêneos

A capacidade de explorar ambientes heterogêneos se deu com a extensão da interface orientada a eventos [Teodoro et al. 2009a]. Essa nova interface permite ao programador escrever a função que processa eventos para múltiplos dispositivos, como CPU e GPU. O ambiente de execução, por sua vez, verifica quais dispositivos estão disponíveis e dispara a execução de eventos nos processadores para os quais existem tratadores implementados. Isso permite que filtros utilizem diversos dispositivos através da execução concorrente de eventos nesses processadores.

Em Anthill orientado a eventos para ambientes heterogêneos o espaço de decisões sobre escalonamento de eventos é mais abrangente. O assinalamento de eventos em cada filtro é feito sob demanda entre os processadores e eles compartilham uma fila de eventos prontos para executar. A política de escalonamento utilizada é DDFCFS (*demand-driven, first-come, first-served*), que seleciona o evento há mais tempo enfileirado quando um processador fica disponível.

5. Escalonamento de eventos sensível a variação de desempenho

O escalonamento de tarefas em ambientes heterogêneos, como discutido, deve classificar as tarefas conforme o desempenho para cada processador e utilizar essa informação para fazer um assinalamento eficiente das mesmas. A materialização dessas ideias em nosso trabalho é dada através de uma política para escalonamento de eventos chamada *demand-driven dynamic weighted round-robin* (DDWRR) [Teodoro et al. 2009b]. Essa política escalona eventos prontos para executar em um filtro nos processadores utilizados sob demanda durante a execução. Assim, quando um processador torna-se disponível, DDWRR seleciona o evento considerando o desempenho relativo (speedup) estimado dos eventos para cada processador. A estimativa desse speedup acontece durante a execução para cada um dos processadores utilizados, baseando-se em um dispositivo base e através da metodologia discutida na Seção 6.

Finalmente, para cada processador utilizado por um filtro, DDWRR cria uma fila de eventos prontos para executar, ordenada decrescentemente em termos do desempenho relativo previsto em cada dispositivo. Eventos com o mesmo desempenho relativo são enfileirado em ordem crescente de acordo com o melhor desempenho do evento nos outros processadores do sistema. Quando da escolha do evento a ser executado, DDWRR seleciona o primeiro na fila do processador que executará o evento. Também é importante salientar que DDWRR não necessita de um valor exato do desempenho relativo, mas somente de uma estimativa suficiente para ordenar corretamente os eventos.

6. Estimador de desempenho

O assinalamento de tarefas aos processadores é realizado com base em uma estimativa dos desempenhos relativos de cada uma delas. Assim, propomos uma metodologia integrada ao ambiente de execução que automatiza essa tarefa. Esse processo é baseado em duas fases: (i) treinamento, onde a aplicação é executada contra uma carga representativa nos dispositivos existentes (eg. CPU e GPU); (ii) previsão, que estima o desempenho relativo das tarefas nos dispositivos utilizados (feita em tempo de execução). Nessa fase, os k registros mais próximos na base de conhecimento gerada durante a primeira fase são retornados (baseando-se em distância entre os

parâmetros) e o desempenho relativo da nova tarefa é estimada como a média do desempenho relativo dos registros retornados.

A metodologia foi avaliada utilizando seis aplicações com implementação para CPU e GPU: Black-Scholes; N-body; Heart Simulation; k-NN; Eclat; Neuroblastoma - NBIA. Detalhes sobre as aplicações e a avaliação podem ser obtidos no texto completo dessa tese. A avaliação da metodologia proposta obteve precisão surpreendentemente alta, com erro menor que 14% e média de 8,52% para as 6 aplicações. A mesma metodologia foi empregada para estimar tempo de execução na CPU das mesmas tarefas da carga de trabalho anterior, mas os erros nesse caso são muito maiores: máximo de 102,6% e média de 62,6%. Essa avaliação empírica é interessante por pelo menos duas razões: (i) mostra que políticas de assinalamento que baseiam-se em estimativa de desempenho relativo estão em vantagem devido a melhor precisão; (ii) o desempenho relativo pode ser utilizado para estimar tempo de execução de uma aplicação executando em dispositivos diferentes, contudo com melhor precisão.

7. Controle de fluxos de dados em ambientes heterogêneos

A utilização eficiente de ambientes distribuídos heterogêneos é ainda mais desafiador quando as máquinas utilizadas são diferentes - a exemplo de quando algumas são equipadas com CPU e GPU e outras possuem apenas CPU. Nesse contexto, as seguintes questões ganham destaque e devem ser consideradas: (i) desbalanceamento de carga entre as cópias de um filtro que são replicadas em máquinas com processadores diferentes; (ii) necessidade de selecionar os dados que maximizam o desempenho dos filtros no lado do envio, de modo que o receptor utilize os processadores adequadamente.

Esses problemas são abordados através da criação de uma política de comunicação entre filtros que é chamada de *on-demand dynamic selective stream (ODDS)* [Teodoro et al. 2010]. Essa política é implementada baseando-se em dois algoritmos: Algoritmo dinâmico de adaptação de filas (*Dynamic Queue Adaptation Algorithm (DQAA)*) e Algoritmo de seleção de *buffers* de dados (*Data Buffer Selection Algorithm (DBSA)*). Nesse tipo de política sob demanda, sempre que um *buffer* de dados é enviado por um filtro ele é enfileirado no lado do envio, sendo enviado na prática quando uma instância do filtro receptor requisitar dados.

O DQAA é executado no lado do receptor de um fluxo e verifica a taxa na qual cada instância de filtro consome dados (processa eventos), modificando a quantidade de requisições ($streamRequestsSize$) por dados em tempo de execução. O objetivo é manter o menor volume de dados no receptor que seja suficiente para manter a ocupação integral dos processadores da máquina, evitando assim desbalanceamento de carga e subutilização dos recursos. O DBSA, por sua vez, é executado nas cópias do filtro no lado do envio do fluxo e seleciona os dados que maximizarão o desempenho dos processadores no lado da recepção. O algoritmo implementado na seleção é o mesmo empregado para o escalonamento DDWRR.

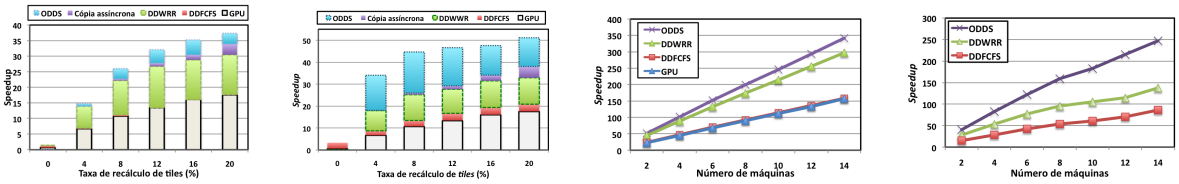
8. Resultados experimentais

Como exemplo utilizamos uma aplicação real (NBIA) usada para a avaliação do prognóstico de neuroblastoma, um tipo de câncer que acomete principalmente crianças. A entrada da aplicação é um conjunto de *slides* de microscópio digitalizadas, que são particionados em pedaços (*tiles*)

menores que podem ser processados independentemente. Os *tiles* são representados em diversas resoluções utilizando uma estratégia piramidal. A análise para cada *tile* começa na resolução mais baixa e é interrompida quando o mesmo for corretamente classificado ou processado na resolução mais alta disponível. NBIA foi executada utilizando 26.742 *tiles* que são representados em duas dimensões: 32×32 e 512×512 $pixels$. Durante a avaliação variamos a taxa de recálculo de $tiles$: percentual de $tiles$ calculados na segunda resolução. É importante ressaltar que outras aplicações foram avaliadas e os resultados são apresentados no texto completo da tese.

Configuração dos experimentos: A avaliação utiliza duas configurações de *clusters*: (i) *cluster* homogêneo com 14 máquinas equipadas com CPU Intel dual-core e GPU NVidia GeForce 8800GT; (ii) *cluster* heterogêneo com as mesmas 14 máquinas, entretanto as GPUs são desligadas em 7 delas. Dessa maneira, criamos heterogeneidade entre as máquinas, pois 7 nós são equipados com CPU e GPU e outros 7 têm apenas a CPU dual-core. ODDS foi comparada a outras duas políticas de comunicação sob demanda, sendo que ambas mantêm um número fixo de requisições de dados entre filtros ligados no fluxo ($streamRequestsSize$): (i) DDFCFS seleciona, no envio, o *buffer* de dados enfileirado há mais tempo e na recepção faz a mesma escolha do próximo evento a processar; (ii) DDWRR seleciona, no envio, o *buffer* de dados enfileirado há mais tempo e na recepção utiliza o escalonamento que observa variações no desempenho relativo. Os *speedups* são calculados com base na versão sequencial.

Resultados: Na Figura 1(a) apresentamos os resultados de desempenho para o caso base do *cluster* homogêneo (uma máquina equipada com CPU e GPU). Primeiramente, nota-se que a utilização de um núcleo CPU adicional em cooperação com a GPU e a política DDFCFS contribui pouco para o desempenho em relação a versão apenas GPU, sendo que DDWRR quase dobra o desempenho da GPU. ODDS superou o desempenho de DDWRR, inclusive o caso base homogêneo, o que deve-se em razão da sua habilidade em selecionar *buffers* de dados que maximizam o desempenho do processador receptor. Isso ocorre mesmo em uma máquina por que os *buffers* são enfileirados no lado do envio para ambas políticas, entretanto, ODDS seleciona os dados que maximizam o desempenho dos processadores receptores no momento do envio.



(a) Base homogêneo. (b) Base heterogêneo. (c) *Cluster* homogêneo. (d) *Cluster* heterogêneo.

Figura 1. Avaliação das políticas de escalonamento - melhor visualizado ao ampliar.

Os resultados no caso base heterogêneo (um computador equipado com CPU dual-core e GPU e o segundo com apenas uma CPU dual-core) são apresentados na Figura 1(b). Primeiramente, quando comparamos os resultados do caso base homogêneo e heterogêneo, vistos nas Figuras 1(a) e 1(b), respectivamente, nota-se que DDFCFS e DDWRR alcançam pequenos ganhos de desempenho com a adição do segundo nó de computação equipado com CPU apenas, enquanto a utilização da segunda máquina e ODDS melhora significativamente o desempenho.

A seguir, apresentamos o desempenho de NBIA à medida em que aumentamos o número de máquinas para cada caso base. No caso base heterogêneo mantivemos a relação de 50% de

máquinas equipadas com CPU e GPU e 50% somente CPU. Na Figura 1(c), apresentamos os *speedups* de NBIA no *cluster* homogêneo. Os resultados mostram que DDFCFS tem quase o mesmo desempenho da versão somente GPU de NBIA, enquanto DDWRR foi capaz de aproximadamente dobrar o desempenho da versão GPU. Adicionalmente, ODDS é 15% mais rápido que DDWRR até mesmo no ambiente homogêneo. Os resultados no caso heterogêneo, Figura 1(d), apresentam ODDS com ganhos ainda maiores — 2× melhor que DDWRR.

9. Conclusões

Esse trabalho aborda o problema de exploração eficiente de ambientes *multi-core* e heterogêneos, utilizando como caso de uso o modelo de programação filtro-fluxo. Os resultados obtidos mostram que a adequada utilização em cooperação de processadores heterogêneos, como CPU e GPU, pode melhorar efetivamente o desempenho de aplicações intensivas em computação. As observações feitas nesse trabalho reabrem a discussão sobre escalonamento em ambientes heterogêneos modernos, pois pouco benefício havia sido alcançado até então com a utilização cooperativa desses dispositivos. As técnicas criadas nesse trabalho também podem ser aplicadas em outros modelos de programação. O software desenvolvido nesse trabalho encontra-se em uso em diversos projetos e pode ser acessado através do seguinte endereço em construção: $http://www.speed.dcc.ufmg.br/anthill/$.

Referências

Catalyurek, U., Ferreira, R., , Sachetto, R., and Teodoro, G. (2010). *Scientific Computing with Multicore and Accelerators*, chapter Dataflow Frameworks for Emerging Heterogeneous Architectures and Their Applications to Biomedicine. Chapman & Hall / CRC Press.

Ferreira, R., Jr., W. M., Guedes, D., Drummond, L., Coutinho, B., Teodoro, G., Tavares, T., Araujo, R., and Ferreira, G. (2005). Anthill:a scalable run-time environment for data mining applications. In *SBAC-PAD*.

Luk, C.-K., Hong, S., and Kim, H. (2009). Qilin: Exploiting parallelism on heterogeneous multiprocessors with adaptive mapping. In *42nd Int. Symp. on Microarchitecture (MICRO)*.

Teodoro, G. (2006). Suporte a fluxos de trabalho de aplicacões intensivas em dados (**Best Master Thesis**). CTD-SBAC/WSCAD. 2006.

Teodoro, G., Fireman, D., Guedes, D., Jr., W. M., and Ferreira, R. (2008a). Achieving multilevel parallelism in the filter-labeled stream programming model. *Inter. Conf. on Parallel Processing (ICPP)*, 0:287–294.

Teodoro, G., Hartley, T. D. R., Catalyurek, U., and Ferreira, R. (2011). Optimizing dataflow applications on heterogeneous environments — accepted for publication. *Cluster Computing: The Journal of Networks, Software Tools and Applications*.

Teodoro, G., Hartley, T. D. R., Catalyurek, U. V., and Ferreira, R. (2010). Run-time optimizations for replicated dataflows on heterogeneous environments. In *Proc. of the 19th ACM Inter. Symposium on High Performance Distributed Computing (HPDC)* - **Best student paper**.

Teodoro, G., Sachetto, R., Fireman, D., Guedes, D., and Ferreira, R. (2009a). Exploiting computational resources in distributed heterogeneous platforms. In *21st Intel Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*.

Teodoro, G., Sachetto, R., Sertel, O., Gurcan, M., Jr., W. M., Catalyurek, U., and Ferreira, R. (2009b). Coordinating the use of GPU and CPU for improving performance of compute intensive applications. In *IEEE Int. Conf. on Cluster Computing*.

Teodoro, G., Tavares, T., Ferreira, R., Kurc, T., Meira, W., Guedes, D., Pan, T., and Saltz, J. (2008b). A run-time system for efficient execution of scientific workflows on distributed environments. *International Journal of Parallel Programming*, 36.

Sobre medidas de desempenho da Internet para o uso em aplicações de redes

Antonio Augusto de Aragão Rocha[1*]
Orientadores: Edmundo A. de Souza e Silva[1]
Rosa M. Meri Leão[1]
Co-orientador: Donald F. Towsley[2]

[1]COPPE/Prog. de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro (UFRJ) – Brasil
{arocha,rosam,edmundo}@land.ufrj.br

[2]Computer Science Department
University of Massachusetts (UMass-Amherst) – USA
towsley@cs.umass.edu

Palavras-chave: avaliação de desempenho, medições em redes, aplicações peer-to-peer, atraso em um sentido, capacidade de transmissão, disponibilidade, tempo de download.

Resumo. *Este artigo apresenta as principais contribuições da tese de doutorado de Antonio Augusto de Aragão Rocha. O trabalho original trata do desenvolvimento de novas técnicas e avaliações de medidas de desempenho para o funcionamento adequado de algumas aplicações da Internet.*

1. Apresentação

A tese de doutorado apresentada em [Rocha, A.A.A. 2010] aborda problemas relacionados à elaboração de novas técnicas e avaliações de medidas de desempenho da Internet, fundamentais para o funcionamento adequado de algumas das mais populares aplicações utilizadas na rede atualmente. Este artigo relata, de forma resumida, as principais contribuições alcançadas no desenvolvimento da tese de doutorado, que foram realizados em duas etapas distintas. A primeira etapa ocorreu, exclusivamente, na Universidade Federal do Rio de Janeiro, sob a orientação dos professores Rosa Maria Meri Leão e Edmundo A. de Souza e Silva. A segunda etapa teve início durante o período de estágio de doutoramento (doutorado sanduíche) do aluno, realizado na Universidade de Massachusetts-Amherst, sob a co-orientação do professor Donald F. Towsley, e se estendeu após o seu retorno ao Brasil. Durante a segunda etapa, o aluno integrou um grupo de pesquisa em aplicações *peer-to-peer* (P2P) daquela universidade e alguns dos trabalhos realizados pelo grupo tiveram participação ativa dos seus orientadores brasileiros.

O restante deste artigo encontra-se estruturados da seguinte forma. Na Seção 2 é apresentada uma visão geral sobre o tema abordado pela tese. A Seção 3 discorre sobre as motivações, objetivos e principais contribuições do trabalho desenvolvido. Por fim, na Seção 4 são listadas as principais publicações obtidas com as contribuições da tese.

2. Visão geral sobre o tema de tese

Os serviços mais populares da Internet deixaram de ser exclusivamente aqueles tradicionais, como correio eletrônico, Web, acesso remoto e transferência de arquivo. Os

[*]Atualmente, professor adjunto do Instituto de Computação da UFF (arocha@ic.uff.br).

usuários, acessando à Internet com taxas de transmissão cada vez mais altas, estão agora interessados também em uma nova gama de serviços como voz sobre IP (VoIP), vídeo sob demanda ou em tempo real, aplicações P2P, jogos "on-line", dentre outros. Ao contrário das aplicações tradicionais que são menos sensíveis ao atraso e mais intolerantes à perda de pacotes, alguns desses novos serviços possuem estreitos requisitos quanto ao desempenho da rede. Por exemplo, para que usuários do Skype possam utilizar o serviço de VoIP oferecidos por esta aplicação de forma satisfatória, a taxa de perda e o atraso dos pacotes dessas aplicações não podem ser muito altos. Do contrário, a qualidade do som e a interatividade da conversa serão insatisfatórias.

As aplicações P2P são, sem dúvida, as mais populares dentre todas da "nova geração". Recentes estudos apresentados por [Schulze and Mochalski 2009] indicam que as aplicações P2P (tais como, BitTorrent, Emule, PPLive e Sopcast) são responsáveis por mais da metade do tráfego gerado atualmente na Internet.

As aplicações P2P revolucionaram o modelo de disseminação de conteúdo na Internet. Os sistemas P2P possuem diversas vantagens em relação ao modelo cliente/servidor e aparecem como principal opção para a distribuição de conteúdo digital que visam as melhorias de desempenho (por exemplo, menor tempo de download e maior disponibilidade), redução de custos para grandes servidores (como economia no consumo de banda) e aumento da escalabilidade. A tendência é que cada vez mais empresas de entretenimento como a CNN, Netflix e Globo utilizem soluções P2P que explorem a capacidade ociosa de seus clientes para auxiliar na disseminação do conteúdo pela Internet.

O princípio adotado no desenvolvimento da arquitetura da Internet foi de um modelo simples e descentralizado de conectividade "fim-a-fim". Embora esse modelo de arquitetura tenha possibilitado o rápido crescimento da popularidade da Internet, essa característica resultou também em limitados serviços oferecidos pelo sistema às aplicações. Algumas dessas limitações são:

- As aplicações não são informadas pela rede a respeito das medidas de desempenho (por exemplo, atraso ou taxa de perda) no caminho entre as duas máquinas;
- As aplicações também não sabem detalhes sobre as características do caminho de rede até a máquina remota. Não têm conhecimento da capacidade de transmissão dos enlaces dos roteadores ao longo do caminho de rede, nem se a máquina remota está conectada à Internet por uma conexão de alta ou baixa capacidade de transmissão, ou mesmo se a largura de banda dos enlaces entre as estações finais satisfazem os requisitos daquela aplicação;
- O serviço oferecido não provê garantias de que os dados serão entregues ao destino ou que os pacotes das aplicações com maior restrição de desempenho terão prioridade.

2.1. Por que é importante medir o desempenho da rede?

Em se tratando de sistemas distribuídos, o desempenho da rede é fundamental para a eficiência do funcionamento de algumas aplicações. Diferentes aplicações exigem distintos requisitos de desempenho da rede. Devido à inexistência na Internet de meios automáticos para garantir a reserva de recursos da rede, ou que ao menos forneçam informações sobre o desempenho da rede, realizar medições e analisar os resultados de desempenho obtidos são tarefas fundamentais para algumas aplicações, além de importante também para usuários e provedores.

Algumas importantes medidas de desempenho da rede são:

- **Atraso em um sentido (ou unidirecional):** tempo que um pacote leva para percorrer um caminho de rede entre a origem e o destino;
- **Capacidade de transmissão:** taxa com que o dispositivo envia dados pela rede;
- **Tempo de download:** tempo necessário para receber por completo um determinado conteúdo (um arquivo, por exemplo);
- **Disponibilidade:** percentual de tempo que determinado serviço fica disponível em relação ao tempo total de observação.

Exemplos práticos do uso de medidas por aplicações, usuários e provedores:

- **Adaptação automática às condições de desempenho da rede:** aplicações multimídia podem, por exemplo, estimar a capacidade de transmissão existente em um caminho de rede e ajustar as taxas de envio de dados ou alterar a codificação de áudio e vídeo de suas transmissões.
- **Escolha de rotas overlay:** Skype e outras aplicações P2P formam redes *overlays* e utilizam máquinas de seus usuários como retransmissores para encaminhar pacotes da aplicação, quando a comunicação direta entre as máquinas originais não é possível ou apresenta qualidade inferior. Como o atraso unidirecional é uma métrica fundamental para a eficiência da interatividade de aplicações VoIP, as escolhas das rotas *overlay* podem levar em consideração essa medida;
- **Criação e validação de modelos:** os resultados de medições são constantemente utilizados para auxiliar analistas na modelagem de sistemas;
- **Verificação de cumprimento dos acordos de serviços:** clientes podem utilizar ferramentas de medições para verificar o cumprimento, por parte dos provedores, dos acordos de serviços (e vice-versa, provedores podem monitorar clientes para conferir cumprimento de contratos);
- **Identificar e implementar soluções para problemas na rede:** é comum o uso de medições por parte dos provedores para identificar eventuais problemas ou pontos de falha na rede. Por exemplo, dispositivos sem-fio, operando a taxas de transmissão muito inferiores, podem comprometer a qualidade dos demais usuários da WLAN;
- **Reduzir custos:** medidas de desempenho como disponibilidade e tempo de *download* são de grande utilidade para provedores que usam os sistemas P2P para distribuição de seus conteúdos. Para conteúdos com alta disponibilidade, provedores podem reduzir seus custos (com diminuição do tráfego em seus enlaces) deixando a tarefa de disseminar o conteúdo por conta dos clientes do sistema P2P, sem afetar o tempo de *download* do usuário.

3. Motivações, objetivos e contribuições da tese

O desenvolvimento de técnicas de medições que permitam conhecer melhor as características da rede e a análise do desempenho de aplicações na Internet sob diferentes perspectivas são dois importantes tópicos de pesquisa, dentro da comunidade de redes na atualidade. Apenas através das medições é possível estimar as características de desempenho da rede. Embora algumas métricas sejam triviais de serem obtidas, outras medidas requerem algoritmos e/ou dispositivos sofisticados para serem estimadas. A análise experimental da operação das aplicações na Internet permite também compreender melhor o estado atual da rede e, possivelmente, melhorar o desempenho do serviço oferecido.

Os objetivos definidos para a tese foram: (i) desenvolver novas técnicas de medições para estimar algumas métricas de desempenho fundamentais para o funcionamento de aplicações em redes; (ii) analisar, por meio de medições em larga escala, o desempenho de aplicações para disseminação de conteúdo na Internet e identificar soluções eficientes para aumentar a disponibilidade do conteúdo e/ou reduzir o custo para os provedores, utilizando aplicação BitTorrent. Esses objetivos foram alcançados com três conjuntos de contribuições. Motivações e detalhes específicos de cada um desses conjuntos de contribuições são descritos a seguir.

3.1. Uma técnica de medição ativa não cooperativa para estimar a média e a variância da distribuição do atraso em um sentido na Internet

Suponha que pacotes sejam gerados a partir máquinas fonte (e.g., A e B) para uma máquina alvo (D, por exemplo). Computar o atraso unidirecional experimentado pelos pacotes na rede não é trivial. A não ser que dispositivos específicos para sincronização de relógios como GPS sejam utilizados, os relógios de duas máquinas diferentes dificilmente possuem taxas de crescimento idênticas e, portanto, o cálculo do atraso unidirecional requer o uso de algoritmos sofisticados para um tratamento especial às diferenças existentes entre os relógios envolvidos na medição [Zhang et al. 2002]. O problema torna-se ainda mais complexo quando o analista não tem acesso à máquina remota (D, neste caso) [Chen et al. 2005]. Isso porque, sem acesso a esta máquina é impossível coletar informações sobre os instantes de chegada dos pacotes para computar o atraso. Portanto, um dos problemas tratados nesta tese aborda a seguinte questão: como estimar d_{AD} e d_{BD}, isto é, o atraso unidirecional sofrido por pacotes enviados a partir de máquinas fonte (A e B) para uma máquina alvo (D), sem privilégios de acesso à máquina alvo?

O primeiro conjunto de contribuições da tese está relacionado ao desenvolvimento de uma nova técnica de medição ativa que contorna ambos os problemas (falta de acesso e falta de sincronismo), permitindo que um analista estime a média e a variância da distribuição do atraso em um sentido. Para lidar com a falta de acesso à máquina remota, foram desenvolvidas duas versões para a técnica proposta. As versões se distinguem quanto ao pré-requisito para a geração das sondas; no entanto, após coletadas as sondas, os algoritmos aplicados são semelhantes. Uma primeira versão requer que o sistema operacional da máquina alvo implemente um contador global para os valores do campo IPID dos pacotes enviados (e.g., MS Windows). Quando a máquina alvo não possui um sistema operacional com IPID global, uma segunda versão da técnica pode ser utilizada. Neste caso, é necessário que ao menos uma das máquinas fonte envolvidas na medição seja capaz de transmitir pacotes com *spoofing* do endereço IP.

3.2. Proposta de uma técnica para estimar a taxa de transmissão de enlaces em uma rede local sem fio IEEE 802.11

As redes locais sem fio (*WLANs*), baseadas nos padrões IEEE 802.11, têm se tornado uma das formas mais populares de acesso à Internet. As taxas de transmissão alcançadas pelos padrões 802.11a/b/g podem variar de valores relativamente altos (54Mbps) até valores significativamente muito baixos (1 ou 2Mbps), dependendo das características do meio de propagação. Quando o usuário de uma aplicação estiver acessando a Internet através de uma rede local sem fio, conhecer a taxa de transmissão na qual está operando o dispositivo pode auxiliar na qualidade do serviço oferecido pela aplicação ao usuário. Em serviços

de mídia contínua, por exemplo, a estimativa desta taxa pode ser utilizada para ajuste da taxa de transmissão do servidor para o cliente multimídia [Nichols et al. 2004].

O segundo conjunto de contribuições da tese refere-se ao desenvolvimento de uma técnica simples e acurada para estimar a taxa de transmissão (capacidade em bits por segundo) do enlace de acesso à Internet, quando este trata-se de uma rede local sem fio IEEE 802.11. A técnica consiste em uma extensão do método tradicional de pares de pacotes, adaptado para computar a dispersão dos pacotes decorrente da capacidade de transmissão do enlace da rede local. A técnica leva em consideração aspectos como o *overhead* causado pelo protocolo 802.11, a existência de tráfego concorrente, a possibilidade de enlaces de capacidade inferior ao longo do caminho de rede e a variação automática da taxa de transmissão do enlace sem fio.

3.3. Estudo de soluções para aumentar a disponibilidade e reduzir o custo na distribuição de conteúdo através de aplicações peer-to-peer na Internet

O uso de sistemas P2P para disseminação de conteúdo tem algumas vantagens em comparação ao método mais tradicional utilizando uma aplicação cliente/servidor. Tais sistemas contam com a capacidade não utilizada dos clientes para possibilitar uma economia de banda do servidor, um tempo menor de download para o usuário e uma maior escalabilidade para a aplicação. No entanto, outras questões são inerentes a essa arquitetura: arquivos pouco populares têm problemas de indisponibilidade no sistema [Ramach et al. 2007] e a disseminação de conteúdos muito populares são extremamente custosos para servidores de conteúdo [Peterson and Sirer 2009]. Experimentos realizados utilizando o BitTorrent nos levaram a duas descobertas no mínimo intrigantes a respeito da disseminação de conteúdo na Internet por meio de sistemas P2P. A primeira é que distribuir arquivos agrupados (por exemplo, todos os arquivos agrupados em um único ZIP ou em um ISO) pode aumentar significativamente a disponibilidade dos arquivos e até mesmo reduzir o tempo total de download de conteúdo. A segunda descoberta é a possibilidade de reduzir a (quase) zero o custo para disseminar conteúdos muito populares, isso sem afetar o desempenho (tempo de download) para o usuário.

O terceiro conjunto de contribuições da tese é: (i) uma análise, através de simulações do protocolo BitTorrent, sobre as implicações da popularidade de um conteúdo na sua disponibilidade, custo para disseminação e desempenho experimentado pelos usuários; (ii) avaliação experimental dos benefícios da prática de agrupamento de arquivos na disseminação de conteúdo, que comprovam a possibilidade de aumentar significativamente a disponibilidade e reduzir o tempo total de download do conteúdo se os arquivos forem distribuídos na forma agrupada; e, (iii) estudo de soluções para reduzir a (quase) zero os custos para disseminar um conteúdo através de sistemas P2P.

4. Resultados de publicações

A técnica para estimar a média e a variância da distribuição do atraso em um único sentido, juntamente com um conjunto limitado dos resultados de simulação, foram publicados em [i]. Este trabalho foi premiado como melhor artigo do SBRC 2006. Uma versão ampliada deste artigo, que incluía resultados de experimentos reais e novos resultados de simulação, foi publicada em [ii]. O método proposto para estimar a taxa de transmissão de uma conexão de rede sem fio IEEE 802.11, com os primeiros resultados experimen-

tais, foram publicados em [iii]. Uma versão estendida incluindo os demais resultados experimentais foi apresentada em [iv].

A análise sobre as implicações da popularidade do conteúdo na disponibilidade entre os usuários, custo de disseminação e desempenho são parte do trabalho publicado em [v]. Uma versão estendida deste trabalho foi apresentada em [vi]. Os resultados de experimentos que comprovam o aumento da disponibilidade e a redução no tempo de *download* de conteúdos disseminados de forma agrupada foram publicados em [viii]. Este artigo foi eleito o melhor artigo da conferência e, por isso, uma versão estendida dele será publicada no IEEE/ACM Transactions on Networking. A análise dos custos da distribuição de conteúdo em função da sua popularidade, assim como a observação da possibilidade do uso de soluções que possibilitem a redução do consumo de banda de provedores para a distribuição de conteúdo via sistemas P2P, foi publicada em [ix].

Principais artigos publicados:

[i] "Estimando a média e variância do atraso em um sentido utilizando o IPID da máquina remota", SBC/SBRC 2006
[ii] "A non-cooperative active measurement technique for estimating the average and variance of the one-way delay", IFIP/Networking 2007
[iii] "Estimando a taxa de transmissão de redes de acesso sem fio através de medições fim-a-fim na Internet", SBC/WPerformance 2006
[iv] "An End-to-End Technique to Estimate the Transmission Rate of an IEEE 802.11 WLAN", IEEE/ICC 2007
[v] "Modeling content availability in peer-to-peer swarming systems", ACM/Performance Evaluation Review 2009
[vi] "Estimating self-sustainability in peer-to-peer swarming systems", Elsevier/Performance Evaluation 2010
[vii] "Modeling chunk availability in P2P swarming systems", ACM/Performance Evaluation Review 2009
[viii] "Content availability and bundling in swarming systems", ACM/CoNEXT 2009 (Versão estendida a ser publicada no IEEE/ACM Transactions on Networking)
[ix] "On P2P System for Enterprise Content Delivery", SBC/SBRC 2009

Referências

Chen, W., Huang, Y., Ribeiro, B., Suh, K., Zhang, H., de Souza e Silva, E., Kurose, J., and Towsley, D. (2005). Exploiting the IPID Field to Infer Network Path and End-System Characteristics. In *Passive and Active Measurement (PAM)*, pages 108–120, Boston, MA, USA.

Nichols, J., Claypool, M., Kinicki, R., and Li, M. (2004). Measurements of the Congestion Responsiveness of Windows Streaming Media. In *International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, pages 189–202, Cork, Irlanda.

Peterson, R. and Sirer, E. (2009). Antfarm: efficient content distribution with managed swarms. In *USENIX symposium on Networked systems design and implementation*, pages 107–122, Boston, MA, USA.

Ramach, A., Sarma, A. D., and Feamster, N. (2007). BitStore: An Incentive-Compatible Solution for Blocked Downloads in BitTorrent. In *Workshop on The Economics of Networked Systems and Incentive-Based Computing*, pages 29–36, San Diego, CA, USA.

Rocha, A.A.A. (2010). *Sobre medidas de desempenho da Internet para o uso em aplicações de redes*. PhD thesis, COPPE/Prog. de Engenharia de Sistemas e Computação - Universidade Federal do Rio de Janeiro.

Schulze, H. and Mochalski, K. (2009). Internet Study 2008/2009. Technical report, Ipoque. `http://www.ipoque.com/resources/internet-studies/internet-study-2008_2009`.

Zhang, L., Liu, Z., and Xia, C. (2002). Clock Synchronization Algorithms for Network Measurements. In *IEEE/Infocom*, pages 160–169, New York, USA.

On the Generalization of Subspace Detection in Unordered Multidimensional Data*

Leandro A. F. Fernandes[1], **Manuel M. Oliveira**[1] **(Advisor)**

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{laffernandes,oliveira}@inf.ufrgs.br

Abstract. *We present a generalized closed-form framework for detecting data alignments in large unordered noisy multidimensional datasets. In our approach, the intended type of data alignment may be a geometric shape (*e.g., *straight line, circle, conic section) or any other structure, with arbitrary dimensionality, that can be characterized by a linear subspace. We also present an extension of our detection scheme to data with Gaussian-distributed uncertainty. The proposed extension makes the framework more robust to the detection of spurious alignments. In contrast to existing solutions, the proposed approach is independent of the geometric properties of the alignments to be detected. Also, it is independent of the type of input data and automatically adapts to entries of arbitrary dimensionality. This allows application of the proposed framework (without changes) in a broad range of applications as a pattern detection tool.*

1. Introduction

A central component of many computer vision and data mining applications is to identify data alignments that emerge as well-defined structures or geometric patterns in datasets. The identification of data alignments is also key in scientific fields such as particle physics and astronomy, because data alignments define strong local coherence in data, and hence, important features to be analyzed. For this reason, automatic detectors have been developed and used both by computer scientists as well as by researcher in many different areas.

The development of automatic detectors has been explored and extended in many ways in order to produce techniques robust to the presence of noise and discontinuities in large datasets. But, traditionally, detectors have been designed for specific types of alignments in a given type of input data. Such a specialization prevents the development of generally applicable techniques and optimizations due to specificities in their formulations. Thus, improvements on existing solutions need to be done on a case-by-case basis.

This work introduces a more general approach for detecting data alignments in unordered noisy multidimensional data. It is focused on fulfill the lack of generality of existing solutions. The proposed approach is based on the observation that a wide class of alignments (*e.g.,* straight lines, planes, circles, spheres, conic sections, among others), as well as input entries, can be represented as linear subspaces. Thus, instead of defining a different detector for each specific case and input data type, it is possible to design a unifying framework to detect the occurrences of emerging subspaces in multidimensional

*L. A. F. Fernandes was supported by CNPq (process 142627/2007-0).

Keywords: subspace detection, parameterization, generalization, Hough transform, geometric algebra.

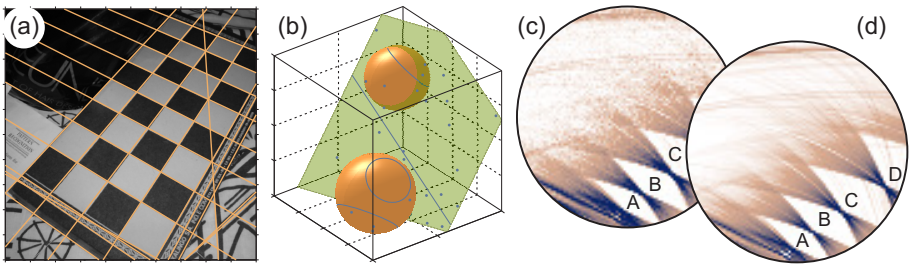


Figure 1. (a) Detection of the 22 lines that best fit the edge pixels of the image. (b) Concurrent detection of plane and spheres. The input dataset is comprised by 43 points, 1 straight line, and 3 circles. The proposed approach was used, without any changes, to automatically detect the structures shown in (a) and (b). (c)-(d) A portion of the accumulator arrays produced for (a) using the sampling-based and the error propagation-based voting schemes, respectively. The blue regions A-D represent peaks of votes (*i.e.,* detected straight lines).

datasets. The versatility of the framework is demonstrated in Figure 1, where it is applied both on a straight line detection case (Figure 1a) as well as on concurrent detection of multiple kinds of alignments with different geometric interpretations, in datasets containing multiple types of data (Figure 1b). Given its general nature, optimizations developed for the proposed framework immediately benefit all the detection cases.

The contributions of this work include: (i) a general framework for subspace detection in unordered multidimensional datasets; (ii) a parameterization scheme for subspaces based on the rotation of a canonical subspace with the same dimensionality; (iii) an algorithm that enumerates all instances of subspaces with a given dimensionality p that either contain or are contained by an input subspace of arbitrary dimensionality; (iv) a procedure that maps subspaces with Gaussian distributed uncertainty to the parameter space characterizing p-dimensional subspaces; (v) a number of experimental evidences supporting that the open affine covering of the Grassmannian (*i.e.,* the set of all p-dimensional linear subspaces of a vector space $\mathbb{R}^n$) can be used as an auxiliary space where the uncertainty of some classes of analytical geometric shapes can be handled in a unified fashion; and (vi) an algorithm that identifies local maxima in a multidimensional histogram.

Due to space limitation, this paper does not present a detailed description of the proposed algorithms or results achieved. The full dissertation [Fernandes 2010] and the list of related publications, courses, pending submissions and implementations [Fernandes and Oliveira 2008, Fernandes and Oliveira 2009, Fernandes and Oliveira 2010, Fernandes and Oliveira 2011] are available in *http://www.inf.ufrgs.br/∼laffernandes/ctd2011*.

2. Geometric Algebra

We have formulated the subspace detector using Geometric Algebra (GA). GA is a powerful mathematical system encompassing many mathematical concepts (*e.g.,* complex numbers, quaternions, and Plücker coordinates) under the same formalism. In GA, subspaces are treated as primitives for computation. As such, it is an appropriate tool for modeling the subspace-detection problem. Also, GA has been proven to be capable of representing many types of geometry. It is because GAs can be constructed over any type of quadratic space, which includes real-valued vector spaces, and also more sophisticated

Hilbert spaces, such as finite Fourier basis, finite random-variable spaces, and basis of orthogonal polynomials, among others. In all cases, the concepts of subspaces, intersections and combinations of subspaces are still valid and independent of the underlying metric space, even though they may not have the same geometric meaning. We explore the generality of these concepts while defining our subspace detection scheme.

By assuming a model of geometry (MOG), one defines the space where data will be encoded and provides a practical (geometric) interpretation to subspaces as input data entries or resulting data alignments. Examples of MOGs successfully encoded by GA include Euclidean, Projective, Spherical, Hyperbolic, and Conic spaces. These MOGs provide practical applications for the proposed technique as a detector of emerging geometric shapes on datasets like, but not limited to, images, volumetric datasets, and point clouds.

Only in the past few years GA became accessible to the computer science researchers through specialized literature. With the aim of disseminating GA within the Brazilian computer science community, the experience obtained from this work has been used in the preparation of courses [Fernandes and Oliveira 2009, Fernandes and Oliveira 2010] for providing an introduction to the fundamental concepts of GA and discussing its great potential as a tool for representing and solving problems in computer graphics, computer vision, and image processing.

3. Related Work

Most of the techniques for detecting data alignments are derived from the Hough Transform (HT), Random Sample Consensus (RANSAC), or Tensor Voting (TV) paradigms. In order to use the HT or RANSAC approaches, one needs to assume a mathematical model for the intended type of data alignment (*e.g.,* the normal equation of the line; the center-radius parameterization of circles) with respect to the expected type of input data (usually points). Although there are partial generalizations of the HT and RANSAC designed to some classes of analytic shapes and HTs for non-analytic shapes, such approaches are still restrictive regarding the assumed input or intended output data. The TV, on the other hand, follows a generalized definition. The TV, however, returns all possible features (with any dimensionality) at the same time. Such a behavior prevents the efficient detection of predefined types of alignments, because it requires a subsequent filtering step.

We propose a voting-based framework for detecting the occurrences of emerging linear subspaces (with a given dimensionality) in multidimensional datasets. The proposed approach is based on the representation of subspaces as primitives in GA (Section 2). By assuming a MOG, such subspaces can be geometrically interpreted as some shape (*e.g.,* straight lines, circles, planes, spheres, among others) or other data alignments (*e.g.,* customer behaviors may emerge as linearly correlated data in e-commerce data). Unlike in conventional HTs, the parameterization used by the proposed approach is independent of the geometric properties of the structure to be detected. When applied to the detection of geometric shapes the proposed framework can be seen as the generalization of the HTs for analytic shapes that can be represented by some linear subspace.

4. Overview of the General Subspace Detection Framework

The proposed subspace detection scheme takes as input a set $\mathcal{X}$ of subspaces (*i.e.,* the input dataset encoded into a MOG), the dimensionality p of subspaces interpreted as the

intended data alignment in the same MOG, and the dimensionality n of the whole underlying space imposed by the MOG. The algorithm outputs the p-dimensional subspaces that best fit the input set $\mathcal{X}$. The detection is performed using a three-step process: (i) create an accumulator array as a discrete representation of the parameter space characterizing p-dimensional subspaces; (ii) perform a voting procedure where the input dataset is mapped to the accumulator array; and (iii) search for the peaks of votes in the accumulator, as they correspond to the p-dimensional subspaces that best fit the input dataset. For the case of uncertain input data, extended mapping and voting procedures are performed in step (ii).

4.1. Parameterization of Subspaces Interpreted as the Intended Data Alignment

In [Fernandes 2010] we show that a p-dimensional subspace in a n-dimensional space can be characterized by a set of $m = p\,(n - p)$ rotations applied to a canonical subspace used as reference, where the values of n and p are related to the MOG where data has been encoded and the type of data alignment one wants to detect, respectively.

By assuming that each one of the m rotation angles (θ_t) related to the sequence of rotation operations are in the $[-\pi/2, \pi/2)$ range, we ensure that such angles define a parameter space for p-dimensional subspaces:

$$\mathbb{P}^m = \{(\theta_1, \theta_2, \cdots, \theta_m) \mid \theta_t \in [-\pi/2, \pi/2)\}, \tag{1}$$

where each parameter vector $(\theta_1, \theta_2, \cdots, \theta_m) \in \mathbb{P}^m$ characterizes an instance of a p-dimensional subspace in the underlying n-dimensional space. The proposed parameterization guarantees the use of the smallest set of parameters in the representation of the intended subspaces. Thus, when applied as a shape detector (Figures 1a and 1b), the proposed approach always represents the intended shapes in the most compact way.

In its first step, the proposed subspace detection framework discretizes $\mathbb{P}^m$, for which an accumulator array is built to receive "votes", and initially set its bins to zero.

4.2. Voting Process for Input Subspaces

The second step maps the input dataset to parameter space. Essentially, the mapping procedure takes each r-dimensional subspace $\mathbf{X}_{\langle r \rangle}$ in the input dataset $\mathcal{X}$ and identifies the parameters (coordinates in $\mathbb{P}^m$, Equation 1) of all p-dimensional subspaces related to it. When $r \leq p$, the mapping procedure identifies in $\mathbb{P}^m$ all p-dimensional subspaces containing $\mathbf{X}_{\langle r \rangle}$. If $r \geq p$, the procedure identifies in $\mathbb{P}^m$ all p-dimensional subspaces contained in $\mathbf{X}_{\langle r \rangle}$. As the input entries are mapped, the bins of the accumulator related to such a mapping are incremented by some importance value of the entry.

In conventional voting-based approaches, such as the HTs, the input data type is known *a priori*. Thus, conventional mapping procedures predefine which parameters of the related parameter vectors must be arbitrated and which ones must be computed. The proposed approach, on the other hand, does not have prior information about input data. It decides at runtime how to treat each parameter. Such a behavior is key for the generality of the proposed detection framework, providing a closed-form solution for the detection of subspaces of a given dimensionality p on datasets that may be heterogeneous and contain elements (*i.e.,* subspaces) with arbitrary dimensionalities ($0 \leq r \leq n$). Such a feature is illustrated by Figure 1b, where the input dataset is comprised by subspaces geometrically interpreted as points, straight line, and circles.

4.3. Voting Process for Input Subspaces with Uncertainty

Experimental (real) data often contain some uncertainty due to imprecision in the instruments used to collect them. Such an uncertainty can be taken into account while performing subspace detection by supersampling input entries according to their distribution of uncertainty and, in turn, by processing each sample with the technique described in Section 4.2. The quality of sampling-based approaches, however, depends on the number of samples, and the computational load increases as more samples are used.

In order to avoid the brute force sampling approach, we propose an extended mapping and voting procedures for input data with Gaussian-distributed uncertainty. The extended mapping procedure is based on first-order error propagation analysis. It transports the uncertainty of each input element throughout the computations into an auxiliary parameter space where the uncertainty is described by a multivariate Gaussian distribution. In turn, such a distribution is mapped to the actual parameter space, leading to non-Gaussian distributions of votes in the accumulator array.

Figures 1c and 1d present a comparison between a portion of the accumulator arrays produced for Figure 1a with a sampling-based voting using the technique described in Section 4.2 and the technique described in the current section, respectively. Notice that error propagation produces smoother distributions of votes than the sampling-based approach. As a result, the latter is less prone to the detection of spurious subspaces.

4.4. Peak Detection

The last step of the subspace detection framework is performed after the voting procedure has been applied to all input data entries $\mathbf{X}_{\langle r \rangle} \in \mathcal{X}$. It consists in identifying the bins that correspond to local maxima in the accumulator array. For this step we propose a sweep-hyperplane-based peak detection scheme developed for accumulator arrays having arbitrary dimensionality. The proposed approach is an extension of the peak detection technique described in [Fernandes and Oliveira 2008] for 2-dimensional accumulator arrays. The technique returns a list with all detected vote peaks, sorted according to their importance (*i.e.,* number of votes). The coordinates of such bins (*i.e.,* parameter vectors) are used to retrieve the most significant p-dimensional subspaces.

5. Results

The proposed approach has been demonstrated by proof of concept implementations of the described algorithms. We have used our own GA library (*i.e.,* Geometric Algebra Template Library, GATL) in such implementations. We intend to make all C++ and MATLAB® code publicly available after the publication of pending submissions.

The implementations have been validated by applying the subspace detection framework to real and synthetic datasets. As a closed-form solution, the same implementation of the proposed framework allows the detection of subspaces that best fit an input set of subspaces with different dimensionalities and different geometric interpretations (*e.g.,* points, straight line and circles – Figure 1b). Also, it allows the concurrent detection of subspaces with different geometric interpretations but with the same dimensionality in a given MOG (*e.g.,* plane and spheres – Figure 1b). Our results have shown that the proposed approach can identify subspaces even in the presence of noise and outliers.

In [Fernandes 2010] we show that an approximation of the dth-order Voronoi diagram of a set of points in $\mathbb{R}^d$ can be retrieved as byproduct of the detection of subspaces geometrically interpreted as circles, spheres, and their higher-dimensional counterparts.

6. Conclusions

We presented a framework for detecting emerging data alignments in unordered noisy multidimensional data. The proposed subspace detector is based on a voting strategy, and it is formulated with GA. By doing so, the technique takes advantage of the conceptual simplicity of the voting paradigm for pattern recognition, while exploring the superior modeling capability of computational primitives and operations in GA.

The time complexity of our approach is the same as of conventional HTs[1] multiplied by p^2, for $r \geq p$: $\mathcal{O}(p^2\,(m-k)\,s^k\,N)$. A naive implementation of our approach suffers from the same drawbacks as HTs: large memory requirement and computational cost. However, as any HT, it is robust to the presence of outliers and is suitable for implementation on massively parallel architectures. Moreover, the generality of our technique should enable new and exciting applications in many different areas, because it avoids tailoring a different solution for each specific case of detection. As a result, we believe it will stimulate research on new optimization approaches for subspace detection. We also hope it will contribute to the popularization of GA among the computer science community.

We have demonstrated the application of the proposed approach on datasets chosen because of their visually-compelling structures. However, one should note that, given its generality, our framework is not restricted to the detection of geometric shapes. It can be applied to any domain in which a problem can be cast as a subspace detection one. For example, the subspace clustering problem in data mining applications. Also, the proposed general parameterization for data alignments may be useful while defining machine learning techniques. Since our approach is independent of the metric properties of the underlying space where data resides, it can be used, without any change, for the detection of subspaces having different interpretations (*e.g.,* different MOGs), *including some that may be defined in the future*.

References

Fernandes, L. A. F. (2010). *On the generalization of subspace detection in unordered multidimensional data*. PhD thesis, PPGC-UFRGS, Porto Alegre, Brazil.

Fernandes, L. A. F. and Oliveira, M. M. (2008). Real-time line detection through an improved Hough transform voting scheme. *Pattern Recognit.*, 41(1):299–314.

Fernandes, L. A. F. and Oliveira, M. M. (2009). Geometric algebra: a powerful tool for solving geometric problems in visual computing. In *Tutorials of Sibgrapi*, pages 17–30.

Fernandes, L. A. F. and Oliveira, M. M. (2011). A general framework for subspace detection in unordered multidimensional data. *IEEE Trans. PAMI*. (submitted).

Fernandes, L. A. F. and Oliveira, M. M. (Jan. 11-22, 2010). Introduction to geometric algebra. Lecture Series of the VISGRAF Laboratory at IMPA, Summer School in CG.

[1]The time complexity of conventional HTs is $\mathcal{O}((m-k)\,s^k\,N)$, where m is the number of parameters, k is the number of arbitrated parameters, s is the number of samples along one dimension of the accumulator array, and N is the number of input entries. It is usually presented as $\mathcal{O}(s^{m-1}\,N)$ in the literature because it is assumed that only one parameter is not arbitrated, *i.e.,* $k = m-1$.

Revisão de Crenças em Lógicas de Descrição e em Outras Lógicas não Clássicas

Márcio Moretto Ribeiro[1]
Renata Wassermann (orientadora)[1]

[1]Instituto de Matemática e Estatística – Universidade de São Paulo (USP)

{marciomr,renata}@ime.usp.br

Resumo. *Revisão de crenças é a área que lida com a dinâmica das bases de conhecimento de agentes racionais. Com o advento da web-semântica tornou-se fundamental o estudo da revisão de crenças em Lógicas de Descrição. Trabalhos recentes mostraram, porém, que a maioria das Lógicas de Descrição não são compatíveis com as técnicas mais usadas de revisão de crenças (o paradigma AGM). Mostramos como adaptar o paradigma AGM para aplicá-lo a outras lógicas não clássicas que incluem diversas Lógicas de Descrição. Apresentamos construções, postulados e teoremas de representação para contração e revisão tanto em bases como em conjuntos de crenças. Além disso, mostramos como usar algoritmos conhecidos da área de depuração de ontologias na construção de determinadas operações e implementamos alguns desses algoritmos.*

1. Introdução

O tema central desse trabalho é revisão de crenças em lógicas não clássicas e, em particular, em Lógicas de Descrição. Considere o seguinte exemplo motivador:

> Desde pequeno João acreditava que alimentos ricos em gordura como margarina e manteiga não são bons para saúde. Em uma aula de biologia ele aprendeu que existem diversos tipos de gordura. A gordura trans presente na margarina é um desses tipos de gordura e é especialmente ruim para a saúde.
> Mais tarde lendo uma revista João descobre a dieta do doutor Atkins. Segundo o Doutor Atkins, tanto a margarina quanto a manteiga podem ser consumidos a vontade, pois não são nocivas a saúde.

Ao lermos o exemplo acima atribuímos crenças ao João. No primeiro momento, por exemplo, atribuímos a ele a crença de que manteiga faz mal a saúde.

A área de revisão de crenças estuda como agentes mudam suas crenças ao receberem novas informações, por exemplo, quando João conheceu a dieta do Dr. Atkins ele mudou suas crenças sobre comidas saudáveis.

Em inteligência artificial, uma *crença* é uma determinada relação entre um agente (e.g.: "João") e uma proposição (e.g.: "alimentos ricos em gordura não são bons para saúde"). Um *agente* é qualquer entidade capaz de perceber o mundo e atuar nele [Russell and Norvig 2003]. Um *agente racional*, segundo Russel e Norvig, é aquele que a cada seqüência de percepções seleciona uma ação que acredita vir a maximizar sua medida de desempenho.

Suponha, por exemplo, que João (um agente racional) acreditava que "Brasília é a capital do Brasil". Ao conhecer a dieta do Dr. Atkins ele pode mudar suas crenças sobre a manteiga ou a margarina, mas seria natural esperar que ele não mude sua crença sobre qual é a capital do país.

A totalidade das crenças que podem ser atribuídas a um agente em um determinado momento é chamada de *estado epistêmico* desse agente. Em revisão de crenças são definidas operações em estados epistêmicos. No exemplo acima primeiramente o estado epistêmico de João contém certas crenças sobre alimentos saudáveis. Quando João aprende sobre gordura trans atribuímos uma nova crença a ele. Ao fazermos isso aplicamos uma operação chamada *expansão* ao estado epistêmico de João. No momento seguinte, quando João conhece a dieta do Dr. Atkins, ele revisa suas crenças. A *revisão* é uma operação que acrescenta novas crenças a um estado epistêmico de forma consistente.

O estudo de revisão de crenças tem dois objetivos centrais. O primeiro é mostrar o que devemos esperar de um agente racional quando este muda suas crenças. O segundo é desenvolver agentes (sistemas) que se comportem de maneira racional ao mudar suas crenças.

Para o primeiro problema, a teoria de revisão de crenças define *critérios de racionalidade* para operações em estados epistêmicos. A minimalidade da mudança é um exemplo de critério de racionalidade: uma operação deve alterar o estado epistêmico de um agente de forma mínima.

Além da revisão e da expansão, a contração é outra operação importante em estados epistêmicos. A contração consiste na remoção de uma crença do estado epistêmico do agente.

O trabalho mais influente da área de revisão de crenças [Alchourrón et al. 1985] define um conjunto de postulados de racionalidade para cada uma das principais operações em estados epistêmicos. Esses conjunto de postulados são conhecidos como *postulados AGM* em homenagem aos autores desse trabalho.

Para o segundo problema, a teoria de revisão de crenças mostra como computar as operações de revisão e contração na base de conhecimento do agente.

A literatura de revisão de crenças é repleta de construções para essas operações. No próprio artigo do trio AGM são apresentadas formas de construir revisão e contração. Os principais resultados de [Alchourrón et al. 1985] (*teoremas de representação*) prova que ambas construções são totalmente caracterizadas pelos postulados AGM.

Um dos trabalhos mais influentes na área de representação de conhecimento é o de Brachman e Levesque [Levesque and Brachman 1987]. Nesse trabalho os autores argumentam por meio de exemplos concretos que existe um balanço entre a expressividade e a complexidade computacional de um formalismo. Portanto, para cada problema devemos tentar usar o formalismo mais adequado.

Nebel em sua tese [Nebel 1990] afirma que existem pelo menos dois importantes critérios de adequação que um formalismo deve satisfazer. O primeiro é a *adequação epistêmica*: um formalismo deve ser capaz de expressar tudo aquilo que for necessário para resolver o problema que se propõe. O segundo é a *adequação heurística*: o formalismo deve ser apropriado para ser usado internamente pelo sistema. Um exemplo

importante desse segundo tipo de adequação é o tempo levado pelo agente para fazer inferências. Um agente que utiliza um formalismo muito expressivo pode levar muito tempo para fazer uma inferência crucial e, portanto, não ser adequado.

Lógicas de Descrição (LDs) são uma família de formalismos usados para representar conhecimento conceitual. A grande vantagem das LDs com relação a seus antecessores (sistemas de frames e redes semânticas) é possuírem semântica bem definida. Diversas LDs com diferentes expressividades e complexidades foram estudadas na literatura e um dos principais desafios da área foi o de encontrar os formalismos mais expressivos em cada classe de complexidade.

Para os propósitos da tese, uma lógica é simplesmente um par $\langle \mathscr{L}, Cn \rangle$ em que $\mathscr{L}$ é a linguagem da lógica e Cn é seu operador de consequência. Ou seja, $\mathscr{L}$ é o conjunto de todas as sentenças admitidas na lógica e Cn é uma função que leva um conjunto de sentenças $B \in 2^{\mathscr{L}}$ ao conjunto de suas consequências $Cn(B)$. Muitos outros aspectos de uma lógica são relevantes, mas de maneira abstrata assumimos apenas que uma lógica seja um par $\langle \mathscr{L}, Cn \rangle$.

A teoria AGM não se aplica a qualquer lógica. Grande parte da literatura da área de revisão de crenças assume que a lógica usada satisfaz certas suposições que chamamos de *suposições AGM*. Essas suposições se aplicam à lógica clássica, mas não se aplicam por exemplo a diversas LDs.

Trabalhos recentes [Flouris 2006, Flouris et al. 2005] mostraram as dificuldades de se aplicar a teoria AGM a LDs. Esses trabalhos mostram que a contração AGM não é nem mesmo definível em determinadas lógicas. A maior parte das LDs são exemplos de lógicas em que a contração AGM não é definível.

O objetivo central da tese é aplicar os resultados da teoria de revisão de crenças às lógicas que não satisfazem as suposições AGM. Nossa abordagem consistiu em estudar propriedades lógicas que garantem determinados resultados de revisão de crenças.

Na seção seguinte apresentamos uma aplicação importante do estudo de revisão de crenças em LDs: a web-semântica. Na última seção enumeraremos as conclusões da tese.

1.1. Ontologias e Web-semântica

Abordamos a aplicação da teoria da revisão de crenças em lógicas não clássicas. Em particular estamos interessados nas LDs. O interesse principal nas LDs está na sua capacidade de representar ontologias.

Uma definição bastante aceita de ontologia é a de Gruber [Gruber 1993]:

> "uma ontologia é uma especificação formal e distribuída de uma conceitualização de um domínio"

Ou seja, uma ontologia explicita conceitos e relações abstratas por meio de uma linguagem formal.

Em computação, o estudo de ontologias ganhou destaque com o desenvolvimento da *web-semântica*. A web-semântica é descrita como uma possível evolução da web em que as informações disponíveis seriam compreensíveis tanto por seres humanos (como a web de hoje) quanto por máquinas.

O papel das ontologias na web-semântica seria o de fornecer um vocabulário comum entre agentes na web. Ontologias seriam responsáveis por declarar os termos usados por cada agente e explicitar como esses termos se relacionam semanticamente. Desta forma, seria possível uma comunicação mais rica entre agentes na web.

Lógicas de Descrição são uma família de formalismos (lógicas) usados para representar conhecimento conceitual (ontologias). O estudo das LDs foi marcado pela busca de um balanço satisfatório entre expressividade e complexidade computacional. Além disso, o fato de possuírem semântica bem definida tornou-as candidatas ideais para representar ontologias na web. A linguagem padrão para representar ontologias na web (OWL) tem como base determinadas LDs.

2. Conclusões da Tese

Diversos autores apontam para a importância do estudo da dinâmica das ontologias [Stojanovic 2004, Flouris 2006]. Como a teoria da revisão de crenças há tempos tem estudado a dinâmica das bases de conhecimento, aplicá-la às lógicas usadas para representar ontologias (Lógicas de Descrição) parece uma ideia promissora. Apesar das dificuldades apontadas por Flouris em [Flouris 2006], mostramos que é possível aplicar a teoria de revisão de crenças às LDs. Para tanto precisamos adaptar os postulados e as construções para revisão de crenças. Essa generalização pode ser útil inclusive para aplicar a revisão de crenças em outras lógicas como a lógica de Horn e a lógica intuicionista.

No caso da contração AGM investigamos o que ocorre ao trocarmos o postulado da recuperação pela relevância [Ribeiro and Wassermann 2006, Ribeiro and Wassermann 2009a, Ribeiro et al. 2009]. O principal resultado obtido foi mostrar que esse novo conjunto de postulados é compatível com uma classe maior de lógicas, a saber toda lógica compacta e tarskiana. Em particular esse resultado se aplica a todas as LDs estudadas. Além disso, mostramos que esse conjunto de postulados é mais adequado para caracterizar a contração partial meet.

Ainda nesse tópico, mostramos que em lógicas chamadas de booleanas a relevância e a recuperação são equivalentes, mas para tantas outras isso não vale. Nessa segunda classe de lógicas a relevância e a recuperação são postulados distintos. Portanto, nesses casos o conceito de minimalidade é completamente distinto do conceito de recuperabilidade. Esse resultado é uma releitura de um trabalho importante de Hansson [Hansson 1991].

O problema central em aplicar a revisão de crenças às lógicas não-clássicas está na negação. Diversas lógicas, LDs em particular, não são fechadas por negação. A falta de negação nos impede de usarmos a chamada identidade de Levi que é a base da construção para revisão mais conhecida na área.

Para resolver esse problema, definimos uma construção que não utiliza a identidade de Levi e não depende da negação. Essa construção satisfaz os postulados AGM para revisão. Apresentamos, além disso, um conjunto de postulados que caracteriza essa construção em uma gama ainda mais ampla de lógicas (qualquer lógica distributiva e compacta). Publicamos esse teorema de representação em [Ribeiro and Wassermann 2009a] e publicamos exemplos de LDs que não são fechadas por negação, mas que são distributivas e compactas em [Ribeiro and Wassermann 2010].

Em relação as LDs, aprendemos que elas possuem características bastante peculiares. Muitas delas não são fechadas por negação, não são compatíveis com os postulados AGM e não são distributivas.

Toda a teoria AGM assume que os estados epistêmicos são representados por conjuntos logicamente fechados. A teoria das bases de crenças não faz essa suposição. Estudamos revisão em bases de crenças [Ribeiro and Wassermann 2007, Ribeiro and Wassermann 2009b] em lógicas não clássicas. Definimos seis tipos de revisão junto do conjunto de postulados que as caracterizam. Nenhuma dessas construções depende da definição da negação e pode ser aplicadas a qualquer lógica tarskiana e compacta.

Mostramos que algoritmos usados na área de depuração de ontologias podem ser usados para as construções em bases de crenças. Assim, por um lado temos algoritmos para as operações em bases de crenças. Por outro, a teoria de revisão em bases de crenças pode ser usada como base teórica para estudar esses algoritmos. Além disso, implementamos essas operações como um plug-in para o Protégé[1]. Apresentamos essa implementação em [Ribeiro and Wassermann 2008c]. Esse trabalho tem sido continuado com participação de outros alunos.

Outros três trabalhos foram frutos dessa tese. O primeiro foi um trabalho desenvolvido coletivamente em que implementamos um plug-in para o Protégé que dada uma ontologia retorna a LD mais adequada para representá-la [Kepler et al. 2006]. No segundo, comparamos duas construções usadas na contração em bases de crenças: kernel e partial meet[2] [Ribeiro and Wassermann 2008a]. No último, mostramos que ao relaxar o postulado da inclusão na contração em bases de crenças obtemos diferentes graus de recuperação que podem ser úteis para tratarmos, por exemplo, de exceções [Ribeiro and Wassermann 2008b].

Referências

Alchourrón, C., Gärdenfors, P., and Makinson, D. (1985). On the logic of theory change. *Journal of Symbolic Logic*, 50(2):510–530.

Flouris, G. (2006). *On Belief Change and Ontology Evolution*. PhD thesis, University of Crete.

Flouris, G., Plexousakis, D., and Antoniou, G. (2005). On applying the AGM theory to DLs and OWL. In Yolanda Gil, E. M., Benjamins, V. R., and Musen, M. A., editors, *Proceedings of the 4th International Semantic Web Conference (ISWC 2005)*, pages 216–231, Galway, Ireland. Springer.

Gruber, T. R. (1993). Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In Guarino, N. and Poli, R., editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, Netherlands. Kluwer Academic Publishers.

Hansson, S. O. (1991). Belief contraction without recovery. *Studia Logica*, 50(2):251–260.

[1] http://protege.stanford.edu/

[2] Uma comparação mais rigorosa está sendo feita por alunos do grupo.

Kepler, F., Paz-Trillo, C., Riani, J., Ribeiro, M. M., Valdivia-Delgado, K., de Barros, L. N., and Wassermann, R. (2006). Classifying ontologies. In *Proceedings of the 2nd Workshop on Ontologies and their Applications (WONTO 2006)*, Ribeirão Perto, SP, Brasil.

Levesque, H. J. and Brachman, R. J. (1987). Expressiveness and tractability in knowledge representation and reasoning. *Computational Intelligence*, 3:78–93.

Nebel, B. (1990). *Reasoning and revision in hybrid representation systems*. Springer-Verlag New York, Inc., New York, NY, USA.

Ribeiro, M. M. and Wassermann, R. (2006). First steps towards revising ontologies. In *Proceedings of the 2nd Workshop on Ontologies and their Applications (WONTO 2006)*, Ribeirão Perto, SP, Brasil.

Ribeiro, M. M. and Wassermann, R. (2007). Base revision in description logics - preliminary results. In *Proceedings of the International Workshop on Ontology Dynamics (IWOD 2007)*, Innsbruck, Austria.

Ribeiro, M. M. and Wassermann, R. (2008a). Degrees of recovery and inclusion in belief base dynamics. In Pagnucco, M. and Thielscher, M., editors, *Proceedings of the 12th International Workshop on Non-Monotonic Reasoning (NMR 2008)*, Sidney, Australia.

Ribeiro, M. M. and Wassermann, R. (2008b). On the relation between remainder sets and kernels. In *XV Encontro Brasileiro de Lógica (EBL08)*, Paraty, RJ, Brasil. Resumo espandido.

Ribeiro, M. M. and Wassermann, R. (2008c). The ontology reviser plug-in for protégé. In *Proceedings of the 3rd Workshop on Ontologies and their Applications (WONTO 2008)*, Salvador, BA, Brazil.

Ribeiro, M. M. and Wassermann, R. (2009a). AGM revision in description logics. In *Proceedings of the 1st Workshop on Automated Reasoning about Context and Ontology Evolution (ARCOE 2009)*, Pasadena, California, USA.

Ribeiro, M. M. and Wassermann, R. (2009b). Base revision for ontology debugging. *Journal of Logic and Computation*, 19:721–743. Special Issue: Recent Advances in Ontology Dynamics.

Ribeiro, M. M. and Wassermann, R. (2010). More about agm revision in description logics. In *Proceedings of the 2nd Workshop Automated Reasoning about Context and Ontology Evolution (ARCOE 10)*, Lisboa, Portugal.

Ribeiro, M. M., Wassermann, R., Antoniou, G., Flouris, G., and Pan, J. (2009). Contraction on the semantic web: On the role of relevance and recovery. In *Proceedings of the International Workshop on Ontology Dynamics (IWOD 2009)*, Washington DC, USA.

Russell, S. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Englewood Cliffs, NJ, 2 edition.

Stojanovic, L. (2004). *Methods and Tools for Ontology Evolution*. PhD thesis, University of Karlsruhe, Germany.

Uma Análise Empírica de Interações em Redes Sociais

Fabrício Benevenuto e Virgílio Almeida

[1]Departamento de Ciência da Computação
Universidade Federal de Minas Gerais
Belo Horizonte - Brasil

{fabricio,virgilio}@dcc.ufmg.br

Resumo. *Redes sociais online se tornaram extremamente populares e vêm causando o surgimento e a crescente popularização de uma nova onda de aplicações na Web. Com uma enorme quantidade de conteúdo criado a cada dia e por permitir o registro de diversos tipos de interações entre os usuários, redes sociais online se tornaram alvo de diversas áreas de pesquisa, não só relacionadas à computação, mas também áreas como a física, a sociologia, a economia, a psicologia e a comunicação. Apesar do grande interesse, pouco se sabe sobre o comportamento dos usuários e suas formas de interação em redes sociais online. Essa tese investiga as diferentes formas de interações em redes sociais, cobrindo aspectos do comportamento e da navegação dos usuários, bem como aspectos relacionados à postagem de conteúdo não solicitado. Para isso, coletamos dados de várias redes sociais atuais em larga escala. Nossas medições revelam padrões de comportamento e interação típicos, que variam de acordo com as mídias utilizadas para compartilhamento de conteúdo e interação. A partir dessas análises de comportamento, um método para a detecção de usuários maliciosos em redes sociais online é proposto.*

1. Introdução

Desde seu início a Internet tem sido o palco de uma série de novas aplicações, incluindo a WWW, aplicações par-a-par (P2P) e email. Hoje em dia a Web tem recebido uma nova onda de aplicações associadas ao crescimento e proliferação de redes sociais e mídia digital. Várias redes sociais surgiram, incluindo redes de profissionais (ex. LinkedIn), redes amizade (ex. MySpace, Facebook e Orkut), e redes voltadas para o compartilhamento de algum tipo específico de conteúdo como blogs (ex. LiveJournal), microblogs (ex. Twitter), fotos (ex. Flickr) e vídeos (ex. YouTube).

Redes sociais têm atraído milhões de usuários. De acordo com uma pesquisa recente da Nielsen Online [Nielsen Online 2009], mídia social passou na frente de email como a atividade online mais popular. Mais de dois terços da população online global visita ou participa de redes sociais ou blogs. Como comparação, se o Facebook fosse um país, seus 600 milhões de usuários registrados o colocariam como o terceiro país mais populoso do mundo [Facebook 2010]. Além da grande popularidade, redes sociais também possuem estatísticas impressionantes em termos de conteúdo gerado por usuários. Usuários do YouTube criam cerca de 10 horas de vídeos a cada minuto [YouTube 2010].

Apesar de tanta popularidade e da enorme quantidade de dados disponível nesses ambientes, o estudo de redes sociais ainda este em sua infância, já que novos problemas,

desafios e oportunidades de pesquisa vêm surgindo continuamente em diferentes áreas. Na computação, redes sociais vêm se tornando um tema chave em pesquisas em várias áreas, incluindo sistemas multimídia, recuperação de informação, mineração de dados e aprendizagem de máquina. Além disso, por oferecem registros de interações e laços de amizades existentes entre usuários, redes sociais permitem que questões de diferentes áreas possam ser quantificadas e abordadas em larga escala. É neste contexto que esta tese está inserida.

Essa tese investiga as diferentes formas de interações existentes em redes sociais online, cobrindo aspectos do comportamento e da navegação dos usuários, bem como aspectos relacionados à postagem de conteúdo não solicitado. Nossa metodologia consiste em coletar dados de redes sociais online em larga escala e caracterizar interações e aspectos do comportamento dos usuários, de forma a obter modelos que descrevem tais comportamentos. Finalmente, um mecanismo de detecção de usuários maliciosos que utiliza os padrões caracterizados é proposto.

As próximas seções deste resumo descrevem algumas contribuições realizadas nessa tese e sumariza as principais publicações científicas realizadas. Finalmente, a seção 6 discute o impacto dessas publicações.

2. Técnicas de coleta de redes sociais em larga escala

No passado o estudo de redes sociais era um domínio de sociólogos e antropólogos, quando ferramentas típicas para obtenção de dados eram pesquisas e entrevistas [Wasserman et al. 1994]. Como consequência, muitos desses esforços foram realizados bases de dados pequenas e pouco representativas. Com o surgimento de redes sociais como Orkut e Facebook, surgiu a oportunidade de estudos sobre redes sociais com o uso de grandes bases de dados. Entretanto, a coleta e o processamento de grandes quantidades de dados de redes sociais muitas vezes requerem o uso de técnicas de introduzam o mínimo viés nos dados coletados. Muitas vezes, para seguirmos uma metodologia científica adequada precisamos coletar o grafo inteiro, o que envolve a coleta de grafos compostos por milhões de nodos e bilhões de arestas.

Existem diferentes estratégias para se coletar dados de redes sociais, que variam desde a construção de coletores distribuídos até mesmo à reconstrução de requisições que passam por um agregador de tráfego (ex. Proxy). Várias técnicas de coleta de redes sociais online foram empregadas ao longo dessa tese. O arcabouço construído foi utilizado em diversos artigos e vem sendo divulgado e compartilhado através de mini-cursos em importantes conferências brasileiras da área de computação como Webmedia'10, SBBD'10, IHC'10 e SBSC'10 [Benevenuto 2010], SBRC [Benevenuto et al. 2011b] e JAI [Benevenuto et al. 2011a].

3. Modelagem do Comportamento dos Usuários

Com o objetivo de estudar como usuários se comportam quando eles se conectam em redes social nós coletamos dados que capturam cada clique realizado por usuários através de um agregador de tráfego. Nossas análises cobrem quatro sistemas: Orkut, MySpace, Hi5 e LinkedIn e provêem uma visão geral do tipo, duração, freqüência e seqüência das atividades realizadas pelos usuários quando eles se conectam a esses sítios Web. Dentre as principais contribuições nesse tópico, podemos citar a modelagem

estatística do processo de chegada de requisições e sessões de usuários nessas redes sociais online. Além disso, nós propomos um modelo baseado em um grafo probabilístico para caracterizar a navegação do usuário de uma atividade para outra (ex. ver foto, postar mensagem, etc.) dentro de diferentes tipos de sessões. Nosso estudo revela várias características do comportamento dos usuários (ex. algumas atividades possuem grandes chances de serem repetidas) que possuem implicações para aspectos relacionados à interface e ao desempenho do sistema. O principal artigo [Benevenuto et al. 2009f] que cobre esses resultados obteve 58 citações de acordo com o Google scholar em um ano e meio (novembro de 2009 a maio de 2011).

4. Caracterização das Interações entre Usuários

Nosso trabalho detalha os diferentes tipos de interações existentes em redes sociais online e caracteriza o uso dessas interações pelos seus usuários. Vários trabalhos foram publicados neste tópico. Em particular, em [Benevenuto et al. 2009f] nós mostramos que 92% das interações dos usuários do Orkut são silenciosas, ou seja, em 92% dos casos os usuários acessaram algum conteúdo de outros usuários sem deixar nenhuma mensagem ou comentário. Curiosamente, cerca de 20% do conteúdo acessado não é conteúdo de amigos imediatos, sendo que a ferramenta de atualizações é a principal responsável por promover interações em usuários que não são imediatamente amigos. Nossos resultados ainda mostram que a maior parte das interações ocorrem em localizações geográficas próximas, o que possui diversas implicações abordagens de distribuição de conteúdo eficientes.

Outras formas de interação exploradas estão relacionadas às interações através de recursos multimídia, em especial vídeos. Vários serviços na Web 2.0 oferecem funções baseadas em vídeos como alternativas para funções que utilizam apenas texto, como revisões e opiniões sobre produtos e propagandas [Shannon 2007]. Sendo assim, nós caracterizamos diversas diferenças entre interações através textuais e interações que ocorrem unicamente através de vídeos. Nosso trabalho caracteriza interações assíncronas através de vídeos resposta no YouTube, revelando diversas diferenças nas interações dos usuários através desses recursos quando comparadas com interações textuais [Benevenuto et al. 2008a, Benevenuto et al. 2009e]. Características geográficas de interações através de vídeos foram exploradas em [Duarte et al. 2007]. Finalmente, em [Benevenuto et al. 2009a, Benevenuto et al. 2010b], foram propostos mecanismos para diferenciar tipos de usuários a partir de seus padrões de navegação em um servidor de vídeos.

5. Identificação de Comportamento Malicioso e Oportunista

A partir dos trabalhos de caracterização foram identificadas diversas formas de comportamentos e interações maliciosas. Sendo assim, nós investigamos mecanismos de detecção de usuários maliciosos a partir de padrões de seus comportamentos.

Inicialmente, abordamos o problema de usuários que postam vídeos spam no YouTube (ex. propaganda, pornografia). Após coletar vídeos e usuários do YouTube de forma apropriada, nós criamos uma grande coleção de usuários rotulados como maliciosos ou não maliciosos e identificamos características desses usuários capazes de distinguir essas duas classes [Benevenuto et al. 2008b, Benevenuto et al. 2008c]. Foram identificados

e caracterizadas 60 métricas com potencial discriminatório, extraídas de características do perfil do usuário no YouTube, de características de seus vídeos e da repercussão que eles tiveram e, finalmente, de características do usuário dentro do grafo social, como amizades recíprocas e amigos conectados entre si. A partir dessas análises criamos um algoritmo baseado em classificação hierárquica para a diferenciação automática de usuários maliciosos e não maliciosos [Benevenuto et al. 2009b, Benevenuto et al. 2009c]. A partir dos métodos propostos foi realizado um pedido de patente nos EUA, que está sob avaliação. Porém, uma patente provisinal já foi concedida [Benevenuto et al. 2009d]. No artigo [Benevenuto et al. 2010c] nós sumarizamos os vários tipos de atividades maliciosas em redes sociais online, em especial redes voltadas para o compartilhamento de vídeos. Finalmente, seguindo uma estratégia semelhante à utilizada para a detecção de spam em vídeo, fomos capazes de identificar spam no Twitter [Benevenuto et al. 2010a].

6. Considerações Finais

Essa tese teve como base vários artigos completos publicados em conferências e revistas importantes, incluindo ACM TOMCCAP, ACM Multimedia, ACM SIGIR, ACM IMC, etc. Apesar de recentes, os trabalhos publicados ao longo dessa tese de doutorado somam mais de 300 citações de acordo com o Google scholar. Dois journals que estendem esses trabalhos estão submetidos e ainda em processo de avaliação (já em estado avançado). Além disso, a partir dos métodos propostos nessa tese para detecção de usuários maliciosos, foi realizado um pedido de patente nos EUA [Benevenuto et al. 2009d] que também está sob avaliação, porém já foi aceito como provisional. Alguns trabalhos publicados tiveram ampla repercussão na mídia, incluindo MTV, revista Info e o jornal "The New York Times" (http://nyti.ms/9NU4fQ).

Cabe ressaltar que essa tese de doutorado tem inspirado projetos de pesquisa [Almeida et al. 2011], além de artigos inseridos no contexto de iniciações científicas [Rodrigues et al. 2009, Rodrigues et al. 2010], dissertações de mestrado [Barbosa et al. 2011, Gomide et al. 2011, Gomide et al. 2011, Cunha et al. 2011] ou mesmo potenciais teses de doutorado [Figueiredo et al. 2011].

Referências

Almeida, J., Gonçalves, M., Prates, R., Hasan, D., Guimaraes, D., Oliveira, D., Belem, F., Figueiredo, F., Langbehn, H., Pinto, H., Lara, R., Ricci, S., and Benevenuto, F. (2011). Greenweb: Melhorando a qualidade da informação na web 2.0. In *Seminário Integrado de Software e Hardware (Semish)*.

Barbosa, G., Santana, I., Gonçalves, G., Prates, R., Benevenuto, F., and Almeida, V. (2011). Characterizing interactions among members of deaf communities in orkut. In *IFIP Conference on Human-Computer Interaction (Interact)*.

Benevenuto, F. (2010). Redes sociais online: Técnicas de coleta, abordagens de medição e desafios futuros. In *Tópicos em Sistemas Colaborativos, Interativos, Multimidia, Web e Banco de Dados*, chapter 2, pages 41–70. Sociedade Brasileira de Computação, Belo Horizonte, Brasil.

Benevenuto, F., Almeida, J., and Silva, A. (2011a). Coleta e análise de grandes bases de dados de redes sociais online. In *Jornadas de Atualização em Informática (JAI)*.

Benevenuto, F., Almeida, J., and Silva, A. (2011b). Explorando redes sociais online: Da coleta e análise de grandes bases de dados às aplicações. In *Mini-cursos do Simpósio Brasileiro de Redes de Computadores (SBRC)*.

Benevenuto, F., Duarte, F., Rodrigues, T., Almeida, V., Almeida, J., and Ross, K. (2008a). Understanding video interactions in YouTube. In *ACM Conference on Multimedia (MM)*, pages 761–764.

Benevenuto, F., Magno, G., Rodrigues, T., and Almeida, V. (2010a). Detecting spammers on twitter. In *7th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*.

Benevenuto, F., Pereira, A., Rodrigues, T., Almeida, V., Almeida, J., and Gonçalves, M. (2009a). Avaliação do perfil de acesso e navegação de usuários em ambientes web de compartilhamento de vídeos. In *Brazilian Symposium on Multimedia Systems and Web (WebMedia)*, pages 149–156.

Benevenuto, F., Pereira, A., Rodrigues, T., Almeida, V., Almeida, J., and Gonçalves, M. (2010b). Characterization and analysis of user profiles in online video sharing systems. *Journal of Information and Data Management*, 1(2):115–129.

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., and Gonçalves, M. (2008b). Detectando usuários maliciosos em interações via vídeos no YouTube. In *Brazilian Symposium on Multimedia Systems and Web (WebMedia)*, pages 138–145.

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., and Gonçalves, M. (2009b). Detecting spammers and content promoters in online video social networks. In *IEEE Int'l conference on Computer Communications Workshops (INFOCOM)*, pages 337–338.

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., and Gonçalves, M. (2009c). Detecting spammers and content promoters in online video social networks. In *Int'l ACM Conference on Research and Development in Information Retrieval (SIGIR)*, pages 620–627.

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., and Gonçalves, M. (2009d). Methods for detecting spammers and content promoters in online video social networks. Patente Provisional USPTO application number 12/286,448. (Patente Provisional).

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., Gonçalves, M., and Ross, K. (2010c). Video pollution on the web. *First Monday*, 15(4).

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., and Ross, K. (2009e). Video interactions in online video social networks. *ACM Transactions on Multimedia Computing, Communications and Applications (TOMCCAP)*, 5(4):1–25.

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., Zhang, C., and Ross, K. (2008c). Identifying video spammers in online social networks. In *Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, pages 45–52.

Benevenuto, F., Rodrigues, T., Cha, M., and Almeida, V. (2009f). Characterizing user behavior in online social networks. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, pages 49–62.

Cunha, E., Magno, G., Comarela, G., Almeida, V., Gonçalves, M., and Benevenuto, F. (2011). Analyzing the dynamic evolution of hashtags on twitter: a language-based approach. In *Int'l Workshop on Language in Social Media (LSM)*.

Duarte, F., Benevenuto, F., Almeida, V., and Almeida, J. (2007). Geographical characterization of YouTube: a latin american view. In *Latin American Web Congress (LAWEB)*, pages 13–21.

Facebook (2010). Facebook Press Room, Statistics. `http://www.facebook.com/press/info.php?statistics`. Acessado em Março/2010.

Figueiredo, F., Benevenuto, F., and Almeida, J. (2011). The tube over time: Characterizing popularity growth of youtube videos. In *ACM Int'l Conference of Web Search and Data Mining (WSDM)*.

Gomide, J., Veloso, A., Jr., W. M., Almeida, V., Benevenuto, F., Ferraz, F., and Teixeira, M. (2011). Dengue surveillance based on a computational model of spatio-temporal locality of twitter. In *ACM SIGWEB Web Science Conference (WebSci)*.

Nielsen Online (2009). Social networks & blogs now 4th most popular online activity, 2009. `http://tinyurl.com/cfzjlt`. Acessado em Março/2010.

Rodrigues, T., Benevenuto, F., Almeida, V., Almeida, J., and Gonçalves, M. (2009). Uma análise contextual de conteúdo duplicado no YouTube. In *Brazilian Symposium on Multimedia Systems and Web (WebMedia)*, pages 141–148.

Rodrigues, T., Benevenuto, F., Almeida, V., Almeida, J., and Gonçalves, M. (2010). Equal but different: A contextual analysis of duplicated videos on youtube. *Journal of the Brazilian Computer Society*, 16(3):201–214.

Shannon, M. (2007). Shaking hands, kissing babies, and...blogging? *Communications of the ACM*, 50(9):21–24.

Wasserman, S., Faust, K., and Iacobucci, D. (1994). *Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press.

YouTube (2010). YouTube fact sheet. `http://www.youtube.com/t/fact_sheet`. Acessado em Março/2010.

CTIC 2011 - XXX Concurso de Trabalhos de Iniciação Científica

PREFÁCIO

O CTIC é um evento anual da Sociedade Brasileira de Computação que tem como objetivo incentivar alunos de Graduação em Computação e Informática a produzirem textos científicos originais sobre temas relacionados à área. Para atender a este objetivo, só é permitida a participação de alunos regularmente matriculados em cursos de gradução, ou recém-diplomados durante o último ano, em Instituições de Ensino Superior brasileiras.

Nesta edição do evento foram submetidos 66 trabalhos, cuidadosamente revisados por três avaliadores do comitê de programa, contando com a colaboração também de alguns revisores externos. O comitê de programa do CTIC contou com a participação de pesquisadores de diferentes instituições, representantes de todas as regiões do país.

Estes anais contêm os 10 trabalhos selecionados dentre todas estas submissões, também recomendados para a publicação na revista REIC (Revista de Iniciação Científica) da SBC. A taxa para esta recomendação foi de 15%. Dentre estes 10 trabalhos mencionados, os 3 primeiros colocados estão apresentados em uma seção de destaque (Seção 1). Eles foram selecionados para apresentação oral no CSBC - 2011 e para premiação de primeiro, segundo e terceiro lugares. A taxa de premiação foi de 4,5%.

Gostaríamos de parabenizar a todos os autores dos trabalhos selecionados e também agradecer a participação de todos os autores que submeteram trabalhos. Agradecemos especialmente a todos os membros do comitê de programa e aos eventuais revisores por eles elencados. Sabemos que sem esta valiosa colaboração não teríamos conseguido concluir o processo de revisão no tempo previsto. Agradecemos também aos organizadores do CSBC-2011 pelo suporte sempre presente, pela infraestrutura disponibilizada e pela oportunidade oferecida. Desejamos boa sorte aos que concorrem à premiação e esperamos que o CTIC possa continuar incentivando alunos a realizar e divulgar seus trabalhos de pesquisa.

Silvia R. Vergilio (DInf -UFPR, coordenadora geral)
Aurora T.R. Pozo (DInf -UFPR, coordenadora geral)
Lyrene F. da Silva (DIMAp-UFRN, coordenadora local)

COMISSÕES

Coordenação Geral

Silvia R. Vergilio (DInf -UFPR, coordenadora geral)
Aurora T.R. Pozo (DInf -UFPR, coordenadora geral)
Lyrene F. da Silva (DIMAp-UFRN, coordenadora local)

Comitê de Programa

Abel Silva-Filho - UFPE
Aldri dos Santos - UFPR
André Ponce de Leon F de Carvalho - USP São Carlos
Andréa Weber - UFPR
Arlindo da Conceição - ICT-UNIFESP
Aurora T.R. Pozo - UFPR
Carlos Maziero - PUCPR
Cesar Melo - UFAM
Celso Kaestner - UTFPR
Cristina Murta - CEFET-MG
Delano Medeiros Beder - UFSCar
Edmundo Spoto - UFG
Edson Cáceres - UFMS
Esteban Clua - UFF
Fabíola Greve - UFBA
Helio Pedrini - UNICAMP
Itana Maria de Souza Gimenes - UEM
Jó Ueyama - USP
Juliano Oliveira - UFG
Julio Nievola - PUCPR
Keiko Veronica Ono Fonseca - UTFPR
Leila Silva - UFS
Leonardo Emmendorfer – FURG
Li Weigang - UNB
Luiz Chaimowicz - UFMG
Lyrene Silva - UFRN
Marcelo Finger - USP
Marcelo Ladeira - UNB
Maria da Graça Campos Pimentel - USP São Carlos
Mario Dantas - UFSC
Paulo Masiero - USP São Carlos
Renata Galante - UFRGS
Renato Fileto - UFSC
Rosângela Penteado - UFSCar
Sílvio Cazella - Unisinos
Silvia R. Vergilio - UFPR
Silvio Guimarães - PUC Minas
Vladimir Di Iorio - UFV

Revisores

Abel Silva-Filho - UFPE
Aldri dos Santos – UFPR
Alneu Lopes USP - São Carlos
Alvaro Fazenda - UNIFESP
Amadeu Barbosa Jr – PUC Rio
André Ponce de Leon F de Carvalho - USP São Carlos
Andréa Weber - UFPR
Arlindo da Conceição - ICT-UNIFESP
Aurora T.R. Pozo - UFPR
Carlos Maziero – PUCPR
Cássio Rodrigues –UFG
Cesar Melo - UFAM
Celso Kaestner - UTFPR
Cristina Murta - CEFET-MG
Delano Medeiros Beder - UFSCar
Edmundo Spoto - UFG
Edson Cáceres - UFMS
Esteban Clua – UFF
Fabiano Ferrari Universidade - UFSCar
Fabíola Greve – UFBA
Guna Santos - UFBA
Helio Pedrini - UNICAMP
Jó Ueyama - USP
Juliano Oliveira - UFG
Julio Nievola - PUCPR
Keiko Veronica Ono Fonseca - UTFPR
Leila Silva - UFS
Leonardo Emmendorfer – FURG
Li Weigang - UNB
Luiz Chaimowicz - UFMG
Lyrene Silva - UFRN
Marcelo Finger - USP
Marcelo Ladeira - UNB
Maria da Graça Campos Pimentel - USP São Carlos
Mario Dantas - UFSC
Paulo Masiero - USP São Carlos
Renata Galante - UFRGS
Renato Fileto – UFSC
Ricardo da Rocha – UFG
Rosângela Penteado - UFSCar
Sílvio Cazella - Unisinos
Silvia R. Vergilio - UFPR
Silvio Guimarães - PUC Minas
Thelma Colanzi - UEM
Tiago Oliveira - UNIFESP
Valerio Rosset - USP

Vladimir Di Iorio - UFV
Wesley Klewerton Guez Assunção - UFPR

ARTIGOS

SESSÃO 1

Avaliação de Sistemas de Busca em Mercados Eletrônicos

Diego M. Duarte [1], **Adriano C. M. Pereira** [2], **Wagner Meira Jr.** [1]

[1] Dpto. de Ciência da Computação – Univ. Federal de Minas Gerais (UFMG)
Caixa Postal 486 – 31.270-901 – Belo Horizonte – MG – Brasil

{diegomd, meira}@dcc.ufmg.br

[2]Dpto. de Computação – Centro Federal de Educação Tecnológica de MG (CEFET-MG)
Av. Amazonas, 7675 – 30.510-000 – Belo Horizonte – MG – Brasil

adriano@decom.cefetmg.br

Abstract. *Electronic markets, such as e-Bay and Amazon.com, have reached great popularity in the last few years. In this scenario, understanding the user behavior is a challenge. This work presents an analysis of the user navigation in TodaOferta, which is a large Brazilian electronic market, with focus on the search system. From the classification of user requests, we distinguish navigation sessions with search functions from the ones that have not. We conclude that the first group has more sessions with larger sizes, such as a higher efficiency to perform transactions. The results can be aplied to improve the search engines and create customized services.*

Resumo. *Mercados eletrônicos, tais como eBay e Amazon.com, têm ampliado sua popularidade nos últimos anos. Neste cenário, um dos desafios existentes é a compreensão da interação dos usuários. Este trabalho apresenta uma análise da navegação de usuários no mercado eletrônico brasileiro TodaOferta, com foco no sistema de busca. A partir da categorização das requisições, distiguimos sessões de navegação que possuem ou não funcionalidades de busca. Observou-se uma superioridade em quantidades e tamanhos do primeiro conjunto, assim como uma eficiência maior das sessões de busca para concretização de transações. Os resultados obtidos podem contribuir para aperfeiçoar os mecanismos de busca e criar serviços personalizados.*

1. Introdução

Nos últimos anos, percebe-se um aumento da popularidade do Comércio Eletrônico (*e-commerce*), o que pode ser confirmado no uso dos mercados eletrônicos (*e-markets*) como o *eBay* e *Amazon.com*. Além disso, com o surgimento do conceito chamado *Web 2.0*, o usuário passa a ser não apenas um espectador, mas também um provedor de informações. Um dos desafios que surgem com essa mudança é a compreensão da interação entre usuário e o sistema. Entender o comportamento de usuários diante de um sistema de compra e venda online pode contribuir para uma melhora de eficiência desse sistema, tanto para o seu provedor, quanto para quem o utiliza.

A partir da navegação de usuários em mercados eletrônicos, é possível concentrarmos em algumas funcionalidades de grande importância, como o processo de busca de produtos. O sistema de busca é um dos mecanismos mais importantes utilizados nesse

ambiente. Os resultados apresentados por essa ferramenta podem possuir uma grande influência para a efetivação de negociação de produtos. Por isso, é de vital importância compreender o funcionamento da busca em um mercado eletrônico, assim como o seu impacto nas transações.

Neste projeto de iniciação científica realizamos uma análise da navegação de usuários no mercado eletrônico brasileiro chamado *TodaOferta*[1], pertencente ao Universo Online (UOL)[2]. Centralizando os estudos no sistema de busca desse mercado, realizou-se uma avaliação detalhada acerca da influência da utilização dessa ferramenta sobre as negociações de produtos online. Observou-se que sessões com funcionalidades de busca possuem uma retenção maior do usuário no sistema comparando-se com sessões que não utilizam tal mecanismo, além de obterem uma quantidade superior de transações efetivadas. Os resultados dessa pesquisa podem ser úteis para a criação de serviços de busca personalizados, bem como para auxiliar o provedor do serviço e os vendedores a aperfeiçoar a sua estratégia de publicação e divulgação de suas ofertas.

Como fruto dos trabalhos relacionados a este projeto de iniciação científica, obtivemos inicialmente a publicação do artigo titulado *"Analyzing User Profiles in Electronic Markets"* [Duarte et al. 2010a], na *IEEE International Conference on Information Society* (*i-society 2010*). Posteriormente tivemos a publicação de resultados parciais deste trabalho no WTIC/WebMedia'2010 [Duarte et al. 2010b].

O restante do trabalho está organizado da seguinte forma: a seção 2 descreve os trabalhos relacionados. A seção 3 apresenta uma descrição e caracterização do nosso estudo de caso. A seção 4 descreve uma análise detalhada do sistema de busca do mercado *TodaOferta*. Finalmente, a seção 5 apresenta as conclusões e trabalhos futuros.

2. Trabalhos Relacionados

Mercados eletrônicos têm apresentado um grande crescimento a cada dia, tanto em volume de transações quanto financeiro. A opção *Compre-já* (*Buy-it-now*), que permite dar um preço fixo para encerrar o leilão, tem alcançado grande popularidade nos mercados eletrônicos. Durham [Durham and Standifird 2004] e Anderson [Anderson et al. 2004] mostraram que vendedores experientes utilizam frequentemente o *Compre-já*, e que esta opção é mais bem vista quando utilizada por vendedores de reputação elevada. No Brasil, esse tipo de negociação direta é muito popular.

A caracterização de carga de trabalho é fundamental para o entendimento e aprimoramento de sistemas *Web*. Há vários estudos que apresentam caracterizações de carga de trabalho de diferentes tipos, tais como servidores *Web* [Arlitt and Williamson 1996], de comércio eletrônico [Menasce and Almeida 2000, Pereira et al. 2006], de blogs [Duarte et al. 2007], de vídeo sob demanda [Costa et al. 2004] e de vídeo ao vivo [Veloso et al. 2006]. Dentre as várias contribuições desses trabalhos, destacamos a criação de valiosos modelos capazes de descrever a carga de trabalho que chega nesses servidores, essenciais para a geração de carga sintética que, por sua vez, possibilita a realização de experimentação e simulação baseadas em distribuições mais realistas.

O trabalho [Benevenuto et al. 2009] apresenta uma análise da carga de trabalho

[1]http://www.todaoferta.com.br
[2]http://www.uol.com.br

de um serviço de vídeos do UOL, utilizando sessões e requisições ao servidor, além de uma avaliação do perfil de navegação dos usuários. Nesse trabalho foram identificados diferentes perfis de usuários que acessam o sistema, o que pode ser utilizado para prover serviços personalizados aos usuários. Nas avaliações dos trabalhos relacionados, não encontramos trabalhos específicos de avaliação de funcionalidades de busca em mercados eletrônicos.

3. Estudo de Caso

Esta seção faz uma descrição geral dos dados do *TodaOferta*. Algumas informações e estatísticas são apresentadas na tabela 1. Devido à confidencialidade desses dados, é importante ressaltar que tivemos que omitir informações quantitativas.

Período da amostra	01/04/2009 a 30/06/2009
Número de categorias de produtos	32
Número de sub-categorias de produtos	2.189
Tipos de negociação	Preço fixo e Leilão

Tabela 1. *TodaOferta* **- Informações Gerais**

Como se pode perceber, o mercado eletrônico dispõe de uma grande variedade de produtos. Apesar da análise realizada na seção 3.2 utilizar toda a base de dados, devido à limitação de espaço, apresentaremos na seção 4 apenas os resultados do mês de *abril* de 2009, que conta com milhões de requisições.

Em uma análise anterior do comportamento do usuário no mercado eletrônico em questão, distinguimos as requisições em dois grupos: de usuários humanos (acessos de um usuário comum, que representa um potencial comprador) e de usuários robôs ou *bots* (programa ou *script* que possui como objetivo simular ações humanas para inúmeras funcionalidades, dentre elas, a coleta de dados). Porém, para a avaliação do sistema de busca apresentada neste trabalho, utilizamos apenas requisições de usuários humanos.

3.1. Caracterização Funcional

Nesta seção, são apresentadas as funcionalidades referentes às requisições de usuários no mercado *TodaOferta*. Ao todo, existem 9 diferentes funções, descritas na tabela 2.

Funcionalidade	**Definição**
Busca	Busca por palavra chave
Busca por Usuário	Listar produtos de um usuário
Página Inicial	Página principal do *TodaOferta*
Página Categoria	Listar de produtos de uma categoria
Comprar	Processo para compra de um produto

Funcionalidade	**Definição**
Confirmação da Compra	Finalização de uma compra
Detalhar Anúncio	Informações de um anúncio
Listar Compradores	Listar compradores de um produto
Perfil Usuário	Detalhar usuário (vendedor ou comprador)

Tabela 2. Funcionalidades do TodaOferta

É importante ressaltar que não estão sendo consideradas as funcionalidades referentes a ações de administração e de cadastramento de ofertas no mercado eletrônico.

3.2. Sessão de Usuário

Uma sessão de um usuário é definida como um série de requisições realizadas a um sítio *Web* durante um determinado período de tempo [Arlitt 2000]. A determinação do início

e término de uma sessão em nosso cenário real de estudo de caso requer uma análise específica dos tempos entre requisições a fim de medir a inatividade do usuário, uma vez que a maioria das sessões não apresenta um registro explícito de operações de início e final (*logout*). Portanto, é necessária uma análise para identificar um limite de tempo entre requisições para que sejam consideradas como sendo da mesma sessão. Assim, duas requisições consecutivas pertencem à mesma sessão se o tempo entre elas é menor do que esse limite, denominado tempo de expiração da sessão.

Seguindo a metodologia proposta em [Menascé et al. 1999], realizamos uma avaliação do tempo de expiração da sessão mais adequado para nossa aplicação. Procuramos um ponto onde a quantidade de sessões comece a se estabilizar, mesmo com o aumento deste tempo. Analisando a figura 1, percebe-se uma queda maior na quantidade de sessões para os usuários humanos do que para robôs. Concluímos também que o melhor tempo para a nossa base de dados é 15 minutos, valor este utilizado nas análises das seções seguintes.

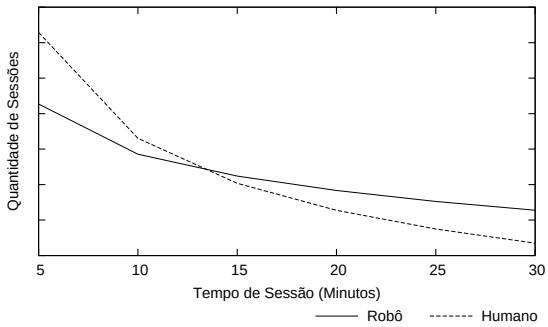


Figura 1. Tamanhos de sessão

Para entendermos o padrão de navegação de usuários dentro de sessões, utilizamos um grafo probabilístico direcionado, onde os vértices representam os possíveis tipos de requisição dos usuários (p. ex., listar compradores, buscar, comprar) e as arestas representam a probabilidade de navegação de um tipo de requisição para outra. Chamamos esse grafo de UBMG (*User Behavior Model Graph*). O UBMG é uma representação genérica do *Customer Behavior Model Graph* (CBMG) [Menasce and Almeida 2000], uma forma de representar o modelo comportamental dos usuários.

4. Sistema de Busca - Caracterização e Análise

Nesta seção, apresentaremos os resultados da avaliação centrada no Sistema de Busca do mercado eletrônico *TodaOferta*.

4.1. Categorização

O primeiro passo para avaliarmos o sistema de busca é identificar as diferentes maneiras de se buscar algo no mercado eletrônico. Existem diversas maneiras de se realizar uma busca além do comumente usado (palavra-chave), como por exemplo a navegação através das categorias, ou até mesmo a busca orgânica através de outros sistemas de busca, como o *Google* e *Yahoo!*. Assim, identificamos oito categorias que abrangem todas as requisições com funcionalidades de busca, que estão detalhadas a seguir:

- **Busca em categoria**: busca de ofertas a partir da navegação em categorias ou sub-categorias do *TodaOferta*;
- **Busca por palavra-chave**: busca apenas por um ou mais termos específicos;
- **Busca com filtros**: qualquer navegação nos resultados do sistema de busca, com a utilização de filtros (ex., preço máximo, localização do produto, etc.);
- **Busca de produtos por vendedor**: listagem de produtos de um vendedor;
- **Busca orgânica (Máquina de Busca)**: buscas advindas de máquinas de busca externas ao mercado;
- **Busca patrocinada** (e.g., *AdWords*): serviço com anúncios em forma de links relacionadas à palavra ou texto utilizados pelo usuário (ex.: *Google AdWords*).
- **Busca por** *Marketing*: anúncios enviados pelo próprio provedor do mercado com o intuito de promover anúncios (ex., Boletim *TodaOferta*).
- **Outros**: busca advinda de sites variados.

Uma vez definidas as categorias, será realizada nas avaliações a seguir uma distinção entre dois tipos de sessões: **Sessões com Busca**, que representam qualquer sessão onde pelo menos uma requisição se enquadra em uma das categorias definidas acima; e o restante das sessões, chamadas de **Sessões sem Busca**.

É importante ressaltar que consideraremos apenas as sessões de humanos nas sessões seguintes, uma vez que a análise da utilização do sistema de busca por robôs não seria relevante para a pesquisa.

4.2. Avaliação Estatística

Com a distinção de sessões que possuem ou não a funcionalidade de busca, foi realizada uma avaliação estatística das requisições e sessões da navegação de usuários. As estatísticas encontradas estão expostas na tabela 3.

	Sessões com Busca	**Sessões sem Busca**
Quantidade de requisições	90,02%	9,98%
Quantidade de sessões	78,38%	21,65%
Média de requisições	3,96 ($\pm 28, 12$)	1,59 ($\pm 7, 29$)
Quantidade de sessões (Tamanho > 1)	87,75%	12,25%
Média de requisições (Tamanho > 1)	6,48 ($\pm 38, 17$)	3,16 ($\pm 13, 91$)

Tabela 3. Estatísticas de sessões

Analisando a tabela 3, percebe-se a superioridade das sessões de busca. Este tipo de sessão corresponde a mais de 90% do total de requisições de nossa base de dados. Além disso, sessões com a funcionalidade de busca representam quase 80% de todas as sessões existentes, obtendo uma média de quatro requisições por sessão, valor este que é maior que o dobro das sessões que não possuem busca (1,59).

Durante a análise dos dados, observou-se que sessões que possuem apenas uma requisição ao *TodaOferta* não agregam muita informação. Dessa forma, para a análise estatística, foram retiradas as sessões que possuem tamanho 1. Nesse contexto, ainda é possível observarmos a grande utilização da ferramenta de busca. Dentre os 87,75% das sessões com busca, a média de requisições por sessão chega a 6,48, novamente um valor bem maior do que as sessões sem busca, que obteve uma média de 3,16.

Com uma grande quantidade de sessões, é necessário realizar uma análise mais profunda do seu tamanho. A quantidade de requisições que compõem uma sessão é a me-

lhor métrica encontrada para esta avaliação, e por isso, expomos na figura 2 a distribuição de tamanhos para sessões com e sem busca.

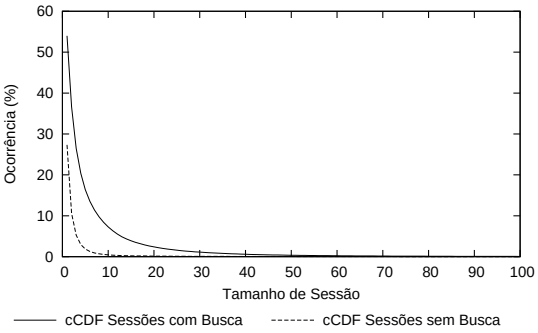


Figura 2. Tamanhos de Sessão

A partir da figura 2, observa-se que sessões com busca apresentam ocorrências maiores do que sessões sem busca. Neste último grupo, mais de 90% das sessões possuem um tamanho menor do que cinco, enquanto nas sessões com busca, esta porcentagem abrange sessões de tamanho dez. Além disso, em ambos os conjuntos, percebe-se um comportamento semelhante com relação à distribuição de ocorrências, onde temos uma grande concentração de ocorrências de tamanhos de sessão pequenos (menores que 10), enquanto sessões maiores ocorrem pouco no mercado.

Com as análises estatísticas, é possível observar que sessões de usuários que utilizam a ferramenta de busca são normalmente bem maiores do que aqueles que não utilizam tal ferramenta. Isto significa que o usuário navega mais pelo mercado ao utilizar a busca, e consequentemente, possui um acesso maior ao seu conteúdo.

4.3. Avaliação Funcional

Utilizando os tipos básicos de requisição definidos na tabela 2, avaliamos a distribuição de tais requisições nas sessões com e sem busca. Nas figuras 3 e 4 estão expostos os histogramas para tal análise. Como uma sessão pode possuir mais de um tipo de requisição, a soma das colunas não necessariamente resulta em 100%.

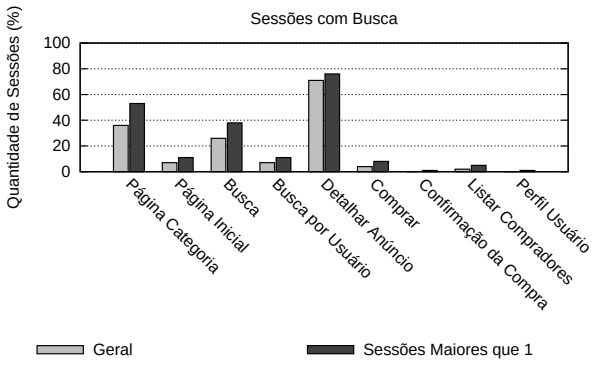


Figura 3. Requisições em sessões com busca

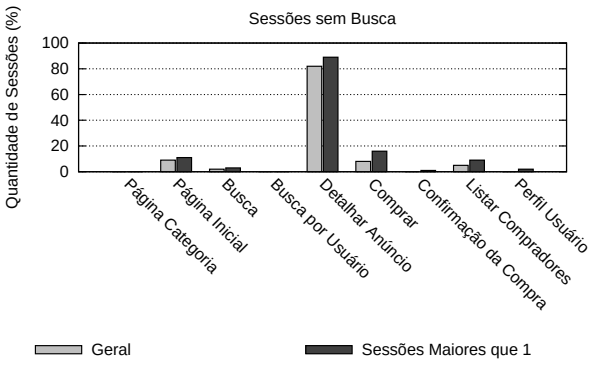


Figura 4. Requisições em sessões sem busca

Nas figuras 3 e 4, percebe-se a grande quantidade de acessos do tipo *Detalhar Anúncio*, que está presente em aproximadamente 80% de todas as sessões em ambos

os casos. Porém, percebe-se que outros tipos de requisições são bem mais acessados nas sessões com busca, como *Página Categoria*. Em alguns casos, não existe nenhuma requisição nas sessões sem busca, como por exemplo, *Busca por Usuário*. Curiosamente, uma porcentagem pequena de requisições do tipo *Busca* aparece nas sessões sem busca. Estas requisições são advindas de sites sem funcionalidades identificadas, assim não foram classificadas como sessões de busca.

Seguindo as categorias definidas na Seção 4.1, realizou-se também uma análise de todas as requisições de sessões com busca. Os resultados estão expostos na figura 5.

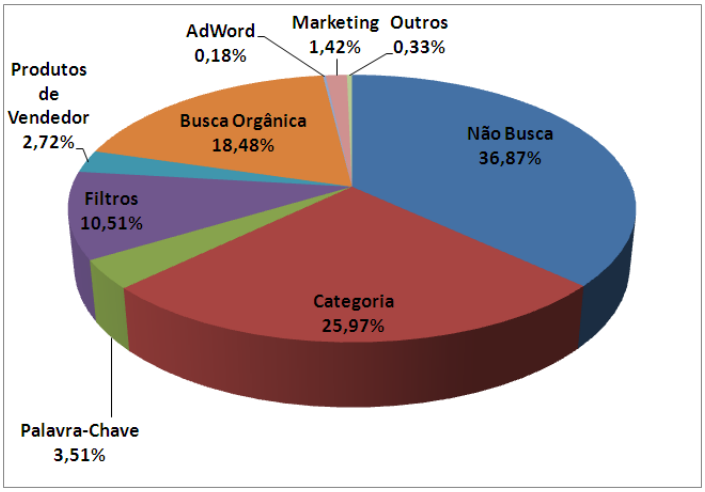


Figura 5. Categorização das requisições

Através da figura 5, percebe-se uma grande quantidade de requisições de Busca Orgânica (18,48%) e por Categoria (25,97%). A utilização de Filtros também é relevante no mercado (10,51%). Porém, outros tipos de busca aparecem em uma porcentagem relativamente pequena, como *Marketing* e Busca Patrocinada (*AdWords*).

Dessa forma, avalia-se que o usuário muitas vezes acessa as ofertas do site através de mecanismos de busca externos. Em outros casos, o usuário prefere uma navegação nas categorias a procurar especificamente um termo na própria ferramenta de busca do mercado. Além disso, o *marketing* utilizado pela *TodaOferta* aparentemente leva a mais navegações do usuário do que a busca patrocinada.

4.4. Análise de Sessões - Modelagem do Comportamento dos Usuários

Nesta seção utilizamos o conceito de UBMG para avaliarmos a ferramenta de busca do *TodaOferta*. Para isto, é necessário identificar na navegação do usuário determinados pontos que possam ser relevantes para o nosso objetivo. Assim, para avaliarmos o real impacto da ferramenta de busca, iremos focar em três tipos de requisições: *Detalhar Anúncio*, *Comprar* e *Confirmação da Compra*. Acreditamos que estas requisições representam os últimos passos do usuário antes da efetivação de uma transação no mercado, uma vez que, para a compra de um produto, é necessário passar pelo anúncio em questão, pelo processo de compra, e então pela sua concretização.

Entretanto, nem sempre uma compra segue à risca esta sequência de requisições. Dessa forma, para a nossa avaliação, iremos utilizar um UBMG modificado, de forma a

garantir que estes três tipos de requisição sejam utilizados em sequência. Esta regra de avaliação por ser compreendida a partir do esquema apresentado na figura 6.

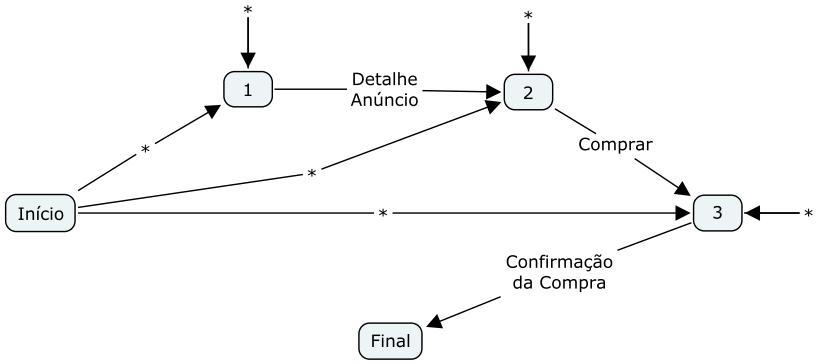


Figura 6. Regra de avaliação

Na figura 6, cada transição representa a execução de uma requisição. Isso foi realizado com o objetivo de compreendermos a regra de avaliação adotada, onde, ao identificarmos uma requisição do tipo *Detalhe Anúncio* mudamos para o estado 2. A partir disto, para que ocorra a mudança para o estado 3, verificamos nas requisições seguintes a ocorrência de uma requisição do tipo *Comprar*.

Como último passo, avaliamos a ocorrência de tal regra nas sessões com e sem busca. Com um simplificação dos resultados obtidos, fomos capazes de gerar o UBMG para ambos os casos, como mostram as figuras 7 e 8.

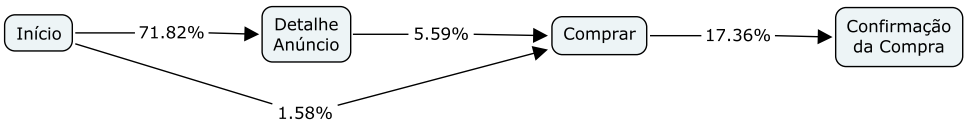


Figura 7. Sessões com Busca

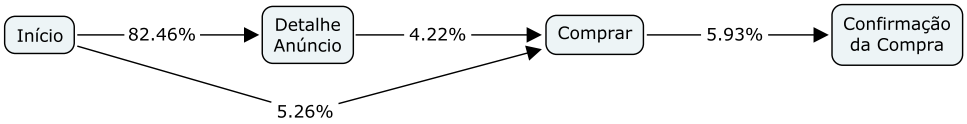


Figura 8. Sessões sem Busca

A partir dos UBMGs gerados, podemos ter uma idéia da ocorrência dos três tipos de requisição citados para esses dois tipos de sessão. Porém, esse grafo comprime algumas informações relevantes. Por exemplo, dentre aqueles que detalham um anúncio, quais chegam a concretizar uma compra? Dessa forma, a seguir são realizadas avaliações a partir de algumas informações que não estão presentes nas figuras 7 e 8, embora estas tenham sido geradas a partir de tais análises.

Iniciando pela requisição de *Detalhe Anúncio*, percebe-se que existe uma leve superioridade na quantidade de sessões que não utilizam o sistema de busca (82,46% contra 71,82%). Isso se deve ao fato de, ao realizarmos uma busca, já estarmos filtrando uma série de anúncios a partir dos seus resultados, o que significa que o usuário não

precisa sair à procura de informação acessando diversos anúncios. Assim, possuímos uma quantidade menor de acessos ao detalhamento de uma oferta, reduzindo a quantidade de sessões com tal função. Porém, o tamanho de cada sessão não é necessariamente menor, já que em sessões com o sistema de busca, o tamanho chega a 4,55 (ou 7,14, desconsiderando sessões de tamanho igual a um), quatro vezes maior das sessões que não utilizam tal ferramenta (1,65 e 3,19 para tamanhos maior que um).

Observando a ação de *Comprar*, ainda existe uma superioridade das sessões que não utilizam o sistema de busca, que chegam a 8,74% de todas as sessões (o dobro do outro conjunto). Porém, dentre as sessões que chegam a detalhar um anúncio, 5,59% das sessões que vieram de uma busca iniciam tal processo de compra, contra 4,22% do outro grupo. Curiosamente, das sessões que possuem busca, 1,58% não detalham a oferta e iniciam o processo de compra. Porém, este valor chega a 5,26% para sessões que não possuem busca. Percebe-se que o refinamento do sistema de busca contribui para a decisão do usuário, mas ainda assim é importante avaliar a *Confirmação da Compra* para analisar o real impacto da ferramenta de busca.

De maneira geral, 0,78% das sessões com busca concluem uma negociação, contra 0,56% das que não utilizam esta ferramenta, o que já demonstra a sua importância. Além disso, resumindo para o grupo que iniciou o processo de compra, esta superioridade sobre para 17,36% contra 5,93% das sessões que não usam busca. Do grupo que detalhou um anúncio, sessões com busca culminam três vezes mais em concretização do negócio. Dessa forma, pode-se dizer que utilizar a ferramenta de busca possui um grande impacto para a concretização de negócios no mercado eletrônico em questão.

5. Conclusão

Neste trabalho de iniciação científica realizamos uma análise da navegação de usuários no mercado eletrônico brasileiro chamado *TodaOferta*, como foco no procedimento de busca, uma das ferramentas mais utilizadas neste cenário.

A partir de uma avaliação geral da carga de trabalho, distinguindo-se requisições de usuários humanos e robôs, percebeu-se o quão distintos são os seus comportamentos. Além disso, foi possível identificar inúmeras funcionalidades atribuídas a cada requisição, permitindo realizar uma análise semântica da carga de trabalho.

Com um foco maior no sistema de busca do mercado, definimos categorias relacionadas à busca para cada tipo de requisição do nosso conjunto de dados. Observamos que boa parte destas são vistas como Buscas Orgânicas ou como navegações nas categorias do *TodaOferta*. A partir disto, realizou-se uma distinção entre sessões com e sem algum tipo de requisição de busca. Com análises estatísticas, mostrou-se que sessões com busca apresentam superioridade tanto em questão de quantidade, quanto em tamanho de sessões. Por fim, modelamos as sessões como um grafo (UBMG), com foco em *Detalhar Anúncio*, *Comprar* e *Confirmação da Compra*. As análises demonstraram que existe uma maior eficiência para concretizar transações para aqueles usuário que utilizam o sistema de busca, além destes possuírem uma navegação maior no site em termos de quantidade de requisições.

Através deste trabalho, conseguimos perceber o quão importante é o sistema de busca em um mercado eletrônico. Os resultados desta pesquisa são importantes para o

provedor e usuários do sistema, pois permite entender melhor a importância do sistema de busca, bem como identificar formas de torná-lo mais eficiente para a aplicação *Web*. Futuramente, iremos analisar mais profundamente algumas categorias definidas, como Busca Orgânica, *AdWords* e *Marketing*, a fim de identificar a importância destes serviços para o mercado.

6. Agradecimentos

Esta pesquisa é parcialmente patrocinada por Universo OnLine S. A. - UOL (www.uol.com.br) e parcialmente apoiada pelo Instituto Nacional de Ciência e Tecnologia para a Web (CNPq no. 573871/2008-6), CAPES, CNPq, Finep e Fapemig.

Referências

Anderson, S., Friedman, D., Milam, G., and Singh, N. (2004). Buy it now: A hybrid internet market institution. Industrial Org. 0412003, EconWPA.

Arlitt, M. (2000). Characterizing web user sessions. *SIGMETRICS Performance Evaluation Review*, 28(2):50–63.

Arlitt, M. and Williamson, C. (1996). Web server workload characterization: the search for invariants. *SIGMETRICS Performance Evaluation Review*, 24(1):126–137.

Benevenuto, F., Pereira, A., Rodrigues, T., Almeida, V., Almeida, J., and Gonçalves, M. (2009). Avaliação do perfil de acesso e navegação de usuários em ambientes web de compartilhamento de vídeos. In *WebMedia'09*, Fortaleza, Brasil. SBC.

Costa, C., Cunha, I., Vieira, A., Ramos, C., Rocha, M., Almeida, J., and Ribeiro-Neto, B. (2004). Analyzing client interactivity in streaming media. In *WWW*.

Duarte, D., Pereira, A., and Meira Jr., W. (2010a). Analyzing user profiles in electronic markets. In *i-Society 2010: Proceeding of the International Conference on Information Society*, Londres, UK. IEEE.

Duarte, D., Pereira, A., and Meira Jr., W. (2010b). Avaliação do sistema de busca em mercados eletrônicos. In *VII Workshop de Trabalhos de Iniciação Científica (WTIC'10)*, Belo Horizonte, MG. Sociedade Brasileira de Computação (SBC).

Duarte, F., Mattos, B., Bestavros, A., Almeida, V., and Almeida, J. (2007). Traffic characteristics and communication patterns in blogosphere. In *Proc. Int'l Conference on Weblogs and Social Media (ICWSM)*.

Durham, Y., M. R. and Standifird, S. (2004). ebay's buy-it-now function: Who, when and how. *Topics in Economic Analysis & Policy*, 4(1):28.

Menasce, D. and Almeida, V. (2000). *Scaling for E Business: Technologies, Models, Performance, and Capacity Planning*. Prentice Hall PTR, NJ, USA.

Menascé, D., Almeida, V., Fonseca, R., and Mendes, M. (1999). A methodology for workload characterization of e-commerce sites. In *ACM Conf. on e-Commerce (EC)*.

Pereira, A., Silva, L., and Meira, Jr., W. (2006). Evaluating the impact of reactive workloads on the performance of web applications. In *Proc. of the 25th IEEE Int'l Performance, Computing, and Communications Conf. (IPCCC)*, Phoenix, USA. IEEE CS.

Veloso, E., Almeida, V., Jr., W. M., Bestavros, A., and Jin, S. (2006). A hierarchical characterization of a live streaming media workload. *ACM Trans. on Network*, 14(1).

Detecção de Spammers na Rede de Origem

Pedro Henrique B. Las-Casas[1], **Humberto T. Marques-Neto**[1]

[1] Departamento de Ciência da Computação
Pontifícia Universidade Católica de Minas Gerais (PUC Minas)
30.535-901 - Belo Horizonte - MG - Brasil

pedro.casas@sga.pucminas.br, humberto@pucminas.br

Abstract. *The volume of unsolicited messages (spam) sent over the Internet represents more than 85% of all e-mails. Even with the evolution of the filtering techniques such as the analysis of the message content and the blocking of IPs, network resources are wasted given that such a filtering is usually performed at the e-mail destination server. This paper proposes a method for detecting spammers in the origin network using a supervised classification technique with metrics which do not require inspection of message contents. Results show that the adopted method is efficient, being able to correctly identify most spammers still in the origin network, saving network resources.*

Resumo. *A quantidade de mensagens não-solicitadas (spams) enviadas na Internet representa mais de 85% de todos os e-mails. Mesmo com a evolução de técnicas de filtragem como a análise do conteúdo de mensagens e o bloqueio de IPs, recursos da rede são desperdiçados, uma vez que essa filtragem é realizada normalmente no servidor de destino dos e-mails. Este trabalho propõe um método para detecção de spammers na rede de origem utilizando uma técnica de classificação supervisionada composta por métricas que não requerem a inspeção do conteúdo das mensagens enviadas. Os resultados mostram que o método utilizado é eficaz, sendo capaz de identificar a maioria dos spammers ainda em sua rede de origem, preservando assim, os recursos da rede.*

1. Introdução

Relatórios recentes [MessageLabs 2010] indicam que cerca de 85% das mensagens eletrônicas (e-mails) que circulam pela Internet são mensagens indesejadas, muitas vezes caracterizadas como spam. Para reduzir o impacto dessas mensagens indesejadas sobre o serviço de correio eletrônico, provedores utilizam filtros de e-mails que as classificam, descartam ou colocam em quarentena, evitando que elas sobrecarreguem as caixas postais dos destinatários. Entretanto, esses filtros não evitam o desperdício de recursos da rede, pois as mensagens recebidas geram tráfego nos links e consomem CPU para serem encaminhadas e processadas.

Uma possível forma de evitar esse desperdício de recursos causado pelos *spams* seria complementar a filtragem no servidor receptor com o uso de técnicas de filtragem prévia, capazes de evitar o envio do *spam* e, com isso, o desperdício de recursos associado. Tais técnicas podem ser aplicadas, por exemplo, em provedores de acesso à Internet de banda larga que, através da análise do tráfego SMTP (*Simple Mail Transfer Protocol*), poderiam detectar a ação de possíveis *spammers*, bloqueando o tráfego na sua origem.

Este artigo descreve a proposta de um novo método para detecção de spammers na rede de origem chamado SpaDeS - *Spammer detection at the Source* - que é baseado em um algoritmo de classificação supervisionada e explora apenas métricas que não requerem a inspeção do conteúdo das mensagens. Para o desenvolvimento da proposta deste método, utilizou-se um estudo prévio do tráfego da rede de origem de um provedor de Internet de banda larga [Castilho et al. 2010]. Este estudo apresenta a caracterização do tráfego SMTP presente na rede de origem, e define métricas capazes de serem utilizadas para distinguir spammers de usuários legítimos. O aluno de Iniciação Científica (IC) Pedro Las Casas participou ativamente deste trabalho, que foi desenvolvido no contexto do projeto ReBu: Sistemas de Redes Robustos - Modelos e Ferramentas. A partir deste trabalho, o aluno de IC desenvolveu a proposta do método de detecção de spammers na rede de origem. Maiores detalhes do método aqui apresentado se encontram em [Las-Casas et al. 2011], artigo científico publicado no XXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2011), sendo o aluno de IC o seu autor principal.

O método proposto foi aplicado e validado utilizando dois conjuntos de dados reais contendo informações agregadas e anonimizadas de transações SMTP de usuários de um provedor brasileiro de Internet de banda larga residencial coletadas em 2009 e 2010. Os resultados apresentados mostram que a utilização do SpaDeS é capaz de diferenciar *spammers* de usuários legítimos ainda na rede de origem, sem inspecionar o conteúdo de suas mensagens. Com o uso da técnica de classificação supervisionada, validada através da comparação com uma base de dados real de denúncias de *spam* e da inspeção de uma amostra dos usuários classificados, estima-se que cerca de 98% dos usuários legítimos e 94% dos *spammers* foram classificados corretamente. Ou seja, as taxas de falsos positivos e de falsos negativos foram, respectivamente, de 2% e 6%. O estudo mostra também que classes de usuários legítimos representaram cerca de 83% dos usuários e realizaram cerca de 1,6% do total de transações SMTP observadas nos dados de 2010. Enquanto isso, os usuários classificados como *spammers*, (cerca de 17%) originaram mais de 98% de todas as transações SMTP no período observado.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta o método de detecção de *spammers* proposto e a Seção 3 discute os resultados mais relevantes do trabalho. Finalmente, a Seção 4 apresenta conclusões e sugere trabalhos futuros.

2. SpaDeS: Detector de Spammers na Rede de Origem

O método proposto para detecção de *spammers* na rede de origem, denominado SpaDeS (*Spammer Detection at the Source*), tem como principal componente um algoritmo de classificação supervisionada, que "aprende" um modelo de classificação de usuários a partir de um conjunto de exemplos (usuários) previamente classificados (conjunto de treino). O classificador recebe como entrada o número de classes distintas C e exemplos de usuários de cada uma. Após a fase de aprendizado, o modelo derivado pode então ser aplicado para classificar novos usuários (conjunto de teste) nas classes pré-definidas. A Seção 2.1 apresenta as classes de usuários consideradas assim como o modelo de representação dos mesmos. O algoritmo de classificação utilizado é apresentado na Seção 2.2, enquanto a Seção 2.3 discute como obter o conjunto de treino.

2.1. Modelo de Representação de Usuários e suas Classes

Cada usuário é representado por um vetor de N atributos que conjuntamente descrevem seu comportamento quanto ao uso do protocolo SMTP. Para detectar *spammers* na rede de origem com eficiência, foram utilizados N=6 atributos, que são métricas que não envolvem processamento do corpo da mensagem. As métricas são: número de transações SMTP realizadas, número de remetentes distintos, número de servidores SMTP distintos acessados, tamanho médio das transações SMTP, distância geodésica média entre origem e destino e tempo médio entre transações consecutivas (aqui refereciado como IATs, *inter-arrival times*).

A escolha das métricas foi inspirada nos resultados do trabalho [Castilho et al. 2010], em que o aluno de IC participou implementando e executando algoritmos, como também analisando resultados. Utilizando o algoritmo de agrupamento *X-means*, demonstrou-se que essas características podem ser utilizadas para distinguir 4 classes de comportamento, sendo que duas refletem padrões de usuários legítimos enquanto as outras refletem padrões abusivos, potencialmente de *spammers*. Por exemplo, o número de transações por usuário é útil para distinguir usuários que fazem pouco uso de SMTP daqueles que o utilizam com grande intensidade. Mais ainda, enquanto o uso de poucos servidores de SMTP é o esperado para usuários legítimos, o acesso a um número muito grande pode indicar a operação de *open proxies* ou de *open mail relays* sendo explorados para o envio de spam por usuários maliciosos ou *bots*. O uso da distância geodésica como métrica se baseia na hipótese de que conexões SMTP de *spammers* tendem a ocorrer entre endereços IPs mais distantes que as conexões de usuários legítimos, já que *spammers* tendem a ocultar suas presença usando máquinas em outros países [Guerra et al. 2008].

2.2. Algoritmo de Classificação Supervisionada

O algoritmo de classificação utilizado neste trabalho é o *Lazy Associative Classifier* (LAC) [Veloso et al. 2006], que tem ótima escalabilidade, com complexidade de tempo polinomial. Diferentemente de outros classificadores, o LAC fornece uma estimativa da confiança na predição feita em cada caso. Essa confiança, que pode ser interpretada como uma probabilidade de acerto da classificação, será explorada na geração do conjunto de treino (Seção 2.3). O LAC explora o fato de que, frequentemente, há fortes associações entre os valores dos atributos e as classes. Tais associações estão geralmente implícitas no conjunto de treino e, quando descobertas, revelam aspectos que podem ser utilizados para prever as classes dos usuários.

2.3. Coleção de Treino

O funcionamento de qualquer método de classificação supervisionada depende primariamente de um conjunto de treino contendo usuários pré-classificados. A obtenção desse conjunto para a classificação de usuários em *spammers* e legítimos é um grande desafio, uma vez que tais dados tipicamente não estão disponíveis publicamente. Um fator complicador é que almeja-se detectar *spammers* ainda na rede de origem. Logo, faz-se necessário um conjunto de treino coletado naquele ponto do sistema. Caso contrário, os padrões levantados poderiam não generalizar para o conjunto de teste, resultando em um desempenho pobre do classificador. São propostas duas estratégias para geração da coleção de treino, uma baseada em informação externa ao algoritmo de aprendizado, enquanto a segunda utiliza a informação de confiança provida pelo LAC.

A primeira estratégia parte das 4 classes de usuários identificadas em [Castilho et al. 2010] e acrescidas de informações sobre usuários apontados como *spammers* por mecanismos de relato de abusos do sistema de correio. Para cada uma das duas classes representando usuários legítimos identificadas, foram selecionados os M usuários mais próximos do centróide de cada classe, de forma a obter bons representantes de cada uma. As duas classes de usuários abusivos (possíveis *spammers*) identificadas naquele trabalho apresentam uma variabilidade maior de comportamentos. Por esse motivo, optou-se por não usar o mesmo mecanismo de seleção, mas basearmos a escolha em uma informação confiável externa. Para isso, utilizou-se a identificação de usuários cujas máquinas foram apontadas como origem de *spam* por relatos oriundos de outros provedores. Tais relatos, enviados para o endereço `abuse` do provedor que forneceu os dados utilizados nesse trabalho, são gerados por provedores tanto a partir de reclamações de seus usuários (como o recurso "*Report spam*" do Gmail) ou por mecanismos automáticos, como listas de bloqueio. Como será visto, um número razoável de usuários denunciados estavam presentes naquelas duas classes, sendo portanto bons representantes das mesmas.

A segunda estratégia parte do pressuposto que o SpaDeS deve ser aplicado continuamente, em diferentes conjuntos de teste (p.ex: dados referentes a diferentes semanas, meses ou anos). Logo, propôs-se retreinar o LAC a partir do resultado da sua execução anterior, explorando as confianças reportadas naquela classificação. Ou seja, considerando sucessivos conjuntos de teste $t_1, t_2 \cdots t_n$, seleciona-se como treino do LAC para a classificação do teste t_i, os usuários do conjunto t_{i-1} que foram classificados com uma confiança superior a um certo limiar. Para a classificação de t_1, um conjunto de treino inicial é necessário, podendo ser obtido pela primeira estratégia. O algoritmo 1 apresenta a estratégia utilizada. Ele garante que pelo menos α% dos usuários de cada classe sejam selecionados, mantendo uma confiança mínima uniforme entre todas as classes.

Algoritmo 1: Dados os usuários classificados pelo LAC na iteração anterior, faça:
1. Ordene os usuários de cada classe em ordem decrescente de confiança;
2. Selecione α% dos usuários de cada classe, ordenados anteriormente;
3. Seja c_i^{min} a menor confiança dos usuários selecionados da classe i ($i = 1..4$);
4. Seja $c = min(c_1^{min}, c_2^{min}, c_3^{min}, c_4^{min})$;
5. Selecione para o conjuto de treino todos os usuários que possuem confiança $\geq c$, mantendo, para cada um, a classe definida pelo LAC na iteração anterior.

Neste trabalho utilizou-se a primeira estratégia, baseada no algoritmo de agrupamento, apenas na iteração inicial e continuou-se o treinamento a partir do resultado da classificação anterior para as iterações seguintes. Esse enfoque se baseia no fato de que, considerando que um número suficiente de bons exemplos de treinos sejam fornecidos, técnicas de classificação supervisionadas (p.ex., LAC) tendem a ser superiores às técnicas não supervisionadas (p.ex., algoritmos de agrupamento) [Veloso et al. 2006]. Note que a coleção de treino pode ser estendida e/ou refinada para incluir exemplos pré-classificados por outros meios, potencialmente mais confiáveis, se tais exemplos estiverem disponíveis. Por exemplo, assim como feito na iteração inicial, havendo conhecimento sobre usuários locais denunciados como *spammers*, os mesmos poderiam ser incluídos no treino das iterações seguintes. Como ficará claro na Seção 3, optou-se por não incluir tais usuários no conjunto de treino nas iterações seguintes para que os mesmos pudessem ser utilizados

para validação do método proposto.

A estratégia proposta é completamente automatizada e não exige esforço manual de classificação. Note que a natureza iterativa do processo, que utiliza resultados da iteração anterior como treino da próxima iteração, pode afetar a classificação ao longo do tempo. Entretanto, nos experimentos realizados, observou-se que os padrões de cada classe de usuários se mantêm estáveis em duas bases de dados incluindo tráfego em 2009 e 2010. Mais ainda, a classificação dos usuários da base de 2010, seguindo a abordagem iterativa descrita, apresentou excelente efetividade (Seção 3). De qualquer maneira, considera-se que, para melhor refletir os padrões de comportamento dos usuários, que podem evoluir com o tempo, e também para interromper uma possível propagação de erros, seja necessário, periodicamente, a aplicação de um conjunto de treino obtido por métodos externos (como a primeira estratégia proposta), reiniciando um novo ciclo de iterações.

3. Avaliação e Resultados

Esta seção apresenta uma avaliação do SpaDeS, descrevendo as bases de dados utilizadas e o procedimento de avaliação, e discutindo os principais resultados obtidos.

3.1. Bases de Dados

Este trabalho utiliza 4 bases de dados diferentes, sendo que duas delas refletem o tráfego SMTP de um provedor de Internet de banda larga e duas contêm listas de usuários daquele provedor que foram denunciados como *spammers* através do endereço `abuse` daquele provedor durante o período considerado.

Cada base de dados de tráfego contém um log de tráfego e um log do serviço DHCP do provedor, ambos cobrindo um mesmo período. Os logs de tráfego são formados por *transações* que representam uma conexão TCP ou um fluxo de dados UDP, contendo informações como endereços IP de origem e de destino, serviço/protocolo utilizado, data/hora inicial, duração e volume de bytes enviados e recebidos. Os logs do serviço DHCP permitem associar transações e usuários através do mapeamento dos endereços físicos de suas máquinas (*MAC addresses*) para os endereços IP fornecidos pelo provedor, com base nas informações de data e hora presentes nos dois logs. Vale ressaltar que os dados dos usuários foram anonimizados, por questões de segurança e privacidade.

As bases de dados de tráfego cobrem os períodos de 01 a 28 de março de 2009 e 12 de junho a 09 de julho de 2010. A base de 2009 contém 40,6 milhões de transações associadas a 44,2 mil usuários. Já a de 2010 contém 45,6 milhões de transações associadas a 48 mil usuários. Foram filtradas as transações SMTP, restando 6,3 milhões de transações SMTP associadas a 5.479 usuários na base de 2009, e 5 milhões de transações SMTP associadas a 5.389 usuários na base de 2010.

As duas outras bases de dados contêm denúncias recebidas pelo endereço `abuse` do provedor durante os períodos das bases de tráfego, identificando certos usuários como *spammers* Os e-mails de denúncia informam o endereço IP de origem do *spam* e a data/hora do seu recebimento e estão no formato ARF (*Abuse Reporting Format*), utilizado para mensagens desse tipo. Foi desenvolvida uma ferramenta de extração para processar essas mensagens e realizar a junção das mesmas com as transações SMTP, possibilitando a identificação de usuários denunciados. Dessa forma, foram identificados 67 e 93 *spammers* nas bases de 2009 e 2010, respectivamente. Para todos esses usuários, os

endereços IPs e as datas/horas listados nas denúncias coincidiram com dados de transações realizadas, listadas nas bases de tráfego utilizadas.

3.2. Procedimento de Avaliação

A avaliação consistiu de dois experimentos de classificação, um com as bases de 2009 e outro com as bases de 2010. Para a classificação da base de 2009, foram selecionados M=30 usuários mais próximos do centróide de cada uma das classes de usuários legítimos (Seção 2.1). Além disso, dos 67 usuários denunciados como *spammers* identificados na base de 2009, 40 são da classe 3 e 27 da classe 4. Assim, essas duas classes são formadas principalmente por *spammers* e são o principal alvo do método de detecção. O conjunto de treino foi então composto por esses 127 usuários, e para teste foram utilizados todos os usuários da base de 2009 *que não estavam no conjunto treino*.

Para o segundo experimento de classificação, realizado sobre a base de 2010, foram utilizados como treino usuários selecionados pelo algoritmo 1 considerando o resultado da classificação da base de 2009 (Seção 2.3). Foi utilizado α = 20%, o que resultou em uma confiança mínima para todos os usuários selecionados de 64%. No total, foram selecionados 787, 605, 621 e 52 usuários das classes 1, 2, 3 e 4, respectivamente.

As seções seguintes apresentam os principais resultados dos dois experimentos. Para o primeiro experimento, não foi possível quantificar a efetividade da classificação, uma vez que os dados do `abuse`, que poderiam servir como base de comparação, foram usados no treino. Optou-se nesse caso por analisar os padrões de comportamento dos usuários classificados em cada classe. A principal validação quantitativa é feita sobre os resultados do segundo experimento: além da avaliação dos padrões identificados, utilizou-se a lista de usuários denunciados como *spammers* em 2010 e foi feita uma inspeção manual de uma amostra dos demais usuários para estimar a efetividade da classificação.

3.3. Classificação da Base de Dados de 2009

O conjunto de treino é composto por usuários selecionados, que representam as quatro classes identificadas anteriormente [Castilho et al. 2010]. Para cada métrica analisou-se a média e o coeficiente de variação CV (razão entre desvio padrão e a média), computados para os usuários de cada classe. As classes 1 e 2 apresentam comportamentos razoavelmente bem uniformes, principalmente com relação aos números de transações, número de remetentes distintos, número de servidores contactados e IAT das transações. A classe 1 compreende usuários que fazem muito baixo uso do correio eletrônico, com apenas uma transação no período coberto pela base. Já a classe 2 compreende usuários com um nível de atividade um pouco mais alto, realizando tipicamente uma transação SMTP a cada 3 dias (IAT médio igual a 70,43 horas). Para ambas as classes, os números de remetentes identificados (1 e 1,70, em média) e de servidores SMTP distintos acessados (1 e 1,53, em média) são baixos, demonstrando um comportamento esperado de usuários legítimos.

Já as classes 3 e 4, representativas de *spammers* e definidas pelos usuários denunciados, revelam padrões bem distintos. Embora ambas apresentem números de transações, remetentes e servidores SMTP superiores aos das classes 1 e 2, a classe 4 revela um padrão muito mais abusivo, com cada usuário enviando 37.841,48 transações SMTP (uma a cada 36 segundos), utilizando 21.660,22 remetentes distintos e acessando

Tabela 1. Classificação dos Usuários da Base de 2009.

	Classe 1	**Classe 2**	**Classe 3**	**Classe 4**
Número de usuários	1.422	2.938	927	65
	Média (CV)	**Média (CV)**	**Média (CV)**	**Média (CV)**
Número de transações SMTP	1 (0)	25,71 (9,61)	2.697,54 (1,53)	39.108,21 (1,05)
Número de remetentes distintos	1 (0)	4,14 (1,93)	1.202,50 (1,39)	16.032,23 (0,98)
Número de servidores SMTP distintos	1 (0)	5,33 (2,32)	1.261,37 (1,17)	11.766,04 (0,60)
Tamanho das transações SMTP (KB)	646,81 (4,34)	611,40 (7,51)	9,64 (12,83)	1,67 (0,64)
Distância geodésica entre os IPs (km)	2.657 (1,32)	3.133 (1,1)	8.081 (0,22)	8.352 (0,16)
IAT das transações SMTP (h)	672 (0)	55,34 (1,02)	0,57 (2,45)	0,01 (0,55)

14.704,89 servidores SMTP distintos, em média. De fato, essas classes revelam dois tipos de *spammers* distintos: um envia o maior número de mensagens possível (classe 4) e o outro (classe 3) envia mensagens utilizando, possivelmente, um controle de fluxo com o objetivo de disfarçar sua presença [John et al. 2009]. É interessante notar também os valores médios de IAT muito baixos e as distâncias geodésicas muito mais altas, para ambas as classes, padrões esperados para *spammers*. Por fim, vale também ressaltar os tamanhos de transações muito menores que os das classes 1 e 2, em consistência com estudos anteriores que demonstraram que *spams* tendem a ser menores que mensagens legítimas [Gomes et al. 2007].

Os resultados da classificação são apresentados na Tabela 1, que sumariza as principais características dos usuários em cada classe. A classe 1 é formada por 1.422 usuários, que representam 26,56% do total de usuários na base de 2009, mas são responsáveis por menos de 1% do total de transações SMTP enviadas. Em consistência com o conjunto de treino, esta classe se mostrou a mais uniforme: todos os usuários fizeram apenas uma transação SMTP, utilizando portanto apenas um remetente e acessando apenas um servidor SMTP no período de 28 dias.

A classe 2, composta por 2.938 usuários (54,9% do total), apresenta uma variação muito grande em seus usuários, principalmente com relação ao número de transações SMTP, que varia de 2 a 11.804. Apesar deste limite superior muito alto, notamos que apenas 2% dos usuários desta classe realizaram mais que 100 transações no período. Além disto, cerca de 90% dos usuários acessaram apenas 10 servidores SMTP distintos (5,33, em média) e utilizaram menos de 10 remetentes distintos (4,14, em média). Todas essas características indicam que grande parte dos usuários que compõem esta classe são legítimos. Alguns poucos usuários com um número de transações, número de remetentes e/ou número de servidores SMTP muito altos podem representar *spammers* erroneamente classificados como legítimos (falsos negativos), ou usuários com redes locais com diversos usuários.

A classe 3, formada por 927 usuários (17,32% do total), também apresenta grande variabilidade entre os usuários. O número de transações SMTP, por exemplo, varia entre 2 e 41.227, embora a média seja bastante alta (2.697,68 transações). De fato, mais de 50% dos usuários realizaram mais de 1.000 transações SMTP, utilizaram mais de 500 remetentes distintos e acessaram mais de 700 servidores SMTP. Além disso, o IAT médio apresentado por 90% dos usuários é menor que 10 minutos, o que indica que a cada 10 minutos o usuário realiza uma transação SMTP. Com base nestes dados, podemos supor que esses usuários estejam infectados por *malwares*, agindo como *bots* que utilizam controle de fluxo [John et al. 2009], ou seja, enviam *spams* com uma frequência relativamente

Tabela 2. Classificação dos Usuários da Base de 2010.

	Classe 1	**Classe 2**	**Classe 3**	**Classe 4**
Número de usuários	1.821	2.656	836	76
	Média (CV)	**Média (CV)**	**Média (CV)**	**Média (CV)**
Número de transações SMTP	2,78 (9,82)	27,99 (8,79)	2.892.91 (1,53)	34.018,17 (0,70)
Número de remetentes distintos	1 (0)	3,32 (1,37)	1.341,38 (1,73)	22.064,07 (0,73)
Número de servidores SMTP distintos	1 (0)	3,04 (1,81)	1.192,24 (1,52)	14.234,76 (0,50)
Tamanho das transações SMTP (KB)	891,55 (4,53)	826,45 (4,29)	22,64 (5,49)	2,23 (0,78)
Distância geodésica entre os IPs (km)	3.968 (0,91)	4.136 (0,82)	7.653 (0,29)	8.572 (0,05)
IAT das transações SMTP (h)	535,51 (0,49)	58,04 (0,96)	1,00 (2,05)	0,02 (0,50)

baixa (do ponto de vista de ferramentas automatizadas) para dificultar sua detecção por sistemas *anti-spam*. É importante ressaltar que essa maior variabilidade dos usuários nas classes 2 e 3 era esperada, uma vez que elas representam comportamentos de fronteira, que podem ser difíceis de serem distinguidos pelo classificador. Assim como discutido para a classe 2, conjectura-se que alguns usuários classificados como da classe 3 podem de fato serem falsos positivos.

Assim como a classe 1, a classe 4 apresenta pouca variabilidade, com CVs variando entre 0,15 e 1,05. Consistente com o conjunto de treino, foram identificados 65 novos usuários com um padrão muito abusivo de uso do SMTP, claramente relacionados à atividade de envio de *spam*. Embora representem apenas 1,21% dos usuários, eles são responsáveis por quase 40% de todas as transações SMTP realizadas.

3.4. Classificação da Base de Dados de 2010

A Tabela 2 mostra os resultados da classificação da base de dados de 2010, utilizando como conjunto de treino usuários selecionados a partir do resultado da classificação da base de 2009, conforme discutido na Seção 3.2. Note que, em termos gerais, os usuários de cada classe mantêm padrões de comportamento bastante semelhantes aos usuários da mesma classe na base de 2009. Por exemplo, as classes 1 e 2, que contabilizam 83% de todos usuários mas apenas 1,6% das transações SMTP realizadas, revelam padrões bastantes consistentes com usuários legítimos: números pequenos de transações, remetentes e de servidores distintos e longos períodos de inatividade.

Já as classes 3 e 4, que representam 17% dos usuários e 98,4% das transações, mais uma vez demonstram padrões muito abusivos. Usuários da classe 4 fazem um uso muito mais intenso de tráfego SMTP que os da classe 3. Ainda assim, a classe 3 possui características muito pouco prováveis para usuários legítimos. Por exemplo, dificilmente um usuário legítimo realizaria 2.800 transações em um período de 28 dias (uma média de 100 transações por dia), utilizando 1.300 remetentes distintos e acessando 1.100 servidores SMTP distintos (pouco mais de 2,5 transações para cada servidor).

No geral, podemos concluir que, consistentemente nas duas bases analisadas (2009 e 2010), as classes de usuários legítimos: (i) realizam poucas transações SMTP; (ii) utilizam poucos remetentes distintos; (iii) acessam poucos servidores distintos; (iv) possuem alto intervalo de inatividade entre as transações; (v) possuem alta variabilidadade do tamanho médio das transações SMTP, uma vez que usuários legítimos podem enviar tanto mensagens apenas de texto, quanto mensagens com anexos extensos, contendo vídeos e imagens; e (vi) realizam suas transações principalmente para servidores brasileiros ou servidores localizados nos Estados Unidos.

Em contraste, as classes de *spammers*: (i) realizam um número alto de transações SMTP; (ii) utilizam um número elevado de remetentes distintos; (iii) acessam vários servidores SMTP distintos; (iv) efetuam as transações com um período de inatividade muito baixo, sendo muitas vezes de apenas segundos; (v) possuem tamanho médio das transações SMTP baixos, uma vez que, normalmente, *spams* possuem apenas texto; e (vi) tendem a apresentar distância geodésica média maior que usuários legítimos.

Como discutido na Seção 3.2, além da avaliação dos padrões de comportamento detectados, também foi feita uma validação da classificação da base de 2010, utilizando a lista de usuários denunciados em 2010 (não utilizada como parte do treino) e inspecionando manualmente uma amostra aleatória de 5% dos usuários de cada classe (exceto classe 4). Essa taxa de amostragem garante, com confiança de 90%, um erro inferior a 12% nas estimativas. Como trabalho futuro, pretendemos realizar a inspeção manual com uma fração maior dos usuários.

Tabela 3. Eficácia da Classificação: Estimativas de Taxa de Acerto, Falsos Positivos e Falsos Negativos (Base de Dados 2010).

	Classe Predita			
Classe Real	**1**	**2**	**3**	**4**
1 (legítimo)	100%			
2 (legítimo)	0,7%	95,5%	3,8%	
3 (*spammer*)		15,9%	84,1%	
4 (*spammer*)				100%

Dos 93 usuários denunciados, 31 se encontram na classe 4 e 62 na classe 3. Ou seja, 40,78% e 7,41% dos usuários classificados nas classes 4 e 3, respectivamente, foram de fato denunciados como *spammers*. Foram inspecionados todos os 45 usuários restantes da classe 4, concluindo que todos apresentam um comportamento consistente com *spammers* bastante abusivos. Logo, todos os usuários classificados na classe 4 foram corretamente identificados como *spammers*. Além disto, verificou-se que 35 de 44 usuários selecionados da classe 3 para inspeção manual foram corretamente classificados, enquanto os 7 restantes apresentavam um comportamento aceitável para usuários legítimos da classe 2, sugerindo assim falsos positivos.

Quanto à classificação de usuários em usuários legítimos, apenas 1 de todos os usuários selecionados da classe 1 foi classificado erroneamente, apresentando um comportamento mais condizente com a classe 2. Note que, ainda assim, esse usuário foi classificado como legítimo. Já para a classe 2, 127 dos 132 usuários selecionados foram corretamente classificados, enquanto 5 tinham um padrão mais próximo de *spammers* da classe 3, sendo assim considerados falsos negativos.

Esses resultados, computados sobre os usuários amostrados das classes 1, 2 e 3 e sobre todos os usuários da classe 4, são sumarizados na Tabela 3. Cada linha representa uma classe atribuída aos usuários por inspeção ou pelo `abuse` (*classe real*), enquanto as colunas representam as classes assinaladas pelo SpaDeS (classes preditas). Os valores indicam as porcentagens das amostras de uma classe real que foram atribuídas à classe predita indicada. Dessa forma, a diagonal indica a taxa de acerto para cada classe e as demais posições indicam predições incorretas. Considerando a classificação de todos usuários nas super-classes "legítimos" (classes 1 e 2) e "*spammers*" (classes 3 e 4), o método SpaDeS apresentou uma excelente taxa de acerto, identificando corretamente 98%

e 94% dos usuários legítimos e *spammers*, respectivamente, apresentando assim taxas de falsos positivos e de falsos negativos de somente 2% e 6%, respectivamente.

Em suma, o SpaDeS apresentou uma excelente efetividade na detecção de *spammers* na rede de origem. Não se conhece nenhum outro método que se proponha a fazer esta detecção tão próximo à fonte dos *spams*, reduzindo assim o tráfego na rede. Uma comparação do SpaDeS com outros métodos de detecção de *spammers* disponíveis na literatura e que abordam o problema em outros pontos do sistema é bastante difícil considerando as bases de dados disponíveis para validação do método proposto, uma vez que tais métodos utilizam informações diferentes das exploradas pelo SpaDeS.

4. Conclusões

Neste trabalho foi apresentado, aplicado e validado com dados reais um método para identificação e detecção de *spammers* na rede origem. Este método, denominado SpaDeS (*Spammer Detection at the Source*), tem como principal componente um algoritmo de classificação supervisionada – *Lazy Associative Classification* (LAC) – e utiliza métricas que não requerem a inspeção do conteúdo da mensagem enviada, para classificar os usuários como sendo legítimos ou *spammers*. O SpaDeS apresentou uma excelente efetividade, com taxa de acerto de 98% para usuários legítimos e 94% na classificação de *spammers*. Como trabalho futuro, propõe-se o aprimoramento do método e a construção e validação de um sistema que viabilize o uso do SpaDeS em tempo real.

Agradecimentos

Esta pesquisa foi financiada pelo Projeto REBU(CTInfo/CNPq 55.0995/2007-2) e pelo PROBIC/PUC-Minas.

Referências

Castilho, L. H. D., Las-Casas, P. H. B., Dutra, M. D., Ricci, S. M. R., Marques-Neto, H. T., Ziviani, A., Almeida, J. M., e Almeida, V. (2010). Caracterização de tráfego SMTP na Rede de Origem. Em *XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, Gramado, Brasil.

Gomes, L. H., Cazita, C., Almeida, J. M., Almeida, V., e Jr., W. M. (2007). Workload Models of Spam and Legitimate E-mails. *Performance Evaluation*, 64(7-8):690–714.

Guerra, P. H. C., Pires, D. E. V., Guedes, D., Jr., W. M., Hoepers, C., e Steding-Jessen, K. (2008). A Campaign-based Characterization of Spamming Strategies. Em *Proceedings of the Fifth Conference on Email and Anti-Spam*, pág. 1–10, Mountain View, CA, USA.

John, J., Moshchuk, A., Gribble, S. D., e Krishnamurthy, A. (2009). Studying Spamming Botnets Using Botlab. Em *6th USENIX Symp. on Networked Systems Design and Implementation*, Boston, EUA.

Las-Casas, P. H. B., Guedes, D., Marques-Neto, H. T., Ziviani, A., e Almeida, J. M. (2011). Detecção de Spammers na Rede de Origem. Em *XXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, Campo Grande, Brasil.

MessageLabs (2010). MessageLabs Intelligence: November 2010. Online.

Veloso, A., Meira, W., e Zakib, M. J. (2006). Lazy associative classification. Em *Sixth International Conference on Data Mining*, Hong Kong, China.

Detecção de *Spams* Utilizando Conteúdo Web Associado a Mensagens *

Marco Túlio C. Ribeiro[1], **Wagner Meira Jr.**[1], **Dorgival Guedes**[1], **Adriano Veloso**[1]

[1]Departamento de Ciência da Computação - Universidade Federal de Minas Gerais
Belo Horizonte, MG

{marcotcr,meira,dorgival,adrianov}@dcc.ufmg.br

Abstract. *In this paper we propose a strategy of spam classification that exploits the content of the Web pages linked by e-mail messages. We describe a methodology for extracting pages linked by spam and we characterize the relationship among those pages and the spam messages. We then use a machine learning algorithm to extract features found in the web pages that are relevant to spam detection. We demonstrate that the use of information from linked pages can significantly outperforms current spam classification techniques, as portrayed by Spam Assassin. Our study shows that the pages linked by spams are a very promising battleground, where spammers do not hide their identity, and that this battleground has not yet been used by spam filters.*

Resumo. *Neste trabalho propomos uma estratégia de detecção de* spams *que explora o conteúdo das páginas Web apontadas por mensagens. Descrevemos uma metodologia para a coleta dessas páginas, caracterizamos a relação entre as páginas e as mensagens de* spam *e, em seguida, utilizamos um algoritmo de aprendizado de máquina para extrair as informações relevantes para a detecção de* spam*. Mostramos que a utilização de informações das páginas mencionadas melhora significativamente a classificação de* spams *e* hams*, gerando um baixo índice de falsos positivos. Nosso estudo revela que as páginas apontadas pelos* spams *ainda são um campo de batalha não explorado pelos filtros, onde os* spammers *não se preocupam em esconder a sua identidade.*

1. Introdução

Spam é um problema que tem acompanhado o desenvolvimento e popularização da Internet e tem sido um meio usual de enviar mensagens relacionadas à obtenção de dados pessoais com objetivos ilícitos (*phishing*) e para a disseminação de códigos maliciosos [Milletary 2005]. Os servidores de correio eletrônico têm que lidar com o fato de que entre 82% e 92% das mensagens recebidas são *spam* [MAAWG 2009]. O prejuízo que essa prática acarreta à sociedade é avaliado em bilhões de dólares [Sipior et al. 2004].

O problema do *spam* é análogo a uma corrida armamentista (chamada comumente de *spam arms race* [Guerra et al. 2010]). Isso significa que há uma evolução constante tanto de técnicas de detecção de mensagens indesejadas como da sofisticação das tecnologias adotadas pelos *spammers*. Nesta corrida, cada um tenta se sobrepor

*Este trabalho é parcialmente financiado por CNPq, CAPES, Finep, FAPEMIG e Bolsa UOL Pesquisa.

ao outro e a mudança na estratégia de um lado induz a mudanças na estratégia do adversário. Os filtros anti-*spam* adotam, comumente, estratégias baseadas em filtragem de conteúdo de mensagens, como o *Spam Assassin* [SpamAssassin 2008] e listas de bloqueio [Cook et al. 2006]. As duas estratégias são complementares, uma vez que a primeira trata do conteúdo da mensagem em si e a segunda trata das estratégias que o *spammer* utilizou para disseminar a mensagem. Vale notar que o objetivo final do *spammer* é ser atrativo o suficiente para o receptor tomar alguma ação — seja esta comprar algum produto ou seguir algum elo de navegação. Os filtros baseados em conteúdo obrigam o *spammer* a ofuscar suas mensagens, de forma que o *spammer* tem um compromisso entre manter o e-mail legível (e atingir menos caixas de entrada) e comprometer a legibilidade, possivelmente atingindo mais usuários. As estratégias dos *spammers* para evitar as listas de bloqueio, por outro lado, não comprometem a "qualidade" das suas mensagens.

Com o advento e popularização de técnicas de contaminação de máquinas por códigos maliciosos que podem transformar qualquer máquina de usuário em um *bot*, uma ferramenta para redistribuição de *spam* (entre outros usos), estratégias baseadas em listas de bloqueio têm se tornado menos eficientes [Ramachandran et al. 2006]. A iminente troca de versão do protocolo IP (da versão 4 para a versão 6) provavelmente criará dificuldades ainda maiores para o sucesso das listas de bloqueio, uma vez o aumento da faixa de endereços disponíveis tornará mais difícil manter as listas de bloqueio atualizadas.

Em um trabalho recente, Guerra et al. (2010) caracterizaram a adaptação dos filtros em relação às estratégias utilizadas por *spammers* e mostraram como certas características são exploradas ao longo do tempo. Um aspecto do *spam* que ainda não é explorado pelos filtros e, consequentemente, não é ofuscado pelos *spammers*, é o conteúdo das páginas Web apontadas pelas *URLs* contidas nos *spams*. Muitas vezes essas páginas estão até mesmo fora do controle dos *spammers*, pertencendo a empresas externas que os contratam para divulgar seus produtos. Pu e Webb (2006) mostraram que pelo menos uma *URL* aparece em 85% a 95% dos *spams* presentes em todos os meses analisados por eles. Já Guerra et al. (2008) reportam que 96,5% das campanhas de *spam* observadas por eles continham pelo menos uma *URL*. Esse números indicam que técnicas que considerem o conteúdo das páginas como evidência para detecção e mitigação do *spam* pode ter impacto bastante positivo. Neste trabalho, mostramos que essas páginas podem oferecer informações valiosas acerca da natureza dos *spams*.

Utilizamos duas bases de dados históricas de *spams* e mensagens legítimas (também chamadas *hams*), *SpamArchive* [Guenter 2010] e *Spam Assassin*[1], para construir uma base de dados que relaciona mensagens e páginas. Utilizando essa base como estudo de caso, mostramos que a utilização das páginas melhora a detecção de *spam* em aproximadamente 10%, sem causar um aumento no índice de falsos positivos. As contribuições deste trabalho, portanto, são (i) a disponibilização de uma base de dados que relaciona mensagens de *spam* às páginas apontadas por elas e (ii) a proposta de uma metodologia para a detecção de *spam* através do conteúdo das páginas apontadas pelas mensagens. Mostramos que as páginas apontadas por mensagens são um campo de batalha promissor e que a informação proveniente das páginas ainda não é explorado pelos filtros.

Este artigo descreve os principais resultados de um projeto de iniciação científica

[1]Disponível em http://spamassassin.apache.org/publiccorpus/

realizado no segundo semestre do ano de 2010. O aluno de iniciação científica, autor deste artigo, foi o idealizador e principal responsável pela realização deste projeto de pesquisa, que foi orientado pelos professores Wagner Meira Jr. e Dorgival Guedes, e contou com o apoio do professor Adriano Veloso, no que diz respeito ao seu algoritmo de classificação de documentos [Veloso et al. 2006]. Uma versão estendida deste trabalho será apresentada como artigo completo no Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos 2011.

2. Trabalhos Relacionados

Em [Upasana and Chakravarty 2010], é apresentado um *survey* sobre técnicas de identificação de *spam* baseadas em classificação de texto. Um filtro que faz uso de grande parte das estratégias conhecidas atualmente é o *Spam Assassin* [SpamAssassin 2008], que utiliza filtros bayesianos e listas de bloqueio DNS. Além disso, o *Spam Assassin* também conta com um conjunto de regras, geralmente representadas por expressões regulares, que são comparadas com os campos *body* ou *header* de cada mensagem. Ou seja, o *Spam Assassin* é um filtro que lida tanto com características do corpo da mensagem quanto características de rede.

Em [Ramachandran et al. 2006], foi feito um estudo sobre a efetividade de listas de bloqueio baseadas em *DNS* em relação a *botnets*. Os resultados preliminares indicam que apenas 5% de todos os *IP*s dos bots estudados apareciam na lista de bloqueio utilizada. Em [Sinha et al. 2008], é feita uma avaliação de várias listas de bloqueio, e mostra-se que as listas de bloqueio apresentam um número significativo de falsos negativos e falsos positivos. Devido aos problemas potenciais de listas de bloqueio, é necessário a descoberta de novas técnicas.

Em [Webb 2006] construiu-se uma base de dados com páginas apontadas por *spams* da base de dados *Spam Archive* [Guenter 2010] no período entre novembro de 2002 e janeiro de 2006. Porém, essa base de dados tem como foco *Web Spam*, e não relaciona cada página a uma mensagem de *spam*, de forma que não pudemos utilizá-la.

Dessa forma, nosso trabalho é o primeiro, até onde sabemos, a empregar o conteúdo de páginas como evidência para identificação de *spams*, e o primeiro a disponibilizar uma base de dados que relaciona *spams* a páginas *Web*.

3. Detecção de Spams utilizando conteúdo Web associado a mensagens

A técnica para detecção de *spam* que propomos se baseia nas páginas apontadas por *URLs* em mensagens de *spam*. Apesar de o acesso a essas páginas implicar em um custo extra, em uma implementação em produção nossa técnica poderia funcionar de forma complementar a outras estratégias de classificação de *spam*. Ao se analisar uma mensagem, carrega-se as páginas identificadas por URLs contidas na mesma e verifica-se se essas páginas possuem conteúdo que seja associado com campanhas de *spam* — da mesma forma que um filtro de conteúdo avalia o corpo da mensagem, mas nesse caso considerando o conteúdo da página. Esse conteúdo é então combinado com as outras características da mensagem (dadas pelo *Spam Assassin*) e o par (mensagem, página) é classificado para identificar *hams* e *spams*. Nesta seção descrevemos as operações em cada uma dessas etapas. O processo é ilustrado pelo diagrama apresentado na Figura 1 e pelo exemplo apresentado ao final.

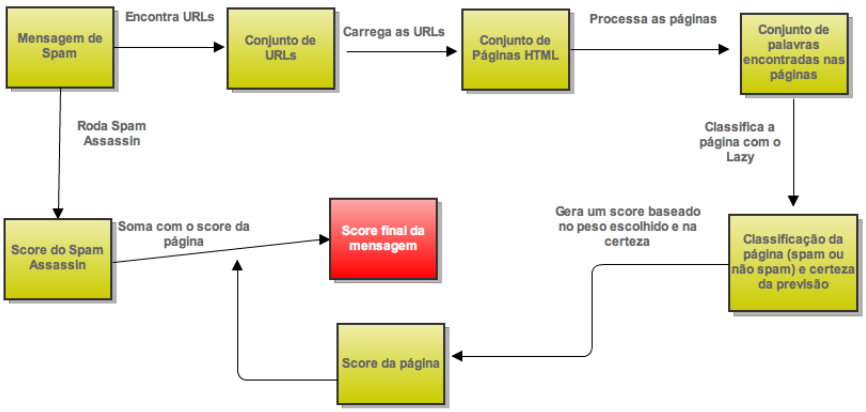


Figura 1. Diagrama ilustrativo do processo de detecção de spams proposto.

3.1. Processamento da página

Após a identificação das URLs nas mensagens e o acesso às páginas por elas identificadas, removemos todas as *tags* HTML e *scripts*, deixando apenas as palavras que o usuário veria se o mesmo carregasse a página.

3.2. Classificação da página

Para classificar a página, utilizamos um algoritmo de aprendizado associativo sob demanda [Veloso et al. 2006]. Optamos por esse algoritmo por (i) ter bom desempenho para utilização em tempo real (o algoritmo consegue classificar, em média, 111 páginas por segundo), (ii) gerar um modelo de boa legibilidade (que pode ser facilmente transformado em um conjunto de expressões regulares, como as do *Spam Assassin*) e (iii) ser bem calibrado [Veloso et al. 2008]. Essa última característica significa que o algoritmo gera uma probabilidade de cada previsão estar certa, ou seja, as previsões com mais certeza são mais confiáveis. O algoritmo produz regras do tipo $\chi \rightarrow c$, onde χ é um conjunto de termos e c é a classe (*spam* ou *ham*). Cada uma dessas regras tem uma certa frequência (que chamamos de suporte) e uma confiança, que é dada pelo número de instâncias que são classificadas corretamente pela regra dividido pelo número de instâncias que contém o conjunto de termos χ. O resultado final da classificação de cada página é a classe predita pelo algoritmo e a certeza da predição, medida entre 0 e 1. Quanto maior o valor da certeza, mais confiável é a predição.

Uma das dificuldades da detecção de *spam* é a assimetria entre o custo associado aos diferentes tipos de erro. Um falso negativo simplesmente causa alguma irritação, *i.e.,* o usuário recebe uma mensagem indesejada. Por outro lado, um falso positivo pode ser crítico: uma mensagem importante pode nunca chegar à caixa de entrada do usuário, se for filtrada pelo servidor [Fawcett 2003]. Em virtude do alto custo de se classificar uma mensagem legítima como *spam*, empregamos também a noção de *custo de classificação*.

O custo de uma classe mede o quão caro é classificar incorretamente uma instância dessa classe. Ao ponderar todas as regras obtidas para uma determinada instância, o algoritmo faz uma soma ponderada das regras levando em conta a confiança e o custo de cada classe, de forma a valorizar mais regras que apontam para classes de custo mais alto. Isso implica que com o aumento do custo, mais indícios são necessários para que o algoritmo classifique uma página como *spam*.

Como o classificador associativo necessita de instâncias de ambas as classes (*spam* e *ham*), utilizamos uma base de dados pública de *ham*s, fornecida pelo próprio *Spam Assassin*, como já citado. Para instâncias da classe *spam*, utilizamos a base de dados descrita na seção anterior.

3.3. Classificação da mensagem

Associamos um peso p à classificação da página (que será somado à classificação da mensagem depois). A pontuação final da página é dada por $p * c$, onde c é a certeza da predição, de forma que páginas com maior chance de serem identificadas como *spam* acarretam em pontuações mais altas para as suas mensagens. Em seguida, a pontuação da página é somada com a pontuação da mensagem (dada pelo *Spam Assassin*). Vale notar que se o classificador associativo classifica uma página como *ham*, a pontuação da página, que é somada à pontuação do *Spam Assassin*, é negativa.

3.4. Exemplo ilustrativo

Apresentamos um exemplo da aplicação da nossa técnica passo a passo. Escolhemos uma mensagem de *spam* obtida do *Spam Archive* no mês de outubro de 2010. A Figura 2 mostra o corpo da mensagem (omitimos grande parte dos cabeçalhos por questão de espaço). Percebe-se que a mensagem é bem concisa e ofuscada. O *Spam Assassin*, com listas de bloqueio ativadas, encontra as seguintes regras:

Regra	Significado	Pontuação
HTML_MESSAGE	Há HTML na mensagem	0.001
RCVD_IN_BRBL_LASTEXT	Lista de bloqueio DNS BRBL	1.644
URIBL_BLACK	URL em lista de bloqueio	1.775

A pontuação resultante é 3,4 – insuficiente para classificar a mensagem como *spam*. Um excerto da página apontada pelas *URLs* dessa mensagem é ilustrado pela Figura 3.

Percebe-se, neste caso, que o conteúdo da mensagem e o conteúdo da página são totalmente diferentes. O conteúdo da página é transformado, então, em um conjunto de palavras (através do navegador *lynx*), que é entregue ao classificador associativo, que já dispunha de um conjunto de páginas de *spam* e não-*spam* como treino (no caso, as outras páginas armazenadas do *Spam Archive* e a base de dados de *ham* do *Spam Assassin*). O classificador associativo encontra um conjunto de regras, das quais alguns exemplos são:

Regra	Suporte	Confiança
$viagra \rightarrow Spam$	36.70%	99.84%
$levitra \rightarrow Spam$	34.01%	99.90%
$rather \rightarrow Ham$	2.97%	67.30%

Por fim, o resultado final do classificador associativo é que a página é *spam*, com 90% de certeza. Supondo que tenhamos pré-definido que o peso das páginas seria 4, a

```
From: Discount Rolex_Etc. <0-hef@hef.fr>
To: ----
Subject: 75% Off on Gucci, Rolex, And Loius Vuitton Handbags and Various Other items
Date: Tue, 26 Oct 2010 19:20:16 +0300
MIME-Version: 1.0
Content-Type: multipart/alternative;
   boundary="----=_hhnlkay_21_61_99"
X-Priority: 3
X-Mailer: ouddj 10
Message-ID: <4281462511.JBNTUZ0E381515@hwvxuglmwirrhn.fnezaq.biz>

------=_hhnlkay_21_61_99
Content-Type: text/plain;
   charset="windows-1250"
Content-Transfer-Encoding: quoted-printable

Stop paying more than you have to!
http://migre.me/1JHMb
------=_hhnlkay_21_61_99
Content-Type: text/html;
   charset="windows-1250"
Content-Transfer-Encoding: quoted-printable

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><HEAD>
<META http-equiv=3DContent-Type content=3D"text/html; charset=3Dwindows-1=
250">
<STYLE></STYLE>
</HEAD>
<BODY>
<a href=3D"http://su.pr/6uuaSy">Stop paying more than you have to!</a>
</BODY></HTML>

------=_hhnlkay_21_61_99--
```

Figura 2. Mensagem de *spam* extraída do *Spam Archive*



Figura 3. Pagina de *spam* apontada pela mensagem extraída do *Spam Archive*

pontuação da página seria $4 * c$, sendo c a certeza do classificador associativo. Percebe-se que o valor de c é determinante na pontuação final da mensagem, de forma que as páginas que o classificador associativo tem menos certeza recebem uma pontuação menor. Essa página, portanto, teria pontuação igual a 3,6. Somando-se a pontuação obtida pelo *Spam Assassin* com a pontuação da página, temos uma pontuação igual a 7,0 – mais do que suficiente para classificar a mensagem como *spam*.

4. Coleta da Base de Dados de Páginas de *Spam*

Em [Pu and Webb 2006], mostrou-se que pelo menos uma *URL* aparece em 85% a 95% das mensagens de *spam* no *Spam Archive* no período entre 2004 e 2006, enquanto [Guerra et al. 2008] reportaram que 96,5% de suas campanhas continham pelo menos uma *URL*. Apesar disso, a coleta de páginas de *spam* ainda é uma tarefa desafiadora. [Anderson et al. 2007] mostra que poucas páginas têm um tempo de vida maior do que 13 dias, ou seja, a coleta das páginas tem que ser feita em um período próximo do instante que a mensagem foi disseminada.

Entre julho e dezembro de 2010 nós ocoletamos as mensagens de *spam* da base de dados *Spam Archive* diariamente (a base também é atualizada diariamente), de forma a obter as mensagens de *spam* mais recentes. Em seguida, extraimos as *URLs* do corpo das mensagens e utilizamos expressões regulares simples para remover imagens e executáveis. Em seguida, carregamos e armazenamos as páginas. No caso de mensagens que continham múltiplas *URLs*, todas as *URLs* foram carregadas e armazenadas. Várias *URLs* continham redirecionamentos; nesse caso, seguimos todos os redirecionamentos e armazenamos o conteúdo final da página.

As características da base de dados obtidas são mostradas na Tabela 1. Percebe-se que a maioria das mensagens contém poucas *URLs*. Vale notar que só consideramos como parte da base de dados as mensagens para as quais pelo menos uma página foi baixada.

Tabela 1. Descrição da base de dados obtida

Número de mensagens	63.034
Número de páginas	157.114
Número médio de páginas baixadas por mensagem	2,49

5. Resultados e Discussão

Para avaliar a aplicabilidade de se construir filtros anti-*spam* a partir do conteúdo das páginas, selecionamos todas as páginas únicas da base de dados. Optamos por avaliar apenas as páginas únicas para impedir que uma campanha de mensagens apontando para a mesma página enviesasse os nossos resultados. Quando várias mensagens diferentes apontavam para a mesma página, uma delas foi selecionada aleatoriamente para a avaliação, de forma que apenas uma instância de cada página permanecesse na avaliação. Ao final, portanto, avaliamos a nossa técnica em 32929 páginas *spam*, apontadas por 12111 mensagens de *spam* e 11134 páginas *ham*, apontadas por 4927 mensagens retiradas da base de *ham* do *Spam Assassin*. Utilizamos validação cruzada para a avaliação, dividindo as páginas em 5 partições. Nas subseções seguintes mostramos o impacto da variação dos parâmetros peso e custo, além da comparação da nossa técnica com o *Spam Assassin* puro. O classificador associativo foi executado com confiança 0.3, o custo das duas classes foi igual e o peso escolhido foi 4, exceto quando indicado diferente. Esses valores foram ajustados na validação cruzada.

5.1. Impacto do parâmetro peso na detecção de *spam*

Mostramos na Figura 4 o impacto de diferentes valores de peso (que é multiplicado com a certeza da previsão do classificador associativo) no índice de falsos negativos e falsos po-

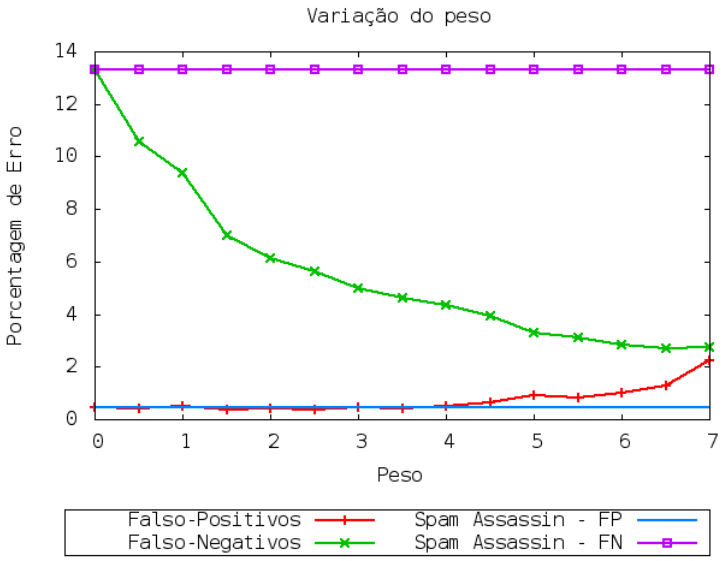


Figura 4. Falsos positivos e falsos negativos x Peso

sitivos. Mostramos também na figura o índice de falsos positivos e falsos negativos gerados através da utilização do *Spam Assassin* sem a nossa técnica, para fins de comparação. Percebe-se que com um peso de até 4, o índice de falsos positivos permanece praticamente igual ao índice de falsos positivos do *Spam Assassin*, embora o índice de falsos negativos seja consideravelmente mais baixo. Selecionamos para os experimentos seguintes, portanto, o valor de peso 4, que representa o menor índice de falsos negativos sem aumentar o índice de falsos positivos.

5.2. Impacto do parâmetro custo na detecção de *spam*

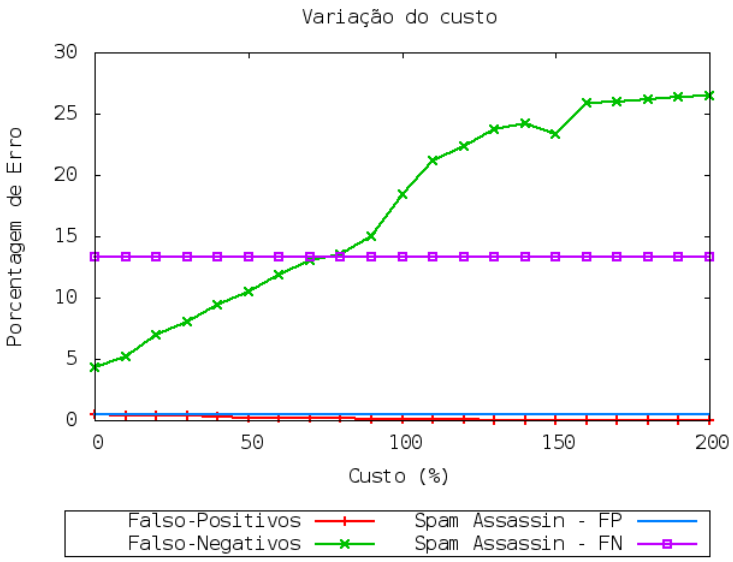


Figura 5. Falsos positivos e falsos negativos x Custo

Mostramos na Figura 5 o impacto da variação do custo no índice de falsos positivos e falsos negativos da nossa técnica. Os valores do eixo x representam a diferença

entre o custo de se classificar um *ham* como *spam* e o custo de se classificar um *spam* como *ham*. Portanto, se o valor no eixo x é 50%, isso significa que é 50% mais custoso classificar um *ham* como *spam* do que vice-versa. Naturalmente, um aumento no custo gera uma redução do índice de falsos positivos e um aumento no índice de falsos negativos. Com um custo maior do que 70%, nossa técnica passa a classificar *spams* com menos eficácia do que o *Spam Assassin*, embora o número de falsos positivos chegue a 0. É interessante notar que esse compromisso é ajustável na nossa técnica, através do parâmetro custo. Cabe ao usuário da técnica definir o custo de acordo com a sua necessidade.

6. Conclusões e Trabalhos Futuros

Neste trabalho, mostramos que as páginas *Web* apontadas por mensagens de *spam* podem ser utilizadas com sucesso para a classificação dessas mensagens. Nossa proposta consiste em utilizar as páginas como complemento a outras estratégias já utilizadas de classificação de mensagens de *spam*.

Mostramos na seção de trabalhos relacionados que estratégias de filtragem de *spam* convencionais não fazem uso das páginas. A grande maioria dos *spams* contém *URL*s [Pu and Webb 2006], e portanto podem ser filtrados pela nossa técnica. Mostramos também que uma das estratégias mais comuns para a filtragem de *spams*, o uso de listas de bloqueio, esta perdendo sua efetividade [Ramachandran et al. 2006] [Sinha et al. 2008], e portanto é necessário que novas técnicas de filtragem sejam estudadas e utilizadas. Neste trabalho propomos uma técnica que explora um aspecto no qual os *spammers* ainda não escondem a sua identidade. Além disso, as páginas muitas vezes não pertencem aos *spammers*, e portanto são um campo de batalha no qual os *spammers* estão em desvantagem.

Avaliamos o uso de um algoritmo de aprendizado de máquina sob demanda [Veloso et al. 2006] para a classificação das páginas, e propomos uma forma de se agregar a classificação das páginas com a classificação tradicional das mensagens com o filtro *Spam Assassin* [SpamAssassin 2008]. Mostramos que a utilização da nossa técnica melhora a filtragem de *spam* em mais de 10%, sem inserir um número significativo de falsos positivos. Mostramos também que a quantidade de falsos positivos pode ser ajustada com a variação do parâmetro custo.

Acreditamos que o trabalho abre novas possibilidades para o desenvolvimento de estratégias de filtragem de *spam*, introduzindo um aspecto totalmente novo e ainda não explorado na literatura. Em outras palavras, as páginas apontadas pelas mensagens constituem-se em um novo campo de batalha, com o qual hoje os *spammers* não precisam se preocupar. Neste novo campo de batalha, diferentes algoritmos podem ser utilizados para a classificação das páginas, e o resultado da classificação pode ser combinado com outras técnicas. Por fim, servidores de correio eletrônico poderiam utilizar as técnicas descritas em [Guerra et al. 2008] para agrupar as mensagens em campanhas, de forma a diminuir o número de páginas a serem classificadas.

Referências

Anderson, D. S., Fleizach, C., Savage, S., and Voelker, G. M. (2007). Spamscatter: Characterizing Internet Scam Hosting Infrastructure. pages 135–148.

Cook, D., Hartnett, J., Manderson, K., and Scanlan, J. (2006). Catching spam before it arrives: domain specific dynamic blacklists. In *ACSW Frontiers '06: Proceedings of*

the 2006 Australasian workshops on Grid computing and e-research, pages 193–202, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.

Fawcett, T. (2003). "in vivo"spam filtering: a challenge problem for kdd. *SIGKDD Explor. Newsl.*, 5:140–148.

Guenter, B. (2010). Spam archive. `http://untroubled.org/spam/`.

Guerra, P. H. C., Guedes, D., Jr., W. M., Hoepers, C., and Steding-Jessen, K. (2008). Caracterização de estratégias de disseminação de spams. In *26o Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, Rio de Janeiro, RJ.

Guerra, P. H. C., Guedes, D., Wagner Meira, J., Hoepers, C., Chaves, M. H. P. C., and Steding-Jessen, K. (2010). Exploring the spam arms race to characterize spam evolution. In *Proceedings of the 7th Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, Redmond, WA.

MAAWG (2009). Email Metrics Program: Report #5 – Third and Fourth Quarter 2008. `http://www.maawg.org/about/MAAWG_2008-Q3Q4_Metrics_Report.pdf`.

Milletary, J. (2005). Technical trends in phishing attacks. Technical report, CERT Coordination Center, Carnegie Mellon University. `http://www.cert.org/archive/pdf/Phishing_trends.pdf`.

Pu, C. and Webb, S. (2006). Observed trends in spam construction techniques: a case study of spam evolution. *Proceedings of the 3rd Conference on Email and Anti-Spam (CEAS)*.

Ramachandran, A., Dagon, D., and Feamster, N. (2006). Can dns-based blacklists keep up with bots? In *In Proceedings of the 3rd Conference on Email and AntiSpam (CEAS) (Mountain View)*.

Sinha, S., Bailey, M., and Jahanian, F. (2008). Shades of grey: On the effectiveness of reputation-based blacklists. In *Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on*, pages 57 –64.

Sipior, J. C., Ward, B. T., and Bonner, P. G. (2004). Should spam be on the menu? *Commun. ACM*, 47(6):59–63.

SpamAssassin (2008). `http://spamassassin.apache.org`.

Upasana and Chakravarty, S. (2010). A survey on text classification techniques for e-mail filtering. In *Machine Learning and Computing (ICMLC), 2010 Second International Conference on*, pages 32 –36.

Veloso, A., Jr., W. M., and Zaki, M. J. (2006). Lazy associative classification. In *ICDM*, pages 645–654. IEEE Computer Society.

Veloso, A., Jr., W. M., and Zaki, M. J. (2008). Calibrated lazy associative classification. In de Amo, S., editor, *SBBD*, pages 135–149. SBC.

Webb, S. (2006). Introducing the webb spam corpus: Using email spam to identify web spam automatically. In *In Proceedings of the 3rd Conference on Email and AntiSpam (CEAS) (Mountain View)*.

SESSÃO 2

A Hybrid Data Mining GRASP with Path-Relinking

Hugo Barbalho[1], **Isabel Rosseti**[1], **Simone L. Martins**[1], **Alexandre Plastino**[1]

[1]Department of Computer Science – Federal Fluminense University
Niterói – RJ – Brazil.

{hbarbalho,rosseti,simone,plastino}@ic.uff.br

Abstract. *The exploration of hybrid metaheuristics — combination of metaheuristics with concepts and processes from other research areas — has been an important trend in combinatorial optimization research. In this work, we developed a hybrid version of the GRASP metaheuristic which incorporates the path-relinking procedure — a memory-based intensification strategy — and a data mining module. Computational experiments showed that employing the combination of path-relinking and data mining allowed GRASP to find better results in less computational time. Another contribution of this work is the application of the path-relinking hybrid proposal for the 2-path network design problem, which improved the state-of-the-art solutions for this problem.*

Resumo. *A exploração de metaheurísticas híbridas — combinação de metaheurísticas com conceitos e processos de outras áreas — vem sendo uma importante linha de pesquisa em otimização combinatória. Nesse trabalho, desenvolvemos uma versão híbrida da metaheurística GRASP que incorpora a técnica de reconexão por caminhos e um módulo de mineração de dados. Experimentos computacionais mostraram que a combinação da técnica de reconexão por caminhos com mineração de dados contribuiu para que o GRASP encontrasse soluções melhores em um menor tempo computacional. Outra contribuição desse trabalho é a aplicação dessa proposta híbrida ao problema de síntese de redes a 2 caminhos, o que proporcionou melhores soluções para esse problema.*

1. Introduction.

Metaheuristics represent an important class of approximate techniques for solving hard combinatorial optimization problems, for which the use of exact methods is impractical. They are general purpose high-level procedures that can be instantiated to explore efficiently the solution space of a specific optimization problem.

A trend in metaheuristics research is the exploration of hybrid metaheuristics. One kind of such hybrid methods results from the combination of concepts and strategies behind two or more classic metaheuristics. Another kind corresponds to metaheuristics combined with concepts and processes from other research areas responsible for improving the original method. An instance of the latter case is the hybrid version of the GRASP metaheuristic that incorporates a data mining process, called DM-GRASP (Data Mining GRASP) [Santos and Plastino 2008].

The GRASP (Greedy Randomized Adaptive Search Procedures) metaheuristic [Feo and Resende 1989, Feo and Resende 1995], since it was proposed, has been successfully applied to solve many optimization problems, in several areas like scheduling, routing, partitioning, location and assignment [Festa and Resende 2009]. GRASP is

easy to implement and is able to obtain very good solutions in acceptable computational times [Festa and Resende 2009]. The solution search process employed by GRASP is performed iteratively and each iteration consists of two phases: construction and local search. A feasible solution is built in the construction phase, and then its neighborhood is explored by the local search in order to find a better solution. The result is the best solution found over all iterations.

Data mining refers to the automatic extraction of knowledge from datasets [Han and Kamber 2006]. The extracted knowledge, expressed in terms of patterns or rules, represents important features of the dataset at hand. The hybridization of GRASP with a data mining process was first applied to the set packing problem [Ribeiro and Martins 2004]. The basic hypothesis was that patterns found in good quality solutions could be used to guide the search, leading to a more effective exploration of the solution space. The resulting method, the DM-GRASP metaheuristic, achieved promising results not only in terms of solution quality but also in terms of execution time required to obtain good solutions. Afterwards, the method was evaluated on three other applications: the maximum diversity problem [Santos and Martins 2005], the server replication for reliable multicast problem [Santos and Plastino 2006] and the p-median problem [Plastino and Salhi 2009], and the results were equally successful.

The first contribution of this work is to show that not only the traditional GRASP metaheuristic but also GRASP procedures improved with the path-relinking heuristic — a memory-based intensification mechanism — can benefit from the incorporation of a data mining procedure to extract patterns of sub-optimal solutions in order to guide more efficiently the search for better solutions.

Path-relinking was proposed by Glover [Glover and Martí 2000] as an intensification strategy exploring trajectories connecting elite solutions obtained by tabu search or scatter search strategies. Starting from one or more elite solutions, path-relinking generates paths leading toward other elite solutions and explores them in the search for better solutions. To generate paths, moves are selected to introduce attributes in the current solution that are present in the guiding solution. Path-relinking is a strategy that seeks to incorporate attributes of high quality solutions, by favoring them in the selected moves.

In this work, we present two path-relinking hybrid strategies, called DM-GRASP-PR and MDM-GRASP-PR, which combine a data mining procedure into the GRASP with path-relinking, and show that these strategies can improve the solution quality and computational time of the original GRASP with path-relinking.

The second contribution is the application of the path-relinking hybrid proposals to solve the 2-path network design problem (2PNDP). This problem has shown to be NP-hard and many applications of this problem can be found in the design of communication networks, in which paths with few edges are sought to enforce high reliability and small delays [Ribeiro and Rosseti 2007a]. GRASP procedures with path-relinking have achieved excellent results for this problem [Ribeiro and Rosseti 2007b]. The computational experiments conducted in this work show that the implemented path-relinking hybrid strategies were able to improve the state-of-the-art solutions for the 2PNDP.

The remaining of this paper is organized as follows. In Section 2, we review the main concepts and the structure of both GRASP metaheuristic and path-relinking strat-

egy. In Section 3, we present the hybrid strategy DM-GRASP-PR developed for the 2PNDP and compare the computational results obtained by this strategy and the traditional GRASP with path-relinking. In Section 4, the strategy MDM-GRASP-PR is described and computational results are presented comparing the DM-GRASP-PR and the MDM-GRASP-PR strategies. Finally, in Section 5, concluding remarks are made and some future works are pointed out.

2. GRASP with path-relinking

GRASP [Resende and Ribeiro 2003] is a metaheuristic already applied successfully to many optimization problems [Festa and Resende 2009]. The first phase of a GRASP iteration is the construction phase, in which a complete solution is built. Since this solution is not guaranteed to be locally optimal, a local search is performed in the second phase. This iterative process is repeated until a termination criterion is met and the result is the best solution found over all iterations.

In the construction phase, the initial solution is the empty set. The components not in the solution are ranked according to a greedy function. The better ranked components form a list, called Restricted Candidate List (RCL). After this step, one component is randomly selected from this list and incorporated into the current solution. This process is repeated until the partial solution is completely built.

The solution obtained in the construction phase becomes the starting point for the local search phase — a hill-climbing process, in which the neighborhood of the solution is explored. The neighborhood of a solution is defined by a function that relates this solution with a set of other solutions. If a better solution is found, local search is performed again, considering the neighborhood of this new solution. Otherwise, the local search terminates.

Path-relinking is a technique proposed by Glover [Glover and Martí 2000] to explore possible trajectories connecting high quality solutions obtained by heuristics.

The GRASP metaheuristic is a memoryless method, because all iterations are independent and no information about the solutions is passed from one iteration to another. The objective of introducing path-relinking to a pure GRASP is to retain previous good solutions and use them as guides in the search of new good solutions. Laguna and Martí [Laguna and Martí 1999] were the first to use path-relinking within a GRASP strategy. Several improvements and successful applications of this technique can be found in the literature [Resende and Ribeiro 2005].

Basically, path-relinking is applied to a pair of solutions $\{s_i, s_g\}$ by starting from the initial solution s_i and gradually incorporating attributes from the guide solution s_g to s_i, until s_i becomes equal to s_g. There are several ways to explore the paths between them [Resende and Ribeiro 2005]: backward relinking, forward relinking, backward-and-forward relinking, periodical relinking, randomized relinking and truncated relinking. To use path-relinking within GRASP, an elite set is maintained, in which good solutions found in previous GRASP iterations are stored.

In this work, path-relinking is performed after each GRASP iteration using a solution from the elite set and a local optimum obtained after the GRASP local search. From this two solutions, the initial (s_i) and guide (s_g) solutions are defined. The set Δ composed of positions in which s_i and s_g differ is then calculated. The initial best solution

and its cost are determined. The steps of path-relinking are performed until the entire path from s_i to s_g is traversed. For every position $m \in \Delta$, let $s_i \oplus m$ be the solution obtained from s_i by changing its m-th position by that of s_g. After this, the component m^* of Δ for which $s_i \oplus m$ results in the least-cost solution is obtained. Then, m^* is removed from Δ and the current solution is updated by changing the value of its m^* position. This solution is more similar to the guide solution because one element from the initial solution was replaced by another from the guide one. If this new solution has a better cost than the current best intermediate solution, then the latter and its cost are updated. The intermediate solution is then set as the initial solution for the next step of the path-relinking.

3. The Hybrid DM-GRASP-PR Proposal

In this section, we describe the 2-path network design problem and the GRASP with path-relinking procedure developed in [Ribeiro and Rosseti 2007b] to solve this problem. Then we present the DM-GRASP-PR heuristic, which is a hybrid version of the GRASP metaheuristic with path-relinking presented in [Ribeiro and Rosseti 2007b] incorporated with a data mining process.

Let $G = (V, E)$ be a connected undirected graph, where V is the set of nodes and E is the set of edges. A k-path between nodes $s, t \in V$ is a sequence of at most k edges connecting them. Given a non-negative weight function $w : E \rightarrow R_+$ associated with the edges of G and a set D of pairs of origin-destination nodes, the *2-path network design problem* (2PNDP) consists in finding a minimum weighted subset of edges $E' \subseteq E$ containing a 2-path between every origin-destination pair in D. Applications of the 2PNDP can be found in the design of communication networks, in which paths with few edges are sought to enforce high reliability and small delays. The decision version of the 2PNDP has been proved to be NP-complete by Dahl and Johannessen [Dahl and Johannessen 2004]. In [Ribeiro and Rosseti 2007b], the authors successfully applied GRASP with path-relinking heuristics for approximately solving this problem.

3.1. GRASP-PR for 2PNDP

In this section, we review the GRASP heuristic with path-relinking (GRASP-PR) for the 2-path network design problem presented in [Ribeiro and Rosseti 2007b].

The construction phase of the GRASP with path-relinking heuristic for the 2PNDP algorithm starts with the computation from scratch of a solution x using edge weights w' that are initially equal to the original weights w. The procedure is performed until a 2-path has been computed for every origin-destination pair. Each iteration starts by the random selection of a pair (a, b) still to be routed. A shortest path Pt from a to b using the modified weights w' is calculated. Then, the weights of the edges in Pt are temporarily set to 0 for the remaining iterations. At last, the pair (a, b) is removed from the set of origin-destination pairs to be routed and the edges in Pt are inserted into the solution under construction.

Each solution x may be viewed as a collection of $|D|$ 2-paths. Given any solution x, its neighbor solutions x' may be obtained by replacing any 2-path in x by another 2-path between the same origin-destination pair.

Each GRASP iteration has three main phases: *Construction*, *Local Search* and *Path-Relinking*. The last one is applied to the solution obtained by local search and to

a randomly selected solution from the pool P twice (one using the latter as the starting solution and the other using the former). The locally optimal solution obtained by local search and the best solutions found along each relinking trajectory are considered as candidates for insertion into P. A solution is inserted in the pool if it is different from all solutions of the pool and its cost is better than the cost of the worst solution of the pool.

3.2. DM-GRASP-PR heuristic

We have already developed heuristics hybridizing GRASP with data mining, called DM-GRASP procedures, for many optimization problems [Plastino and Salhi 2009, Santos and Martins 2005, Santos and Plastino 2006, Santos and Plastino 2008]. The DM-GRASP is composed of two phases. The first one is called the elite set generation phase, which consists of executing n pure GRASP iterations. The d best obtained solutions compose the elite set. After this first phase, the data mining process is applied to extract patterns from the elite set. The patterns to be mined are sets of elements that frequently appear in solutions from the elite set. This extraction of patterns characterizes a frequent itemset mining application [Han and Kamber 2006]. A frequent itemset mined with support s represents a set of elements that occur in $s\%$ of the elite solutions.

Next, the second phase, called hybrid phase, is performed. Another n slightly different GRASP iterations are executed. In these n iterations, an adapted construction phase starts building a solution guided by a pattern selected from the set of mined patterns. Initially, all elements of the selected pattern are inserted into the partial solution, from which a complete solution will be built executing the standard construction procedure.

In this work, we developed the hybrid procedure DM-GRASP-PR, which incorporates a data mining procedure to the GRASP with path-relinking heuristic (GRASP-PR), in order to show that not only the traditional GRASP metaheuristic but also GRASP procedures improved with the path-relinking heuristic — a memory-based intensification mechanism — can benefit from the incorporation of a data mining procedure.

The useful patterns to be mined are sets of edges that commonly appear in suboptimal solutions of the 2PNDP. A frequent itemset mined from the elite set with support s represents a set of edges that occur in $s\%$ of the elite solutions. A frequent itemset is called maximal if it has no superset that is also frequent. In order to avoid mining frequent itemsets which are subset of one another, in the DM-GRASP-PR proposal for the 2PNDP, we decided to extract only maximal frequent itemset.

The adapted construction algorithm is quite similar to the GRASP construction phase code with the difference that, we try to construct a 2-path between a pair (a, b) using only the edges from the pattern or the edges already used which had their weight modified to 0. If a 2-path was not found using just these edges, we compute a 2-path starting from the partial solution found so far and using all edges from E.

3.3. Computational Results for DM-GRASP-PR

In this section, the results obtained for GRASP-PR and DM-GRASP-PR are compared. We generated 25 instances similar to the ones generated in [Ribeiro and Rosseti 2007b]. The instances are complete graphs with $|V| \in \{100, 200, 300, 400, 500\}$. The edge costs were randomly generated from the uniform distribution on the interval (0, 10] and $10 \times |V|$ origin-destination pairs were randomly chosen. The algorithms were implemented in C

and compiled with gcc 4.4.1. The tests were performed on a 2.4 GHz Intel Core 2 Quad CPU Q6600 with 3 Gbytes of RAM, running Linux Kernel 2.6.24. Both GRASP-PR and DM-GRASP-PR were run 10 times with a different random seed in each run. Each strategy executed 1000 iterations. After having conducted some tuning experiments, we set some parameter values: (d) and (t) were set to 10, and (s) was set to 2.

In Table 1, the results related to the solution quality and computational time are shown. The first column presents the identifier of the instance ax-y, where $x = |V|$ and y is the seed used to generate the random instance parameters. The $Best$, Avg and Dev columns present the best cost values, the average cost values and the average cost standard deviation obtained by the strategies. The better results are bold-faced. These results show that the proposed DM-GRASP-PR strategy was able to improve all results obtained by GRASP with path-relinking.

The $Time$ columns show the average execution time (in seconds) of the strategies, obtained for 10 runs, and the columns $TDev$ show its standard deviation. The last column shows the percentage difference between the strategies average times. For all instances, the execution times for DM-GRASP-PR were smaller. The last line of the table presents the average of the percentage differences. We can observe that, on average, DM-GRASP-PR was 20.23% faster than GRASP-PR.

Table 1. GRASP-PR and DM-GRASP-PR quality and time results

	GRASP-PR					DM-GRASP-PR					Time
Instance	Best	Avg	Dev	Time	TDev	Best	Avg	Dev	Time	TDev	%
a100-1	679	687.5	4.06	44.22	0.76	**676**	**682.0**	3.55	**37.39**	0.65	15.44
a100-10	663	669.8	3.25	43.29	0.58	**662**	**668.7**	2.83	**36.14**	0.54	16.51
a100-100	670	674.6	2.65	46.66	0.30	**666**	**670.3**	2.10	**38.89**	0.32	16.66
a100-1000	644	649.9	3.33	42.98	0.55	**641**	**647.0**	4.31	**36.11**	0.79	15.99
a100-10000	664	669.2	3.4	43.57	0.50	**661**	**666.5**	3.58	**36.87**	0.58	15.37
a200-1	1386	1391.9	4.66	201.30	2.57	**1379**	**1384.6**	3.80	**161.87**	1.77	19.59
a200-10	1374	1386.0	8.26	206.32	1.95	**1362**	**1376.1**	8.19	**166.02**	1.85	19.53
a200-100	1361	1369.4	4.27	197.35	2.71	**1354**	**1362.0**	4.80	**157.37**	1.96	20.26
a200-1000	1363	1374.5	7.77	199.61	2.42	**1358**	**1367.9**	8.63	**158.63**	2.35	20.53
a200-10000	1375	1387.4	8.66	207.02	2.20	**1369**	**1377.5**	7.57	**166.49**	1.75	19.58
a300-1	2106	2117.0	7.94	516.63	3.53	**2081**	**2102.4**	9.36	**401.89**	3.01	22.21
a300-10	2134	2148.0	6.88	515.14	3.47	**2122**	**2133.7**	7.89	**401.34**	4	22.09
a300-100	2088	2096.2	7.08	517.84	3.54	**2072**	**2082.3**	7.04	**412.27**	29.71	20.39
a300-1000	2100	2105.7	6.69	516.14	4.42	**2080**	**2094.5**	8.69	**398.99**	4.41	22.70
a300-10000	2077	2092.8	9.13	515.48	3.63	**2067**	**2078.2**	7.70	**399.88**	5.29	22.43
a400-1	2807	2816.2	5.33	1000.79	6.59	**2788**	**2797.5**	4.76	**769.70**	10.15	23.09
a400-10	2848	2864.7	9.24	1003.74	4.89	**2833**	**2847.8**	7.17	**780.44**	17.78	22.25
a400-100	2818	2834.2	7.4	1026.18	4.57	**2803**	**2818.9**	13.36	**854.99**	97.01	16.68
a400-1000	2822	2833.4	7.47	1022.98	3.21	**2800**	**2816.4**	10.28	**824.99**	78.34	19.35
a400-10000	2856	2874.8	9.47	1028.98	5.62	**2844**	**2857.2**	8.94	**808.78**	56.69	21.40
a500-1	3598	3606.6	6.62	1727.36	13.52	**3571**	**3579.6**	6.04	**1330.40**	28.97	22.98
a500-10	3595	3607.7	6.66	1712.67	6.40	**3573**	**3580.7**	7.55	**1302.53**	38.48	23.95
a500-100	3598	3612.4	10.46	1747.14	5.43	**3576**	**3584.7**	7.99	**1396.26**	125.1	20.08
a500-1000	3573	3592.0	7.69	1721.36	9.88	**3554**	**3564.2**	5.84	**1332.65**	26.15	22.58
a500-10000	3605	3625.0	10.85	1760.41	4.69	**3580**	**3597.9**	11.26	**1337.25**	20.1	24.04
Average											20.23

There are two main reasons for the faster behavior of DM-GRASP-PR. First, the computational effort of the adapted construction phase is smaller than the original construction, since a smaller set of edges is processed to find a 2-path for each pair. Second,

the use of patterns leads to the construction of better solutions which will be input for the local search. This incurs in less effort taken to converge to a local optimal solution.

4. The hybrid MDM-GRASP-PR proposal

In the proposed hybrid DM-GRASP-PR, the data mining procedure is executed just once and at the middle point of the whole process. Although the obtained results were satisfactory, we believe that mining more than once, and as soon as the elite set is stable and good enough, can improve the original DM-GRASP framework. Based on this hypothesis, in this work we also propose and evaluate another version of the DM-GRASP for the 2PNDP, called MDM-GRASP-PR (Multi Data Mining GRASP-PR).

The main idea of this proposal is to execute the mining process: (a) as soon as the elite set becomes stable — which means that no change in the elite set occurs throughout a given number of iterations — and (b) whenever the elite set has been changed and again has become stable. We hypothesize that mining more than once will explore the gradual evolution of the elite set and allow the extraction of refined patterns.

4.1. Computational Results

In this section, we report the computational results obtained for the proposed MDM-GRASP-PR strategy using the same kind of execution from the previous section. After performing some experiments using three values for the parameter used to define if the elite set is stable: 1%, 3% and 5% of the total number of iterations, we adopted 1% as this value provided the best cost values.

Since, in the previous analysis, the DM-GRASP-PR outperformed GRASP-PR, we decided to compare the MDM-GRASP-PR only with the DM-GRASP-PR strategy. In Table 2, the results related to quality and computational time are shown. MDM-GRASP-PR found 18 better results for best values and DM-GRASP-PR found four. MDM-GRASP-PR found 24 better results for average values and DM-GRASP-PR just one. These results show that the MDM-GRASP-PR proposal was able to improve the results obtained by DM-GRASP-PR.

We can observe that the DM-GRASP-PR was faster in 18 instances and MDM-GRASP-PR was faster seven instances. However, we observe that MDM-GRASP-PR was, on average, just 1.34% slower than DM-GRASP-PR which is not very significant in terms of the heuristic performance. We conclude that both path-relinking hybrid proposals had a similar behavior in terms of computational time.

In order to verify whether or not the differences of mean values obtained by the strategies presented in Tables 1 and 2 are statistically significant, we employed the unpaired Student's t-test technique. By comparing DM-GRASP-PR with GRASP-PR, we can note that DM-GRASP-PR found better results for all 25 instances and 19 of them are statistically significant, considering a p-value less than 0.01. When comparing MDM-GRASP-PR with GRASP-PR, we can note that MDM-GRASP-PR found better results for all 25 instances and 21 of them are statistically significant. These results show the superiority of the data mining strategies, mainly the good behavior of the MDM-GRASP-PR.

Figures 1(a) and 1(b) show another comparison between the three strategies, based on $Time\text{-}to\text{-}target$ (TTT) plots [Aiex and Ribeiro 2007], which are used to analyze the

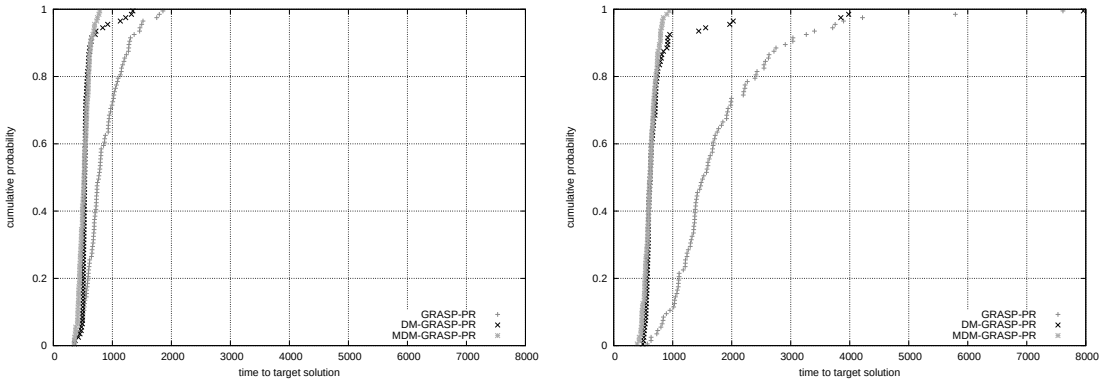
Table 2. DM-GRASP-PR and MDM-GRASP-PR quality results

	DM-GRASP-PR					MDM-GRASP-PR					Time
Instance	Best	Avg	Dev	Time	TDev	Best	Avg	Dev	Time	TDev	%
a100-1	676	682.0	3.55	**37.39**	0.65	**674**	**681.9**	5.28	38.50	0.91	-2.96
a100-10	662	668.7	2.83	**36.14**	0.54	**659**	**665.2**	3.22	37.54	1.38	-3.87
a100-100	**666**	670.3	2.10	**38.89**	0.32	667	**670.0**	2.41	40.41	0.83	-3.91
a100-1000	641	647.0	4.31	**36.11**	0.79	**640**	**646.7**	3.95	37.51	0.51	-3.89
a100-10000	661	666.5	3.58	**36.87**	0.58	**658**	**665.4**	3.56	38.41	1.13	-4.19
a200-1	**1379**	1384.6	3.80	**161.87**	1.77	1380	**1383.9**	4.16	163.19	6.41	-0.81
a200-10	1362	1376.1	8.19	**166.02**	1.85	1362	**1372.5**	5.80	167.06	3.42	-0.63
a200-100	1354	1362.0	4.80	**157.37**	1.96	**1352**	**1360.7**	6.63	162.58	6.85	-3.31
a200-1000	1358	1367.9	8.63	**158.63**	2.35	**1356**	**1364.0**	7.87	160.25	6.65	-1.02
a200-10000	1369	1377.5	7.57	**166.49**	1.75	**1363**	**1374.3**	7.85	166.61	6.77	-0.07
a300-1	**2081**	2102.4	9.36	**401.89**	3.01	2082	**2099.3**	9.23	409.38	12.47	-1.86
a300-10	**2122**	2133.7	7.89	**401.34**	4	2125	**2132.1**	5.05	410.17	12.15	-2.20
a300-100	2072	2082.3	7.04	412.27	29.71	**2069**	**2076.3**	5.40	**404.22**	10.37	1.95
a300-1000	2080	2094.5	8.69	398.99	4.41	**2076**	**2090.3**	7.09	**395.88**	18.55	0.78
a300-10000	2067	2078.2	7.70	**399.88**	5.29	**2060**	**2075.1**	10.38	403.97	14.56	-1.02
a400-1	2788	2797.5	4.76	**749.77**	10.15	**2786**	**2791.4**	4.52	**749.77**	21.24	2.59
a400-10	2833	2847.8	7.17	**780.44**	17.78	**2819**	**2844.1**	11.35	811.97	30.3	-4.04
a400-100	2803	2818.9	13.36	854.99	97.01	2803	**2808.9**	4.39	**799.67**	27	6.47
a400-1000	2800	2816.4	10.28	824.99	78.34	**2793**	**2810.9**	7.91	**797.91**	47.62	3.28
a400-10000	2844	2857.2	8.94	808.78	56.69	**2793**	**2810.9**	10.37	**797.91**	36.19	1.34
a500-1	3571	3579.6	6.04	**1330.40**	28.97	**3567**	**3576.9**	7.27	1349.39	73.55	-1.43
a500-10	3573	3580.7	7.55	**1302.53**	38.48	**3566**	**3580.1**	10.49	1346.80	86.9	-3.40
a500-100	3576	3584.7	7.99	**1396.26**	125.1	**3572**	**3583.1**	9.42	1413.63	65.44	-1.24
a500-1000	3554	**3564.2**	5.84	**1332.65**	26.15	3554	3564.9	4.95	1382.99	64.73	-3.78
a500-10000	3580	3597.9	11.26	**1337.25**	20.1	**3573**	**3596.1**	13.44	1420.02	114.77	-6.19
Average											-1.34

behavior of randomized algorithms. These plots basically show the cumulative probability distributions of running times, i.e., $p(\text{computational_time} < x)$ vs. x.

A TTT plot is generated, initially, by executing an algorithm several times and measuring the time required to reach a solution at least as good as a target solution. In our experiments, each strategy was executed a hundred times. Then, the i-th sorted running time t_i is associated with a probability $p_i = (i - 1/2)/100$ and the points $z_i = (t_i, p_i)$, for $i = 1, ..., 100$ are plotted. Each plotted point indicates the probability (vertical axis) for the strategy to achieve the target solution in the indicated time (horizontal axis). The plots presented in Figures 1(a) and 1(b) were generated by the executions of GRASP-PR, DM-GRASP-PR and MDM-GRASP-PR, for instance a400-100, using the same two target solutions used in the previous experiment, respectively: an average value (2834) and a more difficult one (2820).

For the average target, we observe in Figure 1(a) that GRASP-PR behaves worst than the two other strategies, and that the MDM-GRASP-PR behaves better than DM-GRASP-PR. We can see, for example, that the probability for MDM-GRASP-PR to reach the average target in 800s is 100%, for DM-GRASP-PR is approximately 95% and for GRASP-PR is approximately 58%. For the difficult target, Figure 1(b) shows that MDM-GRASP-PR behaves better than DM-GRASP-PR and both behave better than GRASP-PR. These plots indicate that MDM-GRASP-PR is able to reach difficult solutions faster than DM-GRASP-PR and much faster than GRASP-PR, demonstrating that mining more than once and when the elite set is stable brings robustness to the hybrid strategy.



(a) An average target for instance a400-100 (b) A difficult target for instance a400-100

Figure 1. Time-to-target plotting

5. Conclusions

Hybrid GRASP metaheuristics which incorporate a data mining procedure has been successfully applied for different combinatorial problems. In this work, we proposed to combine a data mining technique into a GRASP metaheuristic with path-relinking in order to show that not only the traditional GRASP can benefit from using patterns to guide the search, but also GRASP improved with the path-relinking heuristic.

The experimental results showed that the first version of the proposed path-relinking hybrid strategy, called DM-GRASP-PR, was able to obtain better solutions in less computational time than the original GRASP with path-relinking developed to solve the 2-path network design problem, which was a state-of-the-art method for this problem.

In this first version of the path-relinking hybrid GRASP, the data mining process occurred just once. To explore the gradual evolution of the elite set of solutions and allow the extraction of better and higher-quality patterns, we proposed another version of the path-relinking hybrid strategy, called MDM-GRASP-PR. This strategy extracts new sets of patterns whenever the elite set changes and becomes stable. The conducted experiments showed that the MDM-GRASP-PR obtained even better results than the DM-GRASP-PR.

These results showed that incorporating a data mining technique is effective, not only to memoryless heuristics, but also to methods that use exchange of information about obtained solutions like the path-relinking strategy.

6. Comments

This work is part of a research project on hybrid metaheuristics with data mining. The student Hugo Barbalho has developed, under supervision of Simone Martins and Alexandre Plastino, both DM-GRASP-PR and MDM-GRASP-PR strategies based on the GRASP-PR, implemented by Isabel Rosseti on her Ph.D. thesis. An extended version of this paper has been submitted to the special issue *GRASP with Path Relinking* of the Computers and Operation Research Journal.

References

A. Plastino, E. R. Fonseca, R. Fuchshuber, S. L. Martins, A. A. Freitas, M. Luis, and S. Salhi, *A hybrid data mining metaheuristic for the p-median problem*, Proceedings

of the SIAM International Conference on Data Mining, pp. 305-316, 2009.

C. C. Ribeiro, S. L. Martins, and I. Rosseti, *Metaheuristics for optimization problems in computer communications*, Computer Communications, 30 (2007), pp. 656-669.

C. C. Ribeiro and I. Rosseti, *Efficient parallel cooperative implementations of GRASP heuristics*, Parallel Computing 33 (2007), pp. 21-35.

G. Dahl and B. Johannessen, *The 2-path network problem*, Networks, 43 (2004), pp. 190–199.

F. Glover, M. Laguna, and R. Martí, *Fundamentals of scatter search and path-relinking*, Control and Cybernetics 39, pp. 653–684, 2000.

T. A. Feo and M. G. C. Resende, *A probabilistic heuristic for a computationally difficult set covering problem*, Operations Research Letters, 8 (1989), pp. 67-71.

T. A. Feo and M. G. C. Resende, *Greedy randomized adaptive search procedures*, Journal of Global Optimization, 6 (1995), pp. 109-133.

P. Festa and M. G. C. Resende, *An annotated bibliography of GRASP Part II: Applications*, International Transactions in Operational Research, 16 (2009), pp. 131–172.

J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2^{nd} Ed., Morgan Kaufmann Publishers, 2006.

L. F. Santos, M. H. F. Ribeiro, A. Plastino, and S. L. Martins, *A hybrid GRASP with data mining for the maximum diversity problem*, Proceedings of the International Workshop on Hybrid Metaheuristics, LNCC 3636, pp. 116–127, 2005.

L. F. Santos, S. L. Martins, and A. Plastino, *Applications of the DM-GRASP heuristic: A survey*, International Transactions in Operational Research, 15 (2008), pp. 387–416.

L. F. Santos, C. V. Albuquerque, S. L. Martins, and A. Plastino, *A hybrid GRASP with data mining for efficient server replication for reliable multicast*, Proceedings of the IEEE GLOBECOM Conference, 2006.

M. Laguna and R. Martí, *GRASP and path relinking for 2-layer straight line crossing minimization*, INFORMS Journal on Computing **11** (1999), pp. 44–52.

M. H. F. Ribeiro, V. F. Trindade, A. Plastino, and S. L. Martins, *Hybridization of GRASP metaheuristic with data mining techniques*, Proceedings of the ECAI Workshop on Hybrid Metaheuristics, pp. 69-78, 2004.

R. Aiex, M. G. C. Resende, and C. C. Ribeiro, *TTT plots: a perl program to create time-to-target plots*, Optimization Letters, 4 (2007), pp. 355–366.

M. G. C. Resende and C. C. Ribeiro, *Greedy randomized adaptive search procedures*, Handbook of Metaheuristics, Kluwer Academic Publishers, 2003.

M. G. C. Resende and C. C. Ribeiro, *GRASP with path-relinking: Recent advances and applications*, Metaheuristics: Progress as Real Problem Solvers (T. Ibaraki et al. editors), (2005), 29–63.

Avaliação da Relevância dos *Warnings* Reportados por Ferramentas para Detecção de Defeitos baseadas em Análise Estática

João Eduardo Montandon[1]
Orientador: Marco Túlio Valente[2]

[1]Instituto de Informática, PUC Minas
[2]Departamento de Ciência da Computação, UFMG

joao.montandon@dcc.ufmg.br, mtov@dcc.ufmg.br

Resumo. *Apesar do interesse e do número crescente de ferramentas de análise estática, ainda não existe um consenso sobre o efetivo papel de tais ferramentas na garantia de qualidade de sistemas de software. Assim, neste Trabalho de Iniciação Científica foram realizados dois amplos estudos envolvendo sistemas importantes de código aberto e com os seguintes objetivos centrais: (a) avaliar a relevância dos warnings reportados por ferramentas de análise estática; (b) investigar eventuais correlações entre warnings emitidos por tais ferramentas e defeitos reportados por usuários finais de um sistema.*

Abstract. *Despite the interest and the increasing number of static analysis tools, there is still no consensus on the actual gains that such tools can introduce in software development projects. In this work, we report the results of two extensive case studies involving major open source systems and with the following central goals: (a) to evaluate the relevance of the warnings reported by static analysis tools; (b) to investigate possible correlations between the warnings reported by such tools and field defects.*

1 Introdução

Recentemente, tem crescido o interesse tanto acadêmico como industrial pelo emprego de técnicas e ferramentas de análise estática para ajudar desenvolvedores de software a detectar defeitos em seus sistemas [12, 8, 1, 3]. Em vez de procurarem verificar se um sistema atende à sua especificação, ferramentas de análise estática – também chamadas de *bug findings tools* – funcionam procurando por violações de padrões de programação. Como exemplo de defeitos detectados por essas ferramentas, podemos citar acesso a referências *null*, uso inapropriado de métodos (como *equals*, *clone* etc), uso incorreto de primitivas de sincronização, *overflow* em vetores, divisão por zero etc. Em resumo, tais ferramentas ampliam e sofisticam as mensagens de *warning* emitidas por compiladores. Adicionalmente, podem contribuir para verificar estilos e boas práticas de programação, como convenções de nomes e de indentação. Dentre as diversas ferramentas de análise estática existentes, podemos citar os sistemas PREfix/PREfast [11] (para programas em C/C++) e FindBugs [10] e PMD [4] (para programas em Java).

Apesar do interesse e do número crescente de ferramentas de análise estática, ainda não existe um consenso sobre o efetivo papel de tais ferramentas na garantia de qualidade de sistemas de software. Em outras palavras, ainda não existem trabalhos que ofereçam respostas consistentes para duas perguntas centrais:

1. *Qual a relevância dos warnings reportados por ferramentas de análise estática?* A fim de contribuir com uma resposta objetiva para essa pergunta, supôs-se como hipótese no trabalho que um defeito introduzido em uma versão i de um sistema é relevante quando ele é removido em uma versão j liberada no máximo t unidades de tempo após a versão i. Em outras palavras, o pressuposto é que se o tempo de vida de um defeito é maior que t, então ele não é relevante. Provavelmente, defeitos não-relevantes não são removidos rapidamente porque eles não originam falhas, não impactam requisitos não-funcionais, não comprometem atributos de qualidade interna etc. Por outro lado, se um defeito é removido rapidamente, isso sugere que ele é relevante.

2. *Warnings emitidos por ferramentas de análise estática são indícios da existência de defeitos que serão posteriormente reportados por usuários finais?* Em outras palavras, quanto maior o número de *warnings* reportados em um determinado *release* de um sistema, maior será o número de defeitos de campo posteriormente observados nesse sistema? Caso as respostas para essas perguntas sejam afirmativas, considera-se que existe uma correlação estatística entre defeitos e *warnings*. A eventual existência de uma correlação – do ponto de vista prático – sugere que um gerente de qualidade pode usar o número de *warnings* como um indicativo da qualidade de um sistema e, por exemplo, somente disponibilizar *releases* cuja densidade de *warnings* seja inferior a um determinado limiar.

Para responder à primeira pergunta, neste Trabalho de Iniciação Científica foi realizada uma avaliação detalhada da relevância dos *warnings* reportados por ferramentas de análise estática [6, 7]. Mais especificamente, a experiência teve como objetivo determinar o tempo de vida dos defeitos apontados pelas ferramentas FindBugs [10] e PMD [4] quando aplicadas retrospectivamente sobre o código fonte de cinco versões da plataforma de desenvolvimento Eclipse. Além de largamente utilizado como ambiente de desenvolvimento, o Eclipse é um projeto complexo e de grande porte. Por exemplo, todas as cinco versões analisadas no estudo possuem pelo menos um milhão de linhas de código. Adicionalmente, para reforçar as conclusões obtidas, o mesmo estudo foi expandido para outros onze sistemas de código aberto.

Para responder à segunda pergunta, foram avaliados vinte cinco sistemas mantidos pela Fundação Apache, perfazendo um total de mais de um milhão e oitocentas mil linhas de código [5]. Basicamente, no estudo foi calculada a correlação estatística entre o número de *warnings* gerados pelo FindBugs e o número de *bugs* reportados pelos usuários finais desses sistemas. Os *bugs* considerados nesta segunda parte do Trabalho de Iniciação Científica foram extraídos do sistema de gerência de *bugs* utilizado pelos sistemas da Fundação Apache. Para avaliar a correlação entre as duas variáveis (defeitos de campo e *warnings*), usou-se o teste de correlação de *ranks* de Spearman [13].

O restante deste trabalho está estruturado conforme descrito a seguir. Na Seção 2, apresenta-se o estudo sobre a relevância dos *warnings* reportados pelo FindBugs e PMD. A Seção 3 resume o estudo realizado para correlacionar estatisticamente *warnings* e defeitos de campo. Por último, a Seção 4 discute os trabalhos relacionados e a Seção 5 apresenta as conclusões do estudo.

2 Relevância dos *Warnings*

A proposta deste estudo é avaliar a relevância dos *warnings* reportados pelas ferramentas FindBugs e PMD. Para isto, o estudo analisou cinco versões do sistema Eclipse, lançadas entre os anos de 2004 (versão 3.0) e 2008 (versão 3.4), totalizando, aproximadamente, sete milhões de linhas de código (MLOC).

Neste estudo, um *warning* foi considerado relevante quando o seu tempo de permanência no sistema – também chamado de tempo de vida – é menor que um determinado tempo t. Isto é, pode-se dizer que o intervalo de vida de um *warning* é definido pelo par (v_p, v_q), onde v_p designa a versão onde o *warning* foi detectado pela primeira vez e v_q designa a versão onde o *warning* foi corrigido. Se o *warning* não foi corrigido, considera-se $v_q = \infty$. Caso contrário, supondo $v_q \neq \infty$, o tempo de vida de um *warning* é determinado pela seguinte diferença: $date(v_q) - date(v_p)$, onde $date(v)$ é a data de liberação da versão. Isto é, se $date(v_q) - date(v_p) < t$, significa que o *warning* é relevante.

2.1 Coleta de Dados

Para obter dados que permitam avaliar a relevância dos *warnings* reportados pelas ferramentas FindBugs e PMD, as seguintes atividades foram realizadas:

Configuração das Ferramentas: As ferramentas foram configuradas para executar em máxima prioridade. Esta configuração filtra os *warnings* reportados de modo que somente os mais graves sejam avaliados, ou seja, elimina previamente prováveis falsos positivos. No caso do PMD, também foram descartadas algumas categorias relacionadas a convenções da linguagem Java, como *Naming Rules* (regras relacionadas a convenções de nomes) e *Brace Rules* (regras relacionadas a utilização de chaves).

Cálculo dos *Warnings* Relevantes: Para execução das ferramentas e cálculo dos *warnings* relevantes reportados por elas, foi implementado um *script* em Perl. Basicamente, esse *script* recebe como entrada um arquivo com a localização das versões a serem analisadas. O *script* então executa as ferramentas sobre cada uma dessas versões e, em seguida, o arquivo `XML` resultante da execução é submetido a um *parser*. Esse *parser* processa o arquivo `XML` e gera uma lista com a assinatura de cada *warning* obtido. Essa assinatura contém as informações de identificação e localização do *warning* no código (pacote, classe e método). A fim de se obter quais *warnings* são relevantes, os arquivos resultantes são submetidos a comparações textuais. Tais comparações permitem identificar quais *warnings* foram removidos entre as versões e assim identificar e calcular sua relevância.

2.2 Resultados

A Tabela 1 mostra a taxa de remoção dos *warnings* reportados pelo FindBugs. Mesmo quando executada para detectar somente *warnings* de alta prioridade, a ferramenta se mostrou pouco eficaz, uma vez que, no seu melhor caso (versão 3.0, 36 meses), menos de 30% dos *warnings* foram removidos durante o limiar de tempo definido.

Para tentar entender as razões pelas quais os desenvolvedores – mesmo após três anos – não removeram a maioria dos *warnings*, realizou-se um estudo mais detalhado dos *warnings* detectados. Ao se analisar os *warnings* presentes na versão 3.0 do sistema Eclipse, descobriu-se que os *warnings* mais comuns detectados pela ferramenta não representavam de fato um defeito no sistema. Isto é, os *warnings* mais comuns reportados

Versão	Limiar de tempo t		
	12 meses	24 meses	36 meses
3.0	21.7	25.3	29.4
3.1	9.4	13.8	12.8
3.2	7.8	15.6	-
3.3	6.2	-	-

Tabela 1. Taxa de remoção dos *warnings* de alta prioridade (FindBugs)

pelo FindBugs denotaram, na verdade, estilos de programação não recomendados ou, no máximo, defeitos relevantes apenas em sistemas que envolvem acesso a dados sigilosos. Ou seja, constatou-se que o critério de classificação dos *warnings* adotado pelo FindBugs dá margem a diversos falsos positivos.

A fim de comprovar essa afirmação, avaliou-se novamente os *warnings* reportados pelo FindBugs, porém considerando somente os *warnings* da categoria *correctness*. Essa categoria inclui avisos com grande probabilidade de se tornarem defeitos reais no sistema. A Tabela 2 mostra a nova taxa de remoção dos *warnings* da categoria *correctness* considerados relevantes. Ao comparar os resultados das Tabelas 1 e 2, pode-se observar que o percentual de *warnings* removidos aumentou consideravelmente. Por exemplo, no melhor cenário da Tabela 1 (versão 3.0, 36 meses), apenas 29.4% dos *warnings* foram classificados como relevantes. Ao restringirmos a análise aos *warnings* da categoria *correctness*, essa taxa aumentou para 63.9%. O mesmo ocorreu no pior caso (versão 3.3, 12 meses), onde a taxa de remoção subiu de 6.2% para 15.2%.

Versão	Tempo de vida		
	12 meses	24 meses	36 meses
3.0	42.6	54.1	63.9
3.1	28.6	34.3	34.3
3.2	29.2	37.5	-
3.3	15.2	-	-

Tabela 2. Taxa de remoção dos *warnings* da categoria *correctness* (FindBugs)

Diferentemente do FindBugs, os resultados obtidos após a análise dos *warnings* detectados pelo PMD não se mostraram efetivos. Além de reportar uma quantidade muito maior de *warnings*, se comparado com o FindBugs, sua taxa de remoção se mostrou muito baixa (menos de 10% em todos os casos). Mais detalhes sobre o estudo com o PMD podem ser obtidos em [7].

2.3 Estudo de Caso Ampliado

Na tentativa de reforçar os resultados obtidos a partir da análise do FindBugs no sistema Eclipse, conduziu-se um experimento mais amplo, envolvendo onze sistemas de código aberto. Diferentemente do Eclipse, onde foram selecionadas cinco versões para análise, a coleta de dados foi limitada a duas versões por sistema, com intervalo de um ano entre as duas, aproximadamente. Além disso, o FindBugs foi configurado para reportar somente *warnings* de alta prioridade da categoria *correctness*, uma vez que essa abordagem se mostrou mais efetiva para o sistema Eclipse.

A Tabela 3 apresenta o resultado da execução do FindBugs sobre esses onze novos sistemas. Como pode ser observado, quatro sistemas – JEdit, PdfSam, Jython e Tomcat

	Sistemas	# HP-CT warnings		Relevância (%) (Tempo de vida)	Relevância (%) (Desenvolvedores)
		Inicial	Final		
1	JEdit	15	3	80.0	-
2	Pdfsam	3	1	66.7	-
3	Jython	10	4	60.0	-
4	Tomcat	24	14	41.7	-
5	JabRef	7	7	0.0	85.7
6	Jetty	4	4	0.0	75.0
7	ArgoUML	11	11	0.0	54.5
8	FreeMind	6	6	0.0	16.6
9	Jung	3	3	0.0	0.0
10	JGraph	2	2	0.0	0.0
11	Xerces	13	13	0.0	0.0

Tabela 3. Relevância dos *warnings* de alta prioridade (HP) da categoria *correctness* (CT)

– obtiveram resultados similares ao Eclipse, alcançando uma taxa de remoção superior a 40%. No entanto, nos outros sete sistemas, a taxa de relevância se mostrou diferente, pois nenhum *warning* detectado pelo FindBugs foi removido entre as versões.

Visto que mais da metade dos sistemas não apresentaram taxas de remoção relevantes, decidiu-se contactar diretamente os desenvolvedores desses sistemas. O contato foi realizado por meio do *bug tracker* de cada sistema, onde foi reportado uma breve descrição dos *warnings* detectados. Ainda, foi pedido aos desenvolvedores para avaliarem relevância dos *warnings* reportados. Como pode ser observado na última coluna da Tabela 3, após a análise dos próprios desenvolvedores, quatro sistemas possuíam *warnings* que mesmo não corrigidos foram considerados relevantes. Com isto, quatro sistemas tiveram alguns de seus *warnings* corrigidos, sendo que três deles – JabRef, Jetty e ArgoUML – obtiveram uma taxa de remoção próxima ou superior àquela obtida no Eclipse.

Por outro lado, os desenvolvedores dos sistemas avaliados também classificaram alguns *warnings* como falsos positivos. No caso de três sistemas – Jung, JGraph e Xerces – todos os *warnings* foram considerados irrelevantes pelos seus desenvolvedores. Frequentemente, isto se deve ao fato de o processo de análise estática não considerar o contexto no qual o código está inserido. Por exemplo, os desenvolvedores do Xerces não consideraram os *warnings* relevantes pois eles foram detectados em código considerado *morto* (isto é, código que, apesar de estar presente, não é mais utilizado pelo sistema).

Conclusões: Com base nos resultados obtidos, considera-se que ferramentas para detecção de defeitos baseadas em análise estática podem ajudar a prevenir *bugs* quando incorporadas ao ciclo de desenvolvimento de software. No entanto, ao adotar tais ferramentas, é fundamental que os desenvolvedores customizem e adaptem as mesmas de acordo com suas necessidades. Dessa forma, a taxa de *warnings* relevantes aumenta substancialmente.

3 Correlação entre Defeitos de Campo e *Warnings*

Este segundo estudo teve como objetivo verificar a existência de uma correlação entre os *warnings* reportados por ferramentas de análise estática e defeitos de campo. Para tanto, avaliaram-se 25 sistemas da Fundação Apache, totalizando mais de um milhão e

oitocentas mil linhas de código. A Tabela 4 apresenta o nome dos sistemas envolvidos neste estudo. Esses sistemas foram selecionados de acordo com os seguintes critérios: (a) representam sistemas de média e grande complexidade (tamanho varia de 21 KLOC a 233 KLOC); (b) o *bytecode* está disponível publicamente; (c) possuem um histórico bem documentado de defeitos de campo. Além disso, foi dada preferência a sistemas que possuíssem versões entre os anos de 2007 e 2008, de modo a obter um tempo de análise de pelo menos um ano dos defeitos de campo reportados pelos usuários.

Para quantificar a relação entre as duas variáveis consideradas (*warnings* e defeitos de campo), utilizou-se o teste de correlação de Spearman, o qual é uma medida de correlação estatística entre duas variáveis [13]. Basicamente, o teste de Spearman produz um coeficiente entre -1 e +1 que informa o grau de correlação entre dois *ranks*. Se existe uma correlação positiva perfeita entre os *ranks* analisados, o valor do coeficiente de Spearman é +1. No caso do trabalho, essa situação ocorrerá se o sistema com i-ésimo maior número de *warnings* for também o sistema com i-ésimo maior número de defeitos de campo (para $1 \leq i \leq 25$). Por outro lado, se existir uma correlação negativa perfeita entre os *ranks* analisados, o valor do coeficiente de Spearman é -1. Em resumo, o teste de Spearman não considera a correlação entre os valores de duas variáveis, mas sim a correlação da ordem desses valores em um *rank*.

3.1 Coleta de Dados

Para obter os dados necessários para correlacionar *warnings* gerados pelo FindBugs com defeitos de campo reportados para os 25 sistemas, as seguintes tarefas foram realizadas:

1. A ferramenta FindBugs foi executada duas vezes sobre o código executável de cada um dos sistemas considerados no estudo. Na primeira execução, a ferramenta foi executada em sua configuração *default*. Na segunda execução, a ferramenta foi configurada para reportar apenas *warnings* de alta prioridade. O número total de *warnings* em cada execução foi coletado.

2. Para coletar os defeitos de campo, acessou-se o *bug tracker* de cada sistema. Para cada um dos sistemas foi coletado o número total de defeitos que atendem às seguintes condições: (a) foram reportados para a mesma versão dos sistemas considerados no estudo e (b) denotam efetivamente defeitos de campo (ou seja, foram desconsiderados registros que denotam solicitações de novas funcionalidades).

A Tabela 4 apresenta os seguintes dados coletados nas tarefas descritas anteriormente: número de *warnings* reportados pelo FindBugs em sua configuração *default* (coluna FB); número de *warnings* de alta prioridade reportados pelo FindBugs (coluna FBH); número total de defeitos de campo registrados na plataforma Jira na data de realização do estudo (coluna JT); número de defeitos de campo reportados na plataforma Jira até seis meses após a disponibilização das versões analisadas (coluna J6) e número de defeitos de campo reportados na plataforma Jira até 12 meses após a disponibilização das versões analisadas (coluna J12). As últimas cinco colunas da Tabela 4 mostram os valores das colunas FB, FBH, JT, J6 e J12 divididos pelo número de milhares de linhas de código (KLOC) dos respectivos sistemas. Em vez de valores absolutos de *warnings* e defeitos, o objetivo é fornecer dados que mostrem esses valores em função do tamanho dos sistemas analisados.

	Sistemas	FB	FBH	JT	J6	J12	FB / KLOC	FBH / KLOC	JT / KLOC	J6 / KLOC	J12 / KLOC
1	Struts2	99	24	91	82	85	2.6	0.6	2.4	2.2	2.3
2	CXF	616	61	145	137	143	3.3	0.3	0.8	0.7	0.8
3	ApacheDS	269	20	78	77	78	3.5	0.3	1.0	1.0	1.0
4	Jackrabbit	543	77	135	109	128	2.9	0.4	0.7	0.6	0.7
5	Myfaces Core	116	14	133	119	131	5.4	0.7	6.2	5.6	6.1
6	Myfaces TH	99	38	106	77	101	2.4	0.9	2.5	1.8	2.4
7	OpenJPA	583	57	117	97	109	5.2	0.5	1.0	0.9	1.0
8	Tuscany SCA	299	53	70	68	70	3.2	0.6	0.7	0.7	0.7
9	UIMA	269	26	64	62	64	2.4	0.2	0.6	0.5	0.6
10	Wicket	799	83	126	123	125	12.6	1.3	2.0	1.9	2.0
11	Hadoop Common	625	177	148	145	148	6.3	1.8	1.5	1.5	1.5
12	Hadoop Hbase	532	198	125	125	125	17.5	6.5	4.1	4.1	4.1
13	Ivy	107	8	63	49	60	4.4	0.3	2.6	2.0	2.5
14	James Server	113	12	71	28	39	4.1	0.4	2.6	1.0	1.4
15	Lucene	391	28	88	72	80	5.0	0.4	1.1	0.9	1.0
16	Roller	344	30	99	67	88	7.0	0.6	2.0	1.4	1.8
17	Shidig	36	5	18	18	18	0.8	0.1	0.4	0.4	0.4
18	Solr	236	100	139	80	121	5.8	2.5	3.4	2.0	3.0
19	Tapestry	113	10	65	54	60	1.8	0.2	1.0	0.9	1.0
20	Axis2	219	30	236	221	230	6.1	0.8	6.6	6.2	6.4
21	Geronimo	309	47	164	147	158	5.4	0.8	2.9	2.6	2.8
22	Xalan	282	51	240	140	208	9.7	1.8	8.3	4.8	7.2
23	Xerces	266	29	123	100	113	6.0	0.7	2.8	2.3	2.6
24	Beehive	426	26	101	68	82	3.5	0.2	0.8	0.6	0.7
25	Derby	937	72	238	138	172	4.2	0.3	1.1	0.6	0.8

Tabela 4. Dados absolutos e relativos sobre *warnings* e defeitos de campo

3.2 Resultados

A Tabela 5 apresenta os coeficientes de Spearman que expressam a correlação entre as duas grandezas consideradas neste trabalho: defeitos de campo (representados pelas seguintes variáveis: `JT/KLOC` , `J6/KLOC` e `J12/KLOC`) e *warnings* reportados pela ferramenta de análise estática FindBugs (representados pelas seguintes variáveis: `FB/KLOC` e `FBH/KLOC`). Todos os coeficientes reportados nesta tabela possuem nível de significância de pelo menos 99% (p-value ≤ 0.01).

	FB / KLOC	FBH / KLOC
JT / KLOC	0.673	0.727
J6 / KLOC	0.665	0.753
J12 / KLOC	0.690	0.779

Tabela 5. Resultados do Teste de Spearman

Conforme pode ser observado na Tabela 5, os resultados do Teste de Spearman mostram que existe uma significativa correlação estatística entre as duas grandezas consideradas no estudo (defeitos de campo e *warnings* emitidos pelo FindBugs). Os coeficientes reportados nesta tabela são sempre superiores a 0.66.

Ainda na Tabela 5, é possível observar que não há uma diferença significativa entre os coeficientes medidos quanto ao tempo utilizado para coletar os defeitos de campo. Em outras palavras, os valores dos coeficientes obtidos para todos os defeitos de campo (JT / KLOC), para os defeitos de campo reportados após seis meses (J6 / KLOC), e para os defeitos de campo após 12 meses (J12 / KLOC) são bem próximos. Além disso, também observa-se que a correlação entre defeitos de campo e *warnings* de alta prioridade é maior

que a correlação entre defeitos e campo e *warnings* de todas as categorias.

Conclusões: Diante dos resultados obtidos, conclui-se que existe uma significativa correlação entre as grandezas avaliadas (defeitos de campo e *warnings*). Particularmente, os resultados obtidos por meio do teste de correlação de *ranks* de Spearman permitem afirmar o seguinte: sistemas com maior número de *warnings* reportados pelo FindBugs têm maior probabilidade de – após serem liberados para uso – apresentar um número maior de defeitos. Assim, conclui-se que ferramentas como FindBugs são importantes instrumentos para avaliar a qualidade das versões disponibilizadas de um sistema.

4 Trabalhos Relacionados

Ayewah et al. avaliaram os resultados reportados pelo FindBugs em três grandes sistemas de software, incluindo a Sun JDK, Glassfish J2EE Server e parte da base de código do Google [2]. Para cada projeto, os autores classificaram manualmente os *warnings* de média e alta prioridade da categoria *correctness* reportados pelo FindBugs nas seguintes categorias: triviais, com algum impacto funcional e com impacto substancial. Por exemplo, dos 379 *warnings* reportados no Sun JDK, apenas 38 foram classificados como *warnings* de impacto substancial. No entanto, conforme discutido no próprio trabalho, a categorização proposta é aberta a interpretação. Isto é, os gerentes de projeto do JDK poderiam ter opinião diferente sobre a proposta de classificação, com base em seu conhecimento do sistema, incluindo a importância relativa de seus módulos e as competências dos colaboradores envolvidos na implementação de cada módulo.

Zheng et al. adotaram o paradigma GQM para determinar "se análise estática pode ajudar a otimizar economicamente a qualidade dos softwares de uma organização" [15]. Para alcançar este objetivo, os autores avaliaram três grandes sistemas em `C++` da Nortel Networks, utilizando três ferramentas de análise estática comerciais: FlexeLint da Gimpel, Illuma da Reasoning e Informe e GateKeeper da Klocwork[1]. Contudo, não está claro se os resultados obtidos podem ser extrapolados para linguagens com tipagem, tais como Java. Além disso, como a Nortel possuía um serviço terceirizado de filtragem manual de defeitos, a experiência conduzida pelos autores considerou somente *warnings* que passaram por este serviço. Isto é, também não está claro se as conclusões podem ser aplicadas a projetos que não incluem este tipo de filtro externo.

Wagner et al. também aplicaram as mesmas ferramentas consideradas no presente trabalho de Iniciação Científica – FindBugs e PMD – em dois estudos de caso [14]. O objetivo principal era determinar se tais ferramentas são eficazes para detectar defeitos de campo, isto é, defeitos que mais tarde serão reportados por usuários finais de software. Particulamente, os autores não conseguiram encontrar um único *warning* reportado pelas ferramentas que poderia estar relacionado a um defeito de campo. A razão se deu pelo fato de grande parte dos defeitos de campo estarem relacionados a falhas lógicas. Apesar desse fato, acredita-se que é importante avaliar se os *warnings* reportados por ferramentas de análise estática denotam violações de práticas não recomendadas de programação. Tais violações podem não contribuir para um funcionamento incorreto do sistema, mas podem levar por exemplo a um código de difícil entendimento e manutenção. Em virtude

[1] `http://www.klocwork.com.`

disto, decidiu-se utilizar outro critério para avaliar a efetividade das ferramentas de análise estática: o tempo de vida dos *warnings* gerados por essas ferramentas.

5 Conclusões

Neste Trabalho de Iniciação Científica, procurou-se esclarecer o papel que ferramentas para detecção de defeitos baseadas em análise estática podem desempenhar na garantia de qualidade de sistemas de software.

O estudo de relevância realizado com as ferramentas FindBugs e PMD apresentou duas contribuições. Primeiro, esse estudo mostrou que, quanto ao FindBugs, é fundamental que essa ferramenta esteja devidamente configurada de acordo com o sistema que será submetido a análise. Deste modo, o número de falsos positivos reportados é reduzido significativamente. Nos estudos de caso realizados, após a configuração do FindBugs para o modo *correctness*, alcançou-se taxas de relevância próximas ou superiores a 50%. Em segundo lugar, quanto ao PMD, o estudo comprovou que essa ferramenta detecta um número muito maior de *warnings* que o FindBugs e, como consequência, sua taxa de relevância não é significativa (menos de 10% em todos os cenários avaliados). Os resultados desse primeiro estudo foram publicados na forma de um relato de experiência no SBQS 2010 [6]. Ainda, os resultados completos do estudo serão publicados em um periódico internacional [7].

O estudo de correlação apresenta uma contribuição em especial, pois foi possível mostrar que existe uma significativa correlação estatística entre *warnings* e defeitos de campo. Mais especificamente, no estudo realizado com vinte e cinco sistemas da Fundação Apache foi constatado que os sistemas com maior número de defeitos de campo eram basicamente aqueles que apresentavam um maior número de *warnings* após a execução da ferramenta FindBugs. Os resultados desse segundo estudo foram publicados na forma de um artigo técnico no SBQS 2010, o qual foi premiado como melhor artigo do simpósio [5]. Uma versão estendida desse artigo encontra-se atualmente submetida a um periódico internacional.

Como trabalho futuro, pretende-se ampliar o estudo realizado, possivelmente acrescentando novos sistemas e/ou ferramentas de análise estática. Pretende-se também investigar outras técnicas estatísticas de correlação, como o Teste de Causalidade de Granger [9].

Agradecimentos: Este trabalho foi desenvolvido por meio de uma Bolsa de Iniciação Científica financiada pelo CNPq. Gostaríamos de agradecer a Silvio José de Souza (ex-aluno de mestrado da PUC Minas), pelas discussões que ajudaram a planejar o primeiro estudo de caso e a César Couto (doutorando em Computação na UFMG), pelas diversas sugestões relativas ao segundo estudo.

Referências

[1] Nathaniel Ayewah, David Hovemeyer, J. David Morgenthaler, John Penix, and William Pugh. Using static analysis to find bugs. *IEEE Software*, 25(5), 2008.

[2] Nathaniel Ayewah, William Pugh, J. David Morgenthaler, John Penix, and YuQian Zhou. Evaluating static analysis defect warnings on production software. In *7th Workshop on Program Analysis for Software Tools and Engineering (PASTE)*, pages 1–8, 2007.

[3] Al Bessey, Ken Block, Ben Chelf, Andy Chou, Bryan Fulton, Seth Hallem, Charles Henri-Gros, Asya Kamsky, Scott McPeak, and Dawson Engler. A few billion lines of code later: Using static analysis to find bugs in the real world. *Communications of the ACM*, 53(2):66–75, 2010.

[4] Tom Copeland. *PMD Applied*. Centennial Books, 2005.

[5] João Eduardo Montandon de Araujo Filho, César Francisco de Moura Couto, Marco Túlio Valente, and Silvio José de Souza. Um estudo sobre a correlação entre defeitos de campo e warnings reportados por uma ferramenta de análise estática. *IX Simpósio Brasileiro de Qualidade de Software*, pages 9–23, 2010.

[6] João Eduardo Montandon de Araujo Filho, Sílvio José de Souza, and Marco Túlio Valente. Os defeitos detectados pela ferramenta de análise estática FindBugs são relevantes? *IX Simpósio Brasileiro de Qualidade de Software*, pages 383–390, 2010.

[7] João Eduardo Montandon de Araujo Filho, Marco Túlio Valente, and Silvio José de Souza. A study on the relevance of the warnings reported by Java bug finding tools. *IET Software*, pages 1–27, 2011 (to appear).

[8] Jeffrey S. Foster, Michael W. Hicks, and William Pugh. Improving software quality with static analysis. In *7th Workshop on Program Analysis for Software Tools and Engineering (PASTE)*, pages 83–84, 2007.

[9] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438, 1969.

[10] David Hovemeyer and William Pugh. Finding bugs is easy. *SIGPLAN Notices*, 39(12):92–106, 2004.

[11] James R. Larus, Thomas Ball, Manuvir Das, Robert DeLine, Manuel Fahndrich, Jon Pincus, Sriram K. Rajamani, and Ramanathan Venkatapathy. Righting software. *IEEE Software*, 21(3):92–100, 2004.

[12] Panagiotis Louridas. Static code analysis. *IEEE Software*, 23(4):58–61, 2006.

[13] Peter Sprent and Nigel C. Smeeton. *Applied Nonparametric Statistical Methods*. Chapman & Hall, 2007.

[14] Stefan Wagner, Michael Aichner, Johann Wimmer, and Markus Schwalb. An evaluation of two bug pattern tools for Java. In *1st International Conference on Software Testing, Verification, and Validation (ICST)*, pages 248–257, 2008.

[15] Jiang Zheng, Laurie Williams, Nachiappan Nagappan, John P. Hudepohl, and Mladen A. Vouk. On the value of static analysis for fault detection in software. *IEEE Transactions on Software Engineering*, 32(4), 2006.

Combinando análise fractal e mineração de séries temporais para identificação de extremos climáticos regionais

Santiago Augusto Nunes[1], **Luciana A. S. Romani** [2], **Ana M. H. Avila** [3], **Elaine P. M. de Sousa** [1], **Agma J. M. Traina** [1]

[1]ICMC - Universidade de São Paulo - São Carlos - Brasil
[2]Embrapa Informática Agropecuária - Campinas - Brasil
[3]Cepagri - Universidade Estadual de Campinas - Brasil

bynael@grad.icmc.usp.br, luciana@cnptia.embrapa.br, avila@cpa.unicamp.br, {parros, agma}@icmc.usp.br

Abstract. *In the last few decades, huge amounts of climate data from ground-based stations and other different sensors have been gathered and stored by several private and public institutions. The analysis of these data has become an important task due to worldwide climate changes and the consequent social and economic effects. In this work, we propose an approach to analyze multiple climate time series in order to identify intrinsic temporal patterns. By dealing with multiple time series as multidimensional data streams, we can integrate different climate variables and discover general behavior changes over time. Experimental studies on real climate time series collected from different regions of Brazil show the applicability of our approach.*

Resumo. *Nas últimas décadas, grandes quantidades de dados climáticos provenientes de estações meteorológicas e de outros tipos de sensores têm sido coletadas e armazenadas por diversas instituições. A análise desses dados tornou-se uma tarefa importante devido às mudanças climáticas e seus efeitos sociais e econômicos. Este trabalho propõe um processo de análise de múltiplas séries temporais climáticas para identificar padrões temporais intrínsecos aos dados. Considerando múltiplas séries como uma data stream, é possível integrar diferentes variáveis climáticas e detectar mudanças de comportamento ao longo do tempo. Estudos em séries climáticas reais coletadas em diferentes regiões do Brasil mostram o potencial de aplicação dessa abordagem.*

1. Introdução

Os avanços tecnológicos relacionados à coleta de dados meteorológicos fizeram com que conjuntos de dados climáticos crescessem rapidamente, trazendo novos desafios para a pesquisa em climatologia e sua aplicação em agricultura. Dada a importância da análise de dados históricos e recentes para compreender e caracterizar fenômenos climáticos, esses dados têm sido armazenados, gerando assim longas séries temporais históricas.

Os meteorologistas têm estudado esses conjuntos de dados com o objetivo de entender e caracterizar o clima global e prever possíveis cenários futuros em diferentes regiões do planeta. Dentre as técnicas estatísticas utilizadas, destacam-se análise de correlação, regressão, análise de componentes principais, distribuição de frequência, geoestatística e estatística não-paramétrica.

Atualmente, os pesquisadores não têm dúvida de que o clima mundial está mudando, com o aumento das temperaturas e a mudança na distribuição de chuvas. Estudos recentes preparados para o relatório do Painel Intergovernalmental de Mudanças Climáticas sugerem um aumento na média global de temperatura de 1,4 a 8,5 graus até o fim deste século. Isso pode levar a uma aceleração do ciclo hidrológico e consequentemente intensificar eventos extremos [Alexander et al. 2006, Ganguly and Steinhaeuser 2008]. Logo, eventos como deslizamentos de terra, inundações e tempestades podem ocorrer com maior frequência e intensidade [IPCC 2007], requerendo assim mais estudos para compreender melhor o clima tanto regional quanto mundial.

Nesse cenário, o estudo de dados climáticos é importante para entender as tendências de fenômenos extremos de modo a preparar especialistas e populações para situações adversas, criando condições para atenuar as consequências e auxiliar nas tomadas de decisões. De acordo com Ganguly e Steinhaeuser [Ganguly and Steinhaeuser 2008], há ainda inúmeros desafios na área de meteorologia e climatologia que podem ser apoiados pela aplicação de técnicas conhecidas de mineração de dados, bem como pelo desenvolvimento de novas técnicas. Eles afirmam ainda que mesmo técnicas simples podem resultar em descobertas científicas com importante impacto social. À medida que o volume de dados cresce, passando da ordem de gigabytes para petabytes, torna-se fundamental o uso de técnicas computacionais para tornar possível análises mais abrangentes e eficientes, como por exemplo:

1. Análise eficiente de múltiplas séries temporais climáticas visando encontrar padrões e mudanças de tendência.
2. Identificação dos eventos climáticos extremos que indiquem alterações do clima regional e global.

Com o objetivo de auxiliar os especialistas do domínio na execução dessas tarefas, neste trabalho de Iniciação Científica foi proposto um processo para a análise de séries temporais climáticas, seguindo uma abordagem que avalia múltiplas séries temporais como uma *data stream* multidimensional, de maneira que cada série temporal define um atributo da *data stream*. Assim, é possível integrar múltiplas variáveis climáticas em um processo unificado que combina:

1. monitoramento da *data stream* para descoberta de padrões e detecção de mudanças de comportamento, utilizando conceitos da Teoria dos Fractais;
2. agrupamento visando encontrar similaridade e desvio de padrões revelados quando os dados são analisados com diferentes granularidades temporais;
3. e uma análise estatística para identificar mudanças em variáveis individuais associadas a uma variação significativa de comportamento.

Estudos experimentais realizados em séries temporais climáticas reais de diferentes regiões do Brasil indicam que essa abordagem pode ser uma ferramenta útil para auxiliar os especialistas na análise de grandes conjuntos de dados climáticos. O trabalho apresentado neste artigo foi desenvolvido no escopo do projeto "Agrodatamine: Desenvolvimento de métodos e técnicas de mineração de dados para apoiar pesquisas em mudanças climáticas, com ênfase em agrometeorologia"[1]. O processo de análise proposto

[1]Projeto apoiado pelo instituto Microsoft Research - FAPESP

foi desenvolvido como trabalho de iniciação científica, com o apoio de especialistas de domínio da Embrapa Informática Agropecuária - Campinas e do Cepagri-Unicamp, parceiros do projeto, para análise e validação dos resultados.

Este artigo está organizado da seguinte maneira: Seção 2 apresenta conceitos da Teoria de Fractais e sua aplicação na análise de *data streams*. Seção 3 descreve a abordagem proposta para análise de séries climáticas. Resultados experimentais em dados reais são discutidos na Seção 4. A Seção 5 apresenta as considerações finais.

2. Conceitos Relacionados

Um fractal pode ser definido pela propriedade de auto-similaridade, isto é, um objeto que apresenta as mesmas características em diferentes escalas [Schroeder 1991]. Assim, um conjunto de dados reais exibe um comportamento fractal se é exatamente ou estatisticamente auto-similar, de modo que as partes de qualquer tamanho dos dados apresentam as mesmas características do conjunto de dados inteiro.

Na Teoria de Fractais, a dimensão fractal é particularmente útil para a análise dos dados, pois fornece uma estimativa da dimensão intrínseca D de conjuntos de dados reais. A dimensão intrínseca fornece a dimensionalidade do objeto representado pelos dados, independentemente da dimensão E do espaço em que está inserido. Em outras palavras, D mede o comportamento intrínseco de dados reais [Faloutsos and Kamel 1994, Traina et al. 2005]. Por exemplo, um conjunto de pontos que definem um plano imerso em um espaço tridimensional ($E = 3$) tem dois atributos independentes e um terceiro correlacionado com os outros, resultando em $D = 2$.

A dimensão fractal de conjuntos de dados reais pode ser determinada pela Dimensão Fractal de Correlação D_2. Uma abordagem eficiente para medir a dimensão fractal de conjuntos de dados imersos em espaços E-dimensional é o método *BoxCounting* [Schroeder 1991], que define D_2 como apresentado na Equação 1, onde r é o lado das células de um hiper-reticulado que divide o espaço de endereçamento do conjunto de dados e $C_{r,i}$ é a contagem de pontos na célula i.

$$D_2 \equiv \frac{\partial log(\sum_i C_{r,i}^2)}{\partial log(r)} \quad r \in [r_1, r_2] \tag{1}$$

Um algoritmo eficiente (com custo linear sobre o número de elementos no conjunto de dados) para calcular D_2 foi proposto em [Traina et al. 2000]. Assim, D_2 pode ser uma ferramenta útil para estimar a dimensão intrínseca D de conjuntos de dados reais, com custo computacional viável. Conceitos da Teoria dos Fractais têm sido aplicados a diversas tarefas em mineração e análise de dados, tais como estimativa de seletividade [Baioco et al. 2007], detecção de agrupamentos [Barbará and Chen 2000], previsão em séries temporais [Chakrabarti and Faloutsos 2002], detecção de correlação [Sousa et al. 2007b] e análise da distribuição de dados [Traina et al. 2005].

As informações de comportamento intrínseco fornecidas pela dimensão fractal D_2 também podem ser aplicadas para detectar mudanças de comportamento em *data streams* evolutivas. Basicamente, a ideia é medir continuamente a dimensão fractal de uma *data stream* ao longo do tempo visando monitorar seu comportamento. Assim, variações significativas nas medidas sucessivas de D_2 podem indicar as mudanças nas características intrínsecas dos dados.

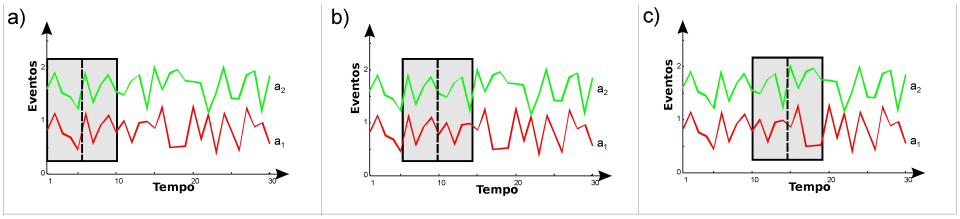


Figura 1. Janela deslizante sobre uma *data stream* bi-dimensional.

Em [Sousa et al. 2007a] é proposta uma técnica para acompanhar as mudanças de comportamento de *data streams* evolutivas, com o algoritmo *SID-meter* para medir continuamente D_2 ao longo do tempo. O *SID-meter* considera uma *data stream* como uma sequência de eventos $< e_1, e_2, ... >$ ordenada e potencialmente infinita onde cada evento é representado por um conjunto de E medidas, isto é, $e_i = (a_1, a_2, ..., a_E)$.

Para medir a dimensão fractal de uma *data stream* ao longo do tempo, o *SID-meter* define uma janela deslizante para limitar eventos sucessivos que serão considerados no cálculo de D_2. A janela é dividida em n_c períodos de tempo (chamados de períodos de contagem), cada qual contendo um número pré-determinado de eventos (n_i). Portanto, $n_i * n_c$ determina o tamanho da janela deslizante e n_i representa o tamanho do deslocamento. O valor de D_2 é, então, continuamente calculado considerando os eventos contidos em cada janela e atualizado quando n_i novos eventos são lidos, descartando os n_i eventos mais antigos. A Figura 1 ilustra uma *data stream* bi-dimensional (atributos a_1, a_2) processados por meio de uma janela deslizante dividida em 2 períodos de contagem ($n_c = 2$). Na Figura 1a), pode-se notar que no intervalo temporal de 1 a 10 (a primeira janela), os atributos a_1 e a_2 apresentam comportamento semelhante, pois são linearmente correlacionados. Nas Figuras 1b) e 1c), os eventos antigos são descartados enquanto novos dados são processados. A partir do tempo 10, o comportamento dos atributos a_1 e a_2 torna-se diferente, ou seja, eles não estão mais correlacionados. Quando ocorre alteração na correlação entre os atributos, o valor de D_2 reflete essas mudanças. Portanto, com o monitoramento da dimensão fractal é possível acompanhar a evolução do comportamento dos dados.

3. Processo de Análise

O processo de análise proposto para detectar padrões de interesse em múltiplas séries temporais combina diferentes técnicas: monitoramento da dimensão fractal de *data streams*, agrupamento de dados e análise estatística.

A Figura 2 ilustra o processo completo. O primeiro passo é associar múltiplas séries temporais em uma *data stream* multidimensional, ou seja, cada série temporal determina um atributo da *data stream*. Por exemplo, séries temporais de temperatura e de precipitação podem ser agregadas em uma única *data stream* bi-dimensional. Essa abordagem permite uma análise integrada das diferentes variáveis climáticas, a fim de descobrir as alterações de comportamento global ao longo do tempo. Em outras palavras, é possível avaliar como as variáveis estão correlacionadas e como essas correlações variam, especialmente quando mudanças de comportamento significativas são identificadas.

O segundo passo é a análise fractal de uma *data stream* utilizando a abordagem proposta no *SID-meter*. O algoritmo *SID-meter* foi inicialmente projetado para trabalhar

com uma única janela deslizante com tamanho pré-definido. Uma extensão do *SID-meter* [Nunes et al. 2010] foi projetada para manipular janelas com diferentes configurações de períodos de contagens (n_c) e número de eventos (n_i) simultaneamente em uma única leitura da *data stream*. Como resultado, pode-se encontrar padrões temporais semelhantes ou distintos que ocorrem em diferentes granularidades de tempo, por exemplo, mensalmente ou anualmente.

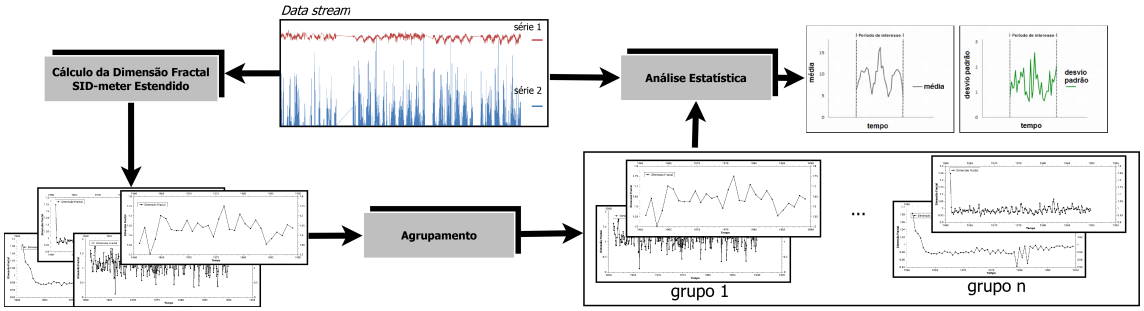


Figura 2. Processo de análise de múltiplas séries temporais.

O *SID-meter* estendido gera janelas deslizantes de tamanhos diferentes a partir de parâmetros iniciais. Basicamente, o menor e o maior tamanho de janela devem ser determinados por meio das seguintes configurações:

- o menor e o maior valor de n_c (períodos de contagem) e de n_i (número de eventos);
- o valor de incremento de n_c e de n_i.

Por exemplo, considere uma *data stream* composta de medidas diárias: configurando n_c variando de 3 a 6 com incremento de 1, e n_i variando de 30 (um mês para cada período de contagem, aproximadamente) a 365 (1 ano para cada período de contagem) com incremento de 30, define-se janelas variando de $3 * 30 = 3$ meses a $6*365 = 6$ anos. Esses parâmetros devem ser determinados considerando a granularidade temporal dos dados e os objetivos da análise a ser realizada. Portanto, os especialistas do domínio são fundamentais nessa tarefa.

A saída do *SID-meter* estendido é um conjunto de gráficos de D_2 gerados para as janelas de diferentes tamanhos. Assim, na terceira etapa do processo de análise, mapeia-se cada gráfico em uma série temporal de medidas de D_2. As séries temporais de D_2 são, então, agrupadas com o objetivo de encontrar padrões semelhantes em granularidades de tempo diferentes e padrões distintos que são detectados em apenas algumas janelas temporais. Além disso, os elementos centrais dos agrupamentos fornecem algumas informações adicionais para os especialistas do domínio sobre como escolher janelas de interesse para uma análise mais aprofundada, ou seja, um elemento central representa o padrão geral do grupo (*cluster*) e, portanto, a janela temporal correspondente (com uma granularidade de tempo específica) pode ser analisada em mais detalhes.

A tarefa de agrupamento combina o *K-Medoids* [Kaufman and Rousseeuw 1990], que é um método de agrupamento baseado em partições, com a função de distância *Dynamic Time Warping* (DTW) [Berndt and Clifford 1994] para medir a similaridade entre as séries temporais de D_2. Essas técnicas são bastante utilizadas em trabalhos de mineração de séries temporais e apresentaram resultados satisfatórios nos primeiros estudos empíricos.

O último passo do processo é a análise estatística, que pode ser aplicada considerando-se apenas as janelas temporais representadas pelos elementos centrais dos agrupamentos. Medidas de estatística básica, como média e desvio padrão, podem indicar a variação individual das variáveis climáticas em períodos com alterações significativas na dimensão fractal. O objetivo principal é descobrir como cada variável climática influencia as mudanças de correlação, em especial aquelas relacionadas a eventos extremos.

Este processo foi aplicado em séries temporais georreferenciadas provenientes de diferentes regiões do Brasil, com o objetivo de identificar mudanças de comportamento significativas ao longo do tempo e relacioná-las aos eventos climáticos extremos e relevantes. Os resultados são apresentados na Seção 4.

4. Experimentos

Foram realizados experimentos com séries temporais reais para validar a abordagem proposta. Dois conjuntos de dados de diferentes regiões do Brasil foram utilizados:

1. Sul: séries climáticas reais, fornecidas pelo Agritempo[2], constituídas de medidas diárias de precipitação e temperatura média mensuradas por 19 estações meteorológicas da região Sul no período 1994-2008.
2. Sudeste: séries climáticas reais, também fornecidas pelo Agritempo, constituídas de medidas diárias de precipitação e temperatura média mensuradas por 10 estações meteorológicas do estado de São Paulo, no período de 1995 a 2009.

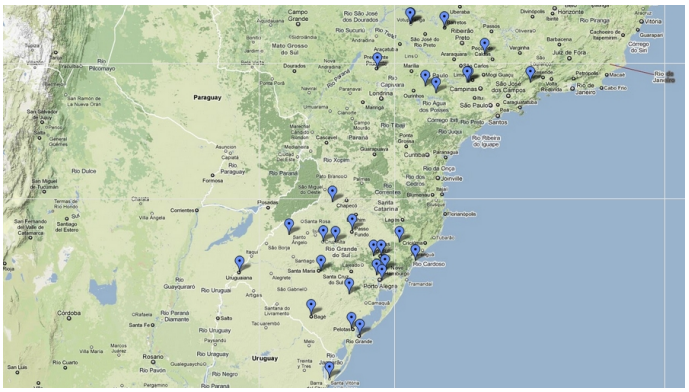


Figura 3. Localização das estações meteorológicas - SP e região Sul.

As séries temporais analisadas nos experimentos são georreferenciadas, uma vez que a posição geográfica das estações meteorológicas afeta significativamente a análise dos resultados. Em geral, o clima apresenta características diferentes em regiões geograficamente distintas. A Figura 3 mostra a localização das estações meteorológicas nas regiões Sul e Sudeste. Vale ressaltar que o estado de São Paulo tem um número significativo de estações, quando comparado aos outros estados da região Sudeste e, portanto, é analisado separadamente.

Para cada conjunto de dados gerou-se uma *data stream* bi-dimensional composta pelos atributos de precipitação e temperatura média. Como parâmetros iniciais do *SID-meter* estendido definiu-se o número de períodos de contagem (n_c) de 2 a 5, com

[2]www.agritempo.gov.br/

deslocamentos (n_i) de 1 mês a 1 ano. Em outras palavras, foram definidas janelas com tamanho variando de dois meses a cinco anos.

Os gráficos que descrevem a dimensão fractal D_2 medida ao longo do tempo pelo *SID-meter* estendido para as duas *data streams* apresentam tendências significativamente diferentes, embora representem as mesmas variáveis climáticas (temperatura e precipitação) e o mesmo intervalo de tempo. As diferenças nos padrões observados nos gráficos mostram as características climáticas distintas das duas regiões. Também observa-se que, para ambos os conjuntos de dados, o valor de D_2 permaneceu em torno de 1, o que indica que temperatura média e precipitação são variáveis correlacionadas. Esse comportamento é consistente com as expectativas dos especialistas, uma vez que a correlação entre essas variáveis alterna de uma correlação mais forte em um determinado período a uma menor correlação em outros.

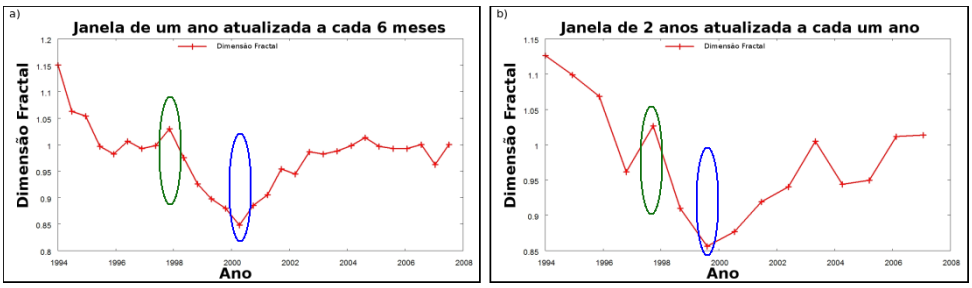


Figura 4. Variação de D_2 para a região Sul: (a) janela de um ano com deslocamento de 6 meses; (b) janela de 2 anos com deslocamento de um ano.

A Figura 4 mostra dois gráficos de D_2 para a região Sul, considerando janelas temporais de diferentes tamanhos. É possível observar que, embora os gráficos estejam relacionados a diferentes granularidades temporais, há um padrão similar de variação de D_2. Ambos os gráficos mostram um pico em 1998 e um declínio no valor de D_2 em torno do ano 2000. Ao avaliar a flutuação da média e do desvio padrão para as variáveis de precipitação e temperatura durante 1997/1998 e 2000/2001, pode-se observar maiores valores de desvio padrão da precipitação (variação 14,7 a 19,8) no primeiro período e maiores valores de desvio padrão da temperatura (variação 6,2 a 8,9) no segundo período. De acordo com especialistas e registros de dados de temperatura, os invernos de 2000 e 2001 foram marcados por baixas temperaturas, clima muito frio e neve no Rio Grande do Sul. Além disso, o período de 1999/2001 coincide com a ocorrência de La Niña, um fenômeno climático caracterizado pelo resfriamento das águas do Pacífico.

Durante o período 1997-1998 ocorreu um El Niño muito forte. El Niño é um fenômeno de grande escala que ocorre no Oceano Pacífico, ao longo da costa da América do Sul, e é caracterizado por um aumento na temperatura da água do mar nessa região. Além disso, os Complexos Convectivos de Meso-escala (CCMs) influenciam fortemente a precipitação ao longo do noroeste do Rio Grande do Sul durante a primavera [Velasco and Fritsch 1987]. Durante o fenômeno El Niño, eventos extremos podem mover-se para o sul ou para o norte, mas ambos os casos indicam a importância da parte ocidental do Sul do Brasil no aumento de eventos extremos durante anos de El Niño. CCMs são frequentes nessa área e aproveitam as anomalias de circulação durante os episódios de El Niño [Grimm and Tedeschi 2008].

A Figura 5 mostra dois gráficos de D_2 para São Paulo, considerando duas janelas temporais de diferentes tamanhos com um padrão semelhante de comportamento. A Figura 5a) corresponde a uma janela de dois anos atualizada a cada ano. É possível observar que o valor de D_2 diminui no período de 2000 a 2002, que coincide com a ocorrência de La Niña. Nesse período (2000/2001), houve um alerta em São Paulo sobre a possibilidade de racionamento de eletricidade no centro Sul, provavelmente devido a uma diminuição de precipitação na região central do Brasil. No Sudeste, os anos de 2000 e 2001 foram caracterizados por uma estação seca pronunciada. Em 2000, os meses de abril, maio e junho foram muito secos com atraso na chegada da estação chuvosa.

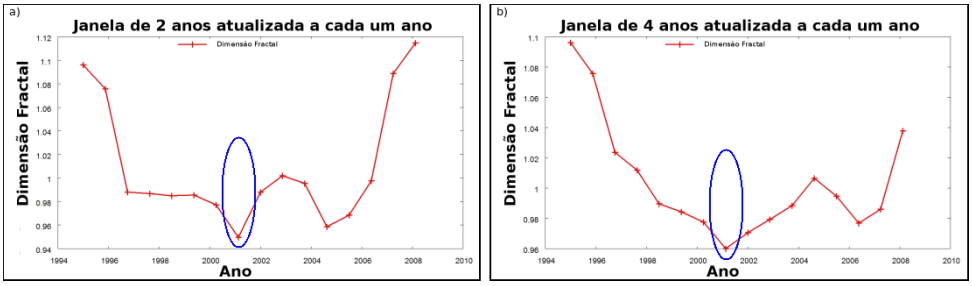


Figura 5. Variação de D_2 para o estado de São Paulo: (a) janela de 2 anos com deslocamento de um ano; (b) janela de 4 anos com deslocamento de um ano.

Os gráficos das Figuras 5a) e 5b) mostram uma mudança de tendência entre 2004 e 2009. Uma análise estatística desse período indica que o ano de 2007 apresentou uma grande variação nos valores da média e desvio padrão para as variáveis temperatura e precipitação. Em 2007, a região teve um verão quente e chuvoso, e um outono mais úmido e um pouco mais quente. Segundo os especialistas, em anos de El Niño há um aumento nas ocorrências de chuvas entre outubro e fevereiro, com uma pausa em janeiro. No entanto, no caso de precipitação mensal total, essa variação é menos pronunciada, ocorrendo em algumas regiões e em outras não.

Em geral, observa-se que, para ambas as regiões analisadas, os anos de El Niño apresentaram uma diminuição na correlação entre a temperatura e a precipitação (maior D_2). Por outro lado, as ocorrências de La Niñas estão relacionadas com um aumento desta correlação (menor D_2). Segundo meteorologistas, em anos de La Niña, as chuvas são causadas pelos CCMs, que são chuvas intensas, tempestades severas, o que pode causar uma descontinuidade na relação entre as variáveis.

O processo de análise inclui também uma etapa de agrupamento sobre os gráficos gerados pelo *SID-meter* estendido. O objetivo principal nesta etapa é encontrar padrões similares que são revelados mesmo que os dados analisados tenham granularidades de tempo diferentes, tais como as variações relevantes de D_2 identificadas nas Figuras 4a) e 4b). Além disso, o especialista também recebe como saída os gráficos selecionados como elementos centrais dos grupos. Assim, pode-se determinar o tamanho da janela apropriada a ser usada para destacar os fenômenos climáticos que estão sendo avaliados. Para ilustrar, a Figura 6 mostra os elementos centrais de cinco agrupamentos gerados para os gráficos de São Paulo. Com os elementos centrais, o especialista pode ter uma visão geral do conjunto de gráficos dentro de cada agrupamento e definir quais grupos poderiam ser utilizados para análises posteriores. Essa abordagem facilita a tarefa de análise, já que o número de possíveis combinações de configurações de janela pode ser elevado.

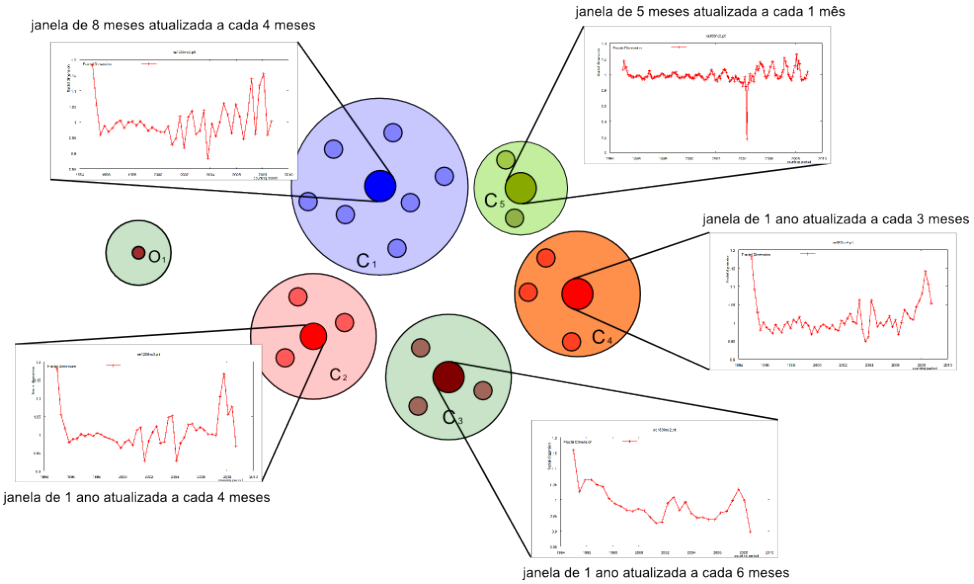


Figura 6. Grupos de gráficos D_2 para São Paulo (C - grupos e O - extremos).

5. Conclusão

A análise de séries temporais de precipitação e temperatura baseada em ferramentas da Teoria dos Fractais revela que existe um padrão rítmico no tempo entre as variáveis. Quando ocorre uma mudança de tendência, a abordagem apresentada neste trabalho é capaz de identificar a mudança e as variáveis responsáveis pela variação no padrão. Assim, é possível relacionar essas variações com as mudanças nos padrões climáticos globais, como o El Niño. Estes resultados indicam que o estudo de séries temporais com mais de uma variável pode permitir a identificação de padrões em mais detalhes do que utilizando apenas uma variável. Além disso, essa abordagem pode ajudar a compreender a interdependência entre as variáveis meteorológicas. Portanto, a análise realizada diretamente sobre dados de estações meteorológicas permite identificar o comportamento dos dados de diferentes estações, detectando eventos extremos em cada região.

Os trabalhos futuros incluem a comparação do comportamento de dados climáticos reais com o de dados simulados gerados a partir de modelos climáticos. Assim, pretende-se detectar tendências no comportamento dos dados reais e possíveis desvios nos modelos climáticos, possibilitando o refinamento desses modelos e maior compreensão dos especialistas do domínio sobre o comportamento dos dados reais.

Agradecimentos

Os autores agradecem Embrapa, Fapesp, CNPq, Capes e a Microsoft Research pelo apoio financeiro e ao Agritempo por fornecer os dados climáticos reais.

Referências

Alexander, L., Zhang, X., Peterson, T., Caesar, J., Gleason, B., Tank, A., Haylock, M., Collins, D., Trewin, B., Rahimzadech, F., Tagipour, A., Kumar, K. R., Revadekar, J., Griffiths, G., Vincent, L., Stephenson, D., Burn, J., Aguilar, E., Brunet, M. Taylor, M., New, M., Zhai, P., Rusticucci, M., and Vasquez-Aguirre, J. (2006). Global

observed changes in daily climate extremes of temperature and precipitation. *Journal of Geophysical Research*, 111:1–22.

Baioco, G. B., Traina, A. J. M., and Traina, C. (2007). Mamcost: Global and local estimates leading to robust cost estimation of similarity queries. In *Proceedings of SSDBM*, pages 6–16, Banff, Canada.

Barbará, D. and Chen, P. (2000). Using the fractal dimension to cluster datasets. In *Proceedings of KDD*, pages 260–264, Boston, USA.

Berndt, D. J. and Clifford, J. (1994). Using dynamic time warping to find patterns in time series. In *Proceedings of AAAI94 Workshop on Knowledge Discovery in Databases*, pages 359–370, Seattle, USA.

Chakrabarti, D. and Faloutsos, C. (2002). F4: large-scale automated forecasting using fractals. In *Proceedings of CIKM*, pages 2–9, McLean, USA.

Faloutsos, C. and Kamel, I. (1994). Beyond uniformity and independence: Analysis of r-trees using the concept of fractal dimension. In *Proceedings of PODS*, pages 4–13, Minneapolis, USA.

Ganguly, A. R. and Steinhaeuser, K. (2008). Data mining for climate change and impacts. In *Proceedings of ICDM*, pages 385–394, Pisa, Italy.

Grimm, A. and Tedeschi, R. G. (2008). Enso and extreme rainfall events in south america. *Journal of Climate*, 22:1589–1609.

IPCC (2007). Intergovernmental panel on climate change. `http://www.ipcc.ch/ipccreports/index.htm`. accessed: March, 2009.

Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data: an Introduction to Cluster Analysis*. John Wiley and Sons.

Nunes, S. A., Romani, L. A. S., Avila, A. M. H., Traina, C., Sousa, E. P. M., and Traina, A. J. M. (2010). Análise baseada em fractais para identificação de mudanças de tendências em múltiplas séries climáticas. In *Proceedings of SBBD Short Papers*, pages 65–72, Belo Horizonte, Brasil.

Schroeder, M. (1991). *Fractals, Chaos, Power Laws*. W. H. Freeman and Company.

Sousa, E. P. M., Traina, C., Traina, A. J. M., and Faloutsos, C. (2007a). Measuring evolving data streams' behavior through their intrinsic dimension. *New Generation Computing Journal*, 25:33–59.

Sousa, E. P. M., Traina, C., Traina, A. J. M., Wu, L., and Faloutsos, C. (2007b). A fast and effective method to find correlations among attributes in databases. *DMKD*, 14(3):367 – 407.

Traina, C., Sousa, E. P. M., and Traina, A. J. M. (2005). Using fractals in data mining. In Kantardzic, M. M. and Zurada, J., editors, *New Generation of Data Mining Applications*, volume 1, pages 599–630 (Chapter 24). Wiley/IEEE Press.

Traina, C., Traina, A. J. M., Wu, L., and Faloutsos, C. (2000). Fast feature selection using fractal dimension. In *Proceedings of SBBD*, pages 158–171, João Pessoa, Brasil.

Velasco, I. and Fritsch, J. (1987). Mesoscale convective complexes in the americas. *J. Geophys. Res.*, 92:9591–9613.

Explorando a Dissimilaridade em Sistemas Colaborativos de Recomendação

Lucas Miranda, Fernando Mourão e Wagner Meira Jr.

[1]Departamento de Ciência da Computação - Universidade Federal de Minas Gerais

{lucmir,fhmourao,meira}@dcc.ufmg.br

Abstract. *The huge amount of options available in various commercial applications became Recommender Systems (RS) crucial tools to assist users in their choices. Despite recent advances in RS, there is still room for more effective techniques which are applicable to a larger number of domains. Most problems arise from the simplified model recurrently used. In this paper, we propose a richer user modeling which allows to extrapolate the usual similarity analysis. Furthermore, we propose a technique that, by exploiting an information type defined as dissimilarity, provides significant improvements over traditional techniques based on collaborative systems, as well as reduces the analysis cost required by such techniques.*

Resumo. *O grande volume de opções existentes em variadas aplicações comerciais tornaram Sistemas de Recomendação (SR) ferramentas cruciais para auxiliar os usuários em suas escolhas. Apesar dos avanços recentes em SR, há ainda uma necessidade por técnicas mais eficazes e aplicáveis a um número maior de domínios. Muitos problemas existentes decorrem do emprego de um modelo restrito das relações entre usuários. Neste trabalho, propomos uma modelagem diferenciada, que permite extrapolar a usual análise de similaridade. Além disso, propomos uma técnica que, explorando informações definidas como dissimilaridade, provê melhorias significativas sobre métodos tradicionais de sistemas colaborativos, bem como reduz o custo de análise requerido por tais técnicas.*

1. Introdução

O grande volume de dados disponível na WEB gerou, recentemente, um cenário desafiador para variadas aplicações. Usuários possuem mais opções que efetivamente podem manipular [Adomavicius and Tuzhilin 2005]. Neste cenário, cresce a necessidade por mecanismos especializados em filtrar conteúdo e auxiliar decisões. Sistemas de Recomendação (SRs) são ferramentas que, considerando o histórico dos usuários, realizam tal filtragem através de indicações personalizadas de itens [Burke 2002].

Uma das técnicas mais simples e eficazes de recomendação existentes são os "Sistemas Colaborativos de Recomendação"[1]. SCR consistem, basicamente, em identificar grupos de usuários com hábitos de escolha similares, de forma que os itens a serem recomendados para um dado usuário sejam aqueles de maior interesse para o seu grupo. A correlação entre indivíduos é, usualmente, representada por meio de uma rede, em que usuários são vértices e as relações são definidas em função do grau de similaridade.

Como argumentado em [Adomavicius and Tuzhilin 2005], embora haja inúmeras propostas para SCRs, os atuais sistemas ainda requerem melhorias para tornar a

[1]Tal como feito em alguns trabalhos [Koren 2009], referenciamos SCR apenas como o conjunto de técnicas baseada em $k\text{-}nearest\text{-}neihghbor$.

recomendação mais eficaz e aplicável a cenários mais diversificados. Algumas das limitações de SCRs decorrem do fato de tais métodos serem aplicados sobre uma modelagem simplificada de rede, que não agrega diversas informações potencialmente relevantes. Uma das informações negligenciadas pela maioria destes sistemas é a existência de níveis distintos de similaridade. Um indivíduo pode ser, ao mesmo tempo, similar a algumas pessoas e ter preferências completamente distintas das de outro conjunto de usuários. Tais níveis de similaridade são particularmente relevantes por endereçar um grande desafio existente em SRs: a escassez de informação sobre usuários ou itens.

Dada a complexidade de se avaliar todos os níveis de similaridade, considerando o balanceamento entre eficácia e eficiência de utilização de tais informações, este trabalho de iniciação científica limita-se a analisar a dissimilaridade em SCR. Definimos como dissimilaridade, informações obtidas a partir do conjunto de usuários menos similares a cada usuário. Como principais contribuições, destacamos: **(1)** a proposta de uma nova modelagem de rede para SCR, que permite considerar distintos níveis de similaridade; **(2)** a caracterização e verificação de duas hipóteses sobre os dados do Netflix[2]: *1- existem diversos níveis de similaridade no cenário de recomendação*; *2- a incorporação da dissimilaridade pode beneficiar a tarefa de recomendação*; **(3)** a proposta de uma nova técnica baseada em SCRs que, através do uso da dissimilaridade, permite endereçar a escassez de informação, bem como uma limitação de SCRs atuais identificada em nossas caracterizações: o problema de *supervalorização*. Este problema refere-se ao fato dos SCRs superestimarem os valores de qualificações preditos, degenerando a qualidade das recomendações, sobretudo dos itens mais desaprovados. Avaliações sobre o Netflix demonstraram que nossa técnica é capaz de prover melhorias sobre técnicas tradicionais, atingindo 8,43% de ganhos, e de reduzir o custo de análise destas técnicas. Além disso, uma discussão sobre as análises realizadas esclarece o que diz respeito ao emprego apropriado da dissimilaridade em SCRs.

2. Trabalhos Relacionados

Grande parte dos esforços presentes na literatura sobre recomendadores constituem os chamados "Sistemas Colaborativos de Recomendação"(SCR), que consideram somente informações históricas dos usuários ou itens. Em particular, a popularidade de técnicas deste tipo deve-se à sua simples modelagem de correlação entre usuários. Em SCR, tipicamente, para cada indivíduo, define-se um conjunto de usuários *"vizinhos"*, cujas avaliações anteriores sejam similares. Pontuações para cada item desconhecido pelo usuário são preditas a partir dos pesos atribuídos pelos vizinhos mais próximos [Breese et al. 1992]. Como SCRs são baseadas em agrupamentos, sua efetividade depende dos grupos gerados expressarem boas correlações entre os usuários.

Apesar dos inúmeros estudos em SCRs, existem poucos questionamentos sobre a limitação de informações agregadas que é imposta pela modelagem quando se trabalha com uma vizinhança restrita aos usuários mais similares. Em [Hu et al. 2008] é proposto um modelo que considera, além da vizinhança formada por usuários similares, informações implícitas, tais como preferências e níveis de confiança. Em [Kagie et al.] são discutidas métricas que podem ser adotadas para incorporação da dissimilaridade na construção do modelo. Nosso trabalho se diferencia dos demais da área por constituir uma proposta simples, mas efetiva, de uma nova modelagem em recomendação. Muitos trabalhos, apesar de apresentarem propostas sofisticadas de algoritmos para recomendação, negligenciam informações importantes ao adotarem um modelo restrito. Além disso, não

[2]A base de dados do Netflix contém qualificações explícitas de usuários sobre filmes.

encontramos trabalhos precedentes que investigaram o emprego da dissimilaridade em recomendação, por uma perspectiva similar à nossa.

3. Sistemas Colaborativos de Recomendação

De uma maneira geral, SCRs seguem três passos: **(1)** cálculo da similaridade entre o usuário u para o qual se deseja recomendar e os demais usuários do sistema; **(2)** determinação do subconjunto de usuários N_u mais similares a u (vizinhos); **(3)** ponderação das avaliações dos usuários em N_u para formação da avaliação predita. Valores de qualificação sobre um item[3], para um dado usuário, são derivados considerando uma soma ponderada dos votos de seus vizinhos. Nestes algoritmos, os votos são, usualmente, tratados de maneira indistinta, atribuindo a mesma importância às avaliações de vizinhos com diferentes graus de similaridade. A equação a seguir descreve o algoritmo tradicional para estimar a avaliação geral $\hat{r}$ que o usuário u concederia ao item i. A variável $r_{v,i}$ representa a qualificação que o usuário v atribuiu, anteriormente, ao item i.

$$\hat{r}_{u,i} = \frac{\sum_{v \in N_u} sim(u,v) r_{v,i}}{\sum_{v \in N_u} sim(u,v)}$$

A qualidade dessa abordagem de recomendação depende, fundamentalmente, da modelagem de rede utilizada. Dessa forma, modelos simplórios podem limitar a qualidade do recomendador. Por exemplo, considerar apenas os comportamentos similares pode restringir a quantidade de informações disponíveis para os recomendadores. A seção seguinte detalha nossa proposta de uma nova modelagem dos dados, capaz de agregar informações potencialmente relevantes para a aplicação das técnicas de SCRs.

4. Modelo de Rede

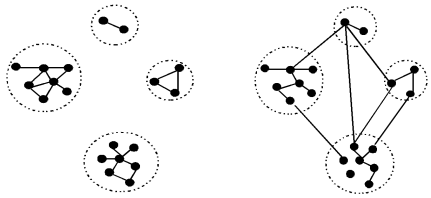
Atualmente, a escassez de informações persiste entre os desafios mais estudados em SRs [Adomavicius and Tuzhilin 2005]. Neste trabalho, a fim de identificar outro tipo de informação relevante para SCRs em cenários de escassez, consideramos as informações fornecidas pelo conjunto de indivíduos menos similares a cada usuário, informações estas definidas como **Dissimilaridade**. Entretanto, para que seja possível avaliá-las, faz-se necessário o uso de um modelo mais rico e representativo de rede.

Em SCR, o modelo de rede consiste em um grafo $G(V, A)$, em que o conjunto de vértices V representa os usuários e o conjunto de arestas A representa suas relações. Relacionamentos entre usuários são determinados por uma função de similaridade, tal como Cosseno ou Pearson [Adomavicius and Tuzhilin 2005]. Para fins de podagem, o número de vizinhos é reduzido para um parâmetro K, definindo os "k-vizinhos mais próximos". Apesar desta modelagem ser muito empregada, por ser simples e intuitiva, alguns problemas podem comprometer sua qualidade. Por exemplo, muitas informações são perdidas ao se considerar somente um grupo restrito de usuários.

Em nossa proposta de modelagem, a rede $G(V, A)$ é composta por relacionamentos entre todos os usuários que possuam uma interseção mínima entre seus históricos de transações. Por exemplo, considerando o domínio de vídeos, haverá uma relação entre dois usuários se existir uma quantidade I mínima (*intensidade* do relacionamento) de filmes que ambos avaliaram. Dessa forma, é possível definir relacionamentos entre usuários com preferências distintas sobre os mesmos itens. Com isso, a similaridade entre

[3] A tarefa de quantificar o interesse de um usuário por um dado item é, frequentemente, chamada de predição. Outra modalidade de recomendação, chamada de "top-N", é derivada no formato de uma lista de itens ordenados pelo potencial de interesse para o usuário.

as preferências deixa de ser o fator determinante dos relacionamentos, para ser um atributo S. Outros dois atributos importantes que associamos a cada relacionamento A_{u_i,u_j}, são denominados *representatividades* do relacionamento: R_{u_i} e R_{u_j}. R_{u_i} é definida como o tamanho da interseção entre os históricos de u_i e u_j sobre o tamanho do histórico de u_i. Essa medida expressa quanto do perfil de cada usuário um relacionamento representa. Assim, nosso modelo consiste de um grafo não direcionado, multi-valorado em que cada relacionamento possui atributos I, S, R_{u_i} e R_{u_j}.



(a) Modelo Tradicional (b) Modelo Proposto

Figura 1. Modelo de Rede

A figura 1 contrasta a construção da rede tradicional de usuários similares *(a)* com a rede segundo o modelo proposto *(b)*. Como podemos observar, deixar que usuários com preferências distintas se relacionem permite ao modelo agregar uma quantidade maior de informações sobre os usuários, beneficiando cenários típicos de SCRs, tais como presença de usuários com gostos peculiares ou com poucas informações em seus históricos. Entretanto, uma possível limitação desta modelagem é que, como o volume de informações agregadas pela rede aumenta, o custo computacional para processar e armazenar tal rede tende a aumentar. Este problema pode ser controlado assumindo um compromisso, que depende do domínio de análise, entre custo e quantidade das informações agregadas.

5. Validação das Hipóteses

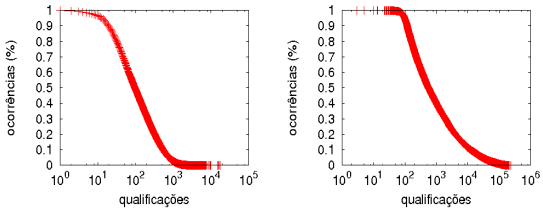
Nesta seção, verificamos, sobre um relevante domínio real, a existência das hipóteses: **(1)** *Existem diversos níveis de similaridade entre usuários em aplicações baseadas em SCR*; **(2)** *O emprego da dissimilaridade pode beneficiar a tarefa de recomendação*. De forma a facilitar o entendimento da discussão que segue, descrevemos a base de dados utilizada e apresentamos uma breve caracterização sobre a mesma.

Caracterização do Netflix

O *Netflix* é um serviço *online* de aluguel de filmes que disponibilizou, para fins de pesquisa em recomendação, uma base de dados com informações coletadas entre 01/10/1998 a 31/12/2005. A base contém características típicas de cenários reais, onde as técnicas de recomendação são frequentemente aplicadas. Seus dados incluem mais de 100 milhões de qualificações atribuídas por 480.189 usuários a 17.770 filmes.

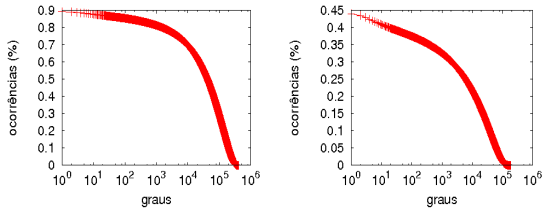
Realizamos uma caracterização da base, a fim de entender os hábitos dos usuários ao avaliar itens. Para tanto, foram extraídas informações estatísticas sobre as qualificações concedidas pelos usuários. Uma qualificação é um valor discreto entre 1 e 5. Investigando a distribuição destes valores, verificamos que os mais freqüentes são 4 (33,60%), 3 (28,67%) e 5 (23,06%). A predominância dos valores mais elevados é um indício de que os usuários tendem a qualificar, preferencialmente, os filmes que lhes agradam mais. Avaliamos também a quantidade de qualificações e a popularidade dos filmes, através da distribuição acumulada complementar do número de qualificações fornecidas por usuário e recebidas por filme, respectivamente, tal como mostra a figura 2. Os gráficos indicam que o número de qualificações 100 representa uma divisão no comportamento de ambas as curvas. Existem muitos usuários/filmes que concederam/receberam pelo menos

100 qualificações. Mas este número reduz drasticamente para valores maiores que 100. Este comportamento observado é comum em cenários que envolvem a interação entre múltiplos usuários e itens [Park and Tuzhilin 2008].



(a) Atribuída por Usuário (b) Recebida por Filme

Figura 2. Distribuição acumulada do número de qualificações



(a) Restrição $i = 20$ (b) Restrição $i = 120$

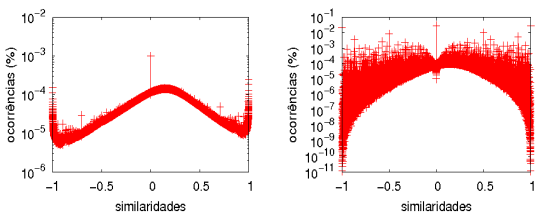
Figura 3. Distribuição de graus para redes formadas com diferentes restrições

Por fim, investigamos o número de relacionamentos entre usuários quando variamos o número mínimo I de interseção de histórico (i.e., intensidade mínima do relacionamento). O número 100 surge, novamente, como divisor do comportamento da curva. Nesse caso, ele indica que o número de pares de usuários que qualificaram pelo menos I filmes em comum reduz drasticamente quando I ultrapassa 100. Nota-se que o valor de I não pode ser muito superior à 100, caso em que a condição seria muito restritiva. Um valor muito abaixo de 100, por outro lado, pode tornar a restrição pouco efetiva, gerando redes gigantescas, porém, pouco informativas. Os gráficos da figura 3 contrastam a distribuição de graus das redes considerando valores da intensidade mínima para um caso pouco restritivo (I igual a 20) com um caso muito restritivo (I igual a 120).

Verificação da Primeira Hipótese

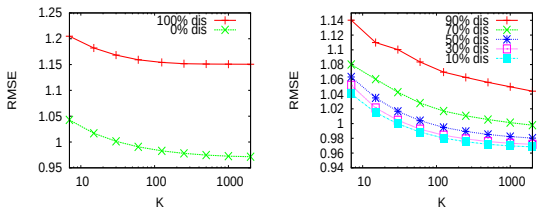
A primeira hipótese proposta defende que, em contextos de recomendação, existem diversos níveis de similaridade. Apesar de seu caráter intuitivo, é necessário verificá-la para quantificar os níveis de similaridade em cada domínio de forma a melhor incorporá-los nos recomendadores. O experimento realizado procurou mostrar que os usuários se relacionam de maneiras diferenciadas, podendo existir graus diversos de semelhanças entre eles. Para isso, todos os usuários foram combinados em pares e a similaridade de cada par foi computada, considerando as duas métricas mais populares (i.e., *Cosseno* e *Pearson*). Os gráficos da figura 4 exibem as distribuições de similaridade obtidas.

As duas distribuições apresentaram muita heterogeneidade em seus valores, o que sugere que, entre os vários relacionamentos de usuários da rede, existe, de fato, uma diversidade de níveis de similaridade. Além disso, é importante observar a existência de pontos próximos aos valores extremos de cada métrica, mostrando que há tanto usuários quase idênticos, quanto quase antagônicos.



(a) Similaridade Cosseno (b) Similaridade Pearson

Figura 4. Análise da Hipótese 1



(a) Similaridade vs. Dissimilaridade (b) Similaridade + Dissimilaridade

Figura 5. Análise da Hipótese 2

Verificação da Segunda Hipótese

A segunda hipótese afirma que a dissimilaridade pode ser utilizada em favor dos SCRs. Verificamos tal hipótese através de uma avaliação comparativa do desempenho de algoritmos de recomendação tradicionais com algoritmos que incorporam a dissimilaridade. Para tanto, é necessário definir como empregar a dissimilaridade e como medir a qualidade das recomendações. A dissimilaridade foi utilizada adaptando o algoritmo tradicional, discutido na seção 3, para considerar além dos vizinhos similares, os vizinhos dissimilares. Ou seja, todos os indivíduos da rede com valores de similaridade com cada usuário u_i a partir de δ são considerados vizinhos similares de u_i, enquanto indivíduos com similaridade até ω, são tidos como vizinhos dissimilares de u_i. Posteriormente, selecionamos apenas K vizinhos de cada usuário u_i para geração das predições para u_i. Em nossos experimentos, avaliamos versões distintas desta estratégia em que variamos percentualmente a quantidade de usuários dissimilares entre os K selecionados. Para avaliar a qualidade da recomendação, utilizamos a métrica RMSE [Koren 2009], que considera a distância entre o valor predito e a qualificação real como o erro da predição.

Nossas análises do uso da dissimilaridade para diferentes tamanhos K de vizinhança são sumarizadas nos gráficos da figura 5. Nestes experimentos, a similaridade foi calculada para todos os pares de usuários, sem restrições (ou seja, para δ=-1 e ω=1). Por questões de espaço, apresentamos apenas os resultados relacionados à utilização da correlação de Pearson como medida de similaridade, mas os resultados para testes com a distância cosseno foram similares. Além disso, por apresentar melhores resultados, utilizamos como medida de similaridade entre cada par de usuários o produto da correlação Pearson e os valores de representatividade entre os históricos de consumo de ambos, tal como definido na nossa rede. As predições foram realizadas para lista de pares $<$*usuário,item*$>$ presentes em um conjunto de teste [4], utilizando informações contidas em um conjunto de treinamento. O valor real da qualificação atribuída pelo usuário ao item, presente no conjunto de teste, foi utilizado para o cálculo de RMSE.

A figura 5 *(a)* contrasta o desempenho do uso restrito da similaridade (i.e., 0% de vizinhos dissimilares) com o uso restrito da dissimilaridade (i.e., 100% de vizinhos dissimilares). Nota-se, claramente, que a similaridade é mais precisa para caracterizar relacionamentos do que a dissimilaridade. A figura 5 *(b)* apresenta resultados para estratégias que combinam vizinhos similares e dissimilares. O melhor resultado foi obtido para combinações de 90% de similaridade e 10% de dissimilaridade. A qualidade alcançada com esta proporção superou, ligeiramente, o resultado obtido considerando apenas a similaridade. Outro resultado interessante é que o RMSE estabiliza-se em valores mais baixos com tamanhos muito menores de vizinhança. Enquanto que com o uso restrito da similaridade o RMSE estabiliza-se com aproximadamente $K = 1000$ vizinhos, com a melhor combinação de similaridade e dissimilaridade o RMSE estabiliza-se com aproximadamente $K = 200$ vizinhos. Isso representa um grande impacto sobre o custo computacional de SCRs, mostrando que com muito menos informações alcançamos resultados quase tão bons que os encontrados considerando-se grandes redes de vizinhos similares. Dessa forma, os resultados sugerem que a dissimilaridade pode beneficiar SCRs.

Discussão

Um entendimento mais geral sobre o impacto da dissimilaridade em SCRs requer analisar mais a fundo as derivações decorrentes da definição de indivíduos dissimilares.

[4] O conjunto de teste utilizado foi concedido, juntamente com a base de dados, pelo Netflix.

Dizer que dois indivíduos são dissimilares quanto ao gosto, consiste em dizer que tais indivíduos, de maneira geral, não gostam do mesmo tipo de itens. Conseqüentemente, estes indivíduos tendem a não consumirem itens do mesmo tipo frequentemente. Com isso, menos informações temos para definir a dissimilaridade, repercutindo em menos "confiança" sobre tal métrica. Essa é uma diferença essencial entre similaridade e dissimilaridade. Enquanto a similaridade possui um conjunto maior de indícios para se apoiar (i.e., indivíduos similares consomem os mesmos tipos de itens), o mesmo não ocorre com a dissimilaridade (embora o caráter multi-gosto dos usuários garanta sua existência). Intuitivamente, isto explica o comportamento pouco eficaz obtido para a dissimilaridade em nossos resultados. Além disso, temos uma certa redundância nas informações contidas na dissimilaridade. Por exemplo, um item que não se adeque ao gosto de um dado usuário possivelmente não se adequaria aos gostos dos vizinhos deste usuário e, conseqüentemente, já não seria recomendado a ele. Tais distinções entre similaridade e dissimilaridade sugerem que a visão local propiciada pela análise da vizinhança de cada indivíduo em uma rede de relacionamentos privilegia a similaridade.

Uma solução para estes problemas seria mudar nosso foco de análise para uma visão global. Identificar tendências globais do domínio pode trazer um conjunto de informações acerca da dissimilaridade muito mais úteis. Nossa proposta reside justamente em realizar tal análise global através do uso de regras de associação que expressam padrões recorrentes entre os usuários do domínio.

6. Sistema Colaborativo com Regras Semânticas

A exploração de informações expressas como regras semânticas, tais como *usuários que gostam do item X, não gostam do item Y*, corresponde à principal idéia do nosso algoritmo. A motivação para isso vem de três observações distintas. Primeiramente, explorar o domínio como um todo provê uma massa muito maior de informações, endereçando o problema de escassez de informação para a dissimilaridade. Segunda, regras similares evidenciam correlações freqüentes e consistentes presentes no domínio. Terceiro, este tipo de análise é complementar à análise local definida por estratégias de recomendação. Combinar aspectos de similaridade locais com fatores de dissimilaridades globais pode ser uma estratégia eficaz, principalmente por prover mais informações em casos onde a rede de vizinhança de um usuário não possui informações sobre determinados itens.

Dessa forma, apresentamos um novo algoritmo de recomendação que combina a técnica de filtragem colaborativa com regras de associação extraídas por técnicas tradicionais de mineração de dados. Descreveremos em detalhes tal proposta, bem como discutiremos seu impacto em termos de custo computacional. Nossa abordagem consiste, basicamente, na definição de dois passos fundamentais: a geração de regras semânticas e a incorporação das regras no algoritmo de predição.

Geração de Regras Semânticas

De forma a gerar as regras semânticas utilizadas no algoritmo, é necessário primeiramente definir o que são transações neste contexto. Uma transação é o conjunto de todos os itens qualificados por um usuário. Dessa forma, teremos tantas transações quanto usuários distintos, bem como um tamanho variado de transação por usuário. Além disso, itens bem qualificados foram diferenciados de itens qualificados negativamente. Ou seja, para cada usuário, geramos uma transação em que, os itens cujo valor de qualificação seja maior que a média de qualificações do usuário mais X desvios padrões são classificados como "positivos", e itens qualificados com valores abaixo dessa média menos X

desvios padrões são tidos como "negativos". Este valor X deve ser obtido empiricamente, uma vez que depende essencialmente do domínio considerado. O algoritmo *FP-Growth* [Han et al. 2000] foi utilizado para gerar *itemsets* freqüentes considerando o conjunto de transações e um parâmetro ρ que define o suporte mínimo. A partir dos *itemsets* gerados, e de um parâmetro τ que define a confiança mínima, obtemos as regras de associação.

Incorporação de Regras em SCRs

O passo seguinte à geração das regras semânticas consiste em incorporá-las ao sistema de recomendação. O algoritmo proposto consiste, basicamente, em uma adaptação do método tradicional de predição, discutido na seção 3. O processo de incorporação das regras possui, como primeira etapa, uma filtragem das regras de interesse. Uma vez mineradas no domínio todas as regras de associação com suporte mínimo maior que ρ e confiança mínima superior a τ, selecionamos, por questão de simplicidade, apenas regras, de qualquer tamanho, com um único item como conseqüente. Selecionadas as regras de interesse para o domínio, nosso próximo passo consiste em derivar as qualificações a partir do conjunto de regras resultantes.

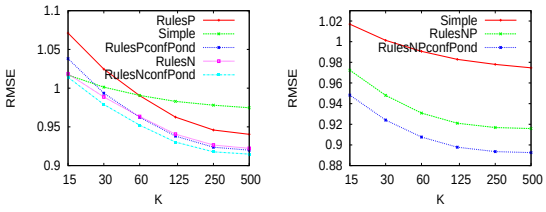
O algoritmo tradicional adota um sistema de votação baseado em informações dos vizinhos de cada usuário u para prever a qualificação que este concederia para um determinado item i. Assim, apenas os vizinhos de u que qualificaram o item i no passado são considerados para a predição. Caso nenhum vizinho tenha qualificado o item i, é retornado um valor derivado a partir das médias de todas qualificações atribuídas pelo usuário u e recebidas pelo item i no sistema. Em nosso algoritmo as regras semânticas são utilizadas como complemento para a predição apenas na ausência de uma qualificação atribuída por um vizinho. Ou seja, para cada vizinho v_j de u que não tenha qualificado o item i, derivamos uma qualificação $'r_{ji}$ de v_j para i a partir das regras semânticas previamente mineradas. Para derivar uma qualificação $'r_{ji}$, é necessário buscar regras que possuam i como conseqüente. Também é necessário que o vizinho v_j atenda aos antecedentes das regras a serem utilizadas (i.e., tenha qualificado da mesma forma, positivamente ou negativamente, cada um dos itens que formam o antecedente da regra). Atendendo a estes critérios, somente a regra resultante com maior confiança ($\mathcal{R}$) é escolhida para derivarmos $'r_{ji}$. Tal derivação pode ser realizada de variadas formas. A estratégia adotada, visando eficiência computacional, consiste em definir $'r_{ji}$ como a qualificação média que todos os usuários do domínio que atendem a regra $\mathcal{R}$ selecionada atribuíram ao item i. Caso nenhuma regra seja seleciona, tal como no algoritmo tradicional, derivamos a qualificação a partir das médias de qualificações dadas pelo usuário v_j e recebidas pelo item i.

Análise experimental

De forma a avaliar a qualidade do algoritmo proposto, foram definidos dois cenários de análise. Primeiramente, no cenário de regras homogêneas, avaliamos o uso de regras semânticas constituídas somente por itens qualificados positivamente ou negativamente. No cenário de regras heterogêneas, foram avaliadas regras constituídas por itens positivos e negativos simultaneamente. Em ambos cenários, avaliamos além das derivações de qualificações $'r_{ui}$ que cada usuário u atribuiria ao item i, tal como definido acima, duas variações do uso das regras semânticas. Uma delas pondera cada qualificação $'r_{ui}$ predita pelo valor de confiança da regra $\mathcal{R}$ selecionada para derivar a qualificação. Na segunda variação, ponderamos $'r_{ui}$ por um fator γ caso o valor predito seja menor que a média de qualificações atribuídas pelo usuário u, ou seja, caso $'r_{ui}$ represente uma predição negativa para u. Como os usuários no Netflix tendem a qualificar mais itens que lhes agradam, regras com itens positivos tornam-se mais freqüentes globalmente. Dessa

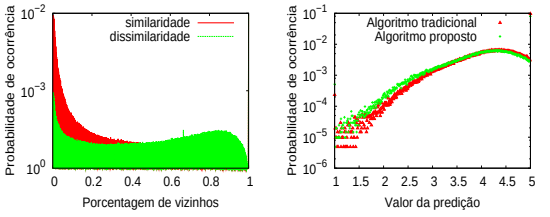
forma, a fim de valorizar ocorrências das qualificações negativas, e tornar a análise mais balanceada utilizamos o fator γ. Tal fator deve ser determinado empiricamente em cada domínio. Para o Netflix o melhor valor encontrado foi 1,4.

A figura 6 apresenta os valores de RMSE para ambos cenário, considerando distintos tamanhos de vizinhança. Nestes gráficos, a curva denominada *'Simple'*, refere-se ao SCR tradicional, tal como definido na seção 3. *'RulesP'* refere-se ao nosso algoritmo de regras semânticas considerando apenas itens classificados como positivo nas regras e *'RulesN'* considera apenas os itens negativos. A curva *'RulesPN'* refere-se ao cenário heterogêneo de análise. Além disso, as curvas contendo rótulos *'conf'* e *'Pond'* referem-se às versões do nosso algoritmo que consideram a confiança da regra e a valorização das qualificações negativas, respectivamente. De maneira geral, todas as variações de algoritmo avaliadas se mostraram superiores ao algoritmo de SCR tradicional. Para o cenário de regras homogêneas, os melhores resultados foram alcançados considerando itens negativos, ponderação de confiança e valorização das regras negativas, atingindo 5,62% de ganho sobre o algoritmo tradicional. Para o cenário heterogêneo, os ganhos foram ainda maiores, atingindo 8,43% para o algoritmo que utiliza a ponderação de confiança e valorização das regras negativas. Deve ser ressaltado que, utilizando apenas 1/4 do tamanho de vizinhança máximo avaliado nos experimentos dos gráficos 5, atingimos valores significativamente melhores. Isso representa um grande impacto sobre o custo computacional para a recomendação. Como a obtenção do conjunto de regras de associação é realizada uma única vez, e o custo de incorporação de tais regras é linear com o número de itens a serem recomendados, temos uma grande redução neste custo computacional por analisar uma vizinhança significativamente menor para cada usuário.



(a) Regras Homogêneas (b) Regras Heterogêneas

Figura 6. Qualidade das Recomendações Usando Regras Semânticas



(a) Ocorrência de Vizinhos (b) Supervalorização das Qualificações

Figura 7. Caracterização dos Resultados

Discussão

De forma a entender os resultados alcançados, caracterizamos dois aspectos acerca das predições realizadas. O primeiro refere-se à porcentagem de vizinhos de cada usuário u que qualificaram no passado os itens a serem recomendados a u. O gráfico da figura 7 (a) apresenta as probabilidades de ocorrência de cada porcentagem de vizinhos similares e dissimilares. Percebemos que a dissimilaridade prioriza porcentagens mais elevadas de vizinhança. A maior parte das vizinhanças dissimilares já qualificaram vários itens a serem recomendados, enquanto entre os vizinhos similares uma porcentagem muito menor já qualificou estes itens. Ou seja, a dissimilaridade é capaz de endereçar o problema de escassez de informação acerca da vizinhança analisada. Um dos motivos das melhorias propiciadas pelo uso das regras decorre desse fato. Uma evidência disso é que os maiores ganhos em qualidade do algoritmo proposto ocorrem para tamanhos menores de vizinhança, situações em que o problema de escassez de informação é mais acentuado.

O segundo aspecto avaliado é a distribuição dos valores de predição. Distribuições tanto para o nosso algoritmo quanto para o algoritmo tradicional evidenciaram o que de-

nominamos problema de supervalorização das qualificações. Ou seja, as predições tendem a gerar valores mais elevados que os valores reais, tal como mostrado no gráfico da figura 7 (b). Entretanto, o uso de regras, sobretudo com valorização das qualificações negativas, atenua este problema. De forma a mostrar isso, geramos o *Third Standardized Moment Skewness* para ambas curvas. A distribuição para o algoritmo tradicional apresentou um *skewness* de $-0,80$, enquanto para nosso algoritmo este valor foi de $-0,82$. Isso mostra que nossa distribuição é mais desbalanceada para valores menores, evidenciando o fato de gerarmos predições ligeiramente menos positivas. Dessa forma, uma segunda explicação para as melhorias alcançadas vem da atenuação do problema de supervalorização. É importante ressaltar que os resultados obtidos mostram o potencial da consideração dos diversos níveis de similaridade associados com uma visão global no processo de recomendação. A relevância de nossa proposta reside não só em sua aplicação para prover melhorias sobre técnicas tradicionais, mas também para reduzir o custo de análise destas técnicas, exigindo um tamanho muito menor de vizinhança.

7. Conclusão e Trabalhos Futuros

Neste trabalho, apresentamos uma nova proposta para modelagem de dados em Sistemas Colaborativos de Recomendação (SCRs), bem como uma avaliação do uso de dissimilaridade no cenário de recomendação de vídeo, além de uma nova técnica que explora a dissimilaridade em SCRs. A partir da modelagem proposta, foi possível verificar a existência de diversos níveis de similaridade em nossa base de dados, além de executar predições considerando, para um dado usuário, vizinhos similares e regras semânticas que definem aspectos globais de dissimilaridade no domínio. Uma análise comparativa de qualidade das predições geradas mostrou que a técnica proposta é particularmente relevante não só por prover melhorias significativas sobre técnicas tradicionais de SCRs, mas por reduzir o custo de análise destas técnicas. Como trabalho futuro, destacamos o emprego da técnica proposta em outros cenários, com escassez de informação mais acentuada.

Referências

Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE TKDE*, 17(6):734–749.

Breese, J., Heckerman, D., and Kadie, C. (1992). Empirical analysis of predictive algorithms for collaborative filtering. *Learning*, 9:309–347.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370.

Han, J., Pei, J., and Yin, Y. (2000). Mining frequent patterns without candidate generation. *ACM SIGMOD Record*, 29(2):1–12.

Hu, Y., Koren, Y., and Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. In *8th IEEE ICDM*, pages 263–272.

Kagie, M., Van Wezel, M., and Groenen, P. An Empirical Comparison of Dissimilarity Measures for Recommender Systems.

Koren, Y. (2009). Collaborative filtering with temporal dynamics. In *Proc. of the 15th SIGKDD*, pages 447–456. ACM New York, NY, USA.

Park, Y. and Tuzhilin, A. (2008). The long tail of recommender systems and how to leverage it. In *Proc. of the 2008 ACM RecSys*, pages 11–18. ACM.

Previsão de Carga para Configuração Dinâmica de Algomerados de Servidores *Web*

Giulio D. Bottari[1], Julius C. B. Leite[1]

[1]Instituto de Computação
Universidade Federal Fluminense – Niterói, RJ

{gbottari,julius}@ic.uff.br

Abstract. *Nowadays, there is a growing awareness about green computing in computer systems. A major source of energy demand in these systems are web server clusters. This paper evaluates a policy to anticipate the load and configure the number of active servers so that a satisfactory quality of service is met, while reducing energy usage. Experiments are performed on a cluster prototype to validate the employed technique.*

Resumo. *Atualmente, há uma crescente conscientização sobre computação verde em sistemas computacionais. Uma das grandes fontes de demanda energética nesses sistemas são os aglomerados de servidores web. Este trabalho avalia uma política para antecipar a carga e configurar o número de servidores ativos de forma que seja mantido uma qualidade de serviço satisfatória, enquanto se reduz o uso de energia. Experimentos são feitos em um protótipo de cluster para validar a técnica empregada.*

1. Introdução

Redes sociais, motores de busca e aplicações de *e-commerce* caracterizam uma grande parte do que se tornou hoje a Internet. Muitas dessas aplicações demandam um processamento de dados contínuo, realizado em grandes aglomerados (*clusters*) de servidores.

Ao longo dos últimos anos, uma grande preocupação vem se formando em torno da demanda energética desses aglomerados. O tema tem um ponto de vista econômico e um ângulo ambiental, devido principalmente ao custeio da energia usada nesses centros e à emissão de gases causadores do efeito estufa, respectivamente.

A demanda de trabalho num aglomerado de servidores *web* varia ao longo do dia, em função da utilização dos usuários. Por esse motivo, *designers* de *datacenters* o dimensionam para condições de tráfego intenso, como medida de segurança. Isso significa que a maior parte do tempo o aglomerado se encontra subutilizado, sendo possível desligar servidores inativos e religá-los no futuro quando houver necessidade de mais performance, economizando energia. Tal mecanismo é conhecido na literatura como *Configuração Dinâmica de Servidores* (que será identificado por *CDS*, daqui por diante).

Trabalhos como [Pinheiro et al. 2001] e [Rusu et al. 2006] utilizam uma proposta de CDS complementada por variação dinâmica de frequência do processador. Ao invés de se tratar o tema de economia de energia em servidores num espectro

mais amplo, aqui será abordado mais minuciosamente a previsão e antecipação de carga para CDS.

Trabalhos mais recentes como [Santana et al. 2008] e [Santana 2010] utilizam um esquema de CDS auxiliado por previsões, obtendo melhorias no atendimento da carga. Estes trabalhos também utilizam um fator de superestimação de carga para antecipar a reconfiguração do sistema. A proposta deste trabalho é separar estas abordagens em dois métodos: Preditivo e Reativo, explicados em detalhes mais adiante, e comparar as contribuições de cada um.

Na Seção 2 será caracterizada a abordagem escolhida para CDS. Posteriormente será apresentado o previsor utilizado, bem como a calibragem de seus parâmetros na Seção 3. As Seções 4 e 5 mostrarão os *traces* e o *cluster* empregados nos testes. Em seguida, serão abordados a metodologia e resultados dos testes realizados na Seção 6. Finalmente, serão apresentados a conclusão e trabalhos futuros, na Seção 7.

2. Configuração Dinâmica

A CDS tem como objetivo diminuir o gasto energético do aglomerado, ao minimizar o conjunto de servidores trabalhadores ativos. Porém, esse novo conjunto deve ter a capacidade de atender ao mesmo volume de requisições que o anterior.

Pensando nisso, foi desenvolvida uma política conservadora com relação ao chaveamento de servidores, ao mesmo tempo em que busca otimizar o gasto energético do *cluster*. O seu funcionamento está resumido no Algoritmo 1, que deverá ser executado em *loop* com período configurável (nos experimentos foi usado 20 segundos).

Os servidores disponíveis são colocados em uma lista, que deve ser ordenada por algum critério (e.g. eficiência energética). Um critério simples, adotado nesse trabalho, é ordenar as máquinas segundo a sua vazão máxima (ver Tabela 2). Ligando primeiro os servidores mais robustos evita-se que outros menos poderosos, que geralmente oferecem uma baixa taxa de performance por Watt, sejam escolhidos.

Uma convenção adotada foi que as máquinas do início da lista têm prioridade para serem ligadas, enquanto que as do fim têm prioridade de desligamento. Sempre que houver a necessidade de mais performance no *cluster*, um servidor da lista é ligado. Caso contrário, tenta-se diminuir o número dos servidores ativos, mas somente se isso não tornar o aglomerado subdimensionado para a carga. A lista é fixa e igual para todos os experimentos.

O chaveamento de servidores é feito em função da carga (*load*), que é a média simples do intervalo passado. Além disso, pode-se utilizar o parâmetro $\gamma \in [0, 1]$ (linhas 1, 6 e 17 do Algoritmo 1) para *antecipar* a reconfiguração do aglomerado. A utilização desse recurso será chamado de método Reativo.

A necessidade de se antecipar reconfigurações em sistemas de CDS surge ao se considerar o tempo de ligamento destes servidores, operação que leva de 6 à 10 segundos. Quando não se consegue ativar um servidor a tempo de tratar um aumento de carga, tende-se a saturar o sistema.

Algoritmo 1 Atuação da Configuração Dinâmica

```
1: if load > throughput(currConfig) · γ then
2:    {Obter o servidor mais eficiente que, uma vez ligado, será capaz de suprir a carga.}
3:    candidates ← (serverPoll − currentConfig)
4:    for server in candidates do
5:       newConfig ← (currentConfig ∪ server)
6:       if throughput(newConfig) ≥ load · γ then
7:          return newConfig
8:       end if
9:    end for
10:   {Nenhum servidor é capaz de suprir load, então o mais robusto será ativado.}
11:   return currentConfig ∪ mostRobustServer(candidates)
12: else
13:   {Tenta-se reduzir o número de máquinas que estão ligadas.}
14:   candidates ← reverse(currentConfig)
15:   for server in candidates do
16:      newConfig ← (currentConfig − server)
17:      if throughput(newConfig) · γ ≥ load then
18:         return newConfig
19:      end if
20:   end for
21:   return currentConfig
22: end if
```

Como pode-se observar, o Algoritmo 1 irá ligar ou desligar no máximo uma máquina de cada vez. Pretende-se evitar que máquinas sejam ligadas quando há grandes variações de carga que se mantêm por um curto período. Nesse caso, quando o nível de requisições aumenta abruptamente, o *overhead* energético do ligamento de um servidor não compensa o benefício de ligá-lo. Um efeito análogo também ocorre quando a carga decresce bruscamente, neste caso em relação a possibilidade de perda de QoS.

3. Previsor Holt

O Método Linear de Holt, utiliza dois componentes s_t e b_t, que representam o valor suavizado da carga e a estimativa da tendência para o instante t, respectivamente. Por ser capaz de detectar tendências nas séries temporais, julga-se que este previsor é adequado para abordar o problema.

$$\begin{aligned} s_1 &= load_0 \\ b_1 &= load_1 - load_0 \\ s_t &= \alpha \cdot load_t + (1-\alpha) \cdot (s_{t-1} + b_{t-1}) \\ b_t &= \beta \cdot (s_t - s_{t-1}) + (1-\beta) \cdot b_{t-1} \\ F_{t+m} &= s_t + m \cdot b_t \end{aligned} \quad (1)$$

O sistema (1) mostra o conjunto de equações deste modelo. O Método de Holt possui dois parâmetros que controlam a suavização: α e $\beta \in [0,1]$. A variável F_{t+m} dá a previsão para m instantes no futuro, em função da carga atual.

O Algoritmo 1 considera a carga atual para realizar a configuração do sistema, como no método Reativo. No caso do método Preditivo, esse valor serve de entrada

para o método de Holt, retornando o valor da carga prevista ($load_{pred}$). Porém, não é vantajoso utilizar o valor de $load_{pred}$ para configurar diretamente o aglomerado. Caso a previsão aponte para uma queda do volume de requisições, não se deve correr o risco de afetar a QoS desligando servidores no momento atual, sendo que a demanda ainda não diminuiu. Sendo assim, o *cluster* é então configurado utilizando a fórmula $max(load_{pred}, load_{curr})$.

3.1. Otimização de parâmetros

Os valores dos parâmetros do previsor Holt interferem drásticamente no resultado das previsões. Neste trabalho não serão abordados métodos de variação dinâmica para otimizar esses parâmetros. Ao invés disso, serão calculados valores fixos utilizados por toda a duração de um experimento.

Uma técnica apropriada para calibrar estes parâmetros (descrita em [Spyros Makridakis e Hyndman 2002]) é a minimização de algum erro entre a previsão e o dado observado, como MSE (*Mean Square Error*) ou MAE (*Mean Absolute Error*). Além desses critérios, foi elaborado um terceiro chamado *Híbrido*. Ele tem o intuito de produzir parâmetros que evitem a subestimação do previsor em relação à carga. Combinando os outros dois critérios, procura-se penalizar mais severamente uma configuração insuficiente do que uma superdimensionada. Um resumo dos critérios utilizados pode ser visto na Tabela 1. Nela, A_t denomina o valor observado, enquanto que F_t indica o valor previsto no instante t.

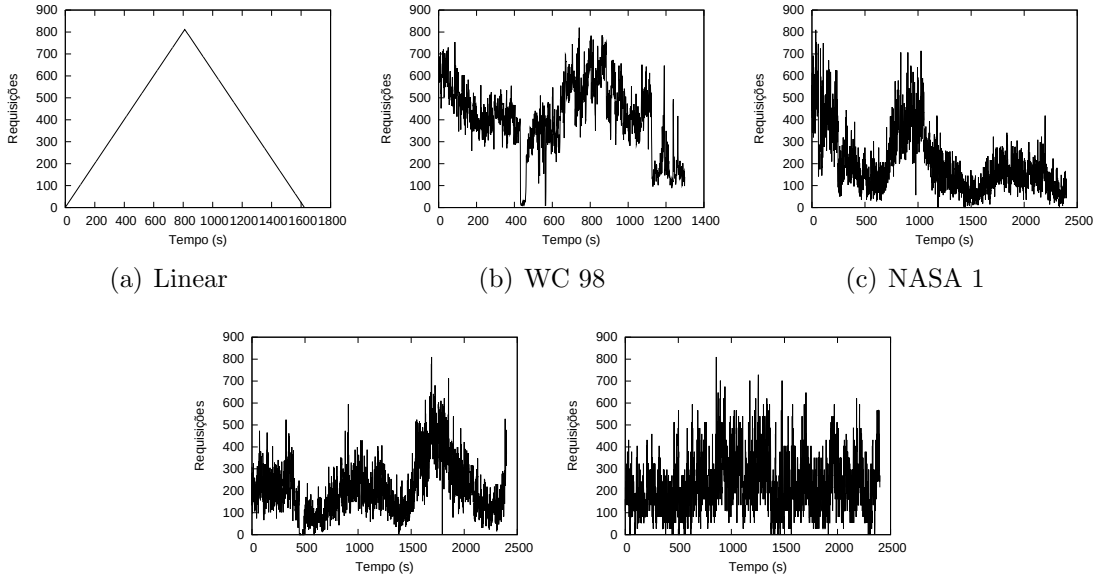
Para realizar a minimização dos erros, foi utilizado um método direto. Gerou-se todas as combinações possíveis de parâmetros α e β no intervalo [0,1] com um passo de 10^{-2}. Para cada combinação, calculou-se o erro entre a estimativa e o resultado real para todo o *trace*. Ao final, foram escolhidos os parâmetros que resultam no menor erro em questão.

Tabela 1. Critérios de minimização

Erro	Fórmula
MSE	$\frac{1}{n} \cdot \sum_{t=1}^{n} (A_t - F_t)^2$
MAE	$\frac{1}{n} \cdot \sum_{t=1}^{n} \lvert A_t - F_t \rvert$
Híbrido	$\frac{1}{n} \cdot \sum_{t=1}^{n} \begin{cases} (A_t - F_t)^2 & \text{se } A_t < F_t \\ \lvert A_t - F_t \rvert & \text{se } A_t \geq F_t \end{cases}$

4. *Traces*

Para obter um perfil de carga realístico foram utilizados *traces* de pedidos feitos a servidores reais, como mostra a Figura 1. O primeiro deles (*Linear*) foi criado sintéticamente para ser fácil de ser previsto usando o método de Holt. Dessa forma, pode-se ter uma base de comparação sobre o potencial desse previsor. Em seguida, temos um trecho da demanda submetida aos servidores durante a Copa do Mundo de Futebol de 1998 (*WC 98*) e outros três realizados aos servidores da NASA. Os *logs* nos quais esses *traces* foram baseados estão disponíveis em [LBNL 2011]. Para construir os *traces*, foi feita a contagem de requisições que foram atendidas a cada segundo nestes *logs*, e em seguida estes valores são dimensionados para a capacidade máxima do *cluster* de testes.



(a) Linear (b) WC 98 (c) NASA 1

(d) NASA 2 (e) NASA 3

Figura 1. Traces do experimento

5. Ambiente de testes

O ambiente de testes consiste de um aglomerado de servidores trabalhadores (*workers*) heterogêneos rodando o servidor *web* Apache versão 2.2.10, sob o SO Gentoo Linux. Um gerador de carga envia requisições durante o experimento para um *frontend*, que distribui a carga entre os trabalhadores através do módulo Apache `mod_proxy_balancer`. A carga é balanceada de acordo com a vazão de cada servidor (Tabela 2), buscando-se obter uma utilização de *CPU* uniforme entre os servidores. A Figura 2 ilustra essa topologia.

Durante o experimento, cada servidor pode ser chaveado entre dois estados: *ativo* e *suspenso*. No primeiro estado o servidor se encontra pronto para atender requisições, enquanto que no outro a máquina salva seu contexto em memória *RAM* e entra num modo econômico de energia (chamado de *Suspend-to-RAM*), que não possibilita o atendimento de requisições. Para transitar entre os estados *suspenso* e *ativo* é utilizado o recurso *Wake-on-LAN*, disponível na placa de rede.

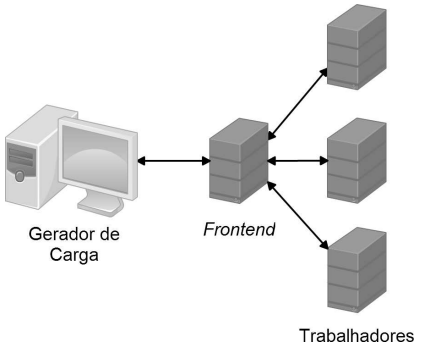


Figura 2. Topologia

O tráfego é gerado nesse sistema através da ferramenta `httperf` [Mosberger e Jin 1998]. A carga consiste de apenas um tipo de requisição, intensiva em *CPU*, implementada através de um *script* PHP que calcula números primos. Além disso, ao longo do experimento as máquinas utilizam o *governor* `performance` do Linux, mantendo o seu *clock* máximo durante todo o tempo. Quando o sistema não se encontra saturado, cada requisição demora cerca de 6 ms para ser atendida.

A Tabela 2 mostra a medida de performance utilizada para os servidores

do experimento na sua frequência de operação máxima: a vazão (*throughput*) de requisições. Além disso, foi medida a potência dos servidores no estado ocioso e ocupado. No estado ocioso nenhuma requisição está sendo atendida, enquanto que no segundo, o processador se encontra cerca de 100% ocupado.

Tabela 2. Características dos servidores utilizados

Nome	Arquitetura (AMD)	Potência ociosa (W)	Potência ocupada (W)	Vazão (req/s)
ampere	Athlon 64 X2 3800+	72,7	109,8	199,6
coulomb	Athlon 64 3800+	75,2	107,7	122,3
hertz	Athlon 64 3800+	71,6	103,2	122,8
joule	Athlon 64 3500+	80,0	110,6	111,4
ohm	Athlon 64 X2 5000+	76,9	140,1	255,2

6. Experimentos

6.1. Simulações

Para comparar a política Reativa com a Preditiva, foi montada uma série de experimentos de simulação com os *traces* da Seção 4. A título de comparação, será utilizado também um previsor "visionário", que conhece exatamente o valor da carga em qualquer instante de tempo. As simulações assumem que o tempo de ligamento ou desligamento de um servidor é instantâneo.

6.1.1. Efeito das previsões

Nesta Seção será utilizado como métrica o número de vezes em que um método superestimou (configurou o *cluster* com mais máquinas do que o necessário para atender a carga), ou subestimou a carga. Em todos os experimentos será usado $\gamma = 1$.

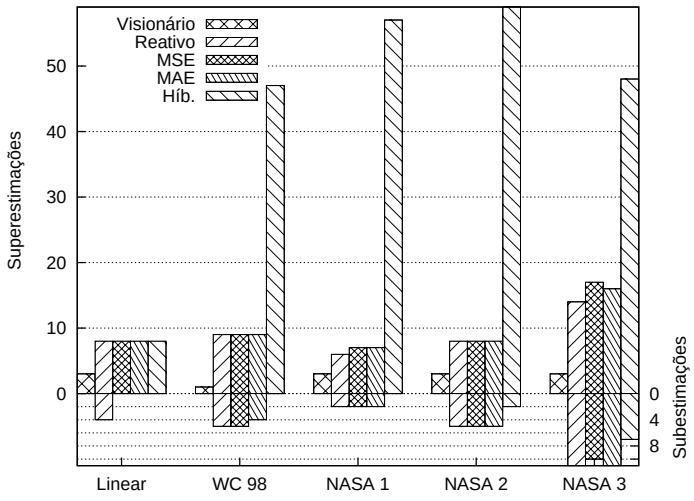


Figura 3. Efeito das previsões

Na Figura 3 é possível observar que o previsor Visionário superestima a carga algumas vezes. Isto ocorre porque o Algoritmo 1 só desliga uma máquina por vez, não importa quão baixa esteja a carga. Os experimentos começam com todos os servidores ligados, fazendo com que esse efeito ocorra apenas no início.

O método Preditivo com critério Híbrido para cálculo de parâmetros apresenta o menor índice de subestimações, mas às custas de um alto número de superestimações em todos os *traces*. Pode-se esperar em experimentos reais no *cluster* que esse comportamento resulte em um melhor nível de QoS, acompanhado de um gasto maior de potência.

Os demais métodos possuem resultados equivalentes, sem conseguir reduzir muito o número de subestimações em comparação com o método Reativo, exceto no caso do *trace* Linear, onde a linearidade do perfil de carga propicia o bom funcionamento do previsor Holt.

6.1.2. Reduzindo subestimações

Na Seção anterior notou-se que apenas o método Preditivo Híbrido é capaz de oferecer alguma melhoria significativa sobre o método Reativo, no caso do subdimensionamento. Por esse motivo, a análise seguirá apenas com esse critério.

Nos experimentos que seguem, foi variado o parâmetro γ a fim de se obter um resultado com zero subestimações com o método Reativo. Na Figura 4, o número que acompanha cada barra é o valor de γ utilizado.

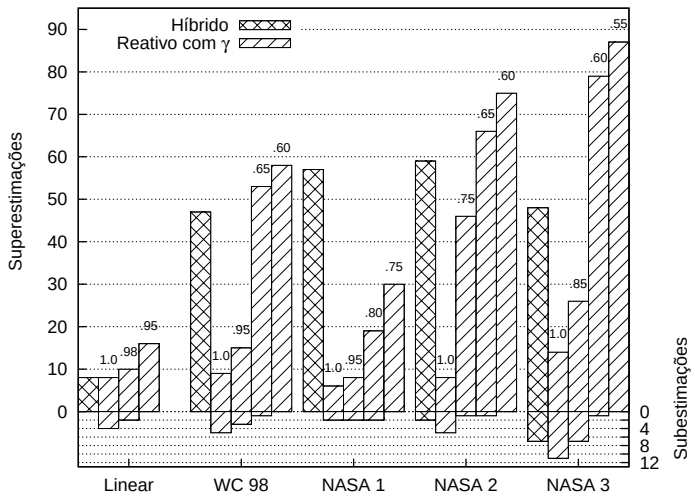
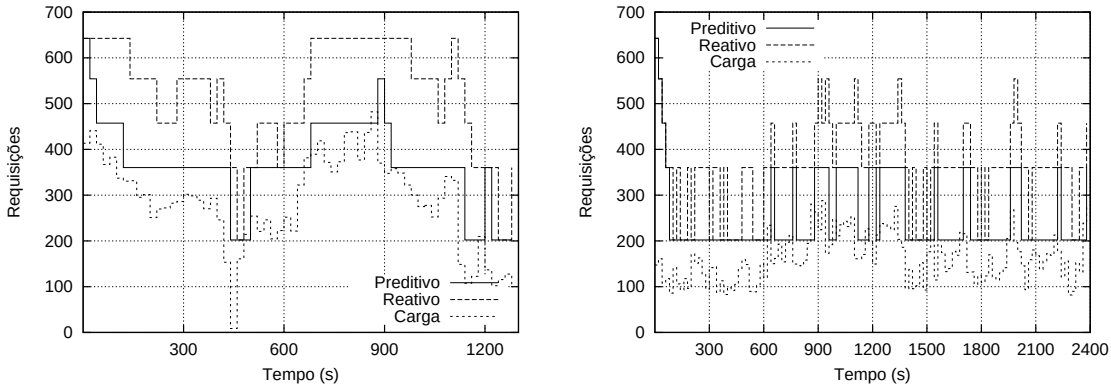


Figura 4. Reduzindo subestimações

Conforme a Figura 4, o método Reativo é capaz de obter zero subestimações em todos os perfis de carga, diferentemente do Preditivo Híbrido. De fato, é possível obter sempre um valor de γ onde o número de subestimações será zero. Dado que o *cluster* está dimensionado corretamente, um valor extremo $\gamma = 0$ significa que todos os servidores permanecerão ligados, sem causar nenhuma subestimação. Além disso, este método consegue menos superestimações para um valor igual ou menor de subestimações nas simulações em todos os casos, fora Linear e WC 98.

A Figura 5 mostra as (re)configurações feitas ao longo de dois *traces*. Nela, as linha de Reativo e Preditivo correspondem a vazão máxima do *cluster* naquele momento. Note que o método Preditivo é mais estável, pois requer menos reconfigurações. Por sua vez, o método Reativo é mais robusto, pois acompanha melhor o perfil da carga.



(a) WC 98 (b) NASA 3

Figura 5. Configurações reativas e preditivas

6.2. Experimentos reais

As simulações cumpriram seu papel de prover uma análise inicial do sistema de CDS desenvolvido e para calibrar uma série de parâmetros, como intervalo de atuação, α, β e γ. Nesta Seção alguns experimentos selecionados da Seção 6.1 serão rodados no ambiente de testes em tempo real.

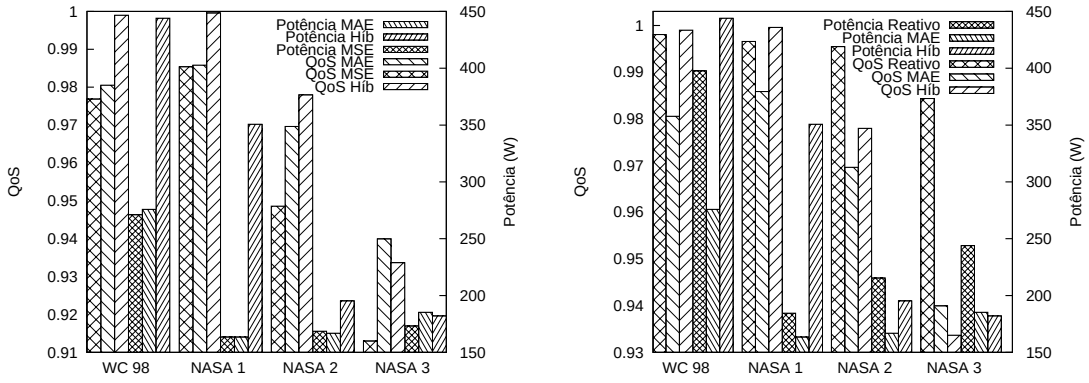
Em vez de medir o número de configurações subestimadas ou superestimadas, serão medidas QoS e potência consumida pelo *cluster*. A QoS instantânea é calculada a cada segundo, e foi definida em função das requisições como $1 - \frac{perdidas}{atendidas}$. O valor para o *deadline* dessas requisições foi ajustado para 20 vezes o valor do seu tempo de resposta médio quando o aglomerado não se encontra saturado (no caso, 120 ms). Essa é uma recomendação do TPC-W ([TPC 2011]) para requisições com *deadline* curto.

Experimentos realizados préviamente em [Li et al. 2009], demonstram que pode-se aproximar por uma relação linear a utilização do processador e a potência gasta por um servidor, quando a carga é intensiva em *CPU*. Por isso, os valores de potência apresentados foram obtidos através das medidas de potência ociosa e ocupada (ver Tabela 2) em função da utilização dos processadores.

6.2.1. Comparação entre parâmetros otimizados

Na Seção 6.1, foi argumentado a favor do método Preditivo Híbrido em relação aos outros métodos preditivos. Resta saber se resultados semelhantes podem ser obtidos em experimentos reais. A Figura 6(a) mostra os resultados de potência e QoS para o método Preditivo com todos os critérios da Tabela 1. Repare que o *trace* Linear não foi mostrado, pois os parâmetros resultantes dos três critérios são os mesmos ($\alpha = \beta = 1$). No eixo vertical esquerdo temos o QoS médio, enquanto que no direito está representada a potência média.

Em alguns casos, os experimentos reais não acompanharam os resultados da Seção 6.1. Por exemplo: os valores de QoS nos *traces* NASA 2 e 3 são diferentes nos critérios MSE e MAE. No entanto, na Figura 3 eles apresentam o mesmo número de



(a) Preditivos (b) Preditivo e Reativo

Figura 6. Experimentos reais

subestimações. Repare que a métrica utilizada era baseada na contagem de subestimações e não no quanto foi subestimado nem por quanto tempo, possibilitando essa diferença. Esse efeito mostra que experimentos reais são muito importantes para validar trabalhos como este.

Os três primeiros *traces* indicados na Figura 6(a) mostram uma melhor QoS com a utilização do método Híbrido. Porém, tal melhoria vem acompanhada de um maior gasto energético, como esperado. Nos experimentos com o NASA 3, o critério MAE apresenta vantagem em QoS e potência consumida em relação aos outros.

6.2.2. Comparação entre Reativo e Preditivo

Tendo em vista que os critérios MAE e Híbrido tiveram melhor desempenho na Seção anterior, serão comparados agora com o método Reativo. Em termos de QoS, o método Reativo é superior ou fica próximo do Preditivo Híbrido em todos os casos representados na Figura 6(b). No segundo caso, a potência consumida pelo método Preditivo Híbrido não justifica a sua pequena vantagem em QoS.

O método Preditivo MAE se coloca numa posição interessante nos três primeiros *traces*, oferecendo um bom compromisso entre potência e QoS. Em compensação, quando uma garantia de QoS mais elevada for uma característica mais valiosa em aplicações *web*, o método Reativo é mais vantajoso pela sua flexibilidade.

7. Conclusão e trabalhos futuros

Este trabalho analisou uma política simples e eficaz para CDS. Foi comparado o uso de duas abordagens que visam manter o aglomerado dessaturado: a reativa e a preditiva. Foi apresentado o previsor Holt e a otimização de seus parâmetros através da minimização de três critérios: MSE, MAE e Híbrido. Através dos experimentos conduzidos, pode-se estudar o impacto de políticas simples em QoS e potência consumida.

Em face dos resultados obtidos, pode-se concluir que a abordagem mais recomendada é a Reativa, com uso do parâmetro γ. Em testes adicionais (não mostrados

aqui), foi observado que abordagem preditiva com o uso de γ é equivalente ou inferior à Reativa. O uso deste parâmetro permite um controle mais simples e intuitivo sobre a superestimação de carga e uma flexibilidade superior ao prover níveis de QoS ajustáveis. Em contrapartida, o uso da política Preditiva por si só não é capaz de realizar o mesmo. Por exemplo, na Figura 6(a), não é possível obter um QoS de 95% em NASA 3, caso isso seja um requerimento da aplicação.

O passo seguinte desta pesquisa será o controle de QoS através de um sistema adaptativo. Neste sistema os parâmetros α e β seriam minimizados periódicamente e γ seria manipulado para se ajustar ao alvo de QoS do fixado. Espera-se obter um controle simples que possa antecipar a carga de maneira eficaz. Além disso, futuramente será importante incorporar políticas de variação de frequência para otimizar o gasto energético e obter um controle fino de QoS.

Em trabalhos futuros serão buscados outros previsores como o Método de Holt-Winters, capaz de detectar sazonalidade em séries temporais. Esta é uma característica presente em alguns *traces* que não foi explorada neste trabalho. Neste trabalho foram realizadas previsões em um período curto de 20 segundos. É necessário analisar períodos mais longos, onde previsões de longo prazo podem ser mais vantajosas que o esquema reativo.

Referências

LBNL (2011). The Internet traffic archive. `http://ita.ee.lbl.gov/index.html`.

Li, L., RuiXiong, T., Bo, Y. e ZhiGuo, G. (2009). A model of web server's performance-power relationship. In *International Conference on Communication Software and Networks*, pp. 260–264, Macau, China. IEEE Computer Society.

Mosberger, D. e Jin, T. (1998). httperf: A tool for measuring web server performance. In *Internet Server Performance Workshop*, pp. 59–67, Madison, WI, EUA.

Pinheiro, E., Bianchini, R., Carrera, E. V. e Heath, T. (2001). Load balancing and unbalancing for power and performance in cluster-based systems. Relatório Técnico DCS-TR-440, Department of Computer Science, Rutgers University, EUA.

Rusu, C., Ferreira, A., Scordino, C. e Watson, A. (2006). Energy-efficient real-time heterogeneous server clusters. In *IEEE Real-Time and Embedded Technology and Applications Symposium*, pp. 418–428. San Jose, CA, EUA.

Santana, C. (2010). Previsão de carga em aglomerado de servidores web aplicada a economia de energia. Dissertação de Mestrado, Universidade Federal Fluminense, Niterói, RJ, Brasil.

Santana, C., Bertini, L., Leite, J. C. B. e Mossé, D. (2008). Applying forecasting to interval based DVS. In *Brazilian Workshop on Real-Time Systems*, pp. 13–20, Rio de Janeiro, RJ, Brasil.

Spyros Makridakis, S. C. W. e Hyndman, R. J. (2002). *Forecasting: Methods and Applications*. Kluwer Academic Publishers.

TPC (2011). Transaction Processing Performance Council. `http://www.tpc.org/`.

Um Mecanismo de Consulta Temporal por Palavras-Chave em Páginas Web

Aline Frozi[1], Edimar Manica[1,3], Renata Galante[1], Carina F. Dorneles[2]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

[2]Depto. de Informática e Estatística – Universidade Federal de Santa Catarina (UFSC)
Campus Universitário Trindade – 88.049-900 – Florianópolis – SC – Brasil

[3]Instituto Federal do Rio Grande do Sul (IFRS) – Campus Avançado Ibirubá
Rua Nelsi Ribas Fritsch, nº 1111, Bairro Esperança – 98.200-000 – Ibirubá – RS – Brasil

{arfrozi, emanica, galante}@inf.ufrgs.br, dorneles@inf.ufsc.br

Abstract. *This paper presents a mechanism for querying text using keywords and time constraints. Throughout this document we show the system architecture and a prototype that works on Web pages: the Chronos search engine. Using this search engine, documents with time information related to the query terms receive higher scores, which is the main difference of the tool in comparison to the current search engines.*

Resumo. *Este artigo apresenta um mecanismo que permite a realização de consultas em textos utilizando palavras-chave e restrições de tempo. Ao longo deste documento é apresentada a arquitetura do sistema e um protótipo desenvolvido que atua sobre páginas Web: o motor de busca Chronos. Com ele, são valorizadas páginas que possuem a informação temporal dentro do mesmo contexto dos termos pesquisados – sendo este o principal diferencial da ferramenta em relação aos mecanismos de busca atuais.*

1. Introdução

Uma página Web possui diversas informações de tempo relacionadas a ela: a data de sua publicação, a data em que foi coletada pelo motor de busca e as datas e outros termos indicando tempo que aparecem no seu conteúdo textual. Motores de busca comerciais têm explorado o valor da informação temporal na busca Web. Eles fornecem maneiras do usuário realizar uma busca na Web baseada no tempo. Por exemplo, o Google permite definir o período de interesse, bem como a opção de visualizar os resultados da busca em uma linha de tempo. No entanto, estes motores de busca apenas consideram a data de coleta das páginas [Jin 2008].

Dessa forma, quando a página contém informações relacionadas ao passado ou ao futuro – por exemplo, uma página que foi publicada em 2011 falando sobre a Copa do Mundo de Futebol de 2002 – o tempo associado à pagina pelo motor de busca difere do tempo abordado no conteúdo da página, o que torna imprecisa a busca baseada no tempo oferecida pelos motores de busca atuais. Há trabalhos que propõem motores de busca temporal, como [Jin 2008], [Vicente-Díez e Martínez 2009] e [Pasca 2008].

Porém, não são adequados para páginas que descrevem várias informações temporais, conforme é discutido na Seção 5.

Para contornar esse problema, foi desenvolvida a ferramenta Chronos. Esta, ao indexar as páginas, mapeia cada fragmento de texto ao intervalo de tempo a que se refere. Por exemplo, o trecho *em 1958, o Brasil sagrava-se pela primeira vez Campeão Mundial de Futebol* é associado ao intervalo de tempo de *01/01/1958* a *31/12/1958*. Dessa forma, o mecanismo de busca pode dar mais valor a páginas que têm a informação temporal desejada próxima aos termos que o usuário submeteu para pesquisa.

Este trabalho foi desenvolvido como uma extensão do trabalho [Manica 2010], e compôs o trabalho[1] de conclusão de curso [Frozi 2010a]. Manica et. al [Manica 2010] propõem uma classificação denominada TKC (Temporal Keyword Classification) que especifica os requisitos temporais que um mecanismo de consulta temporal por palavras-chave deve considerar. Com base nessa classificação, é proposta uma abordagem denominada TPI que suporta consultas temporais por palavras-chave em motores de busca XML, tratando os requisitos temporais definidos em TKC.

O principal diferencial do trabalho [Frozi 2010a] é tratar os requisitos temporais em motores de busca Web. Dois aspectos diferenciam os motores de busca Web dos motores de busca XML: a estrutura dos documentos consultados (não estruturados); e a existência de um mecanismo de *ranking*. A principal contribuição deste trabalho é, portanto, definir como incluir nas medidas de *ranking* a dimensão temporal. A ferramenta Chronos foi apresentada na Sessão de Demos do Simpósio Brasileiro de Banco de Dados de 2010 [Frozi 2010b]. Demais contribuições deste trabalho são: (i) o suporte a um conjunto de consultas temporais; (ii) o tratamento da informação temporal presente no conteúdo das páginas Web, identificando quais termos se referem a cada expressão temporal – permitindo tratar páginas que contêm diversas informações temporais; e (iii) o método de indexação temporal.

Chronos foi desenvolvido em Java e fez uso da biblioteca Lucene [Lucene 2010], que disponibiliza métodos para indexação e consulta de textos. Inicialmente, foi construído um motor de busca convencional. Após, foi desenvolvida uma extensão, de modo a adicionar a funcionalidade de busca temporal. A existência dos dois sistemas tornou possível a comparação da qualidade dos resultados obtidos pelos dois tipos de motores de busca.

O artigo está organizado como segue. Na Seção 2 é descrita a arquitetura do mecanismo de busca. A Seção 3 apresenta a ferramenta Chronos. Na Seção 4 são discutidos os experimentos realizados e resultados obtidos. Os trabalhos relacionados são apresentados na Seção 5. Finalmente, na Seção 6, são descritas as considerações finais e as direções futuras.

2. Arquitetura do Sistema

A ferramenta Chronos foi desenvolvida para suportar consultas temporais por palavras-chave sobre páginas Web. Com ela, o usuário pode restringir os resultados da consulta a um período de tempo de seu interesse.

[1] Esta monografia foi orientada por Renata Galante (UFRGS), co-orientada por Edimar Manica (UFRGS, IFRS) e teve a colaboração de Carina F. Dorneles (UFSC).

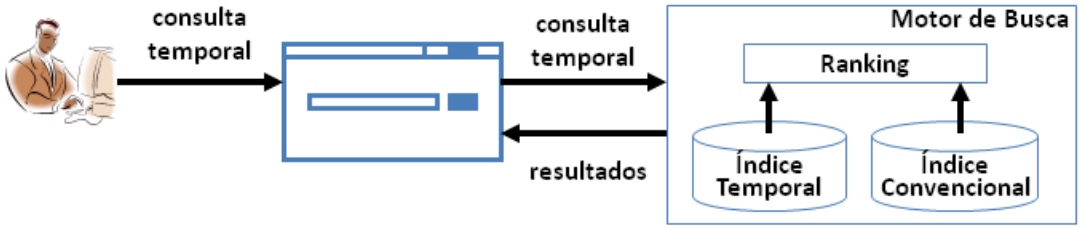


Figura 1. Visão geral da ferramenta Chronos

O funcionamento da ferramenta pode ser dividido em duas fases: a de indexação e a de pesquisa. A Figura 1 apresenta a visão geral da ferramenta. O usuário interage com uma interface fornecendo uma consulta temporal. Esta consulta temporal é constituída de palavras-chave convencionais e de expressões temporais. Por exemplo, seja considerada a consulta *senator Brazil intersect 2005*. Esta consulta é enviada a um motor de busca. O motor de busca, através de seu módulo de *ranking*, processa a consulta, acessa um índice invertido convencional e um índice temporal, e retorna os resultados ordenados de acordo a sua importância (escore), que é obtida por uma média ponderada entre a relevância do texto e a relevância do tempo.

2.1. Especificação das Consultas

As consultas temporais seguem a forma definida no trabalho [Manica 2010]. A sintaxe é mostrada em *Backus–Naur Form* na Figura 2 a seguir. Colchetes [] indicam segmento opcional, chaves {} indicam segmento repetitivo (uma ou mais vezes), parênteses indicam um conjunto de opções, | indica "ou", e palavras em negrito são usadas como termos pré-definidos.

```
tempQuery ::= {kws} tempPredicate
kws ::= string
tempPredicate ::= tempOp op
tempOp ::= ( After | Before | Contains | During | Equals | FinishedBy
           | Finishes | Intersect | Meets | MetBy | OverlappedBy
           | Overlaps | StartedBy | Starts )
op ::= ( instant | interval | compInterval )
instant ::= [ number / ] [ number / ] number
interval ::= [ instant , [ instant ] ] | [ , instant ]
compInterval ::= word [value] granularity
word ::= ( current | last | next )
granularity ::=
   (Day(s) | Week(s) | Month(s) | Year(s) |Decade(s) | Century(ies))
value ::= integer
```

Figura 2. Formato de consulta temporal

Uma consulta temporal (tempQuery) é composta por um conjunto de palavras-chave sem informação temporal (kws) e um predicado temporal (tempPredicate). O

predicado temporal é uma restrição temporal dividida em duas partes: operador temporal (tempOp) e operando (op). O operando é um instante ou um intervalo, que deve seguir o operador temporal. O operador temporal define a relação temporal entre o conjunto de palavras-chave convencional e o operando. É importante notar que o operador temporal cobre todas as relações básicas entre intervalos temporais definidas em [Allen 1983] e, adicionalmente, o operador INTERSECT.

Há três possibilidades para usar o operando: instante (instant), intervalo (interval) ou intervalo composto (compInterval). O instante é um ponto no tempo. Por exemplo, *2008*. O intervalo é definido por um período temporal composto por dois instantes que representam, respectivamente, o instante inicial e final do período. Estes dois instantes devem estar entre colchetes e separados por vírgula. Por exemplo, *[01/01/2008, 30/06/2008]*. Um intervalo composto define um deslocamento a partir da data atual. Por exemplo, *last 10 months* (últimos 10 meses). Alguns exemplos de consultas são: (i) *senator Brazil* ***after*** *2008*; (ii*) senator Brazil* ***intersect*** *[01/01/2002, 30/06/2005]*; e (iii) *senator Brazil* ***before*** *last 4 years*.

2.2. Indexação

A indexação é pré-requisito para que se possam fazer consultas. Deve ser executada de forma antecipada para formar as estruturas (índices) que são posteriormente acessadas pelo motor de busca para definir a importância de cada página para uma dada consulta.

A ferramenta Chronos utiliza dois índices criados a partir das páginas Web coletadas por um *crawler*. Ambos os índices são criados *off-line*, após a coleta dos documentos. De tempos em tempos, após cada coleta, os índices são atualizados. Durante a fase de consultas, estes índices são acessados pelo mecanismo de busca.

O primeiro é um índice invertido convencional construído de acordo com técnicas tradicionais de RI [Baeza-Yates e Ribeiro-Neto 1999]. Este índice é criado através da adição de documentos, armazenando basicamente três informações de cada arquivo: caminho do arquivo, data de modificação e conteúdo.

O segundo é um índice "temporal", que se diferencia do índice convencional principalmente no seguinte aspecto: neste índice, ao invés de serem adicionados documentos, são adicionados *nuggets* temporais. Um *nugget* é um fragmento de uma sentença que informa fatos de domínio aberto associados com alguma entidade [Pasca 2008]. Por exemplo, o fragmento "O Brasil ganhou a Copa do Mundo em 1994" é um *nugget* temporal, pois informa um fato (ganhou a Copa do Mundo) da entidade Brasil. Este fato ainda está associado a sua data de ocorrência (1994). Cada *nugget* temporal é adicionado ao índice temporal, e o intervalo temporal existente no *nugget* é armazenado em um banco de dados. O processo de construção do índice temporal possui seis etapas, conforme é mostrado na Figura 3.

A etapa **A. Processamento Linguístico** rotula as palavras do texto de cada página Web com suas funções morfossintáticas, como verbo e substantivo. Esta etapa é realizada pela ferramenta TreeTagger [TreeTagger 2010]. A etapa **B. Reconhecimento de Expressões Temporais** anota as expressões temporais presentes no texto. A etapa **C. Normalizador** normaliza as expressões temporais para um formato único. As etapas **B** e **C** são executadas pela ferramenta GUTime [GUTime 2010], que detecta e normaliza

expressões temporais em textos por meio de um autômato finito, considerando as funções morfossintáticas das palavras identificadas na etapa **A**.

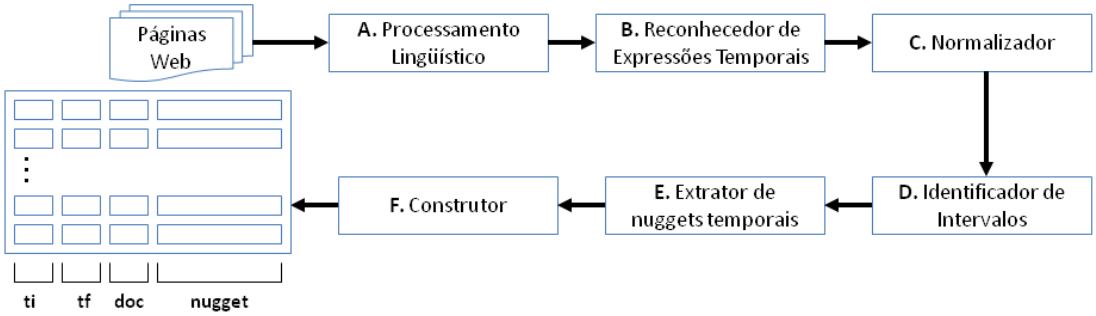


Figura 3. Detalhamento da Indexação Temporal

A etapa **D. Identificador de Intervalos** identifica quando duas expressões temporais representam um intervalo. Esta etapa é realizada a partir de alguns padrões léxico-sintáticos, tais como: *from DATE to DATE* e *between DATE and DATE*. A etapa **E. Extrator de Nuggets Temporais** extrai *nuggets* do texto que estão associados com expressões temporais. Esta etapa é baseada nos padrões léxico-sintáticos definidos em [Pasca 2008] e considera as funções morfossintáticas das palavras identificadas na etapa **A**. Um exemplo de padrão utilizado é *[StartOfSent] Nugget [in|on] Date [EndOfSent]*.

A etapa **F. Construtor** é responsável por criar o índice. Cada *nugget* temporal é considerado como um pseudo-documento que é indexado. Alguns campos adicionais são associados a cada *nugget*: (i) o tempo inicial (ti) do intervalo representado pela expressão temporal associada ao *nugget*; (ii) o tempo final (tf) do intervalo; e (iii) uma referência ao documento que contém o *nugget*. Quando a expressão temporal associada ao *nugget* não constitui um intervalo e sim um ponto no tempo (... *elegeu-se em1994*, por exemplo), atribui-se seu valor tanto para o tempo inicial quanto para o final.

2.3. Ranking das Páginas Web

O módulo de *ranking* das páginas Web recebe como entrada uma consulta temporal e retorna os documentos correspondentes. Estes resultados são classificados de acordo com uma média ponderada entre a relevância do texto e a relevância do tempo.

Dada uma consulta temporal que é submetida no Chronos, são executados dois tipos de busca: convencional e temporal. O objetivo da busca convencional é obter o escore de cada documento que um motor de busca convencional daria para a consulta sem considerá-la temporal. O objetivo da busca temporal é dar mais valor a documentos que possuem frases que contêm, ao mesmo tempo, ambas as partes da consulta: os termos buscados e o intervalo temporal desejado.

O *ranking* convencional é obtido por funções de consulta existentes na biblioteca Lucene, que utiliza uma combinação de modelo Vetorial e modelo Booleano para determinar o escore de um documento.

A forma de obter o *ranking* temporal diferencia-se da anterior por não se basear nos documentos, mas sim nos seus *nuggets* temporais. Para computar o valor de cada documento, primeiro calcula-se o escore segundo o critério de relevância da Lucene para a consulta sem a parte temporal sobre o índice temporal. Com isso, cada *nugget*

temporal indexado recebe um *score*. Em seguida, cada *nugget* temporal cujo intervalo temporal associado não satisfaz o predicado temporal da consulta é eliminado. Finalmente, o escore da página será o escore de seu *nugget* temporal com maior escore.

O escore final de relevância de uma página é computado de acordo com a fórmula

$$rank(P) = \alpha * ksim + \beta * tsim, \quad (1)$$

onde *ksim* representa o escore de relevância do texto, dado pelo *ranking* convencional, e *tsim* representa o escore de relevância do tempo, dado pelo *ranking* temporal. Os valores de α e β representam os pesos das variáveis. A fórmula (1) é baseada na que é apresentada em [Jin 2008], para o motor de busca TISE. As diferenças são: (i) no trabalho de [Jin 2008], não existe o cálculo de tsim, uma vez que não são utilizados os *nuggets* temporais; e (ii) no trabalho de [Jin 2008], é considerada na soma do escore também uma parcela referente à importância da página dada pelo algoritmo PageRank [Page 1998].

3. Apresentação da Ferramenta

A interface da ferramenta Chronos é Web. Assim como a maioria dos motores de busca, a ferramenta Chronos possui uma interface simples para a realização das consultas. A Figura 4 apresenta a página de busca do Chronos, e a Figura 5 apresenta um exemplo de página de resultados. A base de dados usada para teste continha páginas sobre personalidades estadunidenses. Ressalta-se que, como resumo de cada página nos resultados, é apresentado o *nugget* temporal com maior valor de escore; assim, o usuário pode, muitas vezes, encontrar a informação que necessita já nos resumos apresentados na página de resultados.

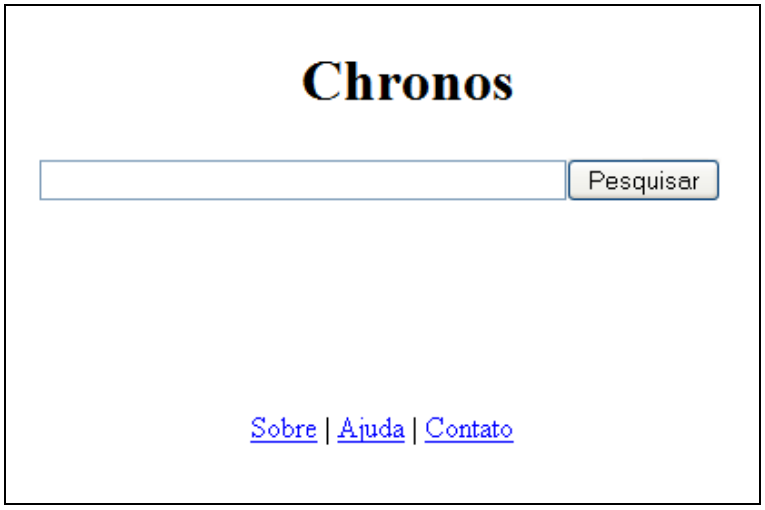


Figura 4. Página de busca

4. Experimentos

O objetivo dos experimentos realizados foi validar a ferramenta Chronos no quesito consultas temporais por palavras-chave. Foi medida a qualidade dos resultados retornados por ela mediante algumas consultas. Seus resultados foram comparados com

os resultados retornados por um motor de busca convencional, ambos desenvolvidos com o apoio da biblioteca Lucene.

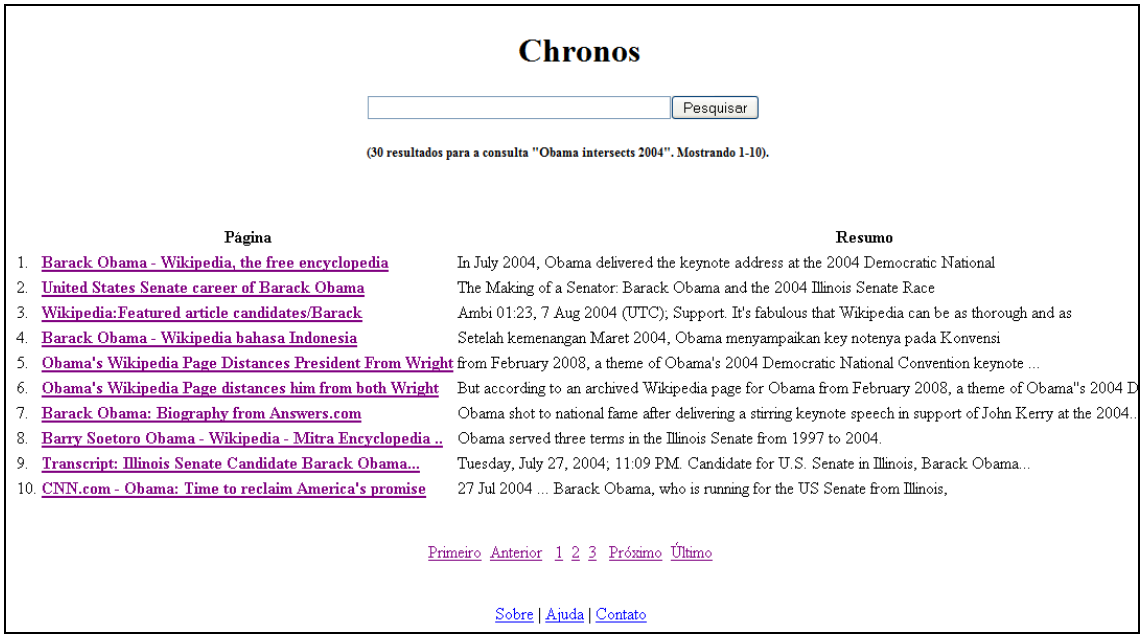


Figura 5. Página de resultados

Para os experimentos realizados, o conjunto de dados selecionado compreendeu quarenta e três páginas HTML coletadas da Wikipedia [Wikipedia 2010] com informações sobre os presidentes dos Estados Unidos. Cada página contém a biografia de um presidente. Previamente, foram construídos os índices convencional e temporal com base nessas páginas. O índice convencional foi criado a partir do conjunto completo, e o índice temporal a partir do conjunto processado.

Foram submetidas 15 consultas temporais aos dois motores de busca, considerando para análise apenas os 10 primeiros resultados. Para encontrar as páginas relevantes foram analisados manualmente os resultados de ambas as ferramentas.

A métrica para avaliação de desempenho utilizada nos experimentos foi *Precision at X*. Essa medida indica a precisão após X documentos recuperados. Foram utilizados X=10, X=5 e X=1.

A Figura 6 mostra o gráfico obtido pelo cálculo da precisão considerando os dez primeiros resultados. É possível notar que não houve um grande aumento na precisão dos resultados do Chronos em comparação com os do mecanismo convencional. Isso se deve ao fato de a medida da precisão considerar apenas a quantidade de documentos relevantes que foram obtidos e não a sua posição na lista de resultados. Os resultados relevantes com Chronos geralmente ficavam mais ao topo da lista, mas isso não é mostrado pela métrica da precisão.

A Figura 7 apresenta o cálculo da precisão para os cinco primeiros resultados. Conforme foi dito, a maioria dos resultados relevantes do Chronos ficou acima dos resultados relevantes do motor de busca convencional. Sendo assim, pode ser visto na Figura 7 que a precisão das consultas 4, 7 e 15 com Lucene diminuíram em relação ao

que foi mostrado na Figura 6, pois elas continham resultados relevantes abaixo da quinta posição do *ranking*.

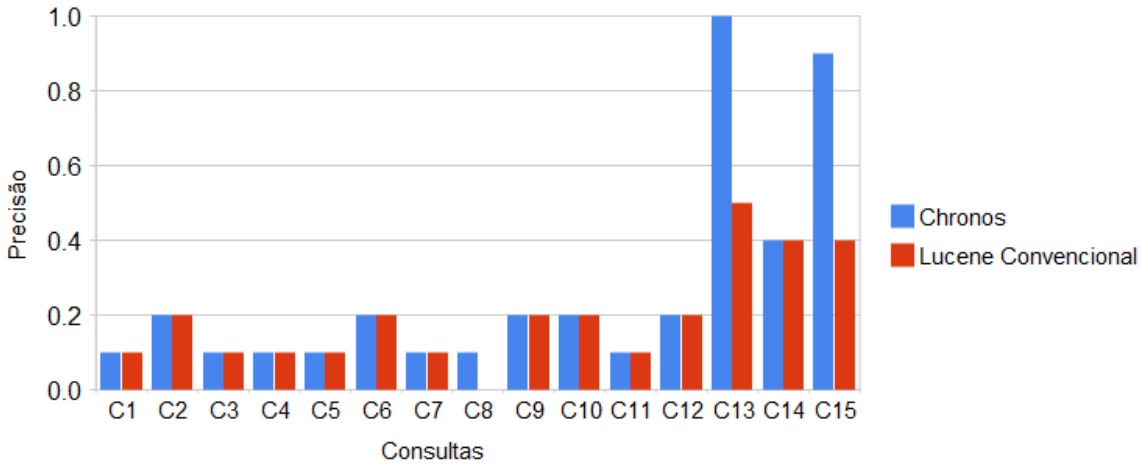


Figura 6. Precision at 10

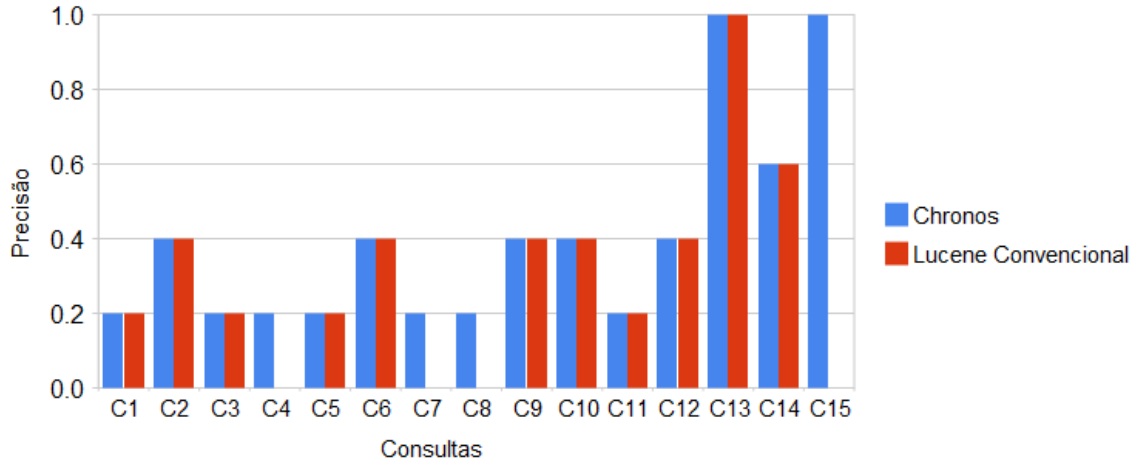


Figura 7. Precision at 5

A seguir é mostrada a Figura 8, que contém a precisão considerando apenas o primeiro resultado de cada ferramenta. Em nove das quinze consultas Chronos retornou um resultado relevante na primeira posição, enquanto que o motor convencional retornou zero. Em cinco houve empate e, em apenas um caso (C5), o mecanismo convencional retornou um relevante contra zero de Chronos.

Pode ser observado pelos resultados que a ferramenta Chronos obteve um aumento significativo na precisão quando são utilizados operadores temporais que não podem ser mapeados em uma consulta convencional (como *after* e *before*). Resultados bons também foram conseguidos quando o intervalo solicitado é maior do que um ano, já que um mecanismo de busca convencional consulta o termo correspondente ao ano que foi informado, enquanto que Chronos consulta todo o intervalo correspondente.

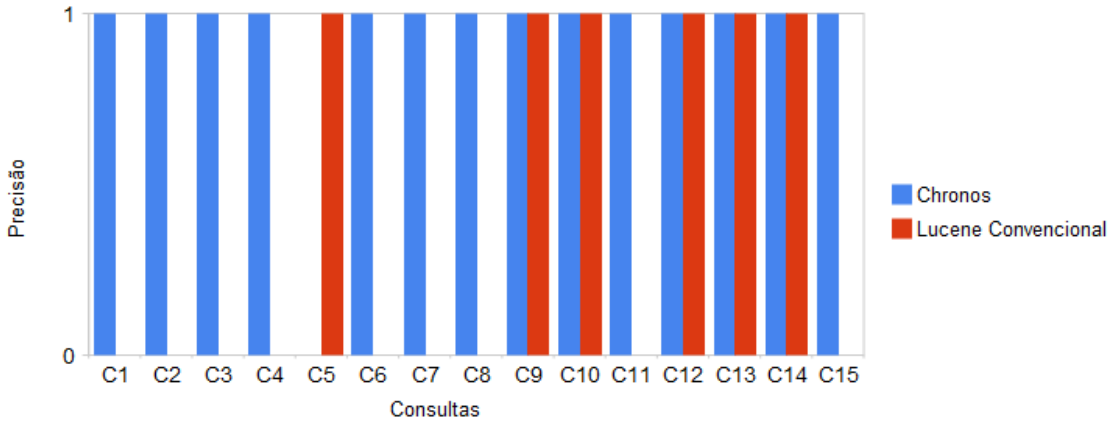


Figura 8. Precision at 1

Também é possível notar que Chronos apresentou os resultados relevantes mais ao topo. Isso ocorreu porque a ferramenta Chronos normaliza as expressões temporais presentes no conteúdo das páginas e associa a cada expressão temporal apenas os termos relacionados com ela.

No entanto, existem alguns casos em que a ferramenta Chronos precisa ser melhorada. Por exemplo, para a consulta *was assassinated intersect 1881*, havia um documento que continha *was mortally wounded in 1881* que não foi recuperado, mesmo contendo a informação desejada. O motivo de não ter sido recuperado é que esta frase não possui ambas as partes temporal e não temporal da consulta, pois, ao invés do termo *assassinated* que havia sido pesquisado, possui um sinônimo. Para resolver esse problema, pode ser utilizada a técnica de expansão de consultas.

5. Trabalhos Relacionados

TISE [Jin 2008], TERN [Vicente-Díez e Martínez 2009] e [Pasca 2008] são propostas de motores de busca temporal. TISE indexa apenas uma expressão temporal por página, selecionando a expressão temporal que descreve mais apropriadamente os eventos da página Web. Isto acarreta em perda de várias informações temporais presentes no conteúdo da página. TERN e [Pasca 2008] indexam todas as expressões temporais do conteúdo textual da página Web. No entanto, TERN associa todas as expressões temporais com a página, enquanto que [Pasca 2008] associa cada expressão temporal com um *nugget* temporal, formando um pseudo-documento. TISE e TERN permitem consultas de seleção temporal, onde o usuário utiliza uma restrição temporal para filtrar os resultados da consulta. Já [Pasca 2008] permite consultas de saída temporal, onde o usuário está interessado em saber qual o tempo em que um determinado evento ocorreu.

O diferencial da ferramenta Chronos é aplicar a indexação de *nuggets* temporais para consultas de seleção temporal, a fim de melhorar os resultados deste tipo de consulta associando a uma expressão temporal apenas termos que se referem a ela.

6. Considerações Finais

Este artigo apresentou a ferramenta Chronos, que permite a realização de consultas temporais por palavras-chave em páginas Web. As principais contribuições deste

trabalho são a identificação dos termos relacionados com cada expressão temporal presente no conteúdo da página Web e a criação de um método de indexação temporal.

Experimentos mostram que Chronos melhora a qualidade dos resultados para consultas temporais em comparação com um motor de busca convencional. Como trabalhos futuros destacam-se: (i) a realização de testes com diversos usuários e inúmeras consultas para validar a ferramenta; (ii) a utilização da estrutura da página Web para melhorar os resultados; e (iii) a utilização de expansão de consultas para tratar sinônimos.

Referências

Allen, J. F. (1983). "Maintaining knowledge about temporal intervals". *Commun. ACM*, 26(11).

ANNIE. (2009). "Open source information extraction", http://www.aktors.org/technologies/annie/. Abril, 2010.

Baeza-Yates, R. A. e Ribeiro-Neto, B. (1999). "Modern Information Retrieval". Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Frozi, A. (2010a). "Um mecanismo de consulta temporal por palavras-chave em páginas Web". Monografia de Graduação, Instituto de Informática, UFRGS.

Frozi, A.; Manica, E.; Galante, R. e Dorneles, C. F. (2010b). "Chronos - um motor de busca temporal". Sessão de Demos, SBBD.

GUTime. "Adding TIMEX3 tags". http://www.timeml.org/site/tarsqi/modules/gutime/index.html. Abril, 2010.

Jin, P.; Lian, J.; Zhao, X. e Wan, S. (2008). "Tise: A temporal search engine for web contents". In *IITA '08*, pages 220–224, Washington, USA. IEEE Computer Society.

Lucene. http://lucene.apache.org. Abril, 2010.

Manica, E.; Dorneles, C. F. e Galante, R. (2010). "Supporting temporal queries on xml keyword search engines". In *SBBD '10*, Belo Horizonte, MG, Brasil. SBC.

Page, L. et al. (1998). "The pagerank citation ranking: Bringing order to the web". Technical report, Stanford Digital Library Technologies Project.

Pasca, M. (2008). "Towards temporal web search". In *SAC*, pages 1117–1121. ACM.

TimeML. "Markup Language for Temporal and Event Expressions". http://www.timeml.org. Abril, 2010.

TreeTagger. "A Language Independent Part-of-Speech Tagger". http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/. Abril, 2010.

Vicente-Díez, M. T. e Martínez, P. (2009). "Temporal semantics extraction for improving web search". In *DEXA Workshops*, pages 69–73. IEEE Computer Society.

Wikipedia. "Wikipedia, the free encyclopedia". http://en.wikipedia.org. Abril, 2010.

Uso de Regressão Logística para Modelar e Avaliar a Credibilidade em Aplicações Web

Rafael Lima [1], **Adriano Pereira** [1]

[1]Departamento de Engenharia de Computação (DECOM)
Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG)
Belo Horizonte - MG, Brasil

rafaellima@decom.cefetmg.br, adriano@decom.cefetmg.br

Abstract. *The popularization of Web applications has given rise to new services every day, demanding mechanisms to ensure the credibility of these online services. In this work we use logistic regression to design and evaluate the credibility of a Web application. In order to validate our proposed methodology, we perform experiments using an actual dataset, from which we evaluated different credibility models using distinct types of information sources. The obtained results are very good, showing representative gains, when compared to a baseline. The results show that the proposed methodology are promising and can be used to enforce trust to users of services on the Web.*

Resumo. *A popularização das aplicações Web tem feito surgir novos serviços a cada dia, demandado mecanismos que assegurem a credibilidade desses serviços. Neste trabalho utilizamos regressão logística para modelar e avaliar a credibilidade de um serviço da Web, considerando diferentes critérios associados ao serviço e seus fornecedores. A fim de validar nossa metodologia, executamos experimentos usando uma base de dados real, a partir da qual avaliamos esses modelos de credibilidade. Os resultados obtidos são muito bons, apresentando ganhos representativos, quando comparados à linha-de-base, mostrando assim que a metodologia proposta é promissora e pode ser usada para fortalecer a confiança dos usuários nos serviços providos na Web.*

1. Introdução

A popularização das aplicações da Web 2.0, onde usuários podem interagir cada vez mais, criando e compartilhando uma diversidade de conteúdos, negociando produtos e estabelecendo novas comunidades, representou a maior revolução da forma como usuários e corporações utilizam a Web. Essa revolução trouxe desafios relacionados à credibilidade dos serviços e informações disponíveis na Web. Assim, nesse cenário, mecanismos que auxiliem os usuários a avaliar a credibilidade tem se tornado cada vez mais essencial.

Bibliotecas digitais e mercados eletrônicos são exemplos de aplicações que requerem mecanismos para avaliar a credibilidade. Muitas dessas aplicações já provêem sistemas para lidar com isso, tais como sistemas de reputação.

Modelos de credibilidade diferem dos modelos de reputação, os quais são bastante estudados na literatura [Jøsang et al. 2007, Sabater and Sierra 2005], porque eles não consideram apenas o *feedback* dos usuários, mas também um conjunto de atributos,

que podem ser relacionados ao serviço provido e ao usuário fornecedor do serviço, como uma forma para obter uma avaliação mais completa e efetiva de um dado serviço da Web.

Neste projeto de iniciação científica, utilizamos regressão logística para modelar e avaliar a credibilidade de aplicações da Web[Lima and Pereira 2011]. Essa avaliação é baseada em uma amostra representativa de transações eletrônicas que possuem opinião dos usuários e um *ranking* que representa uma escala de credibilidade gerada pelo modelo. Quanto maior a capacidade dos modelos em posicionar vendedores que oferecem serviços satisfatórios (que são bem qualificados a partir do *feedback*) no topo desse ranking, melhor é a qualidade do modelo. Realizamos experimentos usando dados reais de um mercado eletrônico, com o qual avaliamos os modelos de regressão logística usando diferentes tipos de fonte de informação, tais como atributos relacionados às características das ofertas, *expertise* do vendedor e sua qualificação. Os resultados apresentaram ganhos representativos, quando comparados a modelos adotados como linha-de-base experimental (*baseline*).

O restante deste artigo está organizado da seguinte forma. A seção 2 descreve alguns trabalhos correlatos ao tema de pesquisa deste projeto. A seção 3 apresenta uma breve descrição sobre regressão logística, base de nosso trabalho. A seção 4 descreve nossa metodologia, a qual será aplicada na seção 5 em nosso estudo de caso com dados reais providos pela empresa Universo Online (UOL). Finalmente, a seção 6 apresenta as conclusões e trabalhos futuros.

2. Trabalhos Relacionados

Nos últimos anos, o conceito de credibilidade na Web passou a ser mais estudado devido à necessidade de mensurar o quanto um usuário confia em um serviço ou informação disponível nesse ambiente digital.

A credibilidade de aplicações da Web se tornou um assunto multidisciplinar, onde pesquisadores das áreas de comunicação tem focado em uma análise mais qualitativa (e subjetiva) da credibilidade [Flanagin and Metzger 2007], enquanto pesquisadores da área de computação tem focado em métricas mais objetivas. Os métodos propostos na área de ciência da computação são fortemente baseados em confiança (trust) e reputação [Guha et al. 2004], além de rankings de credibilidade que levam em conta a fonte da informação [Amin et al. 2009] e o seu conteúdo [Juffinger et al. 2009].

Mecanismos de reputação são baseados em opiniões virtuais, fornecidas por pessoas que geralmente não se conhecem pessoalmente. Portanto, a confiança neste meio eletrônico é mais difícil de ser estabelecida, se comparado ao mundo real. Tomando uma visão ampla, nesses mercados eletrônicos uma reputação do comprador representa a probabilidade do pagamento do bem adquirido e a reputação do vendedor representa a probabilidade da entrega bem sucedida do item anunciado (produto que foi comprado) após o pagamento [Houser and Wooders 2006]. Essas probabilidades estão relacionadas com a confiança [Melnik and Alm 2002].

Mercados eletrônicos a cada dia estão se tornando mais populares. Diversos trabalhos investigam os sistemas de reputação e como eles induzem ao comportamento cooperativo. Dellarocas [Dellarocas 2006] realizou uma profunda revisão sobre este tópico. Ao mesmo tempo que fornece incentivos para um bom comportamento dos

usuários, sistemas de reputação também podem fomentar comportamentos enganosos.b Klos et al. [Klos and Alkemade 2005] analisaram o efeito da confiança e da reputação sobre os lucros obtidos através de intermediários em relações comerciais eletrônicas. Diferentes esquemas de propagação de confiança e desconfiança em comércio eletrônico foram estudados e avaliados em Guha et al. [Guha et al. 2004]. Resnick et al. [Resnick and Zeckhauser 2002] mostrou que os vendedores com maior reputação vendem seus produtos mais facilmente, entretanto os ganhos (preços) são reduzidos.

Os trabalhos aqui brevemente descritos sugerem a crescente necessidade de prover novos modelos de credibilidade que forneçam subsídios a usuários de serviços online, para que assim possam atuar com mais segurança e confiança na Web.

3. Regressão Logística

Regressão logística é uma técnica estatística que produz, a partir de uma série de variáveis explicativas, um modelo que permita a predição de valores tomados por uma variável dependente categórica. Assim através de um modelo de regressão, é possível calcular a probabilidade de ocorrência de um evento, através da função de ligação, conforme descrita na Equação 1:

$$\pi(x) = \frac{e^{(\beta_0+\beta_1 x_1+\beta_2 x_2+...\beta_i x_i)}}{1+e^{(\beta_0+\beta_1 x_1+\beta_2 x_2+...\beta_i x_i)}}, \quad (1)$$

onde $\pi_i(x)$ é a probabilidade de sucesso quando o valor da variável preditiva é x, β_0 é uma constante usada para ajuste e β_i são os coeficientes das variáveis preditivas [Hosmer 2000].

Para encontrar a estimativa dos coeficientes β na Equação 1, é usada a técnica de máxima verossimilhança que maximiza a probabilidade de obter o grupo observado de dados, através do modelo estimado. Na regressão logística essa técnica pode ser resolvida pelo método de *Newton-Raphson*, o qual gera uma sequência de soluções que convergem para o ponto máximo da função de verossimilhança [Casella and Berger 2002].

O modelo de regressão pode assumir natureza ordinal ou nominal, dependendo dos valores que a variável dependente pode assumir [Agresti 1996]. Neste projeto será usado a regressão logística nominal, pois não há uma ordem entre as categorias das variáveis. Neste trabalho a variável dependente contém duas categorias (variável dicotômica), portanto será usado o modelo de regressão logística binária, com multi-variáveis, ou seja, mais de uma variável independente.

Para a realização da regressão logística, é importante explicar o conceito de *modelos lineares generalizados*, que consiste em três componentes:

- Um componente aleatório, que contém a distribuição de probabilidade da variável dependente (Y).
- Uma componente sistemática, a qual corresponde a uma função linear entre as variáveis independentes.
- Uma função de ligação, que é responsável por descrever a relação matemática entre a componente sistemática e a componente aleatória. Existem duas classes de função de ligação, denominadas *logit* e *log-linear* [Dobson 1990].

O modelo de regressão logística binária é um caso especial de modelo linear generalizado, que usa a função de ligação logit para obter as estimativas dos coeficientes da Equação 1 [Venables et al. 2009].

Assim é possível obter um modelo de regressão logística. Entretanto, se torna necessário verificar quais variáveis são mais significativas para o modelo, já que modelos com muitas parâmetros apresentam correlações entre as variáveis e grande variação nas estimativas dos coeficientes. Para reduzir o modelo nós usamos a técnica *Stepwise*, a qual permite selecionar e remover variáveis, que são menos significantes para o modelo [Mccullagh and Nelder 1989]. Finalmente é possível encontrar a probabilidade de sucesso, aplicando na Equação 1 os valores das estimativas dos coeficientes encontrados.

4. Metodologia

O uso de regressão logística para criar um *ranking* de credibilidade, iniciou-se com o tratamento e a adequação da base de dados, a qual será descrita na seção 5.1. Nessa base de dados, cada transação tem sua respectiva variável resposta e outros atributos independentes. A fim de facilitar a análise dos dados todos os atributos foram normalizados.

Para construir o modelo de regressão utilizamos a ferramenta de software *R* [Team 2009], que é um software livre que possui diversos pacotes estatísticos. Para aplicar a regressão logística usamos o pacote *GLM* (*generalized linear models*), contido no software. Foi definido um atributo como variável resposta e os demais como variáveis independentes. É importante explicar que essa variável resposta é o *feedback* da transação. Para determinar que será usada uma regressão logística binária, foram configurados os campos FAMILY como binomial e o LINK como *logit*.

A fim de encontrar o melhor modelo, foram removidas as variáveis independentes menos significativas para o modelo. Sendo que o melhor modelo é o que apresenta o menor índice de *Akaike*(AIC). Usamos a técnica de *Stepwise* para realizar esta otimização. Após definir os melhores modelos foi possível obter as estimativas dos coeficientes para os atributos independentes. Em seguida foi calculado a probabilidade de uma transação obter *feedback* positivo(1), usando a Equação 1.

A metodologia usada em nosso trabalho pode ser melhor entendida pelo diagrama descrito na Figura 1. Para a construção do *ranking* de credibilidade, ordenamos em ordem decrescente as probabilidades calculadas pelo modelo de regressão.

A fim de verificar a qualidade dos modelos, escolhemos 1000 registros nos extremos (fundo e topo) do *ranking*, já que essas são as partes mais relevantes para o modelo (onde gostaríamos que estivesse localizado a maior e menor parte de transação confiáveis, respectivamente).

Cada modelo de credibilidade produz um *ranking*, onde cada posição nesse *ranking* é uma transação que contém uma probabilidade de obter *feedback* positivo e o *feedback* real obtido do usuário, que representa a variável resposta.

Espera-se que o maior número de *feedback* positivo esteja no topo do *ranking*. Analogamente é esperado que um menor número de *feedback* positivo se encontre no fundo do *ranking*. Assim, poderemos obter a precisão de cada modelo, avaliando diferentes faixas do *ranking*, comparando o valor estimado com o valor real.

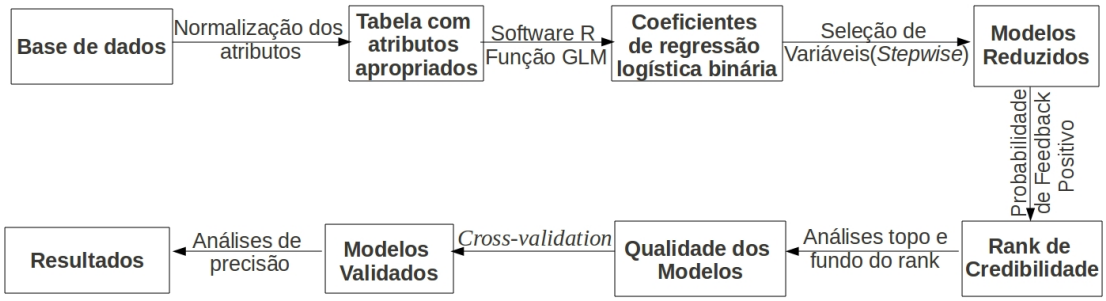


Figura 1. Modelagem da Credibilidade - Definição do Processo

A técnica de *K-fold cross-validation* foi usada para testar a qualidade de cada modelo. Definimos como 5 o número de sub-amostras (K). Assim, o conjunto de dados foi dividido em 5 partes uniformes, onde cada uma foi usada como dados de validação para encontrar os coeficientes do modelo. As outras 4 sub-amostras foram usadas como dados de treinamento, onde o modelo foi aplicado. A precisão foi calculada em cada uma das sub-amostras de treinamento, seguindo os mesmos métodos explicados para a base de dados completa. A precisão final foi obtida através da média aritmética de cada conjunto de valores.

5. Estudo de Caso

Esta seção apresenta nosso estudo de caso, onde aplicamos nossa metodologia em uma base de dados real de um mercado eletrônico para avaliar alguns modelos de credibilidade. Inicialmente, descreveremos brevemente o conjunto de dados na seção 5.1, apresentando em seguida os resultados na seção 5.2.

5.1. Visão geral da base de dados

O *TodaOferta*[1] [Pereira et al. 2009] é um mercado eletrônico desenvolvido, para compra e venda de serviços e produtos através da web, pelo maior provedor de serviços de Internet na América Latina, o Universo Online Inc. (UOL)[2].

A Tabela 1 apresenta um pequeno resumo da base de dados do *TodaOferta*. Essa amostra possui uma quantidade significativa de usuários, anúncios e negociações. Devido ao acordo de confidencialidade dos dados, as informações quantitativas não serão apresentadas neste artigo. O subconjunto da base de dados que iremos utilizar nessa pesquisa consiste de algumas dezenas de milhares de transações.

Cobertura (tempo)	Jul/2007 to Jul/2009
#categorias (*top-level*)	32
#sub-categorias	2189
Média de anúncios por vendedor	42,48
Opções de negociação	Preço fixo e Leilão

Tabela 1. TodaOferta - Sumário da Base de Dados

O mercado eletrônico *TodaOferta* utiliza um simples mecanismo de reputação. Após cada negociação, compradores e vendedores qualificam a outra parte de forma positiva (1), negativa (-1) ou neutra (0). A reputação dos usuários é definida como a soma

[1]http://www.todaoferta.com.br
[2]http://www.uol.com.br

de todas as qualificações recebidas por ele. Os *feedbacks* de um mesmo usuário só são consideradas uma única vez para computar o *score* da reputação.

Ofertas são criadas por vendedores para anunciar produtos ou serviços. As ofertas podem ser ofertadas com um preço fixo ou como um leilão. Ao contrario do *eBay*, onde leilões representam cerca de 50% do total de transações [Holahan 2008], no *TodaOferta* leilões representam menos de 2% do total de transações concretizadas.

Existem 32 níveis de categorias no toda oferta, o qual incluem 2189 sub-categorias provendo uma variedade de produtos e serviços distintos. Atualmente o maior número de vendas são de telefone celulares, *MP3 players* e *pen-drives*.

Selecionamos 15 atributos da base de dados para serem usados como candidatos para o modelo de regressão logística, que estão descritos a seguir.

- **Preço**: Preço do produto/serviço sendo oferecido..
- **Duração**: Duração da oferta (anúncio) estabelecido pelo vendedor (em dias).
- **Destaque**:Indica se a oferta foi anunciada com destaque (pacote de anúncios especiais).
- **Visualizações**: O número de visualizações da oferta.
- **Oferta com Pagamento Seguro**: Indica se uma oferta tem a opção de utilizar o mecanismo de pagamento seguro provido pelo mercado eletrônico.
- **Transação segura**: Identifica a transação que foi efetuada utilizando o mecanismo de pagamento seguro.
- **Itens Vendidos**:O montante total de itens que o vendedor vendeu no mercado eletrônico.
- **Tempo de Cadastro**: Há quanto tempo o vendedor está registrado no mercado eletrônico.
- **Qualificações Positivas**: O montante de qualificações positivas um usuário(vendedor) recebeu.
- **Porcentagem de Qualificações Positiva** :O montante relativo de qualificações positivas que um usuário(vendedor) recebeu.
- **Qualificação Global**: A soma das qualificações do vendedor, considerando diferentes tipos de qualificações recebidas.
- **Valor Total Negociado**: O montante total de dinheiro negociado pelo vendedor no mercado eletrônico.
- **Valor médio negociado**:A media do preço por transação executada pelo vendedor.
- **Mega Vendedor**: Indica se o usuário é considerado um vendedor de destaque pelo *TodaOferta*.
- **Selo Qualidade**: Denota o vendedor que possui um certificado de qualidade, provido por uma empresa à parte.

5.2. Resultados

A otimização usando a técnica de *stepwise* para construir o melhor modelo de regressão logística resultou em diferentes modelos, alguns desses modelos se adaptaram melhor para o topo do *ranking*, outros para o fundo. A seguir iremos apresentar os modelos, onde cada modelo é composto pelos atributos que melhor influenciaram no valor do *feedback*.

O modelo *Baseline* (linha de base) é formado pelos atributos *Porcentagem de Qualificações positivas* e *Qualificação Global*, os quais são considerados os atributos básicos

para gerar o feedback do mercado eletrônico usado no nosso estudo de caso. A fim de melhorar esse modelo, adicionamos novos atributos, preocupando sempre em preservar o valor do *Akaike*. Assim, 4 novos modelos foram criados.

O **modelo A** é composto pelas variáveis do *Baseline* e o atributo *Destaque*. O **modelo B** foi gerado pelo acréscimo da variável *Mega Vendedor* ao *Baseline*. O **modelo C** é composto pelos atributos do *modelo A* com o acréscimo da variável *Visualizações*. Já o **modelo D** foi criado a partir dos atributos do *modelo B* com incremento das variáveis *Itens Vendidos*, *Tempo de Cadastro* e *Oferta com Pagamento Seguro*. Além dos modelos gerados por regressão logística, foi criado um modelo aleatório para facilitar a comparação e análise com os outros modelos, já que o **modelo Aleatório** expressa a probabilidade geral de acordo com a base de dados utilizada.

Analisando os modelos, foi possível calcular a precisão em diferentes partes do *ranking*, onde nos focamos a análise sobre o topo e o fundo do *ranking*, como havíamos explicado. O modelo *Baseline* foi mais preciso que o *modelo Aleatório* em todos os intervalos avaliados. Entretanto o modelo *Baseline* apresentou menores valores de precisão em posições intermediarias do *ranking*.

Os modelos *A* e *C* não foram precisos ao prever valores de probabilidades para o topo do *ranking*. Contudo obtiveram uma boa precisão no extremo fim (fundo) do *ranking*, superando o modelo *Baseline* em muitas faixas do *ranking*.

Os modelos *B* e *D* apresentaram melhor precisão quando usados para prever valores de probabilidades para os integrantes do topo do *ranking*. Esses modelos foram melhores que o *Baseline* em maior parte das faixas do *ranking*.

A fim de avaliar os modelos, comparamos a precisão dos *rankings* obtidos para topo e fundo (Figuras 2 and 3). Esses resultados foram construídos a partir da análise dos dados gerados pela técnica de *K-fold-Cross Validation*. Foram analisados 1000 registros nos extremos (fundo e topo) e gerado um gráfico para cada extremo. Esses comparam os modelos criados pela regressão logística com o *modelo Aleatório*, em faixas variando de 10 em 10 registros (ver gráficos 3 e 2).

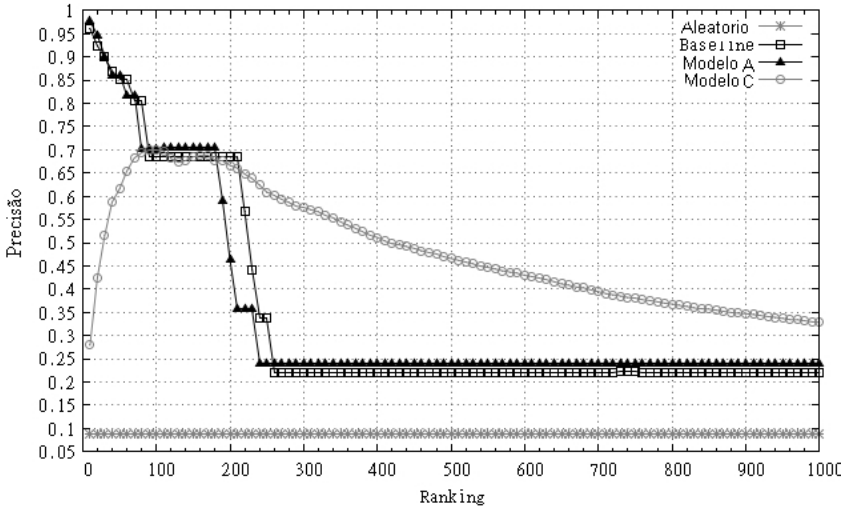


Figura 2. Modelos de credibilidade - Precisão no fundo do *ranking*.

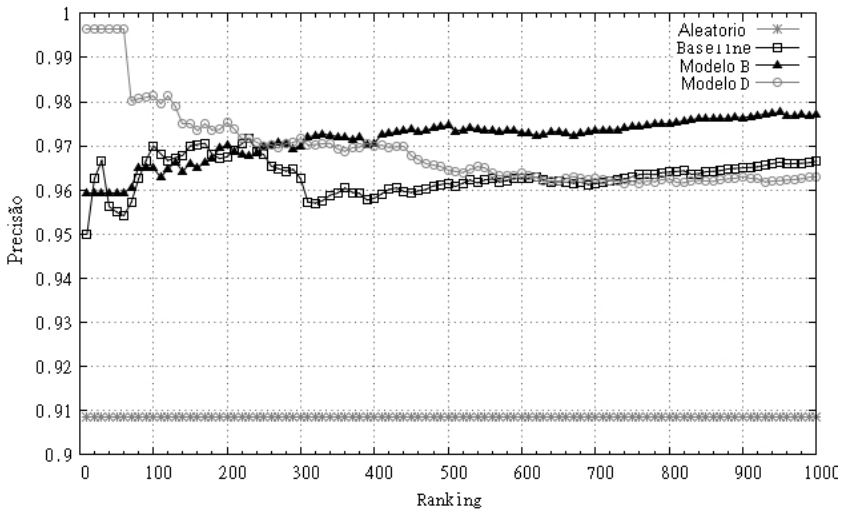


Figura 3. Modelos de Credibilidade - Precisão no topo do *ranking*.

Analisando o gráfico de precisão para o fundo (Figura 2), podemos observar que o *modelo A* obteve melhor precisão no extremo fim (correspondente aos 180 últimos registros) do *ranking*. A melhoria máxima obtida pelo modelo, em comparação ao *Baseline*, foi de 2,3% de precisão, sobre um máximo de 7,6%. O *modelo C* obteve melhor precisão depois da posição dos 220 registros do *ranking*. A melhoria máxima obtida pelo modelo, em comparação ao *Baseline*, foi de 38,2% , sob um máximo de 78%. Podemos observar que o modelo *Baseline* foi o mais preciso entre as faixas de 180 a 220 do *ranking*.

No gráfico de precisão para o Topo (Figura 3), podemos notar que o *modelo D* foi mais eficaz que o *Baseline* nas 600 primeiras posições do *ranking*, ou seja, maiores probabilidades de obter *Feedback* positivo. O *modelo D* alcançou 99,6% de precisão nos 60 primeiros registros e apresentou uma melhoria máxima de 4,6%, em comparação ao *Baseline*, sob um máximo de 5%. O *modelo B* começou a se destacar a partir da posição 250. No intervalo de 250 a 1000, o modelo foi melhor do que os modelos *D* e o *Baseline*. A melhoria máxima em relação ao *Baseline* foi de 1,5% (ante um teto máximo de 3%). Portanto, ao analisar o topo do *ranking*, os novos modelos de credibilidade superaram o modelo *Baseline* em todas as faixas do *ranking*, indicando um melhor probabilidade de obter *feedback* positivo para transações no topo desses modelos de *ranking*.

6. Conclusão

A popularização da Web tem oferecido novos serviços a cada dia, demandando mecanismos para assegurar a credibilidade desses serviços. Até o momento, pouco tem sido feito para medir e entender a credibilidade desse complexo ambiente Web, o que por si só é um grande desafio de pesquisa.

Mercados eletrônicos constituem importante cenário de pesquisa devido à sua popularidade e volume de negócios nos últimos anos. Nesse cenário a reputação assume um importante papel, principalmente para proteger compradores de vendedores fraudulentos. Um mecanismo de reputação procura provê um indicador de quão confiável é o usuário, com base em seu histórico de transações realizadas.

Neste trabalho, usamos regressão logística para modelar e avaliar a credibilidade de aplicações Web. Essa avaliação é baseada em uma amostra representativa de serviços e suas respectivas avaliações dos usuários e um *ranking* que representa uma escala de credibilidade gerada pelo modelo. Chamamos de modelo de credibilidade uma função capaz de determinar o valor de credibilidade para uma transação de uma aplicação Web, considerando diferentes critérios desses serviços e o usuário fornecedor do serviço. Quanto maior a capacidade dos modelos em posicionar vendedores que oferecem serviços satisfatórios (que são bem qualificados a partir do *feedback*) no topo desse *ranking*, melhor é a qualidade do modelo.

Executamos experimentos usando uma base de dados real de um mercado eletrônico, a partir do qual avaliamos o modelo de regressão logística usando diferentes tipos de fonte de informação, tais como atributos relacionados às características das ofertas, *expertise* do vendedor e sua qualificação. Os resultados mostram que nosso trabalho pode ser muito proveitoso e promissor. Os resultados obtidos foram muito bons, apresentando ganhos representativos, quando comparados ao *baseline*. Observamos que existem diferentes modelos para o topo e fundo do *ranking*, assim nos executamos uma análise diferente a fim de classificar as transações online em ambos os cenários.

Como trabalhos futuros queremos melhorar a avaliação e análises dos modelos de credibilidade apresentados neste trabalho. Pretendemos implementar novos modelos de credibilidade baseado em técnicas de aprendizado de máquina e algoritmos genéticos.

Agradecimentos

Este trabalho foi parcialmente patrocinado pelo Universo OnLine S. A. - UOL (www.uol.com.br) e parcialmente patrocinado pelo Instituto Nacional de Ciência e tecnologia para a Web (CNPq no. 573871/2008-6), CAPES, CNPq, Finep, e Fapemig.

Referências

Agresti, A. (1996). *An Introduction to Categorical data Analysis*. John Wiley and Sons, New York.

Amin, A., Zhang, J., Cramer, H., Hardman, L., and Evers, V. (2009). The effects of source credibility ratings in a cultural heritage information aggregator. In *WICOW '09: Proc. of the 3rd workshop on Information credibility on the web*, pages 35–42, NY, USA. ACM.

Casella, G. and Berger, R. (2002). *Statistical Inference*. Pacific Grove, 2nd edition.

Dellarocas, C. (2006). Reputation mechanisms. In *Handbook on Economics and Information Systems*, pages 629–660. Elsevier Publishing.

Dobson, A. J. (1990). *An Introduction to Generalized Linear Models*. London:Chapman and Hall.

Flanagin, A. J. and Metzger, M. J. (2007). The role of site features, user attributes, and information verification behaviors on the perceived credibility of web-based information. *New Media Society*, 9(2):319–342.

Guha, R., Kumar, R., Raghavan, P., and Tomkins, A. (2004). Propagation of trust and distrust. In *WWW '04: Proc. of the 13th international conference on World Wide Web*, pages 403–412, NY, USA. ACM.

Holahan, C. (2008). Auctions on ebay: A dying breed. *BusinessWeek online*.

Hosmer, D. W. (2000). *Applied Logistic Regression*. Wiley, New York, 2nd edition.

Houser, D. and Wooders, J. (2006). Reputation in auctions: Theory, and evidence from ebay. *Journal of Economics & Management Strategy*, 15(2):353–369.

Jøsang, A., Ismail, R., and Boyd, C. (2007). A survey of trust and reputation systems for online service provision. *Decis. Support Syst.*, 43(2):618–644.

Juffinger, A., Granitzer, M., and Lex, E. (2009). Blog credibility ranking by exploiting verified content. In *Proc. of the 3rd workshop on Information credibility on the web*, pages 51–58, NY, USA. ACM.

Klos, T. and Alkemade, F. (2005). Trusted intermediating agents in electronic trade networks. In *AAMAS '05: Proc. of the 4th international joint conference on Autonomous agents and multiagent systems*, pages 1249–1250, New York, NY, USA. ACM.

Lima, R. and Pereira, A. (2011). Applying logistic regression to rank credibility in web applications. In *7th International Conference on Web Information Systems and Technologies*.

Mccullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models*. Chapman and Hall, 2nd edition.

Melnik, M. I. and Alm, J. (2002). Does a seller's ecommerce reputation matter? evidence from ebay auctions. *Journal of Industrial Economics*, 50(3):337–49.

Pereira, A. M., Duarte, D., Jr., W. M., Almeida, V., and Góes, P. (2009). Analyzing seller practices in a brazilian marketplace. In *18th International World Wide Web Conference*, pages 1031–1041.

Resnick, P. and Zeckhauser, R. (2002). Trust among strangers in internet transactions: Empirical analysis of ebay's reputation system. *The Economics of the Internet and E-Commerce*, Elsevier Science B.V.:127–157.

Sabater, J. and Sierra, C. (2005). Review on computational trust and reputation models. *Artif. Intell. Rev.*, 24(1):33–60.

Team, R. (2009). R: A language and environment for statistical computing. `http://www.r-project.org`. The R Development Core Team.

Venables, W. N., Smith, D. M., and the R Development Core Team (2009). An introduction to r. `http://www.cran.r-project.org`.

ENECOMP - XXVIII Congresso Nacional dos Estudantes de Computação

PREFÁCIO

O ENECOMP - Congresso Nacional dos Estudantes de Computação, é um evento anual organizado pela diretoria da ENEC, pelas entidades de base e por estudantes colaboradores. É o maior e mais importante evento estudantil da área de Computação do Brasil e trata de assuntos de interesse dos estudantes, como discussões sobre a regulamentação das profissões de informática, currículo dos cursos de computação, defesa da liberdade do conhecimento, uso e desenvolvimento de software livre, economia solidária e temas ligados as novas tecnologias computacionais.

Durante o ENECOMP também acontece a Assembléia Geral dos Estudantes de Computação, onde são deliberadas questões importantes para a ENEC e é eleita e empossada a Diretoria e o Conselho Fiscal da Executiva.
O XXVIII ENECOMP será realizado nos dias 20 e 21 de julho de 2011, no Centro de Convenções de Natal - RN.

Excepcionalmente esta edição do ENECOMP voltará a ser um evento satélite dentro do XXXI CSBC - Congresso da Sociedade Brasileira de Computação, fato que não acontecia desde 2002.

Contato
Para entrar em contato com a organização, envie e-email para:
contato@enecomp.org.br

COMISSÕES

Organizadores

José Honorato F. Nunes – Coordenador Geral
Paulo Henrique de Lima Santana – Coordenador Geral
Jacques Jefferson Leão – Coordenação de Atividades Estudantis
Raissa Paiva – Colaboradora – Comunicação e Designer
Vinicius Massuchetto – Colaborador – Site

ENIA - VIII Encontro Nacional de Inteligência Artificial

PREFÁCIO

O ENIA 2011 é o oitavo de uma série de encontros em Inteligência Artificial, que têm sido realizados em conjunto com o CSBC - Congresso da Sociedade Brasileira de Computação. O ENIA, promovido bienalmente pela Comissão Especial de Inteligência Artificial - CEIA, teve edições anteriores realizadas em Brasília - DF (1997), Rio de Janeiro - RJ (1999), Fortaleza - CE (2001), Campinas - SP (2003), São Leopoldo - RS (2005), novamente Rio de Janeiro - RJ (2007) e Bento Gonçalves - RS (2009).
Em 2011, o ENIA, novamente realizado em conjunto com o CSBC, ocorre em Natal - RN, oferecendo um fórum para pesquisadores, profissionais, educadores e estudantes apresentarem e discutirem as inovações, tendências, experiências e evolução no campo da Inteligência Artificial.
Os trabalhos apresentados no ENIA introduzem novas ideias, discussões sobre trabalhos existentes, estudos práticos e experimentos, constituindo trabalhos inéditos e relevantes no campo da Inteligência Artificial, versando, principalmente, sobre os seguintes tópicos:
- Aplicações de Inteligência Artificial
- Aprendizado de Máquina
- Aquisição de conhecimento e representação do conhecimento
- Computação evolutiva e vida artificial
- Ferramentas e software para Inteligência Artificial
- Fundamentos da Inteligência Artificial
- Inteligência artificial distribuída e sistemas multiagentes
- Lógica fuzzy, sistema fuzzy e inteligência computacional
- Interfaces humano-computador inteligentes
- Mineração de dados, mineração de textos e mineração na Web
- Ontologias e processamento de linguagem natural
- Planejamento e programação
- Programação lógica
- Raciocínio automático e raciocínio baseado em modelos
- Redes neurais artificiais
- Robótica
- Sistemas de apoio à decisão
- Sistemas híbridos inteligentes
- Sistemas de informação inteligentes
- Sistemas tutores inteligentes

O ENIA 2011 oferece duas palestras e um tutorial, proferidos por renomados pesquisadores nacionais e estrangeiros:
- **Dr. Mark Klein** (MIT): apresenta o tutorial sobre “Computação Social: Progressos e Perspectivas”, onde explora as razões pelas quais computação social é tão potencialmente ponderosa, revisando sistematicamente os principais tipos e mais interessantes exemplos de tecnologia para computação social, e enumerando lições e padrões que podem ser utilizadas para ajudar a projetar aplicações inovadoras em computação social.

- **Prof. Dr. Estevam Rafael Hruschka Júnior** (UFSCar): profere a palestra “Aprendizado de Máquina Sem Fim e a Leitura da Web”, abordando o projeto *ReadTheWeb* (leitura da Web), desenvolvido em parceria por pesquisadores da Universidade Federal de São Carlos (UFSCar) e da Universidade Carnegie Mellon, dos Estados Unidos. O grupo desenvolveu um sistema computacional capaz de aprender de forma autônoma e de utilizar os conhecimentos já adquiridos para evoluir seu próprio aprendizado, com base em uma nova forma de aprendizado de máquina, denominado de "aprendizado sem fim", caracterizado por ser contínuo e sem fim, auto-supervisionado e auto-reflexivo.

- **Prof.ª Dra. Marley Maria Bernardes Rebuzzi Vellasco** (PUC-Rio): profere a palestra “Modelos Neuro-evolutivos com Inspiração Quântica”, onde propõe a utilização de algoritmos evolucionários de inspiração quântica para resolver a maioria dos problemas encontrados na utilização de redes neurais artificiais, constituindo também em uma alternativa para treinamento dessas redes.

ENIA 2011 contou com um comitê de programa composto por 168 pesquisadores, promovendo uma interação entre renomados e experientes pesquisadores com jovens pesquisadores, possibilitando uma prática de avaliação dinâmica, vibrante e responsável. O Comitê de Programa foi assessorado por um corpo de revisores adicionais composto por 78 pesquisadores.

Nesta edição, o ENIA teve 176 submissões, sendo que cada trabalho recebeu, em média, 3 pareceres, mas não menos do que 2 pareceres. Para as situações em que foi detectada a existência de pareceres conflitantes, promoveu-se um frutífero período de discussões. Este processo de avaliação ofereceu as condições adequadas para a seleção de 46 trabalhos de alta qualidade técnica para apresentação oral, e de 16 trabalhos não menos qualificados para apresentação em poster. Os anais do ENIA 2011 apresentam esses 63 trabalhos em formato completo, onde os autores procuraram atender às sugestões recebidas durante o processo de avaliação.

Gostaríamos de expressar nossos sinceros agradecimentos aos membros do Comitê de Programa e demais revisores, por sua valiosa atuação no processo de avaliação dos trabalhos, retornando pareceres consistentes e respeitando os prazos solicitados, o que com certeza garantiu a alta qualidade da programação técnica do evento, assim como desta publicação.

Agradecemos também à SBC e, em particular, à CEIA, pelo seu apoio técnico e financeiro.

Finalmente, gostaríamos de agradecer aos autores, primeiramente por terem submetido seus trabalhos para o evento, mas também pelo seu tempo e interesse em revisar, corrigir e aperfeiçoar seus trabalhos para poder acomodar todas as sugestões e exigências do Comitê de Programa.

Acreditamos que os trabalhos publicados nestes anais apresentam alta qualidade técnica, refletindo a pesquisa em Inteligência Artificial conduzida no Brasil, e esperamos que os leitores possam apreciar o evento e esta publicação.

Graçaliz Pereira Dimuro (Coordenadora de Programa)
Ronaldo Cristiano Prati (Coordenador Geral)
André Maurício Cunha Campos (Coordenador Local)

COMISSÕES

Coordenação Geral

Graçaliz Pereira Dimuro (Coordenadora de Programa)
Ronaldo Cristiano Prati (Coordenador Geral)
André Maurício Cunha Campos (Coordenador Local)

Comitê de Programa

Adolfo Neto (Universidade Tecnológica Federal do Paraná, UTFPR, Brasil)
Adriano Oliveira (Universidade Federal de Pernambuco, UFPE, Brasil)
Adriano Werhli (Universidade Federal do Rio Grande, FURG, Brasil)
Adrião Duarte Dória Neto (Universidade Federal do Rio Grande do Norte, UFRN, Brasil)
Ahmed A. A. Esmin (Universidade Federal de Lavras, UFLA, Brasil)
Alípio Jorge, (Universidade de Porto, Portugal)
Alessandro de Lima Bicho(Universidade Federal do Rio Grande, FURG, Brasil)
Alessandro Koerich (Pontifícia Universidade Católica do Paraná, PUCPR, Brasil)
Alexandre Agustini (Pontifícia Universidade Católica do Rio Grande do Sul, PUCRS, Brasil)
Alexandre Plastino (Universidade Federal Fluminense, UFF, Brasil)
Aline Villavicencio (Universidade Federal do Rio Grande do Sul, UFRGS, Brazil)
Alvaro Moreira (Universidade Federal do Rio Grande do Sul, UFRGS, Brazil)
Amilcar Cardoso (Universidade de Coimbra, Portugal)
Ana Bazzan (Universidade Federal do Rio Grande do Sul, UFRGS, Brazil)
Ana Lorena (Universidade Federal do ABC, UFABC, Brasil)
Ana Carolina Bertoletti De Marchi (Universidade de Passo Fundo, UPF, Brasil)
Ana Cristina Garcia (Universidade Federal Fluminense, UFF, Brasil)
Ana Teresa Martins (Universidade Federal do Ceará, UFC, Brasil)
Anarosa Brandão (Universidade de São Paulo, USP, Brasil)
André Campos (Universidade Federal do Rio Grande do Norte, UFRN, Brasil)
André Ponce de Leon F. de Carvalho (Universidade de São Paulo, USP, Brasil)
Andreia Malucelli (Pontifícia Universidade Católica do Paraná, PUCPR, Brasil)
Anna Helena Reali Costa (Universidade de São Paulo, USP, Brasil)
Anne Canuto (Universidade Federal do Rio Grande do Norte, UFRN, Brasil)
Antônio Braga (Universidade Federal de Minas Gerais, UFMG, Brasil)
António Branco (Universidade de Lisboa, Portugal)
Antônio Carlos da Rocha Costa (Universidade Federal do Rio Grande, FURG, Brasil)
Ariadne Carvalho (Universidade Estadual de Campinas, UNICAMP, Brasil)
Aurora Pozo (Universidade Federal do Paraná, UFPR, Brasil)
Benjamin Callejas Bedregal (Universidade Federal do Rio Grande do Norte, UFRN, Brasil)
Bianca Zadrozny (IBM Research Brazil, Brasil)
Carlos Ribeiro (Instituto Tecnológico de Aeronáutica, ITA, Brasil)
Carlos Thomaz (Centro Universitario da FEI, Brasil)
Celso Kaestner (Universidade Tecnológica Federal do Paraná, UTFPR, Brasil)
Cleber Zanchettin (Universidade Federal de Pernambuco, UFPE, Brasil)
David Martins-Jr (Universidade Federal do ABC, UFABC, Brasil)
Díbio Borges (Universidade de Brasília, UnB, Brasil)
Diana Adamatti (Universidade Federal do Rio Grande, FURG, Brasil)

Eder Mateus Gonçalves (Universidade Federal do Rio Grande, FURG, Brasil)
Edson Matsubara (Fundação Universidade Federal de Mato Grosso do Sul, UFMS, Brasil)
Edson Prestes e Silva Jr. (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Edward Hermann Haeusler (Pontifícia Universidade do Rio de Janeiro, PUC-Rio, Brasil)
Eliseo Reategui (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Emerson Paraiso (Pontifícia Universidade Católica do Paraná, PUCPR, Brasil)
Estevam Hruschka Jr. (Universidade Federal de São Carlos, UFSCar, Brasil)
Fabiano Silva (Universidade Federal do Paraná, UFPR, Brasil)
Fabio Okuyama (Instituto Federal do Rio Grande do Sul, IFRS - Campus Porto Alegre, Brasil)
Fábio Cozman (Universidade de São Paulo, USP, Brasil)
Felipe França (Universidade Federal do Rio de Janeiro, UFRJ, Brasil)
Fernando Buarque de Lima Neto (Universidade de Pernambuco, UPE, Brasil)
Fernando Osório (Universidade de São Paulo, USP, Brasil)
Flavia Barros (Universidade Federal de Pernambuco, UFPE, Brasil)
Flavia Bernardini (Universidade Federal Fluminense, UFF, Brasil)
Flavio Tonidandel (Centro Universitario da FEI, Brasil)
Flávio Soares Corrêa da Silva (Universidade de São Paulo, USP, Brasil)
Flávio Varejão (Universidade Federal do Espírito Santo, UFES, Brasil)
Francisco de Carvalho (Universidade Federal de Pernambuco, UFPE, Brasil)
Fred Freitas (Universidade Federal de Pernambuco, UFPE, Brasil)
George Cavalcanti (Universidade Federal de Pernambuco, UFPE, Brasil)
Gerardo Schneider (Chalmers | University of Gothenburg, Sweden)
Germano Vasconcelos (Universidade Federal de Pernambuco, UFPE, Brasil)
Gerson Zaverucha (Universidade Federal do Rio de Janeiro, UFRJ, Brasil)
Giancarlo Guizzardi (Universidade Federal do Espírito Santo, UFES, Brasil)
Graça Gaspar (Universidade de Lisboa, Portugal)
Graçaliz Pereira Dimuro (Universidade Federal do Rio Grande, FURG, Brasil)
Guilherme Barreto (Universidade Federal do Ceará, UFC, Brasil)
Gustavo Batista (Universidade de São Paulo, USP, Brasil)
Gustavo Giménez-Lugo (Universidade Tecnológica Federal do Paraná, UTFPR, Brasil)
Helder Coelho (Universidade de Lisboa, Portugal)
Helena Caseli (Universidade Federal de São Carlos, UFSCar, Brasil)
Heloísa Camargo (Universidade Federal de São Carlos, UFSCar Brasil)
Huei Diana Lee (Universidade Estadual do Oeste do Paraná, UNIOESTE, Brasil)
Inês Dutra (Universidade do Porto, Portugal)
Ivandré Paraboni (Universidade de São Paulo, USP, Brasil)
Jaime Sichman (Universidade de São Paulo, USP, Brasil)
João Alcântara (Universidade Federal do Ceará, UFC, Brasil)
João Balsa (Universidade de Lisboa, Portugal)
João Luis Tavares da Silva (Universidade de Caxias do Sul, UCS, Brasil)
João Marcos (Universidade Federal do Rio Grande do Norte, UFRN, Brasil)
Jomi Hübner (Universidade Federal de Santa Catarina, UFSC, Brasil)
José Alfredo F. Costa (Universidade Federal do Rio Grande do Norte, UFRN, Brasil)
José Demisio Simões Silva (Instituto Nacional de Pesquisas Espaciais, INPE, Brasil)
Júlio Cesar Nievola (Pontifícia Universidade Católica do Paraná, PUCPR, Brasil)

Karina Machado (Universidade Federal do Rio Grande, FURG, Brasil)
Karina Valdivia-Delgado (Universidade de São Paulo, USP, Brasil)
Karla Vittori (Universidade Federal do ABC, UFABC, Brasil)
Kate Revoredo (Universidade Federal do Estado do Rio de Janeiro, UNIRIO, Brasil)
Katti Faceli (Universidade Federal de São Carlos, UFSCar Brasil)
Laécio Carvalho de Barros (Universidade Estadual de Campinas, UNICAMP, Brasil)
Lúcia Drummond (Universidade Federal Fluminense, UFF, Brasil)
Lúcia Giraffa (Pontifícia Universidade Católica do Rio Grande do Sul, PUCRS, Brasil)
Lúcia Helena Machado Rino (Universidade Federal de São Carlos, UFSCar Brasil)
Leliane Nunes de Barros (Universidade de São Paulo, USP, Brasil)
Leonardo Emmendorfer (Universidade Federal do Rio Grande, FURG, Brasil)
Li Weigang (Universidade de Brasília, UnB, Brasil)
Liang Zhao (Universidade de São Paulo, USP, Brasil)
Luís Correia (Universidade de Lisboa, Portugal)
Luis Torgo (Universidade do Porto, Portugal)
Luis Cunha Lamb (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Luis Otavio Alvares (Universidade Federal de Santa Catarina, UFSC, Brasil)
Luiz Calôba (Universidade Federal do Rio de Janeiro, UFRJ, Brasil)
Luiz Chaimowicz (Universidade Federal de Minas Gerais, UFMG, Brasil)
Luiz Satoru Ochi (Universidade Federal Fluminense, UFF, Brasil)
Maíra Ribeiro Rodrigues (Universidade Federal de Minas Gerais, UFMG, Brasil)

Mara Abel (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Marcelino Pequeno (Universidade Federal do Ceará, UFC, Brasil)
Marcelo Finger (Universidade de São Paulo, USP, Brasil)
Marcelo Ladeira (Universidade de Brasília, UnB, Brasil)
Marcílio de Souto (Universidade Federal de Pernambuco, UFPE, Brasil)
Marcos Castilho (Universidade Federal do Paraná, UFPR, Brasil)
Marcos Quiles (Universidade Federal de São Paulo, UNIFESP, Brasil)
Marcos Eduardo Valle (Universidade Estadual de Londrina, UEL, Brasil)
Maria Carolina Monard (Universidade de São Paulo, USP, Brasil)
Maria Claudia Boeres (Universidade Federal do Espírito Santo, UFES, Brasil)
Maria do Carmo Nicoletti (Universidade Federal de São Carlos, UFSCar Brasil)
Marilton Aguiar (Universidade Federal de Pelotas, UFPEL, Brasil)
Mario Fernando Campos (Universidade Federal de Minas Gerais, UFMG, Brasil)
Márcia Häfele Islabão Franco (Instituto Federal do Rio Grande do Sul, IFRS - Campus Porto Alegre, Brasil)
Márcia Cristina Moraes (Pontifícia Universidade Católica do Rio Grande do Sul, PUCRS, Brasil)
Márcio Basgalupp (Universidade Federal de São Paulo, UNIFESP, Brazil)
Mário Benevides (Universidade Federal do Rio de Janeiro, UFRJ, Brasil)
Marley M. B. R. Vellasco (Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio, Brasil)
Neli Ortega (Universidade de São Paulo, USP, Brasil)
Nuno David (ISCTE-IUL, Instituto Universitário de Lisboa, Portugal)
Patrícia Tedesco (Universidade Federal de Pernambuco, UFPE, Brasil)
Patricia Jaques (Universidade do Vale do Rio dos Sinos, UNISINOS, Brasil)

Paulo Almeida (Centro Federal de Educação Tecnológica, CEFET – MG, Brasil)
Paulo Ferreira Jr. (Universidade Federal de Pelotas, UFPEL, Brasil)
Paulo Quaresma (Universidade de Évora, Portugal)
Paulo E. Santos (Centro Universitário da FEI, Brasil)
Paulo Trigo (Instituto Superior de Engenharia de Lisboa, Portugal)
Raquel Barbosa (Universidade Federal do Rio Grande, FURG, Brasil)
Regivan Hugo Nunes Santiago (Universidade Federal do Rio Grande do Norte, UFRN, Brasil)
Reinaldo Bianchi (Centro Universitario da FEI, Brasil)
Rejane Frozza (Universidade de Santa Cruz do Sul, UNISC, Brasil)
Renata Vieira (Pontifícia Universidade Católica do Rio Grande do Sul, PUCRS, Brasil)
Renata Wassermann (Universidade de São Paulo, USP, Brasil)
Ricardo Gudwin (Universidade Estadual de Campinas, UNICAMP, Brasil)
Ricardo Linden (Faculdade Salesiana Maria Auxiliadora, Brasil)
Ricardo Prudêncio (Universidade Federal de Pernambuco, UFPE, Brasil)

Ricardo Rabelo (Universidade Federal de Santa Catarina, UFSC, Brasil)
Ricardo Azambuja Silveira (Universidade Federal de Santa Catarina, UFSC, Brasil)
Ricardo Tanscheit (Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio, Brasil)
Ronaldo Cristiano Prati (Universidade Federal do ABC, UFABC, Brasil)
Ronnie Alves (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Rosa Viccari (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Rosana Sueli Jafelice (Universidade Federal de Uberlandia, UFU, Brasil)
Rosario Girardi (Universidade Federal do Maranhão, UFMA, Brasil)
Roseli Romero (Universidade de São Paulo, USP, Brasil)
Rui Camacho (Universidade do Porto, Portugal)
Sandra Sandri (Instituto Nacional de Pesquisas Espaciais, INPE, Brasil)
Sérgio Antônio Andrade de Freitas (Universidade de Brasília, UnB, Brasil)
Sílvio Cazella (Universidade do Vale do Rio dos Sinos, UNISINOS/Universidade Federal da Ciência da Saúde, UFCSPA, Brasil)
Sheila Veloso (Universidade Estadual do Rio de Janeiro, UERJ, Brasil)
Silvia Botelho (Universidade Federal do Rio Grande, FURG, Brasil)
Solange Rezende (Universidade de São Paulo, USP, Brasil)
Stanley Loh (Universidade Católica de Pelotas, UCPEL, Brazil)
Teresa Ludermir (Universidade Federal de Pernambuco, UFPE, Brasil)
Thiago Pardo (Universidade de São Paulo, USP, Brasil)
Tsang Ing Ren (Universidade Federal de Pernambuco, UFPE, Brasil)
Vasco Furtado (Universidade de Fortaleza, UNIFOR, Brasil)
Vera Lúcia Strube de Lima (Pontifícia Católica do Rio Grande do Sul, PUCRS, Brasil)
Vitor Santos Costa (University of Wisconsin-Madison, USA)
Wagner Meira Jr. (Universidade Federal de Minas Gerais, UFMG, Brasil)
Walter Carnielli (Universidade Estadual de Campinas, UNICAMP, Brasil)
Wamberto Vasconcelos (University of Aberdeen, Great Britain)
Wilson de Oliveira (Universidade Federal Rural de Pernambuco, UFRPE, Brasil)

Revisores Adicionais

Ailson Almeida (Universidade Federal do Espírito Santo, UFES, Brasil)
Aline Brum Loreto (Universidade Federal de Pelotas, UFPEL, Brasil)

Anderson Costa (Universidade Federal de Mato Grosso do Sul, UFMS, Brasil)
André Abs da Cruz (Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio, Brasil)
André Balan (Universidade Federal do ABC, UFABC, Brasil)
André Gustavo Maletzke (Universidade Estadual do Oeste do Paraná, UNIOESTE, Brasil)
Andréa Konzen da Silva (Universidade de Santa Cruz do Sul, UNISC, Brasil)
Antônio Carlos Becceneri (Instituto Nacional de Pesquisas Espaciais, INPE, Brasil)

Antônio Carlos de Arruda Junior (Universidade de Brasília, UnB, Brasil)
Antônio Selvatici (Universidade Nove de Julho, Brasil)
Ígor Assis Braga (Universidade de São Paulo, USP, Brasil)
Carlos Ferrero (Universidade Estadual do Oeste do Paraná, UNIOSTE, Brasil)
Carlos Holbig (Universidade de Passo Fundo, UPF, Brasil)
Carlos Eduardo Dantas de Menezes (Faculdade de Tecnologia do Ipiranga, FATEC, Brasil)
Carmelo Bastos Filho (Universidade de Pernambuco, UPE, Brasil)
Cesar Augusto Tacla (Universidade Tecnológica Federal do Paraná, UTFPR, Brasil)
Cícero Almeida(Universidade de Brasília, UnB, Brasil)
Claudia Martins (Universidade Federal de Mato Grosso, UFMT, Brasil)
Cristiano Pitangui (Universidade Federal do Rio de Janeiro, UFRJ, Brasil)
Daniela Bagatini (Universidade de Santa Cruz do Sul, UNISC, Brasil)
Davi Vasconcelos (Universidade Federal do Ceará, UFC, Brasil)
Denis Franco (Universidade Federal do Rio Grande, FURG, Brasil)
Diego Pereira (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Edson Pinheiro Pimentel (Universidade Federal do ABC, UFABC, Brasil)
Eduardo Bogue (Universidade Federal de Mato Grosso do Sul, UFMS, Brasil)
Eduardo Borges (Universidade Federal do Rio Grande, FURG, Brasil)
Fabrício Breve (Universidade de São Paulo, USP, Brasil)
Fernando Castilho (Pontifícia Universidade Católica do Rio Grande do Sul, PUCRS, Brasil)
Geraldo Ribeiro Filho (Universidade de Mogi das Cruzes, UMC, Brasil)
Giampaolo Luiz Libralon (Universidade de São Paulo, USP, Brasil)
Gustavo Semaan (Universidade Federal Fluminense, UFF, Brasil)
Isabela Drummond (Universidade Federal de Itajubá, Unifei, Brasil)
Jerônimo Pellegrini (Universidade Federal do ABC, UFABC, Brasil)
João Bertini (Universidade de São Paulo, USP, Brasil)
João Manuel R. S. Tavares (Universidade do Porto, Portugal)
José Aires Castro Filho (Universidade Federal do Ceará, UFC, Brasil)
José Augusto Baranauskas (Universidade de São Paulo, USP, Brasil)
Karin Becker (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Karla Figueiredo (Pontifícia Universidade Católica do Paraná, PUC-Rio, Brasil)
Larissa Freitas (Pontifícia Universidade Católica do Rio Grande do Sul, PUCRS, Brasil)
Laurence Amaral (Universidade Federal de Goiás, UFGO, Brasil)
Leonardo Luiz Barbosa Vieira Cruciol (Universidade de Brasília, UnB, Brasil)
Leticia Mara Peres (Universidade Federal do Paraná, UFPR, Brasil)
Luciana Buriol (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Luciano Silveira (Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio, Brasil)

Luciano Vitoria Barboza (Instituto Federal Sul-rio-grandense, IFSul, Campus Pelotas, Brasil)
Luis Kowada (Universidade Federal Fluminense, UFF, Brasil)
Luis Nardin (Universidade de São Paulo, USP, Brasil)
Luiz Mirisola (Universidade Federal do ABC, UFABC, Brasil)
Luiz Henrique de Campos Merschmann (Universidade Federal de Ouro Preto, UFOP, Brasil)
Manoel França (Universidade Federal do Rio de Janeiro, UFRJ, Brasil)
Márcio M. Ribeiro (Universidade Estadual de Campinas, UNICAMP, Brasil)
Marcos de Oliveira (Universidade Federal do Ceará, UFC, Brasil)
Marcos Augusto Menezes Vieira (Universidade Federal de Minas Gerais, UFMG, Brasil)
Mauricio Pilla (Universidade Federal de Pelotas, UFPEL, Brasil)
Natasha Lino (Universidade Federal da Paraíba, UFPB, Brasil)
Nelson Duarte Filho (Universidade Federal do Rio Grande, FURG, Brasil)
Neusa Grando (Universidade Tuiuti do Paraná, UTP, Brasil)
Newton Spolaôr (Universidade de São Paulo, USP, Brasil)
Norton Trevisan Roman (Universidade de São Paulo, USP, Brasil)
Patricia Oliveira (Universidade de São Paulo, USP, Brasil)
Paulo Mateus (Instituto Superior Técnico, Portugal)
Paulo Schreiner (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Paulo Lilles Jorge Drews Junior (Universidade Federal do Rio Grande, FURG, Brasil)
Pedro Coelho (Universidade Estadual do Rio de Janeiro, UERJ, Brasil)
Priscila M. V. Lima (Universidade Federal Rural do Rio de Janeiro, UFRRJ, Brasil)
Raphael Cobe (Universidade de São Paulo, USP, Brasil)
Richardson Ribeiro (Universidade Tecnológica Federal do Paraná, UTFPR, Brasil)
Rodrigo Barros (Universidade de São Paulo, USP, Brasil)
Rodrigo Calvo (Universidade de São Paulo, USP, Brasil)
Rodrigo Wilkens (Universidade Federal do Rio Grande do Sul, UFRGS, Brasil)
Roger Granada (Universidade Católica de Pelotas, UCPEL, Brasil)
Tatiane Nogueira (Universidade de São Paulo, USP, Brasil)
Valéria Feltrim (Universidade Estadual de Maringá, UEM, Brazil)
Valdinei Silva (Universidade de São Paulo, USP, Brasil)
Veronica Carvalho (Universidade Estadual Paulista, UNESP, Brasil)
Viviane Leite Dias de Mattos (Universidade Federal do Rio Grande, FURG, Brasil)
Wagner Botelho (Universidade Federal do ABC, UFABC, Brasil)

ARTIGOS

Semi-supervised Clustering in Fuzzy Rule Generation

Priscilla A. Lopes[1], **Heloisa A. Camargo**[1]

[1]Departamento de Computação – Universidade Federal de São Carlos (UFSCar)
Caixa Postal 676 – 13.565-905 – São Carlos – SP – Brazil

{priscilla_lopes,heloisa}@dc.ufscar.br

Abstract. *Inductive learning approaches traditionally categorized as supervised, which use labeled data sets, and unsupervised, which use unlabeled data sets in learning tasks. The great volume of available data and the cost involved in manual labeling has motivated the investigation of different solutions for machine learning tasks related to unlabeled data. The approach proposed here fits into this context: a semi-supervised clustering algorithm is applied to a partially labeled data set; the obtained results are used to automatically label the remaining data in the set; following, a supervised learning algorithm is used to generate fuzzy rules from the labeled data. The experiments show that this may be a promising solution for tasks that have encountered difficulties due to partially labeled data.*

1. Introduction

In recent years, the fields of machine learning and knowledge acquisition have been frequently focusing research related to Fuzzy Systems. These systems are based in the fuzzy sets theory and are mainly characterized by the use of linguistic variables in representing at least part of the knowledge. The utility of such systems is recognized in literature and still motivates investigations on methodologies to automatically build and optimize these systems aided by data sets.

Inductive learning distinguishes itself between supervised, making use of data with known classes (labeled data), and unsupervised learning, which utilizes data without classes (unlabeled data). Often enough, the scientific community addresses the need for research efforts to treat for the great volume of available data in the most diverse knowledge areas, due to the facility in obtaining this data accompanied by its high cost of manual interpretation and labeling. This situation refers to the study of learning mechanisms that consider both labeled and unlabeled data, which is known as semi-supervised learning [Klose and Kruse 2005, Zhu 2005].

Both techniques of supervised and unsupervised learning have been applied to fuzzy rule-based systems generation, the particular fuzzy systems of interest for this proposal. The current stage of research evidences that the combination of methodologies benefits the learning process as a whole, because it permits the balancing of advantages of complementary methods.

Among the supervised learning techniques, approaches that involve hybrid methods combining fuzzy set theory and other methodologies are highlighted, especially evolutionary algorithms [Cordón et al. 2004b] and neural networks [Jang et al. 1997].

As for unsupervised learning techniques, the analysis of clusters is a mechanism often explored for generation of fuzzy rules and sets from unlabeled data. The frequent

approach is to apply a clustering algorithm and transform the result in a fuzzy rule base, in which each cluster defines one rule. The fuzzy sets in each dimension are obtained by projection of the clusters. Most times these projections are approximated to triangular and trapezoidal sets.

This work investigates the combination of semi-supervised clustering algorithms with the generation of fuzzy rules. The proposed approach applies a semi-supervised clustering algorithm to a partially labeled data set, using the results to automatically label the remaining data in the set. Later, a supervised learning algorithm is used to generate fuzzy rules from the labeled data.

2. Semi-supervised Data Clustering

Methods of supervised and unsupervised learning, although useful in determined applications, present disadvantages. In supervised learning there is the need of an already labeled data set for extraction of rules. Unsupervised learning depends on the selection of appropriate metric and distribution function, number of clusters to be created and appropriate representation of significant clusters in a data set.

With the increasing complexity of data sets in many domains, complete and manual data labeling becomes more difficult and onerous. In this scenario, the use both types of data, labeled and unlabeled, becomes more interesting. A growing number of publications and conferences about semi-supervised learning is observed, reporting successful applications, especially, in image processing [Grira et al. 2006, Pedrycz et al. 2008] and text classification [Liu and Huang 2003, Geng et al. 2009].

The publications suggest and analyze modifications to already known methods aiming to consider their application to a data set with little amount of labeled and mostly unlabeled data. The work of Zhu [Zhu 2005] presents some tendencies and characteristics of semi-supervised classification, such as self-training, co-training and generative models [Chapelle et al. 2006], and points towards other forms of semi-supervised learning, by means of clustering and regression tasks.

Typically, clustering methods do not use previous knowledge, such as class labels. Semi-supervised clustering algorithms on other hand include mechanisms to consider previous information in the process of generating clusters. These mechanisms include: modification of the objective function, as to include restrictions satisfaction [Pedrycz and Waletzky 1997]; reinforcement of restrictions during the clustering process [Wagstaff et al. 2001, Grira et al. 2005]; initialization and restriction of clustering based on labeled examples [Bensaid et al. 1996, Basu et al. 2002].

The clustering methods that incorporate semi-supervision may be divided in two approaches, depending of available knowledge: seed-based approach [Bensaid et al. 1996, Pedrycz and Waletzky 1997, Basu et al. 2002] and pair-restriction-based approach [Wagstaff et al. 2001, Basu et al. 2004, Grira et al. 2008]. Seeds are labeled examples within a data set that may be used to establish restrictions to the algorithm, restrictions between pairs of examples and to define cluster labels. The pair-wise restrictions may be of type *must-link*, indicating that a pair of examples must belong to the same cluster, or *cannot-link*, indicating that a pair of examples must belong to distinct clusters.

An example of semi-supervised clustering is the semi-supervised Fuzzy C-Means

algorithm (ssFCM) [Bensaid et al. 1996]. This method is an adaptation of the Fuzzy C-Means (FCM) algorithm, that considers information in the form of seeds to improve clustering performance.

The suggested modifications were made aiming to solve three problems from the application of original FCM. The first would be the difficulty in selecting the c number of clusters, due to lack of knowledge about the data set. The second is the problem of defining appropriate labels to each cluster after the clustering process. The last problem relates to objective functions that tend to equal the number of examples in each cluster.

The ssFCM algorithm tries to minimize the described problems. Firstly, the initial FCM data set would be substituted by a union of the labeled data, X^l, and unlabeled data, X^u. Let n be the number of labels represented in X^l, $c = n$. The definition of fuzzy pseudo-partitions is changed so that the membership of labeled examples is defined as 1 and is not altered during the updates of the pseudo-partition. Lastly, a weight is added to each labeled example (w_k) to calculate cluster prototypes, which is defined according to the degree of influence of each example.

3. Semi-supervised Clustering in Fuzzy Rule Generation

The goal of this work is to explore the use of clustering algorithms in the generation of fuzzy rules. This section initiates with a brief description of the most used mechanisms in fuzzy rule generation. Following, there is a discussion on the role of clustering methods in this process, mainly, by the unsupervised version.

3.1. Fuzzy Rule-based Systems Generation

Fuzzy Rule-based Systems (FRBS) are composed by two main components: the Knowledge Base (KB) and the Inference Mechanism (IM). The automatic generation of these systems is a frequent topic in recent researches and characterizes the approximation between the areas of fuzzy modeling and machine learning [Hullermeier 2005]. As the KB of a FRBS includes a Data Base (DB), which contains the fuzzy sets, and a Rule Base (RB), which contains the rules, the methods for generation of these systems vary from adjustment of fuzzy sets, going through generation of one of the components of the KB, to the simultaneous generation of all the components, including parameters of the IM.

Many approaches have been used to automatically generate the KB from numeric data that represent samples or examples of the problem. Among the most successful techniques are the clustering algorithms [Liao et al. 2003], the gradient-based methods [Nomura et al. 1992], the fuzzy decision trees [Janikow 1998], the neural networks [Jang et al. 1997] and the genetic algorithms [Cordón et al. 2004a].

The use of clustering aiming to build a FRBS consists in identifying regions in an input space that may form the antecedent of a rule. The results of the fuzzy clustering may be transformed in a fuzzy RB such that each cluster represents a fuzzy rule. This approach is discussed in the next section.

The hybrid approaches, which combine different methodologies, specially neural networks and evolutionary computing, with fuzzy sets theory, are particularly important for fuzzy rule learning. Evolutionary algorithms in general and genetic algorithms in particular have been extensively used for FRBS learning. In this approach there is a great

variety of mechanisms to optimize or build a FRBS or its parts, sequentially, simultaneously or even as a combination of other methods.

Fuzzy Systems combined with a learning process based on genetic algorithms are named Genetic Fuzzy Systems (GFS). The research on genetic rule-base fuzzy systems has already originated an expressive number of work and continues in activity nowadays. Important contributions that characterize the state of the art and the main tendencies may be found in the many special editions on the subject, such as [Cordón et al. 2004b, Casillas et al. 2007, Herrera 2008, Casillas and Carse 2009].

The GFS are particularly relevant to this work, as long as some of the supervised methods for rule generation utilized in this work were chosen among the ones that fit this GFS proposal.

3.2. Fuzzy Clustering in Generating Fuzzy Rule-based Systems

The most common approach to generate rules by fuzzy clustering is to consider each cluster as a possible rule. Thus, after finding clusters through an algorithm, the antecedent of the rule is determined by projecting the cluster on each of the input space dimensions to obtain propositions in the form "V is A". The conjunction of these propositions forms the antecedent of the rule and each cluster is associated to a class. The disadvantage of this approach is that each rule makes use of its own fuzzy sets, generated by the projection, which may cause linguistic interpretability problems [Klose and Kruse 2005]. Examples of pioneer work following this approach are [Sugeno and Yasukawa 1993, Yager 1993, Babuska et al. 1994].

In recent work it is frequently found approaches that combine clustering methods with other learning mechanisms for the generation of fuzzy rules. In [Juang 2005], for instance, the proposed method combines on-line clustering and genetic algorithm in generating FRBS, with reinforcement. Other approaches that combine genetic algorithms with clustering may be found in [Saez et al. 2008]. In [Lee et al. 2008] a new iterative fuzzy clustering algorithm is presented. The fuzzy rules are obtained by an iterative process of selecting clusters with supervision based on notions of purity and separability of clusters.

Finally, it must be mentioned that clustering methods are very utilized in the generation of fuzzy sets, combined with the fuzzy rule generation process made by another method [Liao et al. 2003].

The proposal developed here concerns the combination of clustering methods with fuzzy systems generation in a way that is different from the ones usually encountered in the literature. The proposal is presented in the following section.

4. Proposed Approach

The goal of the proposed approach is to explore and evaluate the use of semi-supervised clustering methods in generating fuzzy rule bases. The main idea is to automatically label partially labeled data sets, based on the partition obtained by the application of semi-supervised clustering methods and, then, generate a fuzzy rule-based system with the aid of a supervised classification method.

The main motivation for this work was to investigate a hybrid mechanism that use partially labeled data, by the combination of two complementary methodologies. While the generation of fuzzy rules is usually done by mechanisms that require labeled data, the increasing availability of unlabeled data evidences the need to research methods that are adequate for such situations. On the other hand, clustering methods, traditionally applied to unlabeled data, present difficulties relative to validation and interpretation of groups resulting from the partitioning, which might not favor its utilization to generate rules. The proposed method aims to study alternatives to surpass those difficulties.

The method developed in this work is divided in two steps: (1) clustering and labeling; (2) rule generation . These steps are described in the following sections.

4.1. Clustering and Labeling Step

This step aims to automatically label the training data, according to the resulting partition of the semi-supervised algorithm. The labeled data can then be used in supervised learning, specifically, in the generation of fuzzy rules.

The clustering was based on the seeded-based ssFCM algorithm. The FCM algorithm was used for comparison and clustering validation. The comparisons between the clustering algorithms were based on the heterogeneity index R [Carvalho et al. 2006].

4.2. Rule Generation Step

The rule generation step consists of the application of a supervised fuzzy rule generation method on the labeled data set resulting from the previous step. Aiming at providing enough results to support the comparative analysis, four different rule generation algorithms have been used in this work.

One of the algorithms used is a non-evolutionary fuzzy rule base generation method, Fuzzy Rule Learning Model [Chi et al. 1996, Cordón et al. 1999]. The Fuzzy Rule Learning Model (CRW) builds a fuzzy rule base by means of a technique adapted from Wang & Mendel method, also considering weights that improve the classification. The other algorithms selected for the rule generation task are evolutionary methods. The hybrid fuzzy genetics-based machine learning algorithm (IH) [Ishibuchi et al. 2005] is a method that implements a hybridism between two famous approaches: Michigan and Pittsburgh. The steady-state algorithm for extracting fuzzy classification rules from data (SG)[Mansoori et al. 2008] is a genetic algorithm with a finite number of generations, which is related to the dimension of the evaluated problem. The goal of the SG is to extract a more compact and legible rule base. The structural learning algorithm on vague environment (SL) [González and Perez 2001] method is a genetic algorithm based on the iterative approach.

The selected methods were applied using the KEEL (Knowledge Extraction based on Evolutionary Learning) tool [Alcala-Fdez et al. 2009]. The evaluation of obtained results were based on the error rate, i.e., the percentage of incorrectly classified test instances.

The main goal of the experiments is to compare the results of the rule bases generated from automatically labeled data set and originally labeled data set.

Table 1. Data sets used in experiments

Data Set	Instances	Attributes	Classes
Apendicitis	109	9	2
Balance	625	4	3
Bupa	345	6	2
Ecoli	336	7	8
Glass	214	9	7
Haberman	306	3	2
Ionosphere	351	33	2
Iris	150	4	3
Monk 2	432	6	2
New Thyroid	215	5	3
Pima	768	8	2
Sonar	208	60	2
Spambase	4597	57	2
SPECTF Heart	267	44	2
Texture	5500	40	11
Vehicle	846	18	4
WDBC	569	30	2
Wine	178	13	3
Yeast	1484	8	10

5. Experiments and Results

Thirteen popular data sets, shown in Table 1were used to validate the proposal and evaluate the results obtained by the clustering and classification methods. The numerical data sets used are available at the UCI repository for machine learning [Frank and Asuncion 2010]. Cross-validation for 5 folds was adopted to compare the performance of clustering and classification algorithms.

The goals of the proposal allow the validation to be divided in two parts: the validation of results obtained by clustering methods and the validation of the fuzzy rule-base for classification generated by the rule base generation methods. Specifications of each part are described is sections 5.1 and 5.2.

5.1. Clustering Validation

The validation of clustering methods is given by an index obtained from the results of experiments considering two different fuzzy clustering algorithms:

FCM Unsupervised method. The original labels were removed from the data set. The metric for instance dissimilarity metric was the Euclidean distance.

ssFCM Semi-supervised method, based on seeds. The original labels of the training sets were partially removed and the remaining labels were used as seeds. Experiments were made with sets of seeds sized at approximately 10%, 20% and 30% of instances. The selection of seeds happened in two different ways: random selection of n instances per label and random selection of instances from the complete training set.

The implementation of the clustering algorithms uses a few structures of the WEKA [Hall et al. 2009] data mining software package.

This work utilizes an index based on sum of squares (SSQ) to evaluate the fuzzy clustering results applied to quantitative data sets. The overall SSQ and between-cluster SSQ metrics [Carvalho et al. 2006] are used in the definition of the interpretation index.

The overall SSQ metric (T) evaluates the general heterogeneity for all n examples in the data set, according to the distance function used in the clustering algorithm. This metric relates each example of the data set with the general data set prototype.

Between-cluster SSQ metric (B) evaluates the dispersion of cluster prototypes and, thus, the difference among all the clusters obtained with the application of the clustering algorithm. It relates the cluster prototypes with the general data set prototype and uses cluster membership degrees obtained by the clustering.

The general heterogeneity index (R) is obtained by $R = \frac{B}{T}$ were high values of R indicate clusters that are more homogeneous and better represented by the cluster prototype.

5.2. Classification Validation

The validation of the fuzzy rule-base generation is given by comparing and evaluating results obtained by the application of classification algorithms to the data sets with the original labels and on the data sets that have gone through the semi-supervised clustering and labeling.

Comparison of the results is given by the Friedman test [Demšar 2006]. A post-hoc test is applied when significant difference between results is verified.

Generally, the Friedman test is applied to evaluate algorithms. In this work, the goal is to verify whether the application of classification algorithms to data sets that were automatically labeled after being clustered by a semi-supervised algorithm influences the performance of the classification algorithm.

The results for the Friedman test and the post-hoc were given by the data analyses software GraphPad (http://www.graphpad.com/).

5.3. Results

The goal of a clustering algorithm is finding the optimal partition of a determined data set. This implies in partitioning the data as to obtain groups that are most homogeneous and compact as possible, given a proximity measure.

One of the difficulties in clustering tasks is that sets with high dispersion of data may influence the results of the clustering in a negative way. Table 2 indicate the R index, as presented in section 5.1, for results on the clustering of the used data sets. The applied clustering methods were FCM and ssFCM. In Table 2, <data-set-name>$_{(a,b,c)}$, a, b and c are the number of groups in which the data was partitioned and ssFCM$_i$, i is the percentage of seeds relative to the number of total instances in the training data set

Low values for R indicate a considerably disperse set. It is noted, in Table 2, that, as the number of groups in FCM clustering increases, the value R also increases. Data partitioning using higher numbers of groups may generate better values for the R index, although that does not always indicate good partitioning. The value of R, for some data sets, is inversely proportioned to the number of initial seeds. The disperse characteristic

Table 2. General heterogeneity index (R) for FCM and ssFCM

Data Set	$\mathbf{FCM}_a$	$\mathbf{FCM}_b$	$\mathbf{FCM}_c$	$\mathbf{ssFCM}_{10\%}$	$\mathbf{ssFCM}_{20\%}$	$\mathbf{ssFCM}_{30\%}$
$\text{Balance}_{(2,3,4)}$	0.0042	0.0366	0.0818	0.0192	0.0168	0.0121
$\text{Bupa}_{(2,3,4)}$	0.0001	0.0187	0.0717	0.0000	0.0000	0.0000
$\text{Haberman}_{(2,3,4)}$	0.8935	0.9432	0.9607	0.9049	0.8778	0.8628
$\text{Ionosphere}_{(2,3,4)}$	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
$\text{Iris}_{(2,3,4)}$	0.5224	0.6476	0.7188	0.6800	0.6815	0.6825
$\text{Monk2}_{(2,3,4)}$	0.0174	0.0436	0.0697	0.0221	0.0114	0.0070
$\text{New Thyroid}_{(2,3,4)}$	0.6579	0.7499	0.8184	0.6985	0.6842	0.6931
$\text{Pima}_{(2,3,4)}$	0.0000	0.0006	0.0072	0.0000	0.0000	0.0000
$\text{Sonar}_{(2,3,4)}$	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
$\text{SPECTF Heart}_{(2,3,4)}$	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
$\text{Vehicle}_{(3,4,5)}$	0.0026	0.0041	0.0058	0.0019	0.0012	0.0005
$\text{WDBC}_{(2,3,4)}$	0.0025	0.0056	0.0161	0.0020	0.0016	0.0012
$\text{Wine}_{(2,3,4)}$	0.0038	0.0116	0.1196	0.0003	0.0003	0.0003

Table 3. Error percentage for rule-base (O)

Data Set	CRW	IH	SG	SL
Balance	9.30	15.80	25.00	19.00
Bupa	40.87	29.57	39.89	36.20
Haberman	26.23	24.80	25.86	26.06
Ionosphere	18.61	34.61	23.29	5.41
Iris	7.17	4.83	6.08	3.17
Monk 2	30.57	4.09	19.45	2.78
New Thyroid	14.48	5.47	10.23	8.20
Pima	26.01	23.5	28.24	24.78
Sonar	20.41	51.05	27.14	21.27
SPECTF Heart	18.53	26.97	21.11	20.13
Vehicle	36.61	47.50	48.11	28.53
WDBC	5.59	8.22	8.19	7.95
Wine	4.30	8.87	9.19	4.38

of data sets may influence the partitioning using ssFCM, "confusing" the algorithm as the number of seeds increases.

Tables 3 and 4 present the error rate for experiments made with algorithms CRW, IH, SG and SL, with the different types of data. Table 3 contains the results for the rule-base generated from the 100% original labeled data set (O). Table 4 presents the results obtained from data set with planned seeds (P) and random seeds (R), which went through a labeling process after being clustered by the ssFCM algorithm.

An initial evaluation of Tables 3 and 4 suggest that the difference in results for the 3 types of data set (original and automatically labeled) are not significant.

In the first evaluation strategy, the Friedman test was among the rule-base generation method for each data set. The goal was to verify if an algorithm a_1 that performs significantly better than another algorithm a_2, when applied to data set O, will still perform better when applied to data sets P and R. The results obtained confirm the initial evaluation of the error rates table: the results are not significantly different.

In the second evaluation strategy used to analyze the results of the experiments, the comparison was made among the different data types for each rule generation method. Considering that the labeling based on the semi-supervised clustering is successful when

Table 4. Error percentage for rule-base (P and R)

Data Set	$\mathbf{CRW}_P$	$\mathbf{IH}_P$	$\mathbf{SG}_P$	$\mathbf{SL}_P$	$\mathbf{CRW}_R$	$\mathbf{IH}_R$	$\mathbf{SG}_R$	$\mathbf{SL}_R$
Balance	14.44	15.48	15.00	15.18	20.68	19.68	22.14	17.04
Bupa	19.13	11.01	14.89	9.06	19.82	13.74	17.43	10.66
Haberman	32.21	9.88	35.95	16.9	29.74	11.15	70.32	12.25
Ionosphere	25.71	38.50	28.34	16.24	27.60	46.09	28.34	15.33
Iris	8.50	6.17	7.42	6.50	8.67	6.75	8.25	7.09
Monk 2	46.93	11.48	16.42	8.94	45.86	8.51	10.20	7.58
New Thyroid	41.16	41.16	41.16	41.16	22.85	11.63	21.57	18.67
Pima	22.62	14.06	21.26	17.97	21.78	12.50	20.93	15.02
Sonar	25.72	52.76	28.59	22.71	25.73	52.76	28.60	22.71
SPECTF Heart	18.53	26.97	21.11	20.13	16.20	31.07	28.33	17.75
Vehicle	18.91	22.31	15.82	15.38	20.97	23.52	20.91	16.11
WDBC	7.58	8.76	4.55	3.62	7.58	6.69	4.29	3.15
Wine	4.30	8.87	9.19	4.38	23.36	19.22	16.66	14.18

the labels are as close as possible to the original ones, that goal is reached if the performance differences between the algorithms is not significant.

As this second application of the Friedman test showed no significant differences, it is inferred, according to earlier considerations, that the resulting labels obtained after the semi-supervised clustering may be as good as the ones in the original data sets.

6. Conclusion

This work presents a semi-supervised learning proposal that utilizes a semi-supervised clustering approach to label data for posterior application of a method to generate a fuzzy rule-base for classification.

This proposal suggests a solution for machine learning problems in domains that are characterized by great volume of data and varied data types, but have a relatively small set of labeled instances.

The results of this work suggest that considering semi-supervised clustering in labeling data may be a good approach for learning tasks in those contexts. The discussion presented here confirm the relevance and importance of the proposed work.

Relevant questions and considerations were raised due to the research involved in this proposal, which may be explored in future work: adaptation and proposal of semi-supervised clustering methods, aiming partially classified data labeling; comparison of results between distinct semi-supervised learning approaches, evaluation of results obtained by application of the proposed method as a solution for problems in specific domains.

Acknowledgment

The authors would like to thank E-Biz solution and CAPES for their support.

References

Alcala-Fdez, J., Sánchez, L., García, S., del Jesus, M., Ventura, S., Garrell, J., Otero, J., Romero, C., Bacardit, J., Rivas, V., Fernández, J., and Herrera, F. (2009). Keel: A software tool to assess evolutionary algorithms for data mining problems. *Soft Computing*, 13(3):307–318.

Babuska, R., Jager, R., and Verbruggen, H. (1994). Interpolation issues in sugeno-takagi reasoning. In *Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on*, pages 859–863.

Basu, S., Banerjee, A., and Mooney, R. J. (2002). Semi-supervised clustering by seeding. In *ICML '02: Proceedings of the Nineteenth International Conference on Machine Learning*, pages 27–34.

Basu, S., Banerjee, A., and Mooney, R. J. (2004). Active semi-supervision for pairwise constrained clustering. In *In Proceedings of the 2004 SIAM International Conference on Data Mining*, pages 333–344.

Bensaid, A. M., Hall, L. O., Bezdek, J. C., and Clarke, L. P. (1996). Partially supervised clustering for image segmentation. *Pattern Recognition*, 29(5):859–871.

Carvalho, F. A. T., Tenorio, C. P., and Junior, N. L. C. (2006). Partitional fuzzy clustering methods based on adaptive quadratic distances. *Fuzzy Sets and Systems*, 157(21):2833–2857.

Casillas, J. and Carse, B. (2009). Special issue on "genetic fuzzy systems: Recent developments and future directions". *Soft Comput.*, 13(5):417–418.

Casillas, J., Herrera, F., Pérez, R., del Jesús, M. J., and Villar, P. (2007). Special issue on genetic fuzzy systems and the interpretability-accuracy trade-off. *Int. J. Approx. Reasoning*, 44(1):1–3.

Chapelle, O., Schölkopf, B., and Zien, A. (2006). *Semi-Supervised Learning*. MIT Press, Cambridge, MA.

Chi, Z., Yan, H., and Pham, T. (1996). *Fuzzy Algorithms: With Applications To Image Processing and Pattern Recognition*. World Scientific.

Cordón, O., del Jesus, M., and Herrera, F. (1999). A proposal on reasoning methods in fuzzy rule-based classification systems. *International Journal of Approximate*, 20(1):21–45.

Cordón, O., Gomide, F., Herrera, F., Hoffmann, F., and Magdalena, L. (2004a). Special issue on genetic fuzzy systems. *Fuzzy Sets and Systems*, 141.

Cordón, O., Gomide, F., Herrera, F., Hoffmann, F., and Magdalena, L. (2004b). Ten years of genetic fuzzy systems: current framework and new trends. *Fuzzy Sets and Systems*, 141(1):5–31.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, 7:1–30.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository. http://archive.ics.uci.edu/ml.

Geng, C., Yuquan, Z., Jianing, T., and Tianhan, H. (2009). An algorithm of semi-supervised web-page classification based on fuzzy clustering. In *Information Technology and Applications, 2009. IFITA '09. International Forum on*, volume 1, pages 3–7.

González, A. and Perez, R. (2001). Selection of relevant features in a fuzzy genetic learning algorithm. *IEEE Transactions on Systems and Man and and Cybernetics and Part B: Cybernetics*, 31(3):417–425.

Grira, N., Crucianu, M., and Boujemaa, N. (2005). Active semi-supervised fuzzy clustering for image database categorization. In *MIR '05: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, pages 9–16. ACM.

Grira, N., Crucianu, M., and Boujemaa, N. (2006). Fuzzy clustering with pairwise constraints for knowledge-driven image categorisation. *Vision, Image and Signal Processing, IEEE Proceedings* -, 153(3):299–304.

Grira, N., Crucianu, M., and Boujemaa, N. (2008). Active semi-supervised fuzzy clustering. *Pattern Recognition*, 41(5):1851–1861.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18.

Herrera, F. (2008). Genetic fuzzy systems: Taxonomy and current research trends and prospects. *Evolutionary Intelligence*, 1:27–46.

Hullermeier, E. (2005). Experience-based decision making: A satisficing decision tree approach. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 35(5):641–653.

Ishibuchi, H., Yamamoto, T., and Nakashima, T. (2005). Hybridization of fuzzy gbml approaches for pattern classification problems. *IEEE Transactions on Systems and Man and Cybernetics - Part B: Cybernetics*, 35(2):359–365.

Jang, J. S. R., Sun, C. T., and Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing*. Matlab Curriculum Series. Prentice Hall.

Janikow, C. Z. (1998). Fuzzy decision trees: issues and methods. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 28(1):1–14.

Juang, C.-F. (2005). Combination of online clustering and q-value based ga for reinforcement fuzzy system design. *Fuzzy Systems, IEEE Transactions on*, 13(3):289–302.

Klose, A. and Kruse, R. (2005). Semi-supervised learning in knowledge discovery. *Fuzzy Sets and Systems*, 149:209–233.

Lee, H. E., Park, K. H., and Bien, Z. Z. (2008). Iterative fuzzy clustering algorithm with supervision to construct probabilistic fuzzy rule base from numerical data. *IEEE T. Fuzzy Systems*, 16(1):263–277.

Liao, T. W., Celmins, A. K., and II, R. J. H. (2003). A fuzzy c-means variant for the generation of fuzzy term sets. *Fuzzy Sets and Systems*, 135(2):241–257.

Liu, H. and Huang, S. T. (2003). A genetic semi-supervised fuzzy clustering approach to text classification. In *In Proceedings of the 4th International Conference on Advances in Web-Age Information Management (WAIMâ03), Lecture Notes in Computer Science*, pages 173–180.

Mansoori, E., Zolghadri, M., and Katebi, S. (2008). Sgerd: A steady-state genetic algorithm for extracting fuzzy classification rules from data. *IEEE Transactions on Fuzzy Systems*, 16(4):1061–1071.

Nomura, H., Hayashi, I., and Wakami, N. (1992). A learning method of fuzzy inference rules by descent method. In *Proc. IEEE Int Fuzzy Systems Conf*, pages 203–210.

Pedrycz, W., Amato, A., Di Lecce, V., and Piuri, V. (2008). Fuzzy clustering with partial supervision in organization and classification of digital images. 16(4):1008–1026.

Pedrycz, W. and Waletzky, J. (1997). Fuzzy clustering with partial supervision. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 27(5):787–795.

Saez, Y., Zazo, F., and Isasi, P. (2008). A study of the effects of clustering and local search on radio network design: Evolutionary computation approaches. In *Hybrid Intelligent Systems, 2008. HIS '08. Eighth International Conference on*, pages 951–954.

Sugeno, M. and Yasukawa, T. (1993). A fuzzy-logic-based approach to qualitative modeling. *Fuzzy Systems, IEEE Transactions on*, 1(1):7.

Wagstaff, K., Cardie, C., Rogers, S., and Schrödl, S. (2001). Constrained k-means clustering with background knowledge. In *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*, pages 577–584. Morgan Kaufmann Publishers Inc.

Yager, R. (1993). On a hierarchical structure for fuzzy modeling and control. *Systems, Man and Cybernetics, IEEE Transactions on*, 23:1189–1197.

Zhu, X. (2005). Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison.

NebFuzz: Um Novo Algoritmo de Agrupamento Semi-Supervisionado Baseado no *Fuzzy C-Means*

Valmir Macário[1], **Francisco de A. T. de Carvalho**[1]

[1]Centro de Informática, Universidade Federal de Pernambuco;
Av. Prof. Luiz Freire, s/n, Cidade Universitária, Recife-PE; Brazil; 50740-540

{vmf2,fatc}@cin.ufpe.br

Abstract. *Semi-supervised clustering uses unlabeled data, combined with the labeled data, to improve the algorithm performances. This paper presents a new algorithm for semi-supervised clustering based on Fuzzy C-Means algorithm. The new algorithm was evaluated and compared against two semi-supervised clustering algorithms in the context of learning from partially labeled data. The behavior of the proposed algorithm is discussed and the results are validated using accuracy rate, corrected rand index and a 95% confidence interval. Thus, it was possible to certify the better accuracy performance of the new algorithm when a few labeled data are available.*

Resumo. *Agrupamento semi-supervisionado utiliza dados não rotulados, juntamente com dados rotulados, com a finalidade de melhorar o desempenho dos algoritmos. Este trabalho apresenta um novo algoritmo de agrupamento semi-supervisionado baseado no algoritmo* Fuzzy C-Means*. O novo algoritmo é avaliado em relação à dois algoritmos de agrupamento semi-supervisionados a partir de dados parcialmente rotulados nas tarefas de classificação e de agrupamento. Além disso, o comportamento do algoritmo é discutido e os resultados validados com taxa de acerto, índice de Rand corrigido e intervalos com 95% de confiança. Desse modo, foi possível certificar que o novo algoritmo de agrupamento semi-supervisionado apresenta desempenho melhor quando há poucos dados rotulados disponíveis.*

1. Introdução

A aprendizagem semi-supervisionada [Chapelle et al. 2006][Zhu 2008] é uma abordagem intermediária entre a aprendizagem supervisionada e a aprendizagem não-supervisionada. A abordagem semi-supervisionada é importante devido a dificuldade de rotular dados, que pode ser uma tarefa extremamente complexa, cara, demorada e requerer especialistas humanos em algumas aplicações reais. Em contraponto, existe a facilidade de obtenção de dados não rotulados [Nigam et al. 2000]. Na aprendizagem semi-supervisionada são utilizados exemplos rotulados e não rotulados para guiar a aprendizagem com a finalidade de melhorar o desempenho dos algoritmos.

Uma das técnicas semi-supervisionadas, que despertaram grande interesse, foram as de agrupamento semi-supervisionado [Pedrycz and Waletzky 1997]. Essas técnicas são de grande interesse devido a possibilidade de incorporar dados rotulados nos algoritmos tradicionais de agrupamento não supervisionado. A tarefa de agrupamento tem sido aplicada em diversos problemas. Para citar alguns, temos a mineração de textos, agrupamento de expressões gênicas, processamento de imagens, entre outras.

O objetivo deste artigo é apresentar um novo algoritmo de agrupamento semi-supervisionado chamado de NebFuzz. O novo algoritmo demonstrou nos experimentos que é uma boa opção para tarefas com poucos dados rotulados. O algoritmo foi avaliado em duas tarefas: classificação e agrupamento. O algoritmo foi comparado com algoritmos totalmente supervisionados [Mitchel 1997] e com algoritmos de agrupamento semi-supervisionado consolidados na literatura [Pedrycz and Waletzky 1997][Bouchachia 2007].

Este artigo está estruturado da seguinte forma: a Seção 2 apresenta trabalhos relacionados de aprendizagem semi-supervisionada. A Seção 3 apresenta detalhadamente o algoritmo semi-supervisionado proposto neste trabalho. A Seção 4 descreve a metodologia dos experimentos utilizados neste trabalho. Depois, na Seção 5, os resultados dos experimentos são apresentados. Finalmente, a Seção 6 traz as conclusões e trabalhos futuros.

2. Trabalhos Relacionados

O agrupamento é a aglomeração de objetos (ou exemplos) em grupos, de modo que os objetos pertencentes ao mesmo grupo sejam mais similares entre si de acordo com alguma medida de similaridade [Hathaway et al. 2000], enquanto os objetos pertencentes a grupos diferentes tenham uma similaridade menor. O objetivo do processo de agrupamento é maximizar a homogeneidade dos objetos de um mesmo grupo enquanto maximiza a heterogeneidade entre objetos de grupos diferentes [Stepp and Michalski 1986]. Segundo Jain e Dubes [Jain and Dubes 1988], agrupamento é o estudo formal de algoritmos e métodos para agrupar exemplos que não estão rotulados numa classe correspondente.

Alguns algoritmos não supervisionados clássicos foram adaptados para levar em conta informações supervisionadas em seu treinamento. Assim, os algoritmos de agrupamento se tornaram semi-supervisionados, pois em seu treinamento utilizam informações supervisionadas e não supervisionadas. O algoritmo de agrupamento clássico que a maioria desses algoritmos se baseou foi o *Fuzzy C-Means* (FCM) [Bezdek 1981].

O Algoritmo de Pedrycz e Waletzky [Pedrycz and Waletzky 1997], que foi chamado de ISODATA com supervisão parcial (ISODATA-PS), foi um dos primeiros algoritmos que utilizou a abordagem semi-supervisionada para algoritmos de agrupamento. Pedrycz formulou um novo algoritmo modificando a função objetivo do algoritmo FCM. O termo adicionado à função objetivo do algoritmo FCM utiliza os rótulos disponíveis dos exemplos da base de dados para aumentar os graus de pertinência atribuídos pelo algoritmo para os grupos que representam a classe que o exemplo pertence. A utilização desse algoritmo foi numa base de sinais biológicos (eletrocardiogramas), onde havia poucos elementos rotulados.

O algoritmo proposto por Bouchachia e Pedrycz [Bouchachia and Pedrycz 2006] basicamente estende a função objetivo do algoritmo *Fuzzy C-Means*. O objetivo dessa mudança é capturar estruturas dos dados não rotulados e rotulados. As estruturas dos dados não rotulados são adquiridas pelo primeiro termo da função objetivo, que é igual a função objetivo do algoritmo FCM. O segundo termo leva em conta as estruturas refletidas pela avaliação dos rótulos disponíveis. Uma das vantagens desse algoritmo é que a função objetivo leva em conta possibilidade de mais de um grupo representar uma determinada classe. Aqui, esse algoritmo é identificado pela sigla SSC (*Semi-Supervised Clustering*).

3. Algoritmo NebFuzz

O algoritmo proposto neste trabalho é uma nova abordagem para algoritmos de agrupamento semi-supervisionado que acrescenta um segundo termo supervisionado à função objetivo do algoritmo FCM de Bezdek [Bezdek 1981]. A motivação para construir outro algoritmo de agrupamento semi-supervisionado foi a de que existe a hipótese que padrões pertencentes à mesma classe possuem alta similaridade. Como consequência, esses padrões estão próximos entre si, e também, o grau de pertinência do padrão que pertence ao grupo que representa sua classe tem quer ser similares. Assim, desenvolvemos uma função de otimização em que dois padrões são comparados entre si, e caso pertençam à mesma classe, a diferença entre os graus de pertinência para o grupo da classe que os dois pertencem é atenuada para que seus graus de pertinência se tornem similares. Essa manobra faz com que padrões da mesma classe possuam graus de pertinência parecidos para os C grupos. Desse modo, a função objetivo do novo algoritmo de agrupamento semi-supervisionado, dado que N padrões (inclui padrões rotulados e não rotulados) são agrupados em C grupos e existe P classes conhecidas é:

$$J(U,V) = \sum_{k=1}^{C}\sum_{i=1}^{N} u_{ik}^2 d_{ik}^2 + \alpha \sum_{k=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{N} b_i b_j t_{ij}(u_{ik} - u_{jk})^2 d_{ij}^2 \quad (1)$$

onde U representa a matriz de graus de pertinência u_{ik} do padrão $\mathbf{x_i}$ no grupo k. V representa o conjunto de protótipos $\mathbf{v_k}$ $(v_k = (v_{k1}, ..., v_{kz}))$ associado a cada grupo, onde z é o número de atributos. Existe um conjunto de dados representado pelo conjunto $X = \{\mathbf{x_1}, \mathbf{x_i}, ..., \mathbf{x_N}\}$, tal que cada padrão $\mathbf{x_i} = (x_{i1}, ..., x_{iz})$. A distância euclidiana entre o padrão $\mathbf{x_i}$ e o protótipo $\mathbf{v_k}$ é representada por d_{ik}. $b_i = 1$ e $b_j = 1$ se os padrões $\mathbf{x_i}$ e $\mathbf{x_j}$ são rotulados, e $b_i = 0$ e $b_j = 0$ caso contrário. O fator de balanceamento α indica a confiança nos rótulos disponíveis a priori. Caso os rótulos disponíveis não sejam confiáveis, esse valor deve ser próximo de 0, e se for totalmente confiável, deve ser 1. Caso α seja 0, a função objetivo se torna a mesma função do Fuzzy C-Means (FCM) com o termo de fuzzificação $m = 2$. O termo t_{ij} é uma função indicativa que fornece a informação do rótulo utilizada na função objetivo. O valor de t_{ij} é igual a 1 quando o padrão $\mathbf{x_i}$ e o padrão $\mathbf{x_j}$ pertencerem à mesma classe l, e $t_{il} = 0$, caso contrário.

$$t_{ij} = \begin{cases} 1 & se\ (\mathbf{x_i}\ and\ \mathbf{x_j})\ \in\ l \\ 0 & senão \end{cases}$$

A função objetivo obedece as restrições apresentadas na equação 2, que também são encontradas no algoritmo FCM.

$$\sum_{k=1}^{C} u_{ik} = 1\ \forall\ i,\ \ 0 < \sum_{i=1}^{N} u_{ik}\ <\ N\ \forall\ k \quad (2)$$

O multiplicador de lagrange é utilizado para otimizar a equação 1 com respeito à matriz U, obtendo as equações para o cálculo dos graus de pertinência u_{ik}, obtendo a equação 3:

$$u_{ik}^{(s)} = \frac{1 + \alpha b_i \left[\left(\sum_{h=1}^{C} \frac{\sum_{j=1}^{N} b_i b_j t_{ij}(u_{jk}^{(s-1)} - u_{jh}^{(s-1)}) d_{ij}^2}{(d_{ih}^{(s)})^2 + \alpha b_i \sum_{j=1}^{N} b_j t_{ij} d_{ij}^2}\right)\right]}{\sum_{h=1}^{C}\left[\frac{(d_{ik}^{(s)})^2 + \alpha b_i \sum_{j=1}^{N} b_j t_{ij} d_{ij}^2}{(d_{ih}^{(s)})^2 + \alpha b_i \sum_{j=1}^{N} b_j t_{ij} d_{ij}^2}\right]} \quad (3)$$

onde s é o contador de iterações. Uma das características desse cálculo é a necessidade do grau de pertinência de uma iteração anterior. Portanto, há necessidade de preencher a matriz U antes de utilizar esta fórmula pela primeira vez. A equação para o cálculo do protótipo é igual a equação utilizada no algoritmo base FCM dada pela equação 4:

$$\mathbf{v_k} = \frac{\sum_{i=1}^{N} u_{ik}^2 \mathbf{x_i}}{\sum_{i=1}^{N} u_{ik}^2} \tag{4}$$

A execução do algoritmo proposto segue o mesmo processo de algoritmos de agrupamento não supervisionados, como o FCM. Assim, o processo de agrupamento semi-supervisionado é descrito a partir do pseudo-código logo abaixo.

1 *Inicialização.* Iniciar os parâmetros do algoritmo: Fixar C, tal que $1 < C \leq N$; Fixar $MaxIter$ (número máximo de iterações); Iniciar contador de iterações $s = 0$; Atribuir um valor para ϵ, tal que $0 < \epsilon << 1$; Iniciar Matriz de Protótipos escolhendo C exemplos diferentes tal que $\mathbf{v_k} \in X = \{x_1, ..., x_N\}(k = 1, ..., C)$; Iniciar Matriz de Graus de Pertinência, incluindo todos os valores de pertinência conhecidos;

2 *Passo 1. Obtendo os melhores protótipos:* Calcular os protótipos $v_k^{(s)}$ da matriz V utilizando a equação 4;

3 *Passo 2. Obtendo as melhores partições:* Atualizar o valor de cada grau de pertinência dos padrões $\mathbf{x_i}$ a cada um dos C grupos: $u_i^{(s)} = (u_{i1}^{(s)}, ..., u_{iC}^{(s)})(i = 1, ..., N)$ da matriz de partição $U^{(s)}$ utilizando a equação 3;

4 *Critério de parada:* Calcular o valor da função objetivo $J^{(s)}$ utilizando a equação 1. **Se** $|J^{(s)} - J^{(s-1)}| \leq \epsilon$ ou $s \geq MaxIter$) e $(s > 1)$ **então** PARE. **Senão** Atualizar $s = s + 1$ e ir para 2 (Passo 1).;

Algoritmo 1: Algoritmo Semi Supervisionado NebFuzz

O algoritmo NebFuzz é uma nova abordagem de algoritmos de agrupamento semi-supervisionado. Uma das características do novo algoritmo é a otimização levando em conta apenas os padrões que são de uma mesma classe, e ainda, a utilização da distância entre estes dois padrões ao invés da distância entre o padrão e o protótipo do grupo. Ainda, o algoritmo leva em conta na função de otimização, que o número de grupos poder ser maior que o número de classes conhecidas previamente.

4. Avaliação do Algoritmo

4.1. Base de Dados

Neste artigo, apresentamos a avaliação dos algoritmos para 3 bases de dados, sendo 2 delas tiradas do repositório digital UCI (*http://archive.ics.uci.edu/ml/*) e uma base de dados sintética. As bases de dados retiradas do UCI foram: câncer de mama e iris. A base de dados sintética foi gerada utilizando uma distribuição normal com os parâmetros média e desvio padrão (μ, σ) apresentadas na Tabela 1. A primeira classe p_1 consiste de um grupo e a segunda classe p_2 consiste de dois grupos descritas por dois atributos. Cada grupo é formado por 100 padrões.

Tabela 1. Base de dados Sintética

Características		μ_1	μ_2	σ_1	σ_2
Classes	p_1	3.0	1.0	7.0	0.5
	p_2	4.0	4.5	1.0	1.0
		2.0	-2.5	1.0	1.0

4.2. Validação Cruzada

Um dos métodos para avaliar métodos de aprendizagem é a validação cruzada com k *folds* [Mitchel 1997]. Esse método consiste numa divisão da base de dados em k grupos de tamanhos aproximadamente iguais. Os padrões contidos nos grupos $k-1$ são utilizados para treinar o algoritmo e os padrões do grupo k são utilizados para testar o algoritmo. Esse processo é repetido para todas as combinações de $k-1$ grupos. O desempenho do algoritmo é medido através das médias das taxas de erro dos resultados das k repetições.

Na aprendizagem não supervisionada, quando existe uma classificação prévia disponível do conjunto de dados, a relevância estatística da diferença entre médias de taxa de erros de classificação do conjunto de testes também pode ser utilizada [Costa et al. 2003]. O conjunto de treinamento é apresentado ao método de agrupamento, o resultado é uma partição de treinamento. Depois, a técnica de centróide mais próximo é utilizada para construir um classificador a partir da partição de treinamento. A técnica de centróide mais próximo calcula a proximidade de cada padrão do conjunto de teste ao protótipo, ou o centro, de cada agrupamento formado na partição de treinamento. Desta forma, cada padrão do conjunto de teste é atribuído ao agrupamento cuja proximidade seja mínima. Depois, o conjunto de teste é comparado com a partição prévia utilizando um índice externo.

Para a aprendizagem semi-supervisionada com métodos baseados em agrupamento, adaptamos a metodologia da validação de métodos não supervisionados. A técnica de centróide mais próximo é adaptada. No trabalho original, proposto por Costa [Costa et al. 2003], o padrão pertence ao grupo cuja similaridade for maior. A similaridade é calculada utilizando a forma original do algoritmo, ou seja, pela equação do algoritmo que calcula a similaridade entre o exemplo e o protótipo. No entanto, devido a característica *fuzzy*, um exemplo pertence a todos os grupos, quantificado pelo grau de pertinência do exemplo para cada grupo. Assim, o exemplo é atribuído ao grupo cujo grau de pertinência for maior.

4.3. Taxa de Acerto

A saída obtida com métodos de agrupamento é composta por estruturas como partições ou hierarquias, que não podem ser diretamente utilizadas para classificar outros objetos. Assim, é importante saber qual grupo representa cada classe para confirmar se o algoritmo atribuí o padrão para a classe correta. Depois dos grupos formados pelo algoritmo, uma matriz de confusão no modelo da Tabela 2 é construída. Seja $V = v_1, ..., v_k, ..., v_C$ o conjunto de C grupos, $P = p_1, ..., p_l, ..., p_H$ o conjunto de H classes e a entrada n_{kl}, o número de objetos que estão tanto no grupo k quanto na classe p_l. O termo $n_{l\bullet}$ é a soma dos padrões contidos na classe p_l e $n\bullet k$ representa o número de objetos no grupo k e n é o número total de objetos nas partições. A matriz de confusão é caracterizada pelas colunas que representam o grupo atribuído pelo algoritmo e a linha, que representa o grupo ao qual

Tabela 2. Tabela de confusão

Classes	Grupos					
	v_1	$\cdots$	v_k	$\cdots$	v_C	$\sum$
p_1	n_{l1}	$\cdots$	n_{1k}	$\cdots$	n_{1C}	$n_{1\bullet} = \sum_{k=1}^{C} n_{1k}$
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$
p_l	n_{l1}	$\cdots$	n_{lk}	$\cdots$	n_{lC}	$n_{l\bullet} = \sum_{k=1}^{C} n_{lk}$
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$
p_H	n_{H1}	$\cdots$	n_{Hk}	$\cdots$	n_{HC}	$n_{H\bullet} = \sum_{k=1}^{C} n_{Hk}$
$\sum$	$n_{\bullet 1} = \sum_{l=1}^{H} n_{l1}$	$\cdots$	$n_{\bullet k} = \sum_{l=1}^{H} n_{lk}$	$\cdots$	$n_{\bullet C} = \sum_{l=1}^{H} n_{lC}$	$n = \sum_{l=1}^{H} \sum_{k=1}^{C} n_{lk}$

o padrão pertence originalmente. A taxa de acerto é obtida pela equação 5. O conjunto π_l contém todos os grupos que representam a classe p_l. Um grupo k representa apenas uma classe. A classe representada será aquela que tiver mais objetos neste grupo k.

$$Tx = \frac{\sum_{k \in \pi_l}^{C} \max_{1 \leq l \leq H} n_{lk}}{n} \tag{5}$$

4.4. Índice Externo - Índice de Rand Corrigido

É preciso medir a qualidade da partição gerada pelo algoritmo de agrupamento semi-supervisionado. Índices externos são utilizados para avaliar o grau de concordância entre duas partições (U e V), onde a partição U é o resultado do método de agrupamento e a partição V é gerada por uma informação prévia, independente da partição U, como uma classe[Jain and Dubes 1988].

O índice de Rand corrigido [Costa et al. 2003] possui seus valores corrigidos de acordo com acertos nas comparações entre as partições.O índice de Rand corrigido pode assumir valores em [-1,1], onde o valor 1 indica uma coesão perfeita entre as partições, enquanto valores próximos de 0 (ou negativos) correspondem a uma coesão encontrada por acaso. A equação 6 apresenta o índice corrigido de Rand levando em conta a matriz de confusão apresentada na Tabela 2.

$$CR = \frac{\sum_{l}^{H} \sum_{k}^{C} \binom{n_{lk}}{2} - \binom{n}{2}^{-1} \sum_{l}^{H} \binom{n_{l\bullet}}{2} \sum_{k}^{C} \binom{n_{\bullet k}}{2}}{\frac{1}{2}[\sum_{l}^{H} \binom{n_{l\bullet}}{2} + \sum_{k}^{C} \binom{n_{\bullet k}}{2}] - \binom{n}{2}^{-1} \sum_{l}^{H} \binom{n_{l\bullet}}{2} \sum_{k}^{C} \binom{n_{\bullet k}}{2}} \tag{6}$$

5. Experimentos e Resultados

Foram executados 2 tipos de experimentos com algoritmos de agrupamento semi-supervisionado nas tarefas de agrupamento e classificação. Os experimentos foram baseados nos trabalhos de Pedrycz e Waletzky [Pedrycz and Waletzky 1997] e de Amini e Gallinari [Amini and Gallinari 2005]. Nesses dois experimentos, os padrões da base de dados são divididos em 2 conjuntos, o conjunto de dados rotulados e o conjunto de dados não rotulados. O conjunto de dados rotulados contém uma certa porcentagem de exemplos rotulados que varia nas seguintes proporções: $(0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)$, do total da base. O restante do conjunto de dados faz parte do segundo conjunto, o de dados não rotulados. Assim, para os valores respectivos de base rotulada, esses conjuntos utilizam as porcentagens $(100, 90, 80, 70, 60, 50, 40, 30, 20, 10, 0)$ do total da base. Os experimentos são realizados submetendo os dois conjuntos de dados ao algoritmo de

agrupamento semi-supervisionado com as devidas proporções de cada um dos conjuntos. Desse modo, observamos a influência da quantidade de dados rotulados e não rotulados no desempenho dos algoritmos. Os algoritmos comparados foram o algoritmo NebFuzz, o algoritmo ISODATA-PS [Pedrycz and Waletzky 1997] e o algoritmo SSC [Bouchachia and Pedrycz 2006].

O experimento de tarefa de classificação com algoritmos semi-supervisionados emprega a validação cruzada com 10 *folds* para avaliar os algoritmos estudados. A validação cruzada emprega dois conjuntos de dados separados, um para treinamento e outro para teste. Neste experimento foram avaliadas 3 bases de dados: câncer de mama, iris e sintética. A validação cruzada foi repetida 30 vezes. Para cada repetição, a inicialização dos padrões foi repetida 20 vezes para que os parâmetros de inicialização do algoritmo tenham pouca influência no resultado final do algoritmo. Dessas 20 repetições, o resultado escolhido é o da iteração que conseguir obter o menor valor da função objetivo. Desse modo, são gerados os valores dos quais obtemos a média e o desvio padrão para cada uma das configurações do experimento e construímos intervalos com 95% de confiança com a finalidade de comparar os algoritmos estudados.

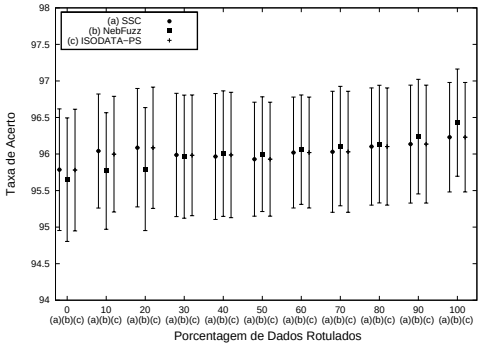
Os resultados dos experimentos para a tarefa de classificação são apresentados na Tabela 3. Na tarefa de classificação os 3 algoritmos tiveram desempenhos muito parecidos para a maioria das configurações de dados rotulados principalmente nas bases de dados câncer de mama e iris. Mesmo com mais dados rotulados, o aumento do desempenho não foi muito alto na base de dados de câncer de mama. Na base de dados iris, o algoritmo proposto NebFuzz obteve o maior ganho no aumento de dados rotulados, também teve o melhor desempenho com todos os dados rotulados. Os algoritmos SSC e ISODATA-PS tiveram um ganho de 2% com a adição de dados rotulados. Na base de dados sintética, o algoritmo proposto obteve os melhores resultados em todas configurações, exceto com 100%, onde não obteve a taxa de acerto máxima. Os algoritmos ISODATA-PS e SSC tiveram desempenhos parecidos nessa base de dados.

Tabela 3. Média da taxa de acerto na tarefa de classificação nas bases de dados câncer de mama, iris e sintética com porcentagens de 0% a 100% de dados rotulados

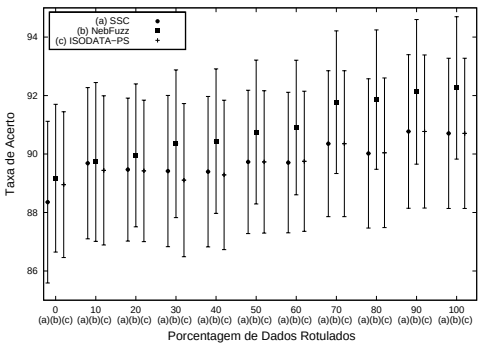
	Câncer			Iris			Sintética		
P	SSC	NebFuzz	ISODATA-PS	SSC	NebFuzz	ISODATA-PS	SSC	NebFuzz	ISODATA-PS
0	95.79	95.65	95.78	88.35	89.17	88.95	79.00	80.70	79.00
10	96.04	95.77	96.00	89.69	89.73	89.44	79.30	82.09	78.70
20	96.09	95.79	96.09	89.47	89.95	89.42	80.91	84.99	79.64
30	95.99	95.96	95.98	89.42	90.35	89.11	85.75	88.40	80.73
40	95.97	96.01	95.99	89.40	90.44	89.29	89.00	91.36	86.70
50	95.93	96.00	95.93	89.73	90.75	89.73	91.92	93.31	91.65
60	96.02	96.06	96.02	89.71	90.91	89.75	93.68	94.50	93.68
70	96.03	96.11	96.03	90.35	91.77	90.35	95.53	95.97	95.53
80	96.10	96.14	96.10	90.02	91.86	90.04	97.30	97.56	97.30
90	96.14	96.24	96.14	90.77	92.13	90.77	98.57	97.81	98.57
100	96.23	96.43	96.23	90.71	92.26	90.71	100.0	97.67	100.0

Os intervalos de confiança do experimento de classificação são apresentados na Figura 1. Podemos perceber quanto os resultados foram equilibrados para praticamente todas as configurações do experimento nas bases de dados de câncer de mama e iris. Na base de dados sintética, o algoritmo proposto obteve os melhores resultados com poucos dados rotulados até 60% de dados rotulados. Os algoritmos SSC e ISODATA-PS tive-

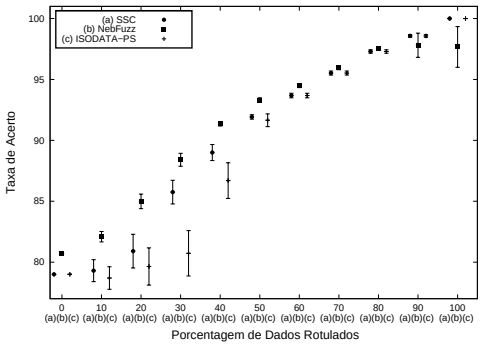
ram desempenhos parecidos, exceto para 30% de dados rotulados onde o algoritmo SSC obteve o segundo melhor desempenho.



(a) Intervalos com 95% de confiança para a base de dados câncer de mama



(b) Intervalos com 95% de confiança para a base de dados iris



(c) Intervalos com 95% de confiança para a base de dados sintética

Figura 1. Intervalos com 95% de confiança para as base de dados câncer de mama, iris e sintética na tarefa de classificação

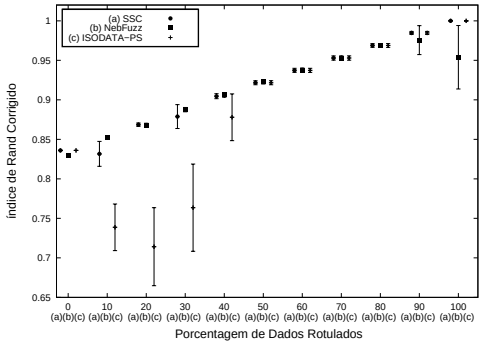
O experimento da tarefa de agrupamento possui como principal característica a utilização de toda a base de dados disponível para o treinamento e para o teste do algoritmo, respeitando as configurações já apresentadas dos conjuntos de dados rotulados e não rotulados. O experimento é repetido 100 vezes para obter uma significância estatística. Para cada iteração do experimento, a inicialização dos padrões foi repetida 20 vezes para que o melhor resultado dessas repetições seja selecionado da iteração cuja função objetivo obter o menor valor em relação aos demais. Aqui, utilizamos o índice de Rand corrigido para validar o desempenho dos algoritmos. Depois, intervalos com 95% de confiança são construídos para comparar o desempenho dos algoritmos de agrupamento semi-supervisionado na tarefa de agrupamento. Apresentamos apenas os resultados do índice de Rand corrigido para as bases de dados câncer de mama, iris e sintética desse experimento na Tabela 4.

O algoritmo proposto NebFuzz obteve os melhores resultados nas bases de dados iris e sintética principalmente quando havia poucos dados rotulados, 0% a 50%. Na base de dados de câncer de mama, os 3 algoritmos tiveram desempenhos muito parecidos para praticamente todas as configurações de dados rotulados do experimento. Na base de dados iris, o algoritmo SSC obteve um resultado melhor que o algoritmo ISODATA-PS.

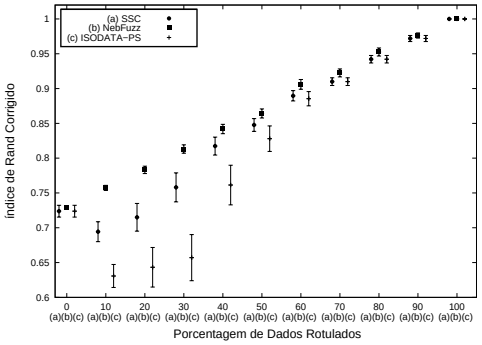
Tabela 4. Média do índice de Rand corrigido global na tarefa de agrupamento nas bases de dados câncer de mama, iris e sintética com porcentagens de 0% a 100% de dados rotulados

	Câncer			Iris			Sintética		
P	SSC	NebFuzz	ISODATA-PS	SSC	NebFuzz	ISODATA-PS	SSC	NebFuzz	ISODATA-PS
0	0.84	0.83	0.84	0.72	0.73	0.72	0.29	0.34	0.29
10	0.83	0.85	0.74	0.69	0.76	0.63	0.32	0.38	0.31
20	0.87	0.87	0.71	0.72	0.78	0.64	0.38	0.47	0.35
30	0.88	0.89	0.76	0.76	0.81	0.66	0.50	0.58	0.37
40	0.90	0.91	0.88	0.82	0.84	0.76	0.60	0.68	0.54
50	0.92	0.92	0.92	0.85	0.86	0.83	0.69	0.74	0.69
60	0.94	0.94	0.94	0.89	0.91	0.89	0.76	0.79	0.76
70	0.95	0.95	0.95	0.91	0.92	0.91	0.83	0.84	0.83
80	0.97	0.97	0.97	0.94	0.95	0.94	0.89	0.90	0.89
90	0.98	0.98	0.98	0.97	0.98	0.97	0.94	0.92	0.94
100	1.00	0.95	1.00	1.00	1.00	1.00	1.00	0.93	1.00

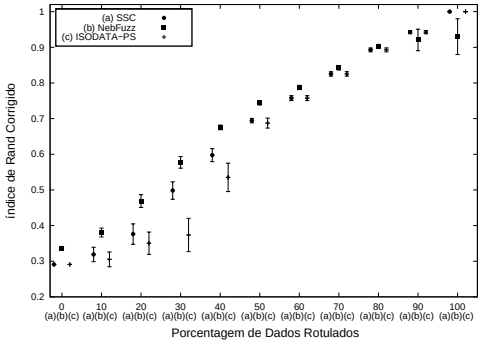
Na base de dados sintética, o algoritmo SSC obteve o segundo melhor desempenho até 40% de dados rotulados, depois os algoritmos ISODATA-PS e SSC tiveram desempenhos similares.



(a) Intervalos com 95% de confiança para a base de dados câncer de mama



(b) Intervalos com 95% de confiança para a base de dados iris



(c) Intervalos com 95% de confiança para a base de dados sintética

Figura 2. Intervalos com 95% de confiança para as base de dados câncer de mama, iris e sintética na tarefa de agrupamento

Intervalos com 95% de confiança para o índice de Rand corrigido são apresentados na Figura 2. Na base de dados de câncer de mama, note que exceto para 10% a 30% de dados rotulados onde o algoritmo ISODATA-PS obteve o pior desempenho, os 3 algo-

ritmos tiveram desempenhos similares. Para a base de dados iris, o algoritmo NebFuzz obteve o melhor resultado com poucos dados rotulados, 10% a 40% de dados rotulados, ficando o algoritmo SSC com o segundo melhor desempenho e o ISODATA-PS com o terceiro melhor. Nas outras configurações os 3 algoritmos empataram em termos de desempenho. Na base de dados sintética, o algoritmo NebFuzz se destacou quando havia poucos dados rotulados, até 60% de dados rotulados disponíveis. Com mais de 60% de dados rotulados, os 3 algoritmos empataram em termos de desempenho. Os algoritmos SSC e ISODATA-PS conseguiram desempenhos parecidos, exceto para 30% e 40% de dados rotulados, onde o algoritmo SSC foi melhor que o algoritmo ISODATA-PS.

Por fim, podemos destacar que o algoritmo NebFuzz de agrupamento semi-supervisionado proposto obteve bons resultados para a maioria das base de dados avaliadas nesse trabalho, principalmente quando havia poucos dados rotulados disponíveis. Também, podemos afirmar que o aumento de dados rotulados disponíveis no treinamento do algoritmo aumenta seu desempenho na grande maioria dos casos.

6. Conclusões

Este trabalho propôs um novo algoritmo de agrupamento semi-supervisionado, chamado NebFuzz, que alcança bons resultados com poucos dados rotulados. Os resultados alcançados pelo algoritmo NebFuzz foram bastante satisfatórios. Para poucos dados rotulados, o novo algoritmo de agrupamento semi-supervisionado obteve o melhor ou esteve entre os melhores desempenhos em relação aos outros algoritmos de agrupamento semi-supervisionado. À medida que mais dados eram disponibilizados para seu treinamento, o algoritmo melhorou seu desempenho na grande maioria dos casos, alcançando desempenho compatível com outros algoritmos semi-supervisionados. Assim, o algoritmo NebFuzz, fruto deste trabalho, pode ser utilizado em aplicações reais onde rotular dados seja custoso.

Como objetivo futuro, pretende-se realizar alterações nessa função de similaridade. As distâncias adaptativas permitem a construção de partições em diversos formatos, além do padrão circular, padrão formado pela distância euclidiana, e desse modo pode aprender estruturas mais complexas de distribuições de dados. Assim, pretende-se utilizar distâncias adaptativas com a finalidade de melhorar o desempenho do algoritmo proposto.

Agradecimentos.

Gostaríamos de agradecer pelo suporte financeiro das agências brasileiras CNPq e FACEPE.

Referências

Amini, M. R. and Gallinari, P. (2005). Semi-supervised learning with an imperfect supervisor. *Knowledge and Information Systems*, 8:385–413.

Bezdek, J. (1981). *Pattern Recognition With Fuzzy Objective Function Algorithms*. Plenum.

Bouchachia, A. (2007). Learning with parly data. *Neural Computing and application*, (16):267–293.

Bouchachia, A. and Pedrycz, W. (2006). Data clustering with partial supervision. *Data Mining and Knowledge Discovery*, (12):47–78.

Chapelle, O., Zien, A., and Scholkopf, B. (2006). *Semi-supervised learning*. MIT Press.

Costa, I. G., Carvalho, F. A. D., and de Souto, M. C. (2003). Comparative study on proximity indices for cluster analysis of gene expression time series. *Journal of Intelligent & Fuzzy Systems*, 13:133–142.

Hathaway, R. J., Bezdek, J., and Hu, Y. (2000). Generalized fuzzy c-means clustering strategies using lp-norm distances. *IEEE Transaction on Fuzzy Systems*, 8(5):576–582.

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice Hall, New Jersey.

Mitchel, T. (1997). *Machine Learning*. McGraw Hill.

Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. (2000). Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39:103–134.

Pedrycz, W. and Waletzky, J. (1997). Fuzzy clustering with partial supervision. *IEEE transactions on system, man and cybernetics*, 27(5).

Stepp, R. E. and Michalski, R. S. (1986). *Machine Learning: An Artificial Intelligence Approach*, volume 2, chapter Conceptual Clustering: Inventing Goal-Oriented Classifictions of Structured Objects, pages 471–478. Morgan Kaufmann.

Zhu, X. (2008). *Semi-Supervised Learning Literature Survey*. Carnegie Mellon University.

On the Use of Fuzzy ART and SOM Networks in Ensemble Classifiers: A Performance Comparison

César L. C. Mattos[1] **, Guilherme A. Barreto**[1]

[1]Federal University of Ceará, Department of Teleinformatics Engineering
Av. Mister Hull, S/N, Campus of Pici, Center of Technology
CP 6005 – CEP 60455-970 – Fortaleza – Ceará – Brazil

cesar@lesc.ufc.br, guilherme@deti.ufc.br

Abstract. *In this paper we introduce two novel ensemble models that are built using Fuzzy ART (FA) and SOM networks as base classifiers. For this purpose, we first describe three different strategies to convert these unsupervised competitive learning algorithms to supervised ones to allow them to be applied to pattern classification tasks. Then, a metaheuristic solution based on a hybrid PSO algorithm is devised for parameter optimization of the proposed ensemble classifiers. A comprehensive performance comparison using 10 benchmarking data sets indicates that the FA- and SOM-based ensemble classifiers consistently outperform ensembles built from standard supervised neural networks, such as the Fuzzy ARTMAP and the Extreme Learning Machine.*

1. Introduction

Neural classifiers are typically built from supervised artificial neural networks (ANNs), such as the MLP and RBF Networks [Haykin 2008], ARTMAP networks [Carpenter et al. 1991a] and, more recently, the Extreme Learning Machine (ELM) [Huang et al. 2006].

In contrast to learning stand-alone classifiers, Ensemble Learning (EL) has been studied as an alternative to improve the generalization power of single classifiers [Dietterich 2000]. The ensemble's output is obtained by combining the outputs of a set of single classifiers. The result is an ensemble learning method which can reduce both the bias and the variance of single learning algorithms and guarantee an error less than or equal to the average quadratic error of the individual classifiers [Krogh and Vedelsby 1995].

Unsupervised competitive learning ANNs, such as the SOM [Kohonen 1982], the ART2 [Carpenter and Grossberg 1987a] and Fuzzy ART [Carpenter et al. 1991b] networks, for not requiring labeled data for training, are commonly applied to data clustering, vector quantization for signal/image compression, and dimensionality reduction tasks. Available algorithms in both SOM and ART families are considered to be unstable algorithms and *weak* learners (indeed, these are typical features of ANNs in general), which are two base preconditions necessary for building efficient ensembles [Hansen and Salamon 2002]. However, the development of SOM- and ART-based ensemble models is still in its first infancy.

SOM-based classification ensembles frequently proceed with the following steps [Corchado et al. 2007, Petrikieva and Fyfe 2002]: after training several SOMs in the

usual unsupervised way, the training patterns are presented once more to the individual maps in order to tag each neuron with the more frequent class label among those mapped to that neuron. Then, each individual SOM in the ensemble outputs the class label associated with the corresponding best-matching unit for a given input pattern. Finally, a majority voting rule is adopted in order to choose the final decision of the ensemble.

ART-based ensemble models are less common. This can be explained partly due to the high number of parameters, such as the vigilance and the choice parameters, that need to be set up in advance for this type of ANN architecture. Anyway, the works found in the literature survey carried out for this research used ARTMAP networks as base classifiers [Santos and Canuto 2008, Loo et al. 2006]. It was not found previous works using ART networks. It is worth remembering that learning in ART networks is unsupervised, while for ARTMAP networks learning is supervised.

From the exposed so far, the main contributions of the present paper is threefold. The first one involves a comprehensive evaluation of the influence of three different supervised variants of the SOM network in the performances of SOM-based ensemble classifiers. This approach gives rise to the MUSCLE (Multiple SOM Classifiers in Ensembles) models. The second contribution consists in developing new techniques, inspired by the supervised variants of the SOM, for building ART-based ensemble models. This approach gives rise to the ARTIE (ART networks in Ensembles) models. The third contribution is related with the proposal of a hybrid Particle Swarm Optimization (PSO) technique for tuning the parameters of the ARTIE and MUSCLE models.

The proposed models are compared with ensembles of two popular supervised algorithms (Fuzzy ARTMAP and ELM) using 10 benchmarking data sets. The obtained results strongly suggest that the performances of ARTIE and MUSCLE architectures are comparable to those achieved by ensembles of standard supervised neural networks.

The remaining part of this paper is organized as follows. In Section 2 we present a brief overview on the basic operations of the Fuzzy ART and SOM networks. In Section 3 we introduce the ARTIE and MUSCLE models, as well as the I-HPSO algorithm. In Section 4 we present and discuss the obtained results for 10 benchmarking classification datasets. The paper is concluded in Section 5.

2. Basics of Fuzzy ART and SOM Networks

This section summarizes the operations of Fuzzy ART and SOM networks as unsupervised learning algorithms. Both networks are comprised of one layer of neurons and their corresponding weight (prototype) vectors and the training process follows the principles of competitive learning. According to this learning paradigm, neurons “compete” in order to find groups of similar unlabeled input patterns (clustering) or, equivalently, to achieve a compact representation of the input patterns (vector quantization).

2.1. The Fuzzy ART Network (FA)

The main idea behind any ART architecture is that, if an input pattern is different enough from the patterns already stored in the long-term memory (i.e. weights) of the network, then create a new category (cluster) and associate the “different input pattern” to it [Keskin and Özkan 2009]. This novelty detection mechanism is highly effective in identifying abnormal (outliers) patterns in the data [Barreto and Aguayo 2009].

The FA network extends ART1, which was originally designed to process only binary data [Carpenter and Grossberg 1987b], being able to also learn analog inputs. The initial network contains only one neuron with its weights all set to 1. Let $\mathbf{a}(k) = [a_1(k) \quad a_2(k) \quad \cdots \quad a_p(k)]^T$, a p-dimensional input pattern at the k-th learning iteration. Then, the FA network algorithm follows the steps below.

Step 1 Complement code the p-dimensional input pattern $\mathbf{a}(k)$ into a $2p$-dimensional vector $\mathbf{x}(k) \in \mathbb{R}^{2p}$: $\mathbf{x}(k) = [\mathbf{a}(k) \quad \mathbf{a}^c(k)]^T$, where $a_i^c(k) = 1 - a_i(k), \quad \forall i \in \{1, 2, \ldots, p\}$.

Step 2 Present $\mathbf{x}(k)$ to the first layer of the network, L_1, and compute the activations $T_j(k), \forall j \in \{1, \ldots, n\}$, where n is the number of neurons of the network. For each neuron j, associated with a weight vector $\mathbf{w}_j(k) \in \mathbb{R}^{2p}$, compute the corresponding activation $T_j(k)$ as follows:

$$T_j(k) = \frac{|\mathbf{x}(k) \wedge \mathbf{w}_j(k)|}{\beta + |\mathbf{w}_j(k)|}, \quad \forall j \in \{1, 2, \ldots, n\}, \tag{1}$$

where $|\mathbf{x}(k)| = \sum_{i=1}^{2p} |x_i(k)|$ is the sum of the absolute value of the components of $\mathbf{x}(k)$, the symbol $\wedge$ denotes the fuzzy minimum operator and β is a small positive real number known as the choice parameter.

Step 3 Find the index of the winning neuron j^*, i.e. the one with maximum activation:

$$j^*(k) = \arg\max_{\forall j} \{T_j(k)\}. \tag{2}$$

Step 4 Check if the winning neuron j^* satisfies the vigilance criterion, i.e. if $|\mathbf{x}(k) \wedge \mathbf{w}_{j^*}(k)| \,/\, |\mathbf{x}(k)| \geq \rho$, where $0 < \rho < 1$ is the vigilance parameter. If the vigilance test is satisfied, go to Step 5; otherwise, reset the activation of neuron j^* to zero (i.e. $T_{j^*}(k) = 0$) and go to Step 3. The search is repeated until a neuron passes the vigilance test or all neurons have been tested. If all neurons fail to pass the vigilance test, create a new category using the current input pattern as its prototype vector and go to Step 1.

Step 5 Update the weight vector of the winning neuron:

$$\mathbf{w}_{j^*}(k+1) = \eta \left(\mathbf{w}_{j^*}(k) \wedge \mathbf{x}(k)\right) + (1 - \eta)\mathbf{w}_{j^*}(k), \tag{3}$$

where $0 < \eta \leq 1$ is the learning rate. Go to Step 1.

Steps 1 to 5 are repeated for each training pattern. In this paper just one training epoch (one cycle through the training set) is used, simulating an on-line operation. It is important to note that after the training phase each neuron defines a cluster by a hyperbox [Carpenter et al. 1991b]. The testing process is done through a simple winner-take-all competition based on Eq. (2).

2.2. The Self-Organizing Map (SOM)

Introduced by [Kohonen 1982], the SOM learns from examples a mapping (projection) from a high-dimensional continuous input space $\mathcal{X}$ onto a low-dimensional discrete space (lattice) $\mathcal{Z}$ of n neurons which are arranged in fixed topological forms, e.g. as a rectangular 2-dimensional array. The map $j^*(\mathbf{x}) : \mathcal{X} \rightarrow \mathcal{Z}$, defined by the weight matrix

$\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_q), \mathbf{w}_j \in \mathbb{R}^p \subset \mathcal{X}$, assigns to each input vector $\mathbf{a}(k) \in \mathbb{R}^p \subset \mathcal{X}$ a winning neuron $j^*(k) \in \mathcal{Z}$, determined by

$$j^*(k) = \arg\min_{\forall j} \|\mathbf{a}(k) - \mathbf{w}_j(k)\|, \tag{4}$$

where $\|\cdot\|$ denotes the Euclidean distance and k is the current iteration of the algorithm.

The weight vector of the current winning neuron and the ones of its neighboring neurons are simultaneously adjusted according to the following learning rule:

$$\mathbf{w}_j(k+1) = \mathbf{w}_j(k) + \alpha(k)h(j^*, j; k)[\mathbf{a}(k) - \mathbf{w}_j(k)] \tag{5}$$

where $0 < \alpha(k) < 1$ is the learning rate and $h(j^*, j; k)$ is a weighting function which limits the neighborhood of the winning neuron. A usual choice for $h(j^*, j; k)$ is given by the Gaussian function:

$$h(j^*, j; k) = \exp\left(-\frac{\|\mathbf{r}_j - \mathbf{r}_{j^*}\|^2}{2\sigma^2(k)}\right), \tag{6}$$

where $\mathbf{r}_j$ and $\mathbf{r}_{j^*}$ are respectively, the coordinates of the neurons j and j^* in the output array, and $\sigma(k) > 0$ defines the radius of the neighborhood function at iteration k. The variables $\alpha(k)$ and $\sigma(k)$ should both decay with time to guarantee stable convergence of the weight vectors. In this paper, we adopt an exponential decay for both variables:

$$\alpha(k) = \alpha_0 \left(\frac{\alpha_T}{\alpha_0}\right)^{(k/T)} \quad \text{and} \quad \sigma(k) = \sigma_0 \left(\frac{\sigma_T}{\sigma_0}\right)^{(k/T)} \tag{7}$$

where α_0 (σ_0) and α_T (σ_T) are the initial and final values of $\alpha(k)$ ($\sigma(k)$), respectively.

Weight adjustment is performed until a steady state of global ordering of the weight vectors has been achieved. In this case, we say that the map has converged. The resulting map also preserves the topology of the input samples in the sense that adjacent patterns are mapped into adjacent regions on the map. Due to this topology-preserving property, the SOM is able to cluster input information and spatial relationships of the data on the map. In this paper, the SOM is trained for one epoch only. The testing phase is similar to FA networks, with a winner-take-all competition based on Eq. (4).

3. The Proposed Approaches

The MUSCLE model extends previous SOM-based EL models [Corchado et al. 2007, Petrikieva and Fyfe 2002] by developing two other ways of designing SOM-based EL classifiers. Previous ART-based EL models [Santos and Canuto 2008, Loo et al. 2006] are built using ARTMAP networks, which are supervised classifiers. The ARTIE model, inspired by the strategies used by the MUSCLE models, is a novel approach for building ART-based EL classifiers from the FA networks. Finally, a novel hybrid PSO method is proposed for parameter optimization of the base classifiers used by the MUSCLE and ARTIE models. More details are given in the following sections.

3.1. Adapting FA and SOM for Classification Problems

In order to use FA and SOM networks for supervised classification, some modifications are necessary in their original learning algorithms. The present paper will focus on three strategies for designing FA- and SOM-based classifiers. These strategies, identified by a suffix Ci, i $\in \{1, 2, 3\}$, have been used to design SOM-based classifiers. However, to the best of our knowledge, it is the first time they are used to design FA-based classifiers.

3.1.1. Strategy C1: Post-Training Neuron Labeling

In this approach the SOM and FA networks are trained in the usual unsupervised way. Afterwards, a neuron labeling process is carried out by presenting the training dataset once again to the corresponding network and determining the winning neuron for each pattern vector, according to Eq. (4) for SOM networks or Eq. (2) for FA networks. The label of the winning neuron is assigned on a majority voting basis. Ties can be broken by random selection of the competing labels.

During the testing phase, the winning neuron for an unknown pattern is searched through Eq. (4) for SOM networks, or through Eq. (2) for FA networks. The predicted class for the current input pattern is then the class associated with the winning neuron.

Regarding the SOM, this strategy has been used for the design of single classifiers [Wyns et al. 2004, Christodoulou et al. 2003]; however, it seems that it was never tested before with the FA network.

3.1.2. Strategy C2: One Network per Class

The second strategy uses one SOM (or FA) network for each available class; for instance, if three classes of data are available, three SOMs (or three FAs) will be trained, one for each class. The several SOMs (or FAs), however, are trained independently, using only the data vectors of the class it represents. There is no need for the several SOMs (or FAs) to have the same number of neurons, unless for the sake of simplicity. During testing, the winning neuron is searched among the neurons of all available SOM (or FA) networks, so that its class label is assigned to the current input vector.

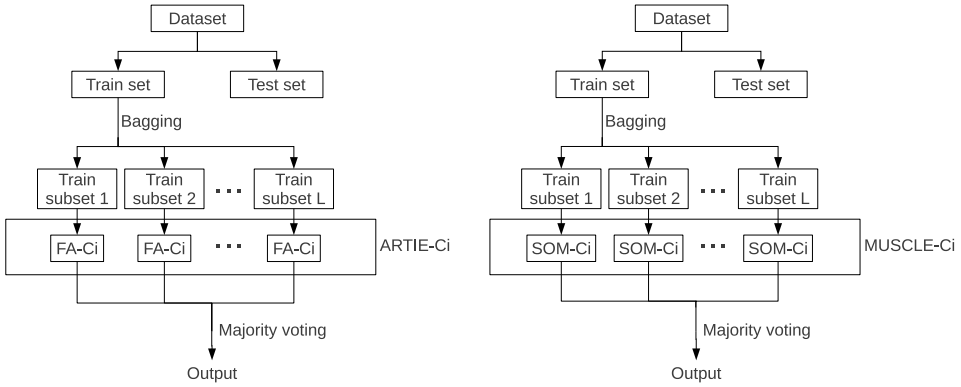
As occurs for the Strategy C1, the Strategy C2 has been used before for the design of SOM-based classifiers [Souza Júnior et al. 2011, Biebelmann et al. 1996]; however, it seems that it was never tested before with the FA network.

3.1.3. Strategy C3: Augmenting Input Space Dimension with Class Labels

In the third strategy, the SOM (or FA) network is made supervised by adding class information to each input data vector. Specifically, the input vectors $\mathbf{x}(k)$ are now formed of two parts, $\mathbf{x}_p(k)$ and $\mathbf{x}_l(k)$, where $\mathbf{x}_p(k)$ is the pattern vector itself and $\mathbf{x}_l(k)$ is the corresponding class label of $\mathbf{x}_p(k)$. During training, these vectors are concatenated to build augmented vectors $\mathbf{x}(k) = [\mathbf{x}_p(k)\ \mathbf{x}_l(k)]^T$ which are used as inputs to the SOM (or FA) network. The corresponding augmented weight vectors, $\mathbf{w}_j(k) = [\mathbf{w}_j^p(k)\ \ \mathbf{w}_j^l(k)]^T$, are adjusted as in the usual SOM/FA training procedure.

During recognition of an unknown vector $\mathbf{x}(k)$, only its $\mathbf{x}_p$ part is compared with the corresponding part of the weight vectors. Then, the class label of the unknown pattern vector is decided on the basis of the $\mathbf{w}_{i^*}^l(k)$ part of the winning weight vector $\mathbf{w}_{i^*}(k)$. The index of the component of $\mathbf{w}_{i^*}^l(k)$ with highest value defines the class label of $\mathbf{x}(k)$.

As occurs for the Strategies C1 and C2, the Strategy C3 has been used before for the design of SOM-based classifiers [Xiao et al. 2005, del-Hoyo et al. 2003, Kohonen 1988]; however, it seems that it was never tested before with the FA network.



(a) Block diagram of the ARTIE model. (b) Block diagram of the MUSCLE model.

Figure 1. ARTIE and MUSCLE models.

3.2. ARTIE: *ART* networks *I*n *E*nsembles

The Fuzzy ARTMAP (FAM) network is a supervised learning method that utilizes FA as building block. For being a supervised classifier, FAM is a natural choice for the base classifier in building EL models [Santos and Canuto 2008, Loo et al. 2006]. However, as described previously, the Strategies C1, C2 and C3 can be used to turn the FA network into a supervised classifier. This way, FA networks can also serve as base learners for building EL models for classification. This approach gives rise to the ARTIE model.

The immediate advantage in using FA-based classifiers instead FAM-based as base learners is related with the computational costs. Training an ARTIE model requires half the number of FA networks required by an ensemble of ARTMAP-like networks and, hence, half the number of parameters. Since FA is simpler than FAM, building ARTIE classifiers is expected to be much easier.

Each one of ARTIE's L base classifiers are trained independently using different subsets of the original training set. Theses subsets are created using the Bagging[1] (Bootstrap Aggregating) method with the objective of generating diversity [Breiman 1996]. Figure 1(a) shows the block diagram of the ARTIE model. The output of ARTIE is decided with a simple majority voting process within the results of the single classifiers.

It is necessary to determine some parameters of FA networks for their use in ARTIE models, namely: the vigilance parameter ρ, the choice parameter β and the learning step η. In order to find almost optimal values for these problem-specific parameters, a hybrid PSO strategy is introduced in Section 3.4.

3.3. MUSCLE: *MU*ltiple *S*OM *CL*assifiers in *E*nsembles

As mentioned in Section 1, the SOM network has been been used in EL classifiers [Corchado et al. 2007, Petrikieva and Fyfe 2002]. For this purpose, the Strategy C1 has been the chosen one. However, Strategies C2 and C3 can also be used for the same purpose. Thus, in doing so, we aimed at extending the range of alternative ways of building SOM-based EL classifiers. This approach gives rise to the MUSCLE model.

[1]Bagging is carried out by sampling (with replacement) training examples, forming new training sets, usually with the same size of the original one. For a training set of N samples and N being large enough, this procedure causes each sample to have a probability of $\left(\frac{N-1}{N}\right)^N \approx 0.368$ of not being chosen.

MUSCLE's base classifiers have the same dimensions and parameters. Diversity is obtained through random weight initialization and using the Bagging for data set selection. The final result is obtained by a simple majority voting among the base classifiers, as with the ARTIE model. Figure 1(b) shows the block diagram of the MUSCLE model.

As the ARTIE model, the MUSCLE model also needs some parameters to be tuned. The main parameters are the dimensions of the maps (P_1 and P_2), the initial and final learning steps (η_0 and η_f) and the initial and final spread parameters (σ_0 and σ_f). In order to find almost optimal values for all these problem-specific parameters, a hybrid PSO strategy is introduced in the next section.

3.4. An Improved Hybrid Particle Swarm Optimization

The Particle Swarm Optimization (PSO) method [Kennedy and Eberhart 1995] is inspired by the social behavior and auto-organization of bird flocking and fish schooling. The exchange of information among the population generates exploration for better solutions, while the individual learning corresponds to the exploitation component, providing a combination of global and local search to the algorithm [Pedersen and Chipperfield 2010].

In [He and Wang 2007] it is proposed a variant of the original PSO algorithm, called Hybrid Particle Swarm Optimization (HPSO), by adding feasibility-based rules for handling constraints and a local search step based on Simulated Annealing (SA). In this paper, for the sake of parameter optimization of the base classifiers of the ARTIE and MUSCLE models, we incorporate the features of the HPSO algorithm into the PSO Standard 2007 [Bratton and Kennedy 2007], resulting in an improved version of the HPSO algorithm, called from now onwards, the Improved HPSO (I-HPSO) algorithm.

Let $\mathbf{x}_i \in \mathbb{R}^d$ and $\mathbf{v}_i \in \mathbb{R}^d$ be, respectively, the position and velocity vectors of the i-th element in a swarm of particles, where d is the number of variables of the problem. Let also $\mathbf{p}_i \in \mathbb{R}^d$ and $\mathbf{pl}_k \in \mathbb{R}^d$ be, respectively, the vectors of best historical individual position of the i-th particle and the best historical position of the neighborhood k.

For a given objective function $f(\cdot)$, the I-HPSO algorithm is implemented according the following steps for a total of L_{PSO} generations:

Step 1 Let $m = 1$ and do $\mathbf{x}_i(0) = \mathbf{x}_{min} + (\mathbf{x}_{max} - \mathbf{x}_{min})\mathbf{U}$, $\mathbf{v}_i(0) = (\mathbf{x}_{max} - \mathbf{x}_{min})\mathbf{U} - \mathbf{x}_i(0)$, $\mathbf{p}_i(0) = \mathbf{0}$ and $\mathbf{pl}_k(0) = \mathbf{0}$, where $\mathbf{U}$ is a d-dimensional random vector whose components are uniformly distributed in the interval $[0, 1]$, $\mathbf{0}$ is a d-dimensional null vector and $\mathbf{x}_{max}$ and $\mathbf{x}_{min}$ are, respectively, the maximum and the minimum limits for the components of $\mathbf{x}_i$.

Step 2 Evaluate all the particles of the swarm. The vector $\mathbf{p}_i(m)$ of each particle receives the current position, as well as its associated value for the objective function. The vector $\mathbf{pl}_k(m)$ and its objective value receive the best position and the best objective function among the particles of the neighborhood k.

Step 3 Considering $\mathbf{x}_i(m)$ and $\mathbf{v}_i(m)$ respectively as the position and velocity of the particle i during the current iteration m and k the neighborhood of the i-th particle, calculate the following update equations:

$$\mathbf{v}_i(m+1) = \chi\{\mathbf{v}_i(m) + c_1 r_1[\mathbf{p}_i(m) - \mathbf{x}_i(m)] + c_2 r_2[\mathbf{pl}_k(m) - \mathbf{x}_i(m)]\}, \quad (8)$$

$$\mathbf{x}_i(m+1) = \mathbf{x}_i(m) + \mathbf{v}_i(m+1). \quad (9)$$

where χ is the constriction factor, c_1 and c_2 are positive constants called acceleration coefficients, while r_1 and r_2 are independent random variables uniformly distributed in the interval $[0, 1]$.

Step 4 Evaluate all the particles of the swarm.

Step 5 For each particle i, if $f(\mathbf{x}_i(m)) > f(\mathbf{p}_i(m))$, $\mathbf{p}_i(m)$ is updated with $\mathbf{x}_i(m)$.

Step 6 For each neighborhood k, let $\mathbf{p}_{k_{max}}(m) = \arg\max(f(\mathbf{p}_i(m)))$ for all i referred to particles within the considered neighborhood. If $f(\mathbf{p}_{k_{max}}(m)) > f(\mathbf{pl}_k(m))$, $\mathbf{pl}_k(m)$ is updated with the solution $\mathbf{p}_{k_{max}}(m)$.

Step 7 Randomly select a fraction of the neighborhoods and perform the local search step based on SA in the solutions $\mathbf{pl}_k$ for each selected value of k.

Step 8 Let $m = m + 1$. If $m > L_{PSO}$, stop and output the best $\mathbf{pl}_k(m)$ of all neighborhoods as the best solution found. Otherwise, go to Step 3.

The local search step is performed only over a fraction of the neighborhoods (in this paper only 10%), randomly chosen, to speedup the search process. Let $\mathbf{pl}_k(m) \in \mathbb{R}^d$ be the best solution found in the k-th neighborhood until the generation m and $p_a \in [0, 1]$ be the acceptance probability of a new solution. Let also $t(m)$ be a temperature parameter. The local search is done according the following steps during the generation m of I-HPSO algorithm for a total of L_{SA} iterations:

Step 1 Do $n = 1$ and $\mathbf{pl}'_k = \mathbf{pl}_k(m)$.

Step 2 Generate a new solution using the following equation:

$$\mathbf{x}' = \mathbf{pl}'_k + \eta_{SA}(\mathbf{x}_{max} - \mathbf{x}_{min})\mathbf{N}(\mathbf{0}, \mathbf{I}), \tag{10}$$

where η_{SA} is an incremental step, $\mathbf{x}_{max}$ and $\mathbf{x}_{min}$ are, respectively, the maximum and the minimum allowed values for the variables of a solution and $\mathbf{N}(\mathbf{0}, \mathbf{I})$ is a d-dimensional Gaussian random vector with zero mean vector and the identity matrix as the covariance matrix.

Step 3 Calculate the acceptance probability $p_a = \min\left\{1, \exp\left[\frac{f(\mathbf{pl}'_k) - f(\mathbf{x}')}{t(m)}\right]\right\}$.

Step 4 If $p_a \geq U(0, 1)$, where $U(0, 1)$ is a random number uniformly distributed in the interval $[0, 1]$, do $\mathbf{pl}'_k = \mathbf{x}'$.

Step 5 Do $n = n + 1$. If $n \geq L_{SA}$, stop and do $\mathbf{pl}_k(m + 1) = \mathbf{pl}'_k$, or else, go to Step 2.

The parameter t is initiated with an empirical value [He and Wang 2007]: $t(0) = -\frac{f_{max} - f_{min}}{\ln(0.1)}$, where f_{max} and f_{min} are respectively the maximum and minimum values for the objective function in the initial swarm. Along the generations, t is reduced exponentially, i.e. $t(m + 1) = \lambda t(m)$, where the annealing rate λ satisfies $0 < \lambda < 1$.

In order to apply I-HPSO to parameter optimization, each particle of the swarm is formed by a vector of values for the tunable parameters and the objective function $f(\cdot)$ is chosen to be the accuracy obtained by the classifier with a possible set of parameters.

4. Simulations and Discussion

Tests with 10 real-world benchmarking datasets were carried out. We used 9 UCI datasets (Balance, Breast-w, Dermatology, Glass, Heart, Sonar, Vehicle, Wall-Following, Zoo) [Frank and Asuncion 2010] and the vertebral column pathologies dataset described in [Rocha Neto and Barreto 2009], named henceforth VCP dataset, which is available upon request. A summary of the evaluated datasets are presented in Table 1.

Table 1. Summary of the benchmarking datasets.

	Instances	Features	Classes
Balance	625	4	3
Breast-w	683	9	2
Dermatology	358	34	6
Glass	214	9	6
Heart	270	13	2
Sonar	208	60	2
VCP	310	6	3
Vehicle	846	18	4
Wall-Following	5456	2	4
Zoo	101	16	7

Table 2. Results from classification problems. The accuracy and standard deviation are percentages.

	Balance	Breast-w	Dermat.	Glass	Heart	Sonar	VCP	Vehicle	Wall-F.	Zoo
ARTIE-C1	86.08 ±5.22	97.23 ±2.20	96.05 ±3.64	65.30 ±10.82	82.22 ±6.49	82.64 ±9.72	77.42 ±6.27	68.60 ±6.24	97.21 ±0.38	**97.09** ±4.69
ARTIE-C2	86.09 ±3.26	96.35 ±2.57	**98.11** ±2.18	73.71 ±8.39	80.0 ±8.76	88.0 ±7.53	80.65 ±6.27	**73.78** ±5.90	**99.95** ±0.09	96.09 ±6.91
ARTIE-C3	84.66 ±4.12	97.09 ±2.72	97.20 ±1.91	73.87 ±5.12	**83.70** ±5.84	84.5 ±11.17	83.87 ±5.89	71.44 ±6.16	99.87 ±0.19	94.09 ±6.94
MUSCLE-C1	85.45 ±3.42	97.22 ±2.23	95.53 ±4.07	72.29 ±9.11	79.63 ±9.44	82.29 ±10.61	**85.81** ±9.40	66.49 ±7.29	95.47 ±0.52	96.09 ±5.05
MUSCLE-C2	86.58 ±3.98	97.08 ±1.92	97.14 ±3.01	70.06 ±3.95	82.59 ±7.82	85.64 ±7.83	84.19 ±6.71	71.10 ±5.35	98.02 ±0.59	94.09 ±6.94
MUSCLE-C3	**90.59** ±3.36	**97.67** ±2.28	98.05 ±1.91	**74.42** ±7.02	78.52 ±7.96	**89.0** ±10.75	85.16 ±6.66	69.45 ±5.91	96.63 ±0.61	94.09 ±5.09
ELM ensemb.	89.96 ±3.36	96.80 ±2.71	96.96 ±2.75	66.72 ±8.44	82.59 ±8.56	77.71 ±7.28	84.84 ±9.26	72.27 ±7.81	94.74 ±1.11	92.09 ±6.30
FAM ensemb.	86.75 ±3.15	96.65 ±2.64	98.0 ±2.35	72.76 ±8.90	79.63 ±8.95	83.36 ±6.36	81.61 ±8.61	72.11 ±6.13	99.73 ±0.28	93.09 ±6.71

For each problem and base classifier a set of parameters was determined using I-HPSO. The optimization process was realized with swarms of 20 particles, a total of 25 generations and 10 iterations of local search. The other parameters of I-HPSO were $c_1 = c_2 = 2.05$, $\chi = 0.72984$, $\lambda = 0.94$ and $\eta_{SA} = 0.001$.

For all simulations, ARTIE-Ci and MUSCLE-Ci, i =1, 2, and 3, are comprised of $L = 10$ base classifiers. The 10 base classifiers are constrained to adopt a single supervision strategy (C1, C2 or C3). This was done in order to have a better idea of the influence of the type of supervision strategy on the EL performance.

For the sake of completeness, performance comparisons between the proposed models and ensembles of ELM and FAM networks are carried out. The ELM/FAM ensembles also used $L = 10$ classifiers and were trained through the Bagging method. Decisions were also made through the majority voting rule.

The performance evaluation procedure follows the approach recommended by Salzberg [Salzberg 1997], which applies 10-fold cross-validation and McNemar's test [Everitt 1977] for comparing the resulting models. The results for each evaluated dataset are shown in the columns of Table 2, where the best values are highlighted in bold.

It is worth noting that ARTIE and MUSCLE presented the best results for all datasets in comparison with ensembles of ELM/FAM classifiers. Indeed, the best among the proposed ARTIE/MUSCLE models performed always better the ensembles

Table 3. McNemar's test for $\alpha = 0.05$. Results are in units of datasets. Winners are in bold. Ties are broken evaluating the best results in Table 2.

Classifier A	Classifier B	A $\approx$ B	A $>$ B	A $<$ B
ELM ensemb.	ARTIE-C1	7	2	1
ELM ensemb.	**ARTIE-C2**	5	2	3
ELM ensemb.	**ARTIE-C3**	6	1	3
ELM ensemb.	**MUSCLE-C1**	7	1	2
ELM ensemb.	**MUSCLE-C2**	7	1	2
ELM ensemb.	**MUSCLE-C3**	8	-	2
FAM ensemb.	ARTIE-C1	7	3	-
FAM ensemb.	**ARTIE-C2**	9	-	1
FAM ensemb.	**ARTIE-C3**	9	-	1
FAM ensemb.	MUSCLE-C1	8	2	-
FAM ensemb.	MUSCLE-C2	9	1	-
FAM ensemb.	**MUSCLE-C3**	8	1	1

of ELM/FAM classifiers. The ARTIE-C2 and MUSCLE-C3 achieved the best overall performances, presenting best accuracy in 30% and 40% of the datasets, respectively.

As a final evaluation, we applied the McNemar's statistical test as described in [Dietterich 1998] for comparing the performances of the classifiers, taken in pairs. The objective is to estimate the probability p of two given classifiers being equivalent (null hypothesis) from a score γ calculated with the McNemar's test. For a significance value of $\alpha = 0.05$, if $p < \alpha$ it is unlikely that two classifiers are equivalent, i.e. the null hypothesis is rejected. In this case, the classifier with better mean accuracy is considered the best for that dataset. Results are summarized in Table 3.

ARTIE-C2, ARTIE-C3 and MUSCLE-C3 models performed better in general than both ELM- and FAM-based ensembles. ARTIE-C1 was the only variant that presented poor results. It is worth mentioning, however, that McNemar's test only claims that two algorithms are different when one of them almost outperforms the other [Dietterich 1998]. Thus, when it suggests that the two classifiers are similar, it is recommended to evaluate the performance metrics shown in Table 2 to choose the best one.

5. Conclusion and Further Work

Although both FA and SOM networks are originally unsupervised competitive learning algorithms, three different techniques were described with the goal of converting them into supervised learning methods. With the supervised variants of the FA and SOM networks at hand, one can build ensemble classifiers with them, giving rise to the ARTIE and MUSCLE models, respectively. In addition, the problem of choosing the appropriate parameters for training base classifiers, for each classification task, was tackled using a metaheuristic approach, which led to the development of the I-HPSO algorithm. A comprehensive performance evaluation was then carried out for 10 different real-world datasets in order to compare the ARTIE and MUSCLE variants with ensembles built from standard supervised classifiers, such as ELM and FAM networks. The obtained results indicates the superior performances of the proposed ARTIE and MUSCLE models.

In this paper we have developed ensembles built using a single supervised variant of the FA and SOM networks. Further work will evaluate the role of diversity of the base classifiers in building efficient ARTIE and MUSCLE models. The goal is to build ensembles whose base classifiers are different variants of the FA and SOM networks.

References

Barreto, G. A. and Aguayo, L. (2009). Time series clustering for anomaly detection using competitive neural networks. In Principe, J. C. and Miikkulainen, R., editors, *Proceedings of the 7th International Workshop on Advances in Self-Organizing Maps (WSOM'09)*, volume LNCS-5629, pages 28–36. Springer.

Biebelmann, E., Köppen, M., and Nickolay, B. (1996). Pratical aplications of neural networks in texture analysis. *Neurocomputing*, 13(2–4):261–279.

Bratton, D. and Kennedy, J. (2007). Defining a standard for particle swarm optimization. In *Proceedings of the IEEE Swarm Intelligence Symposium*, pages 120–127, Honolulu, Hawaii.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Carpenter, G. and Grossberg, S. (1987a). Stable self-organization of pattern recognition codes for analog input patterns. *Applied Optics*, 26:4919–4930.

Carpenter, G., Grossberg, S., and Reynolds, J. H. (1991a). ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4(5):565–588.

Carpenter, G. A. and Grossberg, S. (1987b). A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37(1):54–115.

Carpenter, G. A., Grossberg, S., and Rosen, D. B. (1991b). Fuzzy ART: Fast stable learning, categorization of analog patterns by an adaptive resonance system. *Neural Networks*, 4(6):759–771.

Christodoulou, C. I., Michaelides, S. C., and Pattichis, C. S. (2003). Multifeature texture analysis for the classification of clouds in satellite imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 41(11):2662–2668.

Corchado, E., Baruque, B., and Yin, H. (2007). Boosting unsupervised competitive learning ensembles. In de Sá, J. M., Alexandre, L. A., Duch, W., and Mandic, D. P., editors, *Proceedings of the 17th International Conference on Artificial Neural Networks (ICANN'07), Part I*, volume LNCS 4668, pages 339–348. Springer.

del-Hoyo, R., Buldain, D., and Marco, A. (2003). Supervised classification with associative SOM. In *Proceedings of the 7th International Work-Conference on Artificial and Neural Networks, (IWANN)'03*, pages 334–341.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923.

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40(2):139–157.

Everitt, B. (1977). *The Analysis of Contingency Tables*. Chapman & Hall/CRC.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository.

Hansen, L. K. and Salamon, P. (2002). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):993–1001.

Haykin, S. (2008). *Neural Networks and Learning Machines*. Prentice Hall, 3rd edition.

He, Q. and Wang, L. (2007). A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization. *Applied Mathematics, Computation*, 186(2):1407–1422.

Huang, G. B., Zhu, Q. Y., and Ziew, C. K. (2006). Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1–3):489–501.

Kennedy, J. and Eberhart, R. C. (1995). Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948, Piscataway, NJ, USA.

Keskin, G. A. and Özkan, C. (2009). An alternative evaluation of FMEA: Fuzzy art algorithm. *Quality and Reliability Engineering International*, 25(6):647–661.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69.

Kohonen, T. (1988). The "neural" phonetic typewriter. *Computer*, 21(3):11–22.

Krogh, A. and Vedelsby, J. (1995). Neural network ensembles, cross validation, active learning. *Advances in Neural Information Processing Systems*, pages 231–238.

Loo, C. K., Law, A., Lim, W. S., and Rao, M. V. C. (2006). Probabilistic ensemble simplified fuzzy ARTMAP for sonar target differentiation. *Neural Computing & Applications*, 15(1):79–90.

Pedersen, M. E. H. and Chipperfield, A. J. (2010). Simplifying particle swarm optimization. *Applied Soft Computing*, 10(2):618–628.

Petrikieva, L. and Fyfe, C. (2002). Bagging and bumping self-organising maps. *Computing and Information Systems*, 9(2):69.

Rocha Neto, A. R. and Barreto, G. A. (2009). On the application of ensembles of classifiers to the diagnosis of pathologies of the vertebral column: A comparative analysis. *IEEE Latin America Transactions*, 7(4):487–496.

Salzberg, S. L. (1997). On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery*, 1(3):317–328.

Santos, A. M. and Canuto, A. M. P. (2008). Using ARTMAP-based ensemble systems designed by three variants of boosting. In *Proceedings of the International Conference on Artificial Neural Networks (ICANN'08)*, pages 562–571.

Souza Júnior, A. H., Barreto, G. A., and Varela, A. T. (2011). A speech recognition system for embedded applications using the SOM and TS-SOM networks. In Mwasiagi, J. I., editor, *Self-Organizing Maps - Applications and Novel Algorithm Design*. InTech.

Wyns, B., Sette, S., Boullart, L., Baeten, D., Hoffman, I. E. A., and De Keyser, F. (2004). Prediction of diagnosis in patients with early arthritis using a combined Kohonen mapping and instance-based evaluation criterion. *Artificial Intelligence in Medicine*, 31(1):45–55.

Xiao, Y.-D., Clauset, A., Harris, R., Bayram, E., Santago, P., and Schmitt, J. D. (2005). Supervised self-organizing maps in drug discovery. 1. robust behavior with overdetermined data sets. *Journal of Chemical Information and Modeling*, 45(6):1749–1758.

On Rule Generation Approaches for Genetic Fuzzy Systems

Marcos E. Cintra[1], **Maria C. Monard**[1], **Heloisa A. Camargo**[2]
Trevor P. Martin[3], **Andrei Majidian**[3]

[1]University of Sao Paulo (USP) – Mathematics and Computer Science Institute
CEP13.566-590, São Carlos - SP, Brazil

[2]Computer Science Department – Federal University of Sao Carlos
Rod. Washington Luís, Km 235, PO Box 676, 13.565-905, São Carlos - SP

[3]Department of Engineering Mathematics, University of Bristol, BS8 1UB, UK
BT Innovate and Design, Adastral Park, Ipswich, IP5 3RE, UK

{cintra, mcmonard}@icmc.usp.br, heloisa@dc.ufscar.br

trevor.martin@bris.ac.uk, andrei.majidian@bt.com

Abstract. *Genetic Fuzzy Systems have been researched for two decades and a considerable number of approaches have been proposed in the literature. Depending on the strategy used by the genetic algorithm, the generation of candidate rules is required to form the search space of the genetic algorithm. Specifically for the generation of these rules, proposals in the literature include the exhaustive generation of rules, the use of selection criteria over rules generated exhaustively, such as support, confidence, and degree of coverage, among others. This paper describes some of these methods and present their advantages and disadvantages in order to provide the reader with relevant information when deciding which method to use. A method for rule extraction with competitive advantages based on formal concept analysis is also proposed and preliminary results are discussed in detail. These results show evidence that our proposal is suitable for the task of forming a search space in terms of number of rules and processing time.*

1. Introduction

Fuzzy systems can be described as systems with variables based on the fuzzy logic. They have been successfully applied for the solution of problems in many areas, including pattern classification, optimization, and control of processes [Dumitrescu et al. 2000].

In this work, we focus on the genetic process of the automatic definition of fuzzy systems, specifically on those known as Rule Based Fuzzy Systems (RBFS), which usually have two main components: a Knowledge Base (KB) and an Inference Mechanism (IM). The KB comprises the Fuzzy Rule Base (FRB), *i.e.*, a set of fuzzy rules that represents a given problem, and the Fuzzy Data Base (FDB), which contains the definitions of the fuzzy sets related to the linguistic variables used in the FRB. The IM is responsible for carrying out the required computation that uses inferences to derive the output (or conclusion) of the system, based on both, the KB and the input to the system.

The special term Genetic Fuzzy System (GFS) was coined by the community to refer to fuzzy systems that use a genetic algorithm to create or adjust one or more of

their components [Cordon et al. 2004]. Specifically, the classification of GFSs, according to [Herrera 2008], takes into account if the goal is: i) the genetic tuning of an existing knowledge base, or ii) the genetic learning of components of the KB. This work is focused on the genetic learning of the rule base.

Regarding the generation of rules to form the search space of a Genetic Algorithm (GA), in [Ishibuchi and Yamamoto 2004] the authors proposed an approach based on the rules confidence and support to preselect rules. A predefined number of rules with 0, 1, 2, and 3 antecedent conjunctions were generated for the wine dataset [Frank and Asuncion 2010] and used with a GA. A similar approach, named DOC-BASED, is proposed in [Cintra et al. 2007], in which, after the exhaustive generation of possible rules, a subset of them is selected to form the search space of a GA according to the degree of coverage of the rules. However, since the task of generating all possible rule combinations has exponential complexity, depending on the number of fuzzy variables and sets, the number of possible rules can be very large, interfering with the codification of the chromosomes, and overloading the whole genetic learning process. Thus, for datasets described by many features, a preselection of the most relevant ones is essential.

In this work, we describe some relevant approaches for the generation of the genetic search space proposed in the literature. We also present a proposal for the use of Formal Concept Analysis (FCA) as a competitive alternative to the described methods, including preliminary experiments and results that support our proposal. FCA is a theory on data analysis which identifies conceptual structures among data sets.

The remainder of this paper is described as follows. Section 2 presents the basic concepts of GFS. Section 3 introduces some of the existing approaches in the literature for the generation of the search space for a fuzzy genetic system. Section 4 introduces the theory of FCA and its use for the extraction of rules as well as a proposal for the use of FCA to extract fuzzy rules to be used as the search space of a GA, together with some preliminary experiments and results. The conclusions and future work are presented in Section 5.

2. Genetic Fuzzy Systems

Accordingly to [Herrera 2008], GFSs can be classified as:

1. Genetic tuning: if there exists a KB, a genetic tuning process for improving the RBFS performance is applied without changing the existing FRB;
2. Genetic learning: a component of the KB is learnt, either the FRB, FDB, or both. An adaptive inference mechanism can also be included in the process.

The genetic tuning can be further divided according to the taxonomy provided in Figure 1(a), extracted and adapted from [Herrera 2008], into: i) genetic tuning of KB parameters; ii) genetic adaptive inference engine. Similarly, the genetic learning can also be further divided into: i) genetic KB learning; and ii) genetic learning of DB components and inference engine parameters.

Our work is related to the genetic learning of the FRB, which is a part of the KB. Figure 1(b) provides the complete classification of the genetic KB learning, according to [Herrera 2008], which is also listed with some references next.

2.1 Genetic rule learning with a predefined FDB [Thrift 1991];

2.2 Genetic rule selection with a priori rule extraction [Cintra and Camargo 2007];
2.3 Genetic FDB learning [Cordon et al. 2001];
2.4 Simultaneous genetic learning of KB components [Homaifar and McCormick 1995].

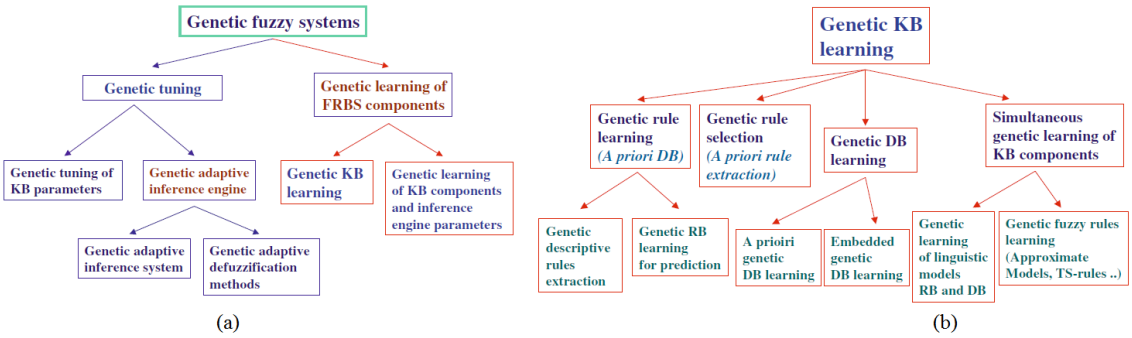


Figure 1. Genetic Fuzzy Systems Classification [Herrera 2008].

More specifically, our work focuses on methods to form the search space of a GA in order to build a GFS, which is part of item 2.2 from the previous list. Next Section surveys some of the key existing approaches for this task, *i.e.*, the generation of the search space for a GFS.

3. Methods for Search Space Rule Generation

In this section we describe some of the most frequently used methods in the literature for the generation of genetic search spaces, presenting and discussing their pros and cons.

3.1. Use of Heuristic Criteria

In [Ishibuchi and Murata 1995], the authors propose the generation of classification rules to form the search space of a GA using the following steps:

1. generation of all possible rule antecedent combinations;
2. calculation of the degree of certainty of each antecedent combination with each possible class using a set of training examples;
3. defining the consequent of each rule antecedent combination as the class with highest degree of certainty.

This approach suits low-dimensional domains, but it is not scalable to larger domains. Thus, in [Ishibuchi and Yamamoto 2004], the authors propose the extensive generation of rule antecedent and class calculation using the degree of certainty with preselection of candidate rules as a more feasible alternative to larger domains. The criteria used to preselect rules are the confidence and support measures, as defined in the data mining context for association rules.

Similarly to the approach in [Ishibuchi and Yamamoto 2004], in [Cintra et al. 2007] it is proposed the generation of the search space by extensively generating all possible rules combining all attributes, then calculating their degree of coverage as a criterium to preselect a subset of them. This way, all rules were composed by a combination of valid linguistic values for each attribute. The Degree of Coverage (DoC) was used as an indication of the classification power of these rules.

In both approaches described previously the main issue is related to the exponential complexity of the exhaustive generation of all possible rules. While these approaches are feasible for domains described by a small number of attributes combined with a reduced set of linguistic values for each attribute, they are not directly scalable for larger domains or when attributes are described by many linguistic values. Another issue is the computational cost required to calculate the degree of certainty and degree of coverage of these rules when combined with the substantial computational effort and processing time required by GAs.

Regarding the advantages of these methods, and others based on heuristic criteria, we can state the following:

1. Since these approaches preselect only relevant rules (according to criteria related to their classification power), the search space can be substantially reduced;
2. Because the rules are previously selected according to their classification power, the search process of the GA is reduced in terms of time;
3. With the preselection of rules, each rule in the search space has an index that is used in the chromosomes, thus, the chromosome codification is simplified (each position of the chromosome has an index to a rule or a special value to represent the absence of a rule). This way, the whole genetic process is optimized saving processing time;
4. Although the approach is not scalable to larger domains, or when a large number of fuzzy terms is used, one might argue that it is, nevertheless, totally feasible for a large number of real domains producing good results. Also, the use of a feature subset selection algorithm reduces the computational cost and time required by this approach, making it scalable to larger domains.

3.2. Association Rules Extraction

Another option for the generation of the rules to form the search space of a GA is to use an association rule extraction algorithm, such as Apriori [Agrawal and Srikant 1994]. Given a set of items (or attributes, for classification problems) and a set of representative examples, the idea is to find rules based on associated items. The support and confidence values of these rules are usually considered for their evaluation. Although association rules are not concerned with supervised domains, association rule algorithms can be used to extract associative classification rules, or simply classification rules, from a set of examples, by simply fixing the consequent as the class attribute .

In the literature it is possible to find proposals to generate fuzzy rule-based classifiers using fuzzy association rules, such as [Hu et al. 2002, Pach et al. 2008]. In [Hu et al. 2002], the authors propose the generation of large fuzzy grids from training examples by fuzzy partitioning each attribute; these grids are then use to generate fuzzy association rules for classification. In [Pach et al. 2008] the authors use the Fuzzy A Priori algorithm to search for frequent fuzzy item sets to form classification rules. The set of these rules is then pruned using the complexity, importance, and generality measures of the rules, forming a fuzzy classifier.

While approaches for the generation of classifiers using fuzzy association rules are abundant in the literature, to the best of our knowledge, the closest proposal using association rules and GAs to a GFS can be found in [Ishibuchi et al. 2006] in which the

authors propose the use of association rule mining to define classifiers using a technique to search for Pareto-optimal rule sets. The authors first mine all possible classification rules using a minimum support and confidence values. These rules are then used by an evolutionary multiobjective optimization algorithm to search for Pareto-optimal rule sets. The three objectives used are: i) number of correctly classified training patterns; ii) number of selected rules (number of rules in the classifier); and iii) total number of antecedent conditions over the selected rules in the classifiers. It is important to notice that the rules generated in this paper are not fuzzy, but classic association rules.

An advantage of this approach based on association rules is related to the fact that the support of these rules can be used in a selection process, discarding rules with very low support, which would not help improve the final rule base, speeding up the genetic search process, and contributing to a better interpretability of the final FRB.

The disadvantages might include the fact that extracting all possible rules is an exponentially complex task. Furthermore, the number of attributes to be included in the extraction process must be defined previously and, thus, the total number of extracted rules might not be sufficient to form the search space, creating a dilemma: the number of combined attributes to be used *versus* the number of possible extracted rules.

Next, we briefly introduce the topic of Formal Concept Analysis (FCA) and present our proposed approach for the generation of the genetic search space using FCA.

4. Formal Concept Analysis

FCA is a mathematical technique for extracting concepts and structure from data introduced in [Wille 1982] which is becoming increasingly popular. Next, we present the basic definitions of FCA.

FCA transforms a formal context into a concept lattice. A formal context is a representation of the relation between objects and their attributes. The basic data structure in FCA is the context, which is normally represented in a table form where the columns represent the attributes and the rows represent the objects. The table contains 1 (true) in cell (i, j) if object i has attribute j, and 0 (false) otherwise. Formally, a context is a triple $k = (G, M, I)$, where G is a set of objects, M is a set of attributes, and I is a binary relation $I \subseteq G \times M$. Given a set of objects $A \subseteq G$, the shared image of A in M is defined as:

$$A^{\uparrow} := \{m \in M | (g, m) \in I \; \forall \; g \in A\} \tag{1}$$

Similarly, given a set of attributes $B \subseteq M$, the shared image of B in G is:

$$B^{\downarrow} := \{g \in G | (g, m) \in I \; \forall \; m \in B\} \tag{2}$$

The pair $(A, B) \in G \times M$ is a formal concept of (G, M, I) if and only if $A \subseteq G, B \subseteq M$ and $A = B \downarrow$, $B = A \uparrow$. A is called the extent of the concept and B is called the intent of the concept [Wille 1982]. In other words, Equation 1 defines the collection of all attributes shared by all objects from A, and Equation 2 defines the collection of all objects sharing all the attributes from B.

In traditional FCA, the relation is binary, although multi-valued contexts are much more common than binary-valued ones. For attributes that can take a range of values, the

idea of "conceptual scaling" that transforms a many-valued attribute (e.g. a number) into a symbolic attribute can be used. For example, an attribute such as "height in centimetres", given an integer or real value between 0 and 200, could be transformed into attributes "height-less-than-50", "height-from-50-to-99", etc. These derived attributes have true/false values and can thus be treated within the FCA framework.

A toy example is illustrated in Table 1, which presents an attribute $\times$ value table with values for name, age, sex, and hair colour of six people.

Table 1. Toy example of a Formal Context.

Name	Age	Sex	Hair Colour
Andy	48	M	Black
Lina	29	F	Black
Mark	23	M	Brown
Martina	46	F	Blonde
Mike	18	M	Brown
Suzy	17	F	Blonde

In order to create the formal context, once FCA only admits binary attributes, attribute Age is discretized into three attributes, Age ≤ 20, > 20 & ≤ 30, and > 30. Since attributes Sex and Hair Colour already present nominal values, these values are used to create single attributes. Table 2 is the resulting table after the transformation.

Table 2. Scaled Version of the Previous Formal Context.

	Age			Sex		Hair Colour		
Name	≤ 20	> 20 & ≤ 30	> 30	Male	Female	Black	Blonde	Brown
Andy			⊛	⊛		⊛		
Lina		⊛			⊛	⊛		
Mark		⊛		⊛				⊛
Martina			⊛		⊛		⊛	
Mike	⊛			⊛				⊛
Suzy	⊛				⊛		⊛	

Using the formal context, it is possible to generate a conceptual lattice that presents the information in a nice visual way. Figure 2 shows the generated conceptual lattice of the information in Table 2. In the lattice structure, formal concepts are represented by the nodes. Attributes are noted slightly above nodes while objects are noted slightly under nodes. Notice that the positioning of the nodes can be arranged in a variety of ways. In the lattice presented in Figure 2, the nodes were arranged in order to minimize intersections, thus, attributes are not displayed in the order they are shown in the formal context. Another option would be to arrange the nodes respecting the order of the attributes or the order of the examples. In order to retrieve extensions, one must simply trace all paths leading down from the node. To retrieve intentions, on the other hand, one must trace all paths leading up from the node. For example, the intention of the formal concept represented by the node named *Mark* is: > 20 & ≤ 30, *Brown*, and *Male*. The extension of the formal concept represented by the node named *Blonde* is: *Martina* and *Suzy*.

As stated previously, the transformation of continuous attributes into binary ones is commonly called *scaling* by the FCA community. The fuzzy theory has also contributed

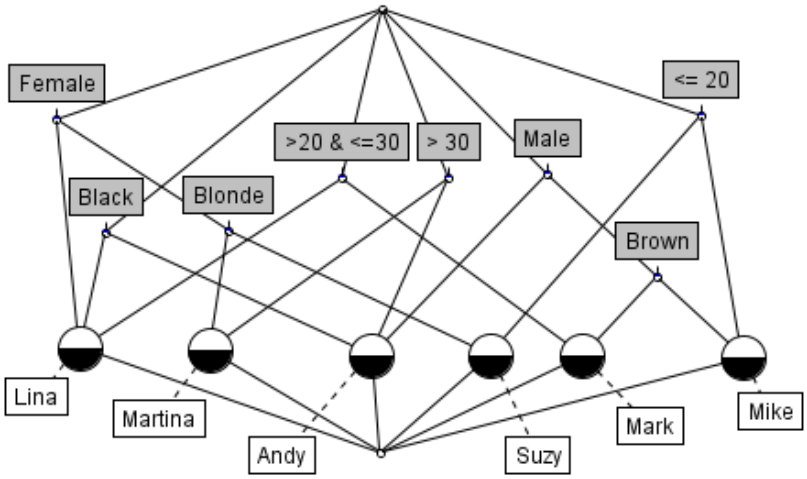


Figure 2. Conceptual lattice.

to this task. In [Wolff 2002] the authors present an introduction to the use of the fuzzy theory for the task of FCA scaling. Some of the advantages of using the fuzzy theory for FCA scaling include:

- the partitioning assume linguistic values, which are easily interpretable by humans, such as *young, old, tall, short*, etc. This way, the formal concepts extracted will convey this interpretability characteristic, which is highly desirable;
- unnatural divisions are avoided, such as the division in Table 2 for the 30 year-old people that are borderlines but must be place in only one category. The fuzzy logic easily avoids this problem with the use of the membership degrees that allow one object to belong to different categories with different degrees of membership;
- it is a natural choice when the interest in FCA is for the extraction of fuzzy rules, since these rules will reflect the fuzzy data base that will also be used by the inference mechanism of the induced classifier.

Next, we present a proposal on the use of FCA for the extraction of fuzzy rules.

4.1. Proposal for Extracting Fuzzy Rules from a Formal Context

As discussed in the previous sections, a formal context is the base for the extraction of formal concepts. These formal concepts can be seen as associations between attributes based on the existence of objects sharing these attributes. It is also important to notice that in the process of extracting formal concepts, their support is calculated automatically. In the literature it is possible to find various algorithms to extract formal concepts from a lattice. Worth mentioning are the NextClosure algorithm which works by finding neighbouring concepts [Ganter 2002], and the one proposed in [Krajca et al. 2010], which has a parallel search process.

Our proposal is based on the fuzzy definition of a problem in an attribute $\times$ value table to create a formal context and then obtain the classification rules. Considering a

general dataset for classification purposes with n examples and m attributes, this fuzzy scaling procedure is performed with the following steps:

1. Define the fuzzy database, *i.e.*, the partitions that will define the fuzzification of the continuous attributes;
2. Create a binary attribute defined by each fuzzy set of each continuous attribute and each value of each discrete attribute;
3. Calculate the membership degree of the input values for each example in each binary attribute (notice that this step is only required for continuous attributes, once discrete ones will be automatically defined as true of false);
4. Define a minimum value A_{min} to guide the scaling of the real values so that if the membership degree of a certain value for a particular fuzzy set is equal or higher than A_{min}, the corresponding attribute is set to true in the formal context;
5. Use an algorithm to extract formal concepts to extract all existing classification rules from the formal context;
6. Define a minimum support value to select a subset of the extracted rules;
7. Use the final rule set as the search space of a GA to induce a fuzzy system.

Please, notice that due to space, the last step of the proposal, the use of a GA to induce a fuzzy system, is not explained in details here.

In order to reduce the number of possible formal concepts, the fuzzy sets defining each attribute can be evenly distributed in the partition, so the maximum possible membership degree in the intersections is 0.5. This way, if A_{min} is set to 0.5, for each original attribute only one of its binary attributes will be activated. Notice that ties must also be handled, thus, in our implementation we used a random variable to activate one of 2 tied fuzzy sets.

An important issue regarding the extraction of the formal concepts when using a binary fuzzification of the formal context is related to the increase in the number of attributes. This total number of attributes will be equal to the sum of all fuzzy sets describing each attribute, each value of each discrete attribute, and the number of classes. This increase in the number of attributes leads to an increase in the total number of formal concepts that can be extracted. Nevertheless, it is important to notice that for our purpose, the total number of formal concepts is much larger than the number of formal concepts we are interested in extracting: since we want to extract classification rules, we are only interested in extracting formal concepts that have a class in their intention.

Regarding the total number of formal concepts existing in a formal context, this number can be estimated using the Metropolis-Hastings algorithm for sampling formal concepts described in [Boley et al. 2010].

The process of extracting formal concepts and, consequently, classification rules can be done by adapting any algorithm for the extraction of formal concepts. In our experiments we used the NextClosure algorithm described in [Ganter 2002]. Since NextClosure can be used to extract formal concepts by analysing either the attributes or the objects, when it uses the attributes, it is called *Next Intent*, and it is called *Next Extent* when it uses the objects. This way, the word *element* is adopted instead of attributes or objects. Regarding the modification of the NextClosure algorithm to extract only formal concepts including a class, the process is quite direct: once the algorithm looks for neighbouring

concepts, we just need to check whether a found concept has a class or not. If it does, then it is stored, otherwise, the process carries on looking for the next neighbouring concept.

Another interesting possibility for the restriction of formal concepts is the evaluation of their support, which, for a formal context, is just the number of objects a given formal concept shares divided by the number of all objects. This way, considering the balancing of the classes, formal concepts with low support could be discarded, reducing the number of extracted formal concepts.

4.2. Preliminary Experiments

Our main goals when carrying on the experiments were threefold: i) to check whether the resulting rule set would contain a suitable number of rules to form the search space of a GA; ii) to evaluate the time taken to extract the formal concepts; and iii) to evaluate the idea of using the support of the rules for a preselecting process.

To evaluate the proposed modification of the NextClosure algorithm we used 10 datasets from the UCI - Machine Learning Repository [Frank and Asuncion 2010] in order to analyse the number of concepts. Table 3 summarizes the dataset characteristics giving the total number of examples (Examples); total number of features (Features), including the number of continuous and discrete features in brackets; number of classes (Classes); the majority error (ME); and the number of fuzzy sets for each of the attributes (FS) predefined using the Fuzzy–DBD algorithm [Cintra et al. 2009]. Examples with missing values were removed from the datasets.

Table 3. General characteristics of the datasets.

Dataset	Examples	Features			Classes	ME	FS
Credit	653	15	6	9	2	45.33	2
Cylinder	277	32	19	13	2	35.74	2
Dermatology	358	34	33	1	6	68.99	2
Diabetes	769	8	8	0	2	34.90	2
Glass	220	9	9	0	7	65.46	7
Heart	270	13	13	0	2	44.44	2
Ionosphere	351	34	34	0	2	35.90	3
Iris	150	4	4	0	3	66.60	3
Vehicle	846	18	18	0	4	74.23	2
Wine	178	13	13	0	3	59.74	3

Table 4 presents the total number of formal concepts (TNFC), the total number of formal concepts with a class (FCwC), *i.e.*, the total number of classification rules, and the percentage it represents of the total number of formal concepts. In order to allow further comparisons, Table 4 also presents the total number of formal concepts for 50%, 20%, 10%, and 5% support values and the percentage they represent of the total number of formal concepts with a class (FCwC). The bigger the support, the smaller the number of rules. Support values greater than 50% were not considered as the number of rules was not enough to form the search space of a GA.

Regarding our first goal (the suitability of the rule set extracted in terms of number), we verified that an appropriate number of rules was extracted (FCwC). Our verification takes into consideration an estimated number of rules to populate 50 chromosomes,

Table 4. Extracted Formal Concepts Information

Dataset	TNFC	FCwC		50%		20%		10%		5%	
Credit	20,083	9,843	49.01	4	0.04	825	8.38	2,410	24.48	4,235	43.03
Cylinder	7,041,110	1,944,271	27.61	884	0.05	113,474	5.84	546,265	28.10	1,236,470	63.60
Dermatology	21,896,570	312,177	1.43	0	0.00	553	0.18	38,141	12.22	140,502	45.01
Diabetes	2,172	1,279	58.89	32	2.50	308	24.08	669	52.31	758	59.27
Glass	4,054	1,863	45.95	0	0.00	38	2.04	301	16.16	670	35.96
Heart	81,935	36,942	45.09	9	0.02	1,648	4.46	8,342	22.58	19,854	53.74
Ionosphere	102,641,179	2,076,229	2.02	12	0.00	1,187,827	57.21	1,197,307	57.67	1,649,406	79.44
Iris	93	65	69.89	0	0.00	11	16.92	27	41.54	41	63.08
Segmentation	10,785	1,437	13.32	0	0.00	0	0.00	162	11.27	54	3.76
Vehicle	86,918	28,979	33.34	0	0.00	91	0.31	1,780	6.14	6,625	22.86
Wine	21,000	9,802	46.68	0	0.00	423	4.32	3,338	34.05	6,676	68.11

which was the total population used in previous experiments with the DOC-BASED algorithm [Cintra et al. 2007]. Considering that the smallest rule set was obtained with the Iris dataset (60 rules), for this specific dataset, the total number of rules in each chromosome must be carefully defined in order to avoid lack of diversity in the population. The number or rules obtained for the remaining datasets can be considered appropriate.

Table 5 shows the time taken, in minutes, for the extraction of the formal concepts and calculation of their support. The process was executed in an Intel® Core™2 Duo T7250 (2.00GHz, 2MB L2 Cache, 800MHz FSB) machine. The time taken to extract the rules can also be considered appropriate for this approach to be used together with a genetic algorithm search process. The whole process took a matter of seconds to finish for all datasets but Cylinder, Dermatology, and Ionosphere, due to the total number of attributes and examples for these databases. It is also important to notice that our experiments were carried out with the NextClosure algorithm [Ganter 2002], but a faster algorithm was recently proposed which has a parallel search process [Krajca et al. 2010].

Table 5. Time (in minutes) taken to extract all formal concepts.

Dataset	Time	Dataset	Time
Credit	0.50	Heart	0.55
Cylinder	126.00	Ionosphere	144.63
Dermatology	168.73	Iris	0.02
Diabetes	0.05	Vehicle	1.90
Glass	0.12	Wine	0.18

If the support is taken into consideration for the selection of formal concepts including a class, it is possible to reduce even more the number of extracted formal concepts, giving the user more flexibility to decide on the number of extracted rules. The only issue one has to bear in mind when using the support in order to reduce the number of extracted rules is related to classes with few examples, which, in turn, will result in relative low support. In fact, as stated previously, in [Ishibuchi and Yamamoto 2004], the authors use the support, confidence, and the product of these two measures to select rules. These measures can also be investigated as future work.

Next, we present the conclusions and future work.

5. Conclusions and Future Work

The field of genetic fuzzy systems is the one with some of the most promising results in the area of fuzzy systems and many new approaches have been proposed in the literature. Specifically for the genetic definition of the rule base, a possible approach is the genetic rule selection, with the previous definition of the fuzzy database and the generation of fuzzy rules to form the search space of the genetic selection process. In this paper we presented some approaches found in the literature for this task of forming the search space of a genetic algorithm.

We also presented a new proposal and preliminary experiments and results on the use of formal concept analysis for this task. Formal concept analysis can be considered a new area of research, with applications in various domains and with an increasing interest due to its visual benefits and powerful mathematical basis.

The preliminary results show that our proposal is suitable for the task of forming the search space of a genetic algorithm in terms of the number of rules extracted, processing time, and also the use of the support measure to preselect a reduced number of rules if necessary.

As future work, we intend to adopt and use the DoC-Based method, proposed in [Cintra et al. 2007, Cintra and Camargo 2007], to generate fuzzy rule bases using the proposal presented here to form the search space. This method includes the number of rules and the accuracy of the rule base in the fitness calculation in order to induce a rule base with high accuracy and interpretability rates. We intend to compare the formal concept extraction time taken by the NextClosure algorithm [Ganter 2002] and the parallel approach proposed in [Krajca et al. 2010]. We also intend to empirically evaluate the use of the support and confidence measures, and the product of them, to select rules.

Acknowledgements

The authors would like to thank CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico and the São Paulo State Research Foundation - FAPESP for the financial support.

References

Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 487–499, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Boley, M., Grosskreutz, H., and Gärtner, T. (2010). Formal concept sampling for counting and threshold–free local pattern mining. In *Proc. of the SIAM Int. Conf. on Data Mining (SDM 2010)*, pages 177–188. SIAM.

Cintra, M. E. and Camargo, H. A. (2007). Fuzzy rules generation using genetic algorithms with self-adaptive selection. *IEEE International Conference on Information Reuse and Integration - IRI*, pages 261–266.

Cintra, M. E., Camargo, H. A., and Martin, T. P. (2009). Optimising the fuzzy granulation of attribute domains. *International Fuzzy Systems Association World Conference*, pages 742–747.

Cintra, M. E., Camargo, H. A., R. Hruschka Jr, E., and Nicoletti, M. C. (2007). Fuzzy rule base generation through genetic algorithms and Bayesian classifiers - a comparative approach. *International Conference on Intelligent Systems Design and Application - ISDA*, 1:315–320.

Cordon, O., Gomide, F., Herrera, F., Hoffmann, F., and Magdalena, L. (2004). Special issue on genetic fuzzy systems.

Cordon, O., Herrera, F., and Villar, P. (2001). Generating the knowledge base of a fuzzy-based system by the genetic learning of the data base. *IEEE Transactions on Fuzzy Systems*, 9(4):667–674.

Dumitrescu, D., Lazzerini, B., and Jair, L. (2000). *Fuzzy Sets and Their Application to Clustering and Training*. Int. Series on Computational Intelligence. CBC Press.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository.

Ganter, B. (2002). Formal concept analysis: Algorithmic aspects. Technical report, Technische Universitat Dresden.

Herrera, F. (2008). Genetic fuzzy systems: taxonomy, current research trends and prospects. *Evolutionary Intelligence*, 1(1):27–46.

Homaifar, A. and McCormick (1995). Simultaneous design of membership functions and rule sets for fuzzy controller using genetic algorithms. *IEEE Transactions on. Fuzzy Systems*, 3(2):129–139.

Hu, Y.-C., Chen, R.-S., and Tzeng, G.-H. (2002). Mining fuzzy association rules for classification problems. *Computing and Industrial Engineering*, 43:735–750.

Ishibuchi, H., Kuwajima, I., and Nojima, Y. (2006). Multiobjective association rule mining. In *Proc. of PPSN Workshop on Multiobjective Problem Solving from Nature*, pages 12–24.

Ishibuchi, H. and Murata, T. (1995). Selecting linguistic classification rules by two-objective genetic algorithms. In *Proceedings of the 1995 IEEE Int. Conf. on Systems, Man and Cybernetics*, pages 1410–1415. IEEE.

Ishibuchi, H. and Yamamoto, T. (2004). Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. *Fuzzy Sets and Systems*, 141:59–88.

Krajca, P., Outrata, J., and Vychodil, V. (2010). Parallel algorithm for computing fixpoints of galois connections. *Annals of Mathematics and Artificial Intelligence*, 59:257–272.

Pach, F. P., Gyenesei, A., and Abonyi, J. (2008). Compact fuzzy association rule-based classifier. *Expert Syst. Appl.*, pages 2406–2416.

Thrift, P. (1991). Fuzzy logic synthesis with genetic algorithms. In Belew, R. K. and Booker, L. B., editors, *Proceedings of the 4th International Conference on Genetic Algorithms*, pages 509–513. Morgan Kaufmann.

Wille, R. (1982). Restructuring lattice theory: An approach based on hierarchies of concepts. In *Ordered Setsand in I. Rivals (Ed.)*, volume 23, pages 445–470.

Wolff, K. E. (2002). Concepts in fuzzy scaling theory: order and granularity. *Fuzzy Sets and Systems*, 132(1):63–75.

Um Método Heurístico para Cálculo de Rotas com Desvios de Obstáculos em Serviços Web de Mapas Digitais

Victor E. D. Basso, Vasco Furtado

Programa de Pós-graduação em Informática Aplicada (DIA/MIA)

Laboratório Engenharia do Conhecimento, NATI – Universidade de Fortaleza (UNIFOR) – Fortaleza – CE – Brasil

vitobasso@gmail.com, vasco@unifor.br

Abstract. *Computing a route between two points that deviate from an obstacle is a typical activity of geoprocessing systems, but that is not supported by services of digital map in the web due to the high computational cost. We propose here an approach to the problem that uses conjointly the shortest route provided by these services and heuristic methods based on visibility algorithms. This method is inspired in route planning for robots what can suggest alternative waypoints to force the contour of the obstacles identified. A comparative assessment of different scenarios indicates the feasibility of the method and its potential for improvement.*

Resumo. *O cálculo de rotas entre dois pontos que desviem de obstáculos é uma atividade típica de sistemas de geoprocessamento, mas que não é provida por serviços web de mapas digitais dado o seu alto custo computacional. Propomos aqui uma abordagem para o problema que consiste em usar conjuntamente o cálculo de rota mais curta provido por esses serviços e métodos heurísticos baseados em algoritmos de visibilidade. Esses métodos são inspirados em planejamento de rotas para robôs, levando a sugerir pontos intermediários alternativos às rotas originalmente traçadas pelos SWMD que forcem o desvio dos obstáculos identificados. Uma avaliação comparativa em diversos cenários indica a viabilidade do método e seu potencial de melhoria.*

1. Introdução

Serviços Web de Mapas Digitais (SWMD) como os proporcionados por Google (www.maps.google.com), Yahoo (www.maps.yahoo.com) e Bing (www.bing.com/maps) são extremamente populares hoje em dia. Tornaram-se essenciais para um grande número de aplicações, em particular, aquelas conhecidas como *mashup* [Duane, 2006], onde informações são automaticamente agregadas aos mapas e funcionalidades típicas de um Sistema de Informações Geográficas são providas. Recuperar informações dentro de um determinado raio, consultar informações para determinada região são exemplos destas funcionalidades.

O cálculo de rotas entre dois pontos é outra funcionalidade típica deste contexto, no entanto, o provimento desse tipo de serviço não se faz sem desafios. O cálculo de uma rota mais curta entre dois pontos é um problema de complexidade $O(v^2)$ onde v é o numero de possíveis pontos intermediários entre o início e destino. Praticamente, essa complexidade é ainda muito custosa para os SWMD que de uma forma geral implementam métodos heurísticos inclusive com a escolha de rotas mais promissoras (avenidas ao invés de ruas pequenas, por exemplo) [Ishikawa, 1991].

Esse problema é mais sério quando se trata do cálculo de rotas em que obstáculos precisam ser contornados. Via de regra, os obstáculos são formados pela aglomeração de eventos vindos tanto da participação voluntária das pessoas, dito VGI [Rouse, 2007] ou vindos de importação de outros sistemas de informações. O dinamismo dessa captura de informações bem como a complexidade de tempo que é ainda maior do que um simples cálculo de rota tornam difícil o provimento deste tipo de serviço pelos provedores de SWMD.

A solução que propomos para esse problema é realizar o cálculo de uma *rota desobstruída* (que contorne os obstáculos) no cliente fazendo uso de serviços dos SWMD. Tal solução impõe novos desafios. O primeiro deles advém do fato que os SWMD não oferecem acesso direto ao mapa com todos os dados geográficos, por exemplo, o eixo de ruas, que são acessíveis apenas visualmente. Conseqüência disto é que o desenvolvimento de métodos alternativos de cálculo de rotas requer o uso do serviço de cálculo de rotas provido pelos SWMD e assim as características das rotas fornecidas pelos SWMD (comprimento, formato, locais por onde passa) só são conhecidas após uma requisição ser feita a este serviço. O segundo problema é que essas próprias requisições ao serviço de cálculo de rotas possuem um custo e devem ser minimizadas. Dadas essas restrições o problema que buscamos resolver é desenvolver um algoritmo de cálculo de rotas que contornem obstáculos, minimizem a distância e o número de chamadas aos serviços de cálculo de rotas dos SWMD.

Propomos uma abordagem inspirada em planejamento de rotas para robôs [Sánchez, 2004], em particular, em algoritmos de visibilidade [Senjuti, 2007] que consiste em definir heurísticas para gerar pontos intermediários alternativos às rotas originalmente traçadas pelas rotas mais curtas sugeridas pelos SWMD.

A aplicação de nosso método foi feita no sistema WikiCrimes [Furtado et al. 2010] onde informações voluntárias sobre ocorrências criminais são informadas no Google Maps. A ideia é desenvolver rotas ditas seguras que contornam áreas de alta densidade de ocorrências criminais calculadas por métodos de Kernel [McLafferty, 2000].

2. Delineamento Básico da Solução Proposta

Em linhas gerais, o método que propomos segue o segunte procedimento iterativo. A cada iteração, o serviço de rotas do SWMD é acessado uma vez e uma nova rota real é descoberta e agregada a um grafo que liga a nova rota com rotas descobertas em iterações anteriores. Em cada iteração, novas rotas são estimadas entre alguns pares de vértices do grafo. Estas estimativas são armazenadas em uma fila de prioridades baseada em uma função de peso da rota e acumulam com as estimativas feitas em outras

iterações. A melhor estimativa da fila em determinada iteração é escolhida para ser parâmetro do cálculo da próxima rota real. Ao ser escolhida uma rota estimada, os pontos que a compõem são inseridos no grafo, abrindo possibilidades para novas estimativas. O processo se repete até que uma rota desobstruída ligando a origem ao destino seja reconhecida por uma função de tolerância.

Nos tópicos a seguir descrevemos em mais detalhe cada aspecto da técnica desenvolvida. Iniciamos descrevendo a forma como as ruas e a obstrução são representadas. Depois apresentamos as funções de peso de rotas e de tolerância. Seguimos apresentando a estrutura geral do algoritmo e por fim as estratégias de estimativa de rotas que experimentamos.

2.1. Representação da *obstrução*

A *obstrução* é representada por um *mapa de calor* gerado a partir dos eventos pontuais.

Um algoritmo de *mapa de calor*, como o estudado em [McLafferty, 2000], divide o mapa em uma grade fina. A cada célula da grade, é associado um valor de densidade representativo para os eventos em sua proximidade. Ao final, cores são atribuídas a cada célula, dependendo do valor, resultando em um *mapa de calor* visual.

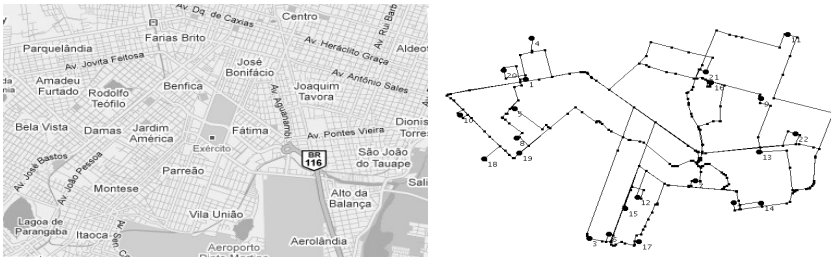
Utilizamos a grade de densidades do *mapa de calor* para representar a obstrução.

2.2. O grafo de rotas

No mapa original, ou seja, o conhecido pelo SWMD, existem informações de ruas, seu nome, sua localização física, sentidos e conexões. No mapa conhecido pelo nosso algoritmo, apenas partes dessas informações estão disponíveis.

Modelamos este mapa em um grafo de rotas como visto na figura 1. Neste grafo, as arestas são rotas unidirecionais e os vértices são origens e destinos das rotas. Cada vértice é definido por um ponto no plano. Cada aresta liga dois vértices e contém a seqüencia de pontos cujo traçado define uma rota gerada pelo SWMD.

Para o problema aqui discutido, as arestas do grafo possuem um peso, que representa tanto o comprimento da rota como a obstrução associada à região pela qual ela passa.



(a) **(b)**

Figura 1. (a) O mapa disponível na interface gráfica do SWMD com a malha viária completa da região. (b) O grafo de rotas que contém parte da malha viária revelada a partir de 20 requisições.

Sejam:

r uma rota;

o(i,j) a densidade da célula do mapa de calor que fica na i-ésima linha e j-ésima coluna;

n e m o número de linhas e colunas do mapa de calor respectivamente;

c(r,i,j) o comprimento da parte da rota que está contida na célula (i,j) do mapa de calor;

O peso da rota r é definido por: $peso(r) = \sum_{i=1}^{n} \sum_{j=1}^{m} \left[c(r,i,j) \cdot o(i,j) \right]$

O peso é usado, a cada iteração, na escolha da rota estimada pela fila de prioridades, para ser enviada ao SWMD, e também na identificação da melhor rota do grafo de rotas, a fim testar sua tolerância.

2.3. A Função Tolerância

Cada vez que uma nova rota é revelada, é possível que um caminho melhor seja encontrado entre a origem e o destino. A função peso quantifica o nível de obstrução e o comprimento da rota. Usamos o peso para decidir se uma rota é melhor que outra, porém, decidir se uma rota já é boa o suficiente, ou seja, se a busca terminou, é outro problema.

Sejam:

c um cenário composto por origem, destino, área e mapa de calor;

$l_{min}(c)$ a distancia entre a origem e o destino de c;

$o_{med}(c)$ a obstrução media em c;

$o_{max}(c)$ a obstrução máxima em c;

$o_n(c) = \frac{o_{med}(c)}{o_{max}(c)}$ a obstrução normalizada;

K uma constante que representa a influencia da distancia no resultado;

F um fator de ajuste;

A tolerância no cenário c é definida como: $t(c) = l_{min}(c) * \left[K + o_n(c) \right] * F$, onde K e F são parâmetros arbitrários de calibração.

3. Heurísticas de geração de rotas estimadas

O ponto principal do algoritmo proposto refere-se às heurísticas para gerar as rotas estimadas. Usamos três heurísticas que têm como função principal gerar o que chamamos de pontos promissores. Trata-se de uma sequência de pontos que, ao ser usada como intermediária em uma chamada ao SWMD, há boas chances de que a rota resultante seja uma rota desobstruída. Um conjunto de pontos é promissor para um mapa de calor, origem e destino específicos. Ou seja, um ponto que é promissor para um subproblema de origem O_1 e destino D_1 pode não ser promissor para outro com O_2 e D_2 distintos.

Sejam (P_1, P_2, ... , P_n) pontos promissores para P_o e P_d, então a sequência de pontos (P_o, P_1, P_2, ... , P_n, P_d) é uma *rota estimada*. Ou seja, uma rota estimada pode ser

usada como parâmetro de uma chamada ao SWMD, tendo boas chances de gerar um bom resultado.

3.1 Mapa de visibilidade

Nesta estratégia, consideramos que um ponto é promissor quando o ponto tem visibilidade em relação à origem e ao destino da rota.

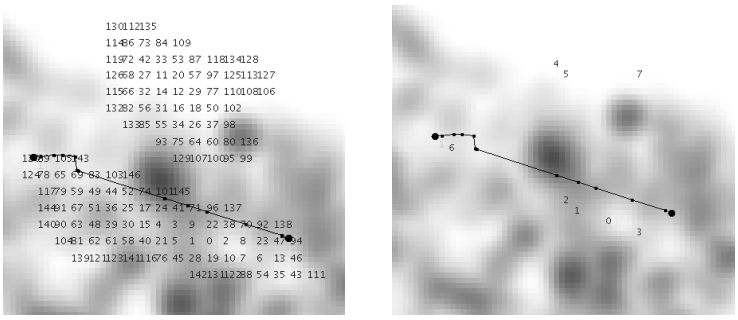
Usamos o termo visibilidade em analogia com o seguinte cenário: imagine que a origem e destino sejam fontes de luz e a obstrução do mapa diminui a passagem da luz. Dizemos que um ponto P tem boa visibilidade quando a região do mapa entre ele e a origem O (e o destino D) não está obstruída, ou seja, P recebe bastante luz de O e D.

Se os obstáculos fossem representados por polígonos opacos, a visibilidade deveria ser representada por uma função binária: "visível"/"não visível". Porém, na estratégia mapa de visibilidade, os obstáculos são representados por manchas de densidade variável, ou seja, o mapa de calor. Neste caso, a visibilidade é representada por um número decimal.

Definimos a função visibilidade para um ponto P, origem O e destino D como segue: $visibilidade(P) = \frac{1}{peso(OPD)}$, onde OPD é a concatenação do segmento de reta que vai de O até P com o que vai de P até D.

A visibilidade é calculada para pontos no mapa inteiro, então os de melhor visibilidade são organizamos numa fila de pontos promissores.

O primeiro passo do processo consiste dividir o mapa em uma *grid* com granularidade arbitrária. Depois, calcular a visibilidade para um ponto em cada célula da *grid*. É obtida uma matriz de valores de visibilidade. O próximo passo é encontrar mínimos locais e armazená-los ordenados por valor numa fila, com referências aos respectivos pontos. Esta fila contém os pontos promissores em ordem. Ver exemplo na Figura 2.



(a) **(b)**

Figura 2. As imagens (a) e (b) representam pontos promissores para um cenário. Em (a) é exibido um número para cada nó da *grid*. O número representa a posição do nó na fila ordenada por valor de visibilidade. Os números de nós com valor de visibilidade muito ruim, comparados com a obstrução da rota original, em preto, estão omitidos. Em (b) estão visíveis apenas os mínimos locais da função de visibilidade.

As rotas geradas pelo Mapa de Visibilidade podem ter apenas um ponto intermediário. Uma vantagem deste método é que ele avalia possibilidades no mapa inteiro e gera muitas rotas estimadas para um cenário.

3.2. Grafo de Visibilidade

Uma variação da estratégia anterior é calcular o Grafo de Visibilidade [O'Rourke, 2003] usado para encontrar menores caminhos desviando de obstáculos poligonais em um contexto mais genérico, ou seja, independente de uma malha viária. Num Grafo de Visibilidade, os vértices são todos os vértices dos obstáculos mais a origem e o destino e as arestas são todas as linhas que ligam dois vértices do grafo exceto as que cortam obstáculos. Sabemos que o menor caminho do grafo é também o menor caminho entre a origem e o destino que desvia de obstáculos.

Neste trabalho, experimentamos usar a mesma estratégia em nosso contexto particular para encontrar pontos promissores. Para obter obstáculos poligonais a partir da obstrução do mapa, definimos um valor de obstrução limite que separa as células do mapa que farão parte de obstáculos das outras. Tendo estas células identificadas, usamos um algoritmo que caminha pela borda de agrupamentos de células e define polígonos. A partir dos polígonos, é construído o Grafo de Visibilidade, e seu menor caminho é calculado. Os vértices contidos no menor caminho compõem uma rota estimada.

Note que os pontos promissores gerados por esta técnica tangem os obstáculos. É necessário definir uma densidade limite "frouxa" o bastante para que os pontos não fiquem em áreas obstruídas.

Uma vantagem desta técnica é que pode gerar uma rota estimada com múltiplos pontos intermediários, o que não é possível com o Mapa de Visibilidade.

3.3. Grafo de Caminhos Livres

Em [Lima, 2008], um Grafo de Caminhos Livres é usado num passo intermediário de um algoritmo de cálculo de menor caminho desviando de obstáculos como meio de demonstrar a existência de, pelo menos, um caminho entre a origem e o destino.

Os vértices de um Grafo de Caminhos Livres são equidistantes dos obstáculos diretamente acima e abaixo. Os caminhos neste grafo, portanto, tendem a passar longe dos obstáculos. Por esse motivo, experimentamos este método em nosso trabalho, como uma alternativa ao Grafo de Visibilidade, para encontrar pontos promissores.

4. Um algoritmo heurístico para rotas desobstruídas em SWMD

Feitas as definições dos principais conceitos e das heurísticas de geração de rotas estimadas, podemos descrever a idéia geral do algoritmo. Seus passos estão descritos no Quadro 1. Cada um dos métodos de cálculo de pontos promissores para rotas estimadas apresentados é usado. O resultado deles é inserido na fila de prioridades. Já que o gargalo do algoritmo está na requisição remota ao SWMD, vale à pena pagar este custo adicional em processamento local para obter mais rotas estimadas alternativas dentre as quais a melhor será escolhida.

No início do algoritmo, o grafo de rotas contém apenas dois vértices: origem P_o e destino P_d. Na primeira iteração, é calculada a rota real rota(P_o,P_d), se a rota não for tolerável, são calculadas rotas estimadas ligando a P_o e P_d. Destas, uma é escolhida para gerar a rota real da segunda iteração. Para cada ponto da rota estimada escolhida um vértice é adicionado ao grafo .

Nas iterações seguintes, cada par de pontos adjacentes da rota estimada escolhida na iteração anterior é considerado origem e destino de um problema de cálculo de rota com desvio de obstáculos menor. Novas rotas estimadas são calculadas para o par de pontos e assim sucessivamente. Se, por exemplo, uma rota estimada com três pontos P_1, P_2 e P_3 gerar uma rota real que é desobstruída apenas entre P_1 e P_2, este resultado parcial é aproveitado e o algoritmo passará a se preocupar com a rota entre P_2 e P_3.

O algoritmo A* [Fu, 2006] atribui a cada nó i uma função de avaliação heurística $F_{(i)} = L_{(i)} + e_{(i,d)}$, onde $L_{(i)}$ é o peso do caminho até então avaliado da origem até o nó i e $e_{(i,d)}$ é uma estimativa do peso do caminho não avaliado ainda entre o nó i e o destino. De maneira análoga, o algoritmo para rotas desobstruídas em SWMD, ao calcular rotas estimadas em sub-problemas, ou seja, com origens e destinos P_o' e P_d', leva em conta no cálculo do peso de cada rota estimada os trechos de rotas reais de menor peso que ligam P_o a P_o' e P_d' a P_d.

Sejam:
G, o grafo de rotas descrito em 2.2;
Q, a fila de rotas estimadas, uma fila de prioridade ordenada pelo peso das rotas definido em 2.2.;
i_{max}, o limite máximo de iterações definido arbitrariamente;
o e d, os pontos de origem e destino respectivamente; e
rota(P), a função de rotas oferecida pelo SWMD, onde P é uma sequência de pontos.
O algoritmo inicia:
R_1 é obtida de rota(o,d).
Para cada iteração i:
    R_i é inserida em G;
    S, a rota de peso mínimo em G, é calculada, como em 2.2;
    Se S for tolerável, conforme definido em 2.3, então:
        Retorna S;
        Fim.
    Senão:
        E, um conjunto de rotas estimadas, é calculado a partir de R_i usando uma das estratégias apresentadas em 3;
        Os elementos de E são inseridos em Q;
    e_{min}, a rota estimada de menor peso em Q, é retirada;
    R_{i+1}, é obtida de rota(e_{min});
    Se i = i_{max}:
        Não foi encontrada rota desobstruída;
        Fim.
    Senão:
        Continua na iteração i+1

Quadro 1. Pseudo-código do algoritmo de cálculo de rotas desobstruídas

5. Avaliação

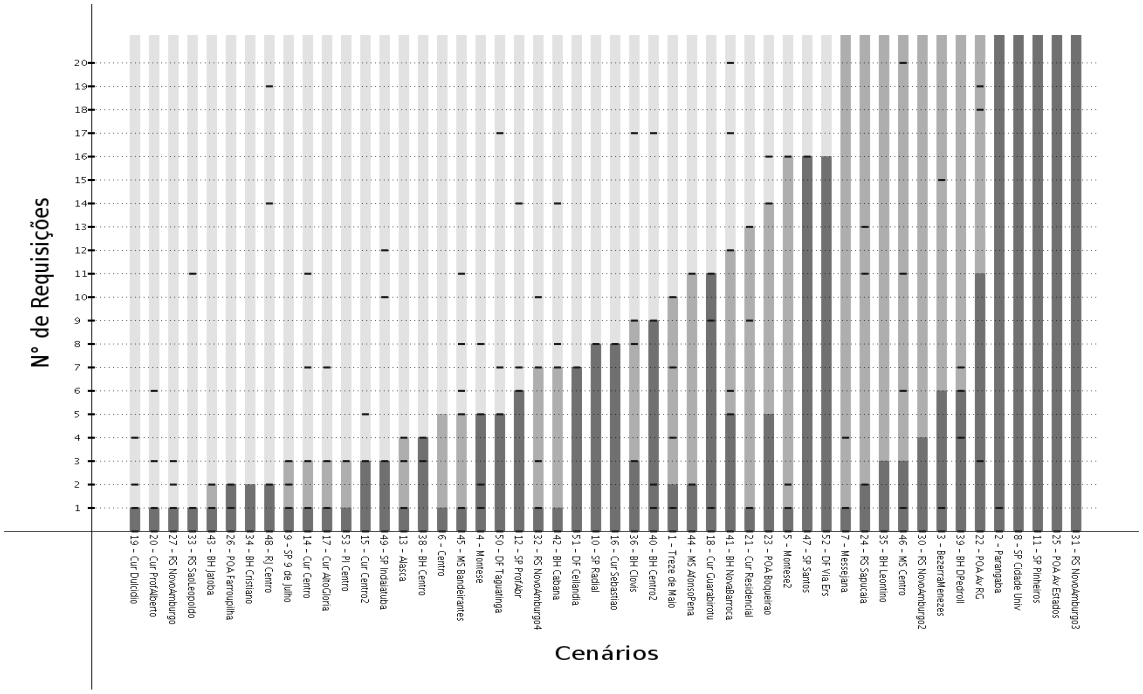
A técnica apresentada foi utilizada em Wikicrimes, um sistema web baseado em GoogleMaps que mapeia eventos de criminalidade tanto a partir de dados providos por entidades governamentais como da colaboração de usuários comuns, fornecendo uma visão geográfica da distribuição de criminalidade em qualquer região do globo. Uma das funcionalidades do Wikicrimes é a apresentação da concentração criminal em um mapa de calor, que é representado internamente por uma matriz de valores reais.

Para implementar uma funcionalidade de rotas seguras no Wikicrimes, foi aplicada a técnica apresentada neste trabalho. Uma rota é considerada segura se evitar regiões de alta criminalidade. Fazendo correspondência com o problema descrito na seção 2, a função de obstrução é representada pela matriz do mapa de calor no Wikicrimes. O *GoogleMaps*, disponibiliza em sua API, uma função de cálculo de rotas que recebe como parâmetro uma lista de pontos, assim como r(x) descrita na mesma sessão.

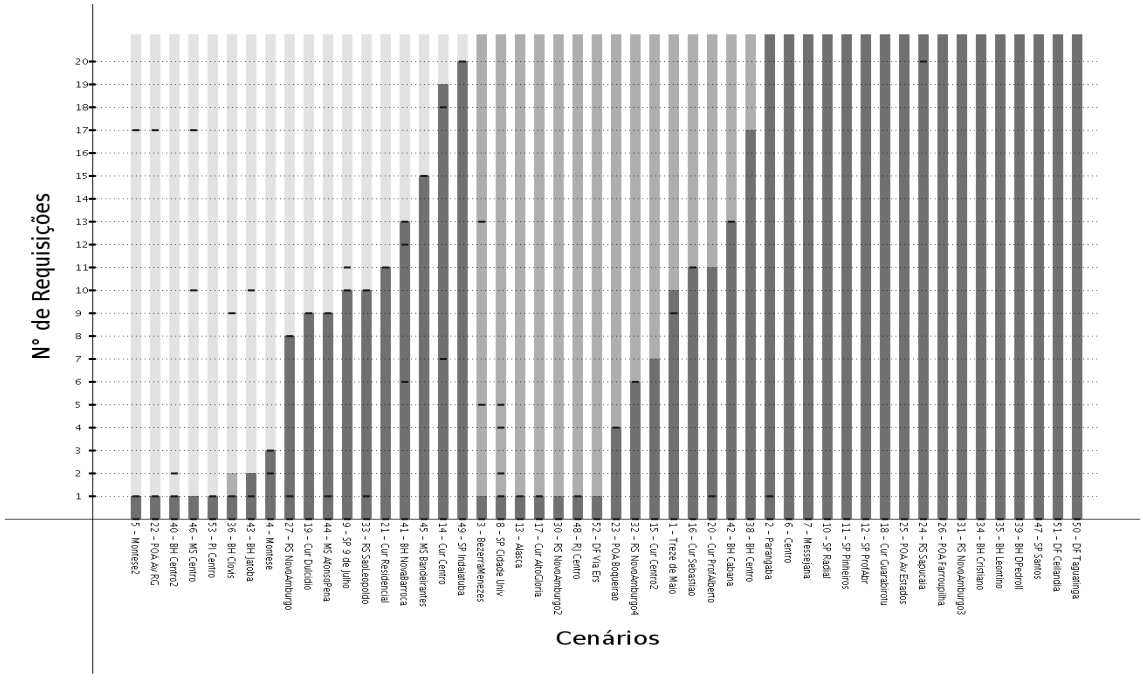
Fizemos testes no Wikicrimes com 50 cenários em 13 cidades diferentes. Cada cenário tem uma origem e um destino para os quais a o SWMD retorna uma rota obstruída. O critério de avaliação adotado é o número de requisições feitas ao SWMD até uma rota satisfatória ser obtida. O julgamento do quão satisfatória é uma rota é feito manualmente. Consideramos que o algoritmo obteve sucesso em um cenário se fizer no máximo 20 requisições.

Os gráficos nas Figuras 3 e 4 relacionam cada cenário ao número de requisições feitas. As barras marcam a qualidade das rotas durante o progresso do algoritmo. O tom mais escuro indica que as rotas conhecidas durante a iteração correspondente são todas obstruídas. O intermediário indica que houve evolução, ou seja, foi encontrada alguma rota melhor do que a original, mas ainda não satisfatória. O tom mais claro indica a descoberta de uma rota boa. As marcas pretas indicam evoluções reconhecidas pelo algoritmo com a função peso. O gráfico está ordenado pelos resultados mais satisfatórios, da esquerda para a direita.

Por exemplo, a Figura 3a mostra que, segundo a avaliação visual feita, o cenário "1. Treze de Maio" (próximo à metade do eixo de cenários) tinha uma rota obstruída antes da execução do algoritmo. Na segunda iteração começaram a aparecer rotas que desviam de alguns obstáculos, mas não desviam de todos ou são muito longas. E na décima iteração foi descoberta uma rota satisfatória. Segundo a avaliação do algoritmo (marcas pretas) houve evoluções nas iterações 1, 4, 7 e 10, ou seja, o menor caminho no grafo de rotas ponderadas pela função peso mudou nestas iterações. O resultado visto nesta figura refere-se ao algoritmo que usa as três estratégias descritas em 3: mapa de visibilidade, grafo de visibilidade e caminhos livres.



(a)



(b)

Figura 3. (a) resultado obtido com uso das três heurísticas propostas. (b) resultado obtido com pontos promissores aleatórios

A Figura 3b mostra o resultado de quando os pontos promissores são gerados aleatoriamente. Serve como parâmetro de comparação mínimo (*baseline*).

A primeira versão, que usa a melhor estratégia disponível, obteve sucesso em 74% dos cenários. O método proposto melhorou a rota em todos os cenários. A segunda obteve em apenas 36%, um resultado bem menos satisfatório, conforme esperado.

É importante observar que pode não ser possível encontrar uma rota satisfatória em alguns cenários. Considere por exemplo um cenário em que a própria malha viária oferece poucas alternativas de caminhos e todas são obstruídas entre a origem e o destino. Neste caso, não seria possível encontrar rota desobstruída.

Decidimos fazer uma variação do método aleatório. Continuamos gerando pontos aleatórios mas com uma fila comum no lugar da fila de prioridades, ou seja, não ordena as rotas estimadas antes de escolher uma para mandar ao SWMD. Os resultados foram ainda inferiores aos obtidos na primeira estratégia, mas aumentos consideráveis foram conseguidos. Obteve-se cerca de 56% de sucesso. Atribuímos este resultado ao fato de que a fila de prioridades frequentemente escolhe rotas estimadas que tenham trechos reais, induzindo o algoritmo a insistir em aproveitar um trecho real encontrado anteriormente e buscar alternativas para outra parte da rota que esteja em uma região difícil, ao invés de descartar o trecho real e tentar encontrar rotas totalmente diferentes.

No futuro, pretendemos experimentar variações da fila de prioridades que priorizem a escolher de rotas inteiramente alternativas.

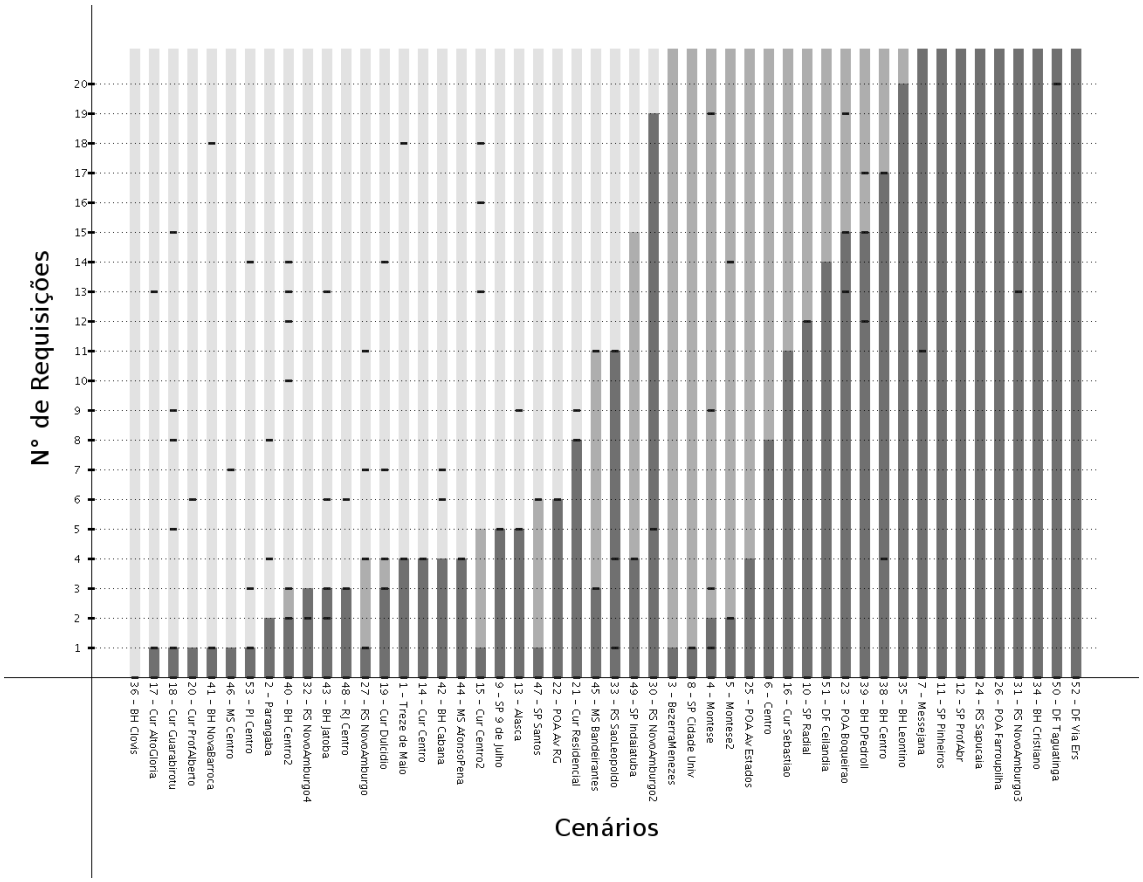


Figura 4. Estimando rotas com pontos aleatórios e sem usar fila de prioridade para ordenar as estimativas.

6. Trabalhos relacionados

Buscas heurísticas para cálculo de caminhos mais curtos é um tema extensivamente estudado em IA e pesquisa operacional. Uma revisão e classificação destes trabalhos pode ser encontrada em [Fu, 2006].

Pesquisas que tratam do desenvolvimento de planejamento de rotas para robôs compartilham igualmente muita similaridade com o problema que estamos atacando. Um exemplo relevante pode ser visto em [Sánchez Miralles, 2004], onde obstáculos são modelados por mapas de distribuição probabilística contínua. O problema e, em especial, a forma como os obstáculos são modelados têm muito em comum com a forma como modelamos a criminalidade no trabalho aqui exposto. No entanto, há uma diferença fundamental no que diz respeito à liberdade de movimento, já que somos limitados a traçar as rotas pelas ruas que só são conhecidas gradualmente a partir das chamdas ao SWMD.

Outros trabalhos relacionados neste contexto buscam usar o conceito de visibilidade como se o ambiente fosse um campo 2d vigiado por sensores e com a presença de obstáculos. [Senjuti, 2007], por exemplo, utiliza formas especializadas dos diagramas de Voronoi e Delaunay para a geração de potencias campos visíveis.

7. Conclusão

Apresentamos o progresso alcançado por nossa pesquisa em estratégias para encontrar rotas que desviam de obstáculos usando Serviços Web de Mapas Digitais. Foram descritas três heurísticas para estimativa de rotas alternativas visando minimizar as requisições remotas e também a estrutura de algoritmo que as põe em prática. Os testes realizados apontam um sucesso significativo e o potencial de melhoria. A aplicação do método está sendo feita no sistema WikiCrimes e está permitindo calcular uma rota segura a partir do aparelho celular.

Diversas linhas de trabalho futuro serão seguidas. A principal delas é inserir um fator estocástico no método, de forma que problemas de máxima local sejam minimizados. O *ranking* da fila de prioridade é um desses alvos para se introduzir a aleatoriedade.

Agradecimentos

Este trabalho foi parcialmente financiado pelo CNPq (559977/2010 e 306266/2008) e FUNCAP.

Referencias Bibliográficas

Cormen, Thomas H. et al. (2001). “Section 24.3: Dijkstra's algorithm”. In: “Introduction to Algorithms”, 2nd ed. MIT Press and McGraw Hill. pp. 595-601.

Dobrin, Adam (2005). “A Review of Properties and Variations of Voronoi Diagrams”, Whitman College.

Fu, L., Sun D. and Rilett L. R. (2006) “Heuristic shortest path algorithms for transportation applications: state of the art”, In: Computers and Operations Research, Volume 33, Issue 11, Elsevier Science Ltd., Oxford, UK.

Furtado, Vasco, Ayres, Leonardo, de Oliveira, Marcos, Vasconcelos, Eurico, Caminha, Carlos, D'Orleans, Jonathan, Belchior, Mairon. (2010) "Collective Intelligence: The WikiCrimes System. Information Science", v(180), 1.

Ishikawa et al (1991): "Map Navigation Software of the Electro Multivision of the '91 Toyota Soarer". In: IEEE Int. Conf. Vehicle Navig.Inform. Syst. pp. 463–473.

Joseph O'Rourke (1993) "Computational geometry column 18", In: ACM SIGACT News, Volume 24, Issue 1, New York NY, USA.

Lima, Natan Costa and Ferreira, Carlos Eduardo (2008). "Algoritmos e estrutura de dados para problemas de deslocamento no plano", In: IV Simpósio de Iniciação Científica e Pós-Graduação, Instituto de Matemática e Estatística, USP, São Paulo, SP, Brasil.

McLafferty (2000). "Identification, development and implementation of innovative crime mapping techniques and spatial analysis". Washington, D.C: U.S. Department of Justice, p.27.

Merril Duane. "Mashups: The new breed of Web app" at http://www.ibm.com/developerworks/xml/library/x-mashups/index.html

Rosen, Kenneth H. "Handbook of Discrete and Computational Geometry and its Applications".

Rouse, J., Bergeron, S.J., Harris, T.M., "Participating in the geospatial web: collaborative mapping, social networks and participatory GIS," in Scharl, A., Tochtermann, K. (Eds.), The Geospatial Web: How Geobrowsers, Social Software and the Web 2.0 are Shaping the Network Society, Springer, New York, NY, pp. 153-8, 2007

Sánchez Miralles, A. and Sanz Bobi (2004), M. A.: "Global Path Planning in Gaussian Probabilistic Maps", Journal of Intelligent and Robotic Systems, Volume 40, Issue 1, Kluwer Academic Publishers, Hingham, MA, USA.

Senjuti, B. R., Gautam, D. and Sajal D. (2007) "Computing Best Coverage Path in the Presence of Obstacles in a Sensor Field", In: WADS.

BLMSumm – Métodos de Busca Local e Metaheurísticas na Sumarização de Textos

Marcelo Arantes de Oliveira[1], Marcus Vinícius Guelpeli[2]

[1]Centro Universitário de Barra Mansa (UBM)
Barra Mansa, RJ - Brasil

[2]Universidade Federal Fluminense (UFF)
Niterói, RJ - Brasil

marceloarantes19@gmail.br, mguelpeli@ic.uff.br

Abstract. *This paper describes the BLMSumm, a summarizer that can use local search methods and Metaheuristcs to create a summary. The summarization problem is presented as optimization problem and a architecture to identification and choice of attributes, based on bipartite graphs, is shown in this work.*

Resumo. *Este artigo descreve o BLMSumm, um sumarizador que tem por objetivo a utilização de diversos métodos de busca local e Metaheurísticas na construção de sumários. O problema de sumarização é apresentado como um problema de otimização e uma arquitetura para identificação e escolha de atributos utilizando grafos bipartidos também é apresentada aqui.*

1. Introdução

A sumarização é um processo que busca criar uma versão reduzida do texto original (Mark e Maybury 1999). Sumário é um resumo que tem o objetivo de encontrar a idéia principal do autor e representar esta em poucas linhas (Hutchins 1987). Sumários são textos reduzidos que transmitem as idéias principais e mais importantes de um texto original, sem perda da informatividade (Delgado *et al* 2010).

A taxonomia dos sumários pode ser dividida em: indicativos, informativos e sumários de críticas. Os sumários indicativos podem ser úteis para descobrir a idéia principal do autor mostrando-a ao leitor, caso haja interesse em aumentar o conhecimento sobre o tema, busca-se a leitura do texto original. Os sumários informativos possibilitam ter todas as informações, não havendo a necessidade de remeter-se ao texto original. Os sumários de críticas são comparativos (Hutchins 1987).

Processos automáticos de sumarização podem obter sumários (abordagem profunda) e extratos (abordagem superficial). Os sumários alteram a estrutura das frases originais reescrevendo-as com a finalidade de generalizar ou especificar as informações. Já os extratos seguem transposições das frases dos textos originais, e por algum tipo de método, escolhem as frases com maior relevância no texto e as colocam no extrato (Guelpeli *et al* 2008).

O maior problema da área de Sumarização Automática (SA) é gerar um sumário que não perca sua informatividade[1]. Sendo assim a avaliação de SA é muito subjetiva, pois depende fundamentalmente da avaliação humana e não existe um consenso entre os avaliadores humanos frente aos resultados obtidos com os sumarizadores automáticos.

A representação do texto como um grafo bipartido e a representação do problema de sumarização automática como um problema de otimização combinatória são as inovações apresentadas e serão explicadas com detalhes em seções posteriores.

Este trabalho está organizado como segue. A Seção 2 apresenta modelo utilizado na aplicação dos métodos de melhoria iterativa em sumarização. A Seção 3 descreve a metodologia, o corpus utilizado, a forma de avaliação e os testes estatísticos. Na Seção 4 os experimentos e os resultados obtidos são descritos. Na Seção 5 o trabalho é concluído sugerindo trabalhos futuros.

2. Modelo BLMSumm

A idéia deste trabalho é apresentar um modelo que torne possível a utilização de heurísticas e metaheurísticas na solução do problema de geração de sumários automáticos, o BLMSumm. A Figura 1 apresenta o modelo desenvolvido, e seus detalhas são descritos nesta seção.

O BLMSumm, é composto por três fases: Pré-processamento, Processamento e Pós-Processamento.

2.1. Pré Processamento no BLMSumm

No BLMSumm, a fase de Pré-Processamento é executado em duas etapas: preparação e geração de um grafo bipartido que tem por finalidade fazer a identificar e seleção dos atributos como descrito na seção 2.1.2. O pré-processamento entrega a fase de Processamento uma lista de sentenças devidamente classificadas e "valoradas" a partir do grafo bipartido.

[1] Informatividade são as informações veiculadas através dos textos escritos ou visuais. A informatividade de um texto é medida de acordo com o conhecimento de mundo das pessoas a que ele se destina. Ou seja, um texto possui um alto grau de informatividade quando a compreensão mais ampla desse texto depender do repertório cultural do leitor. Um texto é mais informativo quanto menor for sua previsibilidade, e vice-versa. Para que haja sucesso na interação verbal, é preciso que a informatividade do texto seja adequada ao interlocutor.

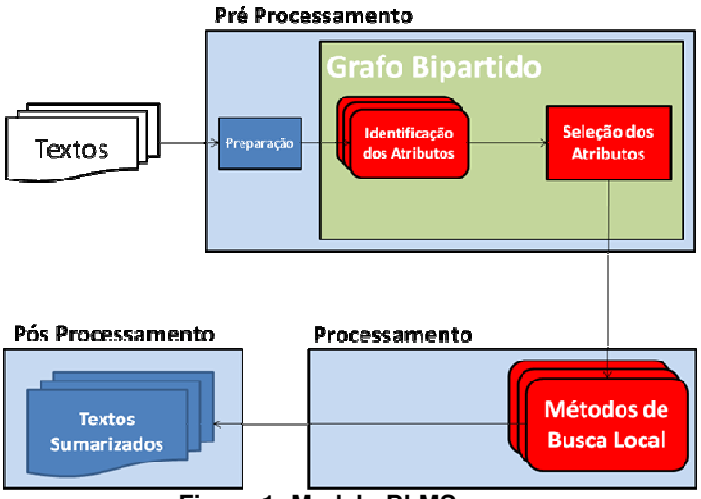


Figura 1: Modelo BLMSumm

2.1.1. Preparação

Foram utilizadas as técnicas de case *folding*, que consiste em transformar todas as letras para minúsculas (ou maiúsculas) e *stopwords*, que consiste na retirada de artigos, preposições e palavras pertencentes às classes abertas. Ao terminar a preparação do texto, um grafo bi-partido é gerado para auxiliar no processo de identificação e seleção dos atributos.

2.1.2. Grafo Bipartido

Um grafo $G=(V, E)$ consiste de um conjunto não vazio de vértices (V) e um conjunto de arestas (E), cada aresta sendo um conjunto contendo dois vértices de V. Um grafo Bipartido é o particionamento de V em dois subconjuntos V_1 e V_2 tais que qualquer aresta possui uma extremidade em V_1 e a outra em V_2 (Drozdek 2001).

No grafo bipartido gerado, o conjunto de vértices $V_1 \in V$ é dado pelo conjunto de sentenças do texto, enquanto o conjunto de vértices $V_2 \in V$ é dado pelo conjunto de palavras válidas do texto (palavras que não fazem parte do conjunto de *stopwords*). Uma aresta $E_x \in E$ é traçada no grafo para determinar que uma palavra faz parte de uma sentença. Considere o texto da Figura 2.

1. **Luxemburgo rebate especulações.**
2. **Em momento algum existiu veto ao Adriano.**
3. **O problema de Adriano é na questão filosófica.**
4. **Luxemburgo afirma que momento não cabe a discussão sobre Adriano: "Não sei o por que dessa discussão no momento."**
5. **Luxemburgo afirma que Adriano ainda pode ser contratado.**

Figura 2: texto fictício baseado na reportagem de globoesporte.globo.com em 18/03/2011

O texto apresentado na Figura 2 gera o grafo bipartido apresentado na Figura 3, onde $V_1 = \{1, 2, 3, 4, 5\}$ e $V_2 = \{$Luxemburgo, rebate, especulações, momento, existiu, veto, Adriano, problema, questão, filosófica, afirma, cabe, discussão, sei, porque, pode,

ser, contratado}, as arestas são: *E* = {(1, Luxemburgo), (1, rebate), (1, especulações), (4, Luxemburgo), (2, momento), (2, existiu), ...}.

Se uma palavra ocorre mais de uma vez em uma sentença, é adicionada a freqüência desta ocorrência na aresta que a representa, como nas arestas: {(4, discussão), (4, momento)}.

2.1.2.1. Identificação dos Atributos

O identificador de atributos do BLMSumm percorre o grafo bipartido determinando o valor de cada palavra e/ou sentença a partir de um cálculo de freqüência previamente estabelecido. Neste trabalho foram utilizados dois tipos de cálculo de frequência de palavras:

1. Cada sentença terá o valor do somatório das frequências de suas palavras.
2. Cada sentença terá o valor do somatório do peso de suas palavras. As palavras da primeira sentença pesam dez vezes sua freqüência e as demais possuem o peso coincidindo com sua freqüência no texto.

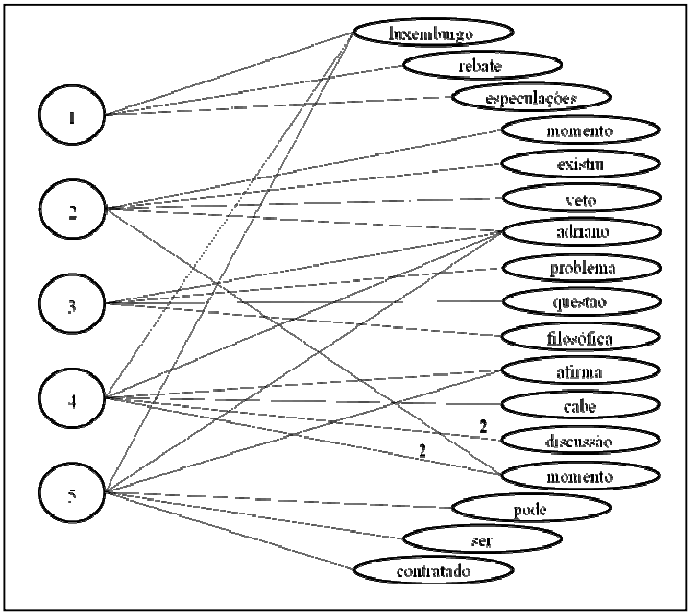


Figura 3: grafo bipartido que representa o texto: à esquerda V1 e à direita V2.

As tabelas, Tabela 1 e Tabela 2 apresentam a aplicação dos cálculos de frequência de palavras ao texto apresentado na Figura 2.

Tabela 1: Cálculo de Freqüência 1 aplicada ao texto.

Sentença	Total de Palavras na Sentença	Lista de sentenças valoradas pelo cálculo de freqüência 1
1	3	3 + 1 + 1 = 5
2	7	3 + 1 + 1 + 4 = 9
3	8	1 + 4 + 1 + 1 = 7
4	19	3 + 2 + 3 + 3 + 1 + 2 + 2 + 4 = 20
5	8	3 + 2 + 4 + 1 + 1 + 1 = 12

Tabela 2: Cálculo de Freqüência 2 aplicada ao texto.

Sentença	Total de Palavras na Sentença	Lista de sentenças valoradas pelo cálculo de freqüência 2
1	3	30 + 10 + 10 = 50
2	7	3 + 1 + 1 + 4 = 9
3	8	1 + 4 + 1 + 1 = 7
4	19	30 + 2 + 3 + 3 + 1 + 2 + 2 + 4 = 47
5	8	30 + 2 + 4 + 1 + 1 + 1 = 39

É possível utilizar outros cálculos de frequência que levem em consideração: a localização da palavra, a localização de sentenças, a relação entre as frequências e localização de sentenças e palavras do texto, Frequência Relativa, Termo de Frequência Inversa por Sentença (TF-ISF), *Ranking* pela frequência de termos (RTF), *Ranking* pela frequência de Sentença (RFS).

2.1.2.2. Seleção dos Atributos

Após a etapa de identificação do atributo começa a fase de seleção, esta fase caracteriza pela escolha de um subconjunto a partir de um conjunto. Este subconjunto permanece fiel a sua posição original melhorando o entendimento do modelo gerado (Liu e Yu 2005). Para um conjunto de dados que possui *n* atributos, existem 2^n - 1 possibilidades de obter um subconjunto. Fica claro que na seleção de atributos existem cortes, que ocorrem mediante algum critério segundo (Dash e Liu 1997).

Na proposta deste trabalho não existe a criação de subconjuntos, mas sim uma lista de sentenças valoradas por algum cálculo de frequência adotado no item 2.1.2.1. A utilização de um grafo bipartido, para que a identificação e a seleção de atributos sejam feitas, proposta neste trabalho torna possível a utilização desta configuração.

2.1.3. Processamento no BLMSumm

Na fase de processamento tem a tarefa de escolher as sentenças que farão parte do sumário, entregando à fase pós processamento a lista de sentenças pertencentes ao sumário.

2.1.3.1. Sumarização como um problema de otimização

Considerando que cada sentença no sumário receberá um valor que determine a "importância" da mesma em função do texto, escolher o melhor sumário pode ser considerado como o processo de escolher o conjunto de sentenças de maior "importância" do texto, sem que a quantidade de palavras deste conjunto ultrapasse uma taxa de compressão. Neste caso, a sumarização superficial pode ser vista como um problema de otimização:

$$\text{Maximize} \sum_{i=1}^{n} h_i b_i \qquad (1)$$

$$\text{Sujeito a} \sum_{i=1}^{n} p_i b_i \leq C \qquad (2)$$

Onde:

- n – é o número total de sentenças em um texto.
- i – é a i-ésima sentença de um texto.
- p_i – é a quantidade de palavras da i-ésima sentença de um texto.
- h_i – é o valor da i-ésima sentença de um texto, dada pela função de avaliação.
- b_i – é um valor binário que determina se a i-ésima sentença de um texto, faz parte ($b_i = 1$) ou não faz parte ($b_i = 0$) de um sumário.
- C – é a quantidade máxima de palavras permitida a um sumário, que é dada em função da taxa de compressão.

Assim, cada estado é um sumário e pode ser visto como uma lista de sentenças, um número binário e/ou um número inteiro. Problemas

2.1.3.2. Métodos de Busca Local e Metaheurísticas

Algoritmos de busca local são construídos como uma forma de exploração do espaço de busca. Nestes um estado inicial é gerado por algum método (construtivo ou aleatório) e este é melhorado a cada iteração até que um estado ótimo seja atingido. Métodos de busca local tendem a ficar presos em ótimos locais, não gerando, conseqüentemente, uma solução ótima global para um determinado problema. As Metaheurísticas são desenvolvidas para fugirem destes ótimos locais e convergirem para uma solução ótima global.

2.1.3.2.1. Algoritmos Gulosos

Um algoritmo guloso sempre faz a escolha que parece ser a melhor no momento, ou seja, ele faz uma escolha ótima para as condições locais na esperança de que esta escolha leve a uma solução ótima para a situação global (Cormen e Leiserson 2002).

O algoritmo guloso escolhe a melhor solução localmente, sendo assim, a cada iteração o algoritmo seleciona a sentença de maior valor e, se o somatório de palavras contido no sumário atual adicionado à quantidade de palavras da sentença não ultrapassar à taxa de compressão, esta é adicionada.

Os cálculos de freqüência apresentados na seção 2.1.2.1., geram os valores das sentenças que, serão tratadas aqui como função de avaliação.

Considerando a taxa de compressão de 46%, ou seja, 24 palavras permitidas de um total de 45, e a função de avaliação 1 (cálculo de freqüência 1: seção 2.1.2.1) em associação com o algoritmo guloso escolheria as seguintes sentença:

1. Sentença 4: 19 palavras e 20 pontos;
2. Sentença 1: 3 palavras e 5 pontos;

Que totaliza de 22 palavras e 25 pontos (número binário correspondente: 01001).
Entretanto, outra combinação de sentenças poderia ser mais eficiente:

1. Sentença 2: 7 palavras e 9 pontos;
2. Sentença 3: 8 palavras e 7 pontos;
3. Sentença 5: 8 palavras e 12 pontos;

Que totaliza de 23 palavras e 28 pontos (número binário correspondente 10110)

Esta solução não seria encontrada pelo algoritmo guloso utilizando a função de avaliação acima.

2.1.3.2.2. *Simulated Annealing*

A "localidade" empregada em algoritmos como o algoritmo Guloso, pode levar a soluções excelentes localmente, ou seja, a máximos locais que, em muitos casos, podem estar distantes da solução ótima para um determinado problema.

O *Simulated Annealing* é uma técnica de maximização / minimização que imita o processo de recozimento. A cada iteração o *Simulated Annealing* gera uma possível solução, se esta é a melhor do que a armazenada anteriormente, ela passa a ser a opção corrente. Existe a probabilidade de uma solução pior que a atual ser escolhida como corrente, com isto, o *Simulated Annealing* tenta fugir de máximos locais (Raynor 1999).

Cada sumário é representado por um número binário os bits válidos determinam que sentenças fazem parte do sumário. A cada iteração um número inteiro é escolhido aleatoriamente e convertido para binário, próximo sumário, este número é então associado à solução que ele representa, são verificadas as quantidades de palavras (se o número de palavras é maior que a taxa de compressão, a solução é descartada) e a pontuação do sumário (levando em consideração a função de avaliação). Se o próximo sumário tiver pontuação maior que o sumário corrente, então ele passa a ser o sumário corrente, caso contrário existe uma probabilidade de o próximo sumário se tornar o sumário corrente, esta probabilidade diminui com o tempo.

2.1.4.Pós Processamento no BLMSumm

O Pós-processamento, por sua vez, gera um arquivo texto sumarizado a partir das sentenças recebidas da fase de Processamento. Estes textos são os resultados dos algoritmos de sumarização. Para averiguar a grau de informatividade[1] foi utilizado o pacote ROUGE, que será descrito no item 3.3 deste trabalho

3. Metodologia

A contribuição deste trabalho é a de que métodos de busca local e metaheurísticas podem gerar sumários automáticos extrativos de qualidade similar aos sumarizadores presentes na literatura.

Para demonstrar esta contribuição, os métodos de busca foram implementados no BLMSumm e a seguinte metodologia de simulação foi empregada: foram usados corpus que são descritos no item 3.1 deste trabalho. Foram escolhidos sumarizadores que são detalhados no item 3.2. No aferimento da hipótese foi escolhido o pacote ROUGE que usa métricas *Recall*, *Precision* e *FMeasure* que são comumente usadas na mensuração do processo de sumarização e comentadas detalhadamente no item 3.3.

Foram definidos percentuais de compressão para serem usados nos sumarizadores. Cada texto foi sumarizado pelos algoritmos de sumarização com 50%, e 80%

3.1. Corpus

O corpus utilizado neste trabalho, foi o TeMário, com 100 textos extraídos de dois grandes jornais nacionais Folha de São Paulo e Jornal do Brasil. Eles têm as seguintes categorias: Especial, Mundo, Opinião da Folha de São Paulo e, Internacional e Política do Jornal do Brasil (Pardo e Rino 2004). Todos os textos possuem sumários de

referência criados por especialistas humanos. A estatística do corpus pode ser observada na Tabela 3.

Tabela 3. Mostra a estatística dos 100 textos do domínio jornalístico composto por 5 categorias e no idioma Português.

Itens	Nº de Palavras	Nº de Palavras + Numeral	Nº (%)	Caracteres	Caracteres + Espaços	Sentenças	Palavra por Sentença		Caracteres por Palavra	
Mínimos	424	431	0	2.160	2.649	14	424	14	2.160	431
Máximos	1.310	1.350	10,29	6.827	8.155	71	1.310	71	6.827	1.350
Totais	62.014	63.132	-	323.919	391.014	3.212	62.014	321.219,31	323.919	63.132
Médias	620,14	631,32	1,77	3.239,19	3.910,14	32,12	19,31		5,22	

3.2. Algoritmos de Sumarização

Para a simulação foram escolhidos sumrizadores da literatura. Como critério para escolha dos algoritmos de sumarização destes experimentos foram definidos aqueles que tinham a possibilidade de definir percentuais de compressão por palavra, para isto, deveria possibilitar uma compressão de 50% e 80% do texto original.

3.2.1. Gist Average Keyword

No Gist_Average_Keyword (Pardo 2006) - a pontuação de sentenças pode ocorrer por um de dois métodos estatísticos simples: métodos *keywords* ou o método *average keywords*, isto é, o método *keywords* com normalização em função do tamanho das sentenças (medido em número de palavras) e depois ocorre o *ranqueamento* das sentenças em função da pontuação obtida, sendo que a sentença de maior pontuação é eleita gist *sentence*, isto é, a sentença que melhor representa a idéia principal do texto. Selecionam-se as sentenças que: (a) contenham pelo menos um *stem* em comum com a gistsentence selecionada na etapa anterior e (b) tenham uma pontuação maior do que um *threshold*, que é a média das pontuações das sentenças. Por (a), procura-se selecionar sentenças que complementem a idéia principal do texto; por (b), procura-se selecionar somente sentenças relevantes, Isto baseado no percentual de compressão determinado.

3.2.2. Gist Intrasentença

No Gist_Intrasentença (Pardo 2006) - é realizada em todas as sentenças pela exclusão das *stopwords* com a finalidade de redução da dimensionalidade.

3.2.3. Supor

O Supor (Módolo 2003) - seleciona para compor o extrato as sentenças que incluem as palavras mais frequentes do texto-fonte, pois elas representariam os conceitos mais importantes desse texto. A escolha de cada sentença é feita mediante a classificação de sua representatividade no texto. Para isso, é atribuído um escore a cada sentença, baseado na soma das frequências de suas palavras em todo o texto. Feito isso, determina-se um escore de corte (*threshold*) com base em medidas estatísticas e são, então, selecionadas as sentenças com as palavras mais frequentes do texto.

3.3. Avaliação dos sumários gerados

ROUGE (*Recall Oriented Understudy for Gisting Evaluation*) é, atualmente, uma das maneiras mais utilizadas para avaliar sumários automáticos é a medida (Lin e Hovy 2003). Basicamente, ROUGE é um pacote de avaliação de sumários, que surgiu para possibilitar a comparação direta entre um SA e o respectivo sumário humano usando como métrica n-gramas. Com essa ferramenta é possível analisar a proximidade de qualidade dos SA em relação aos sumários de referência humano. De uma forma geral, a ROUGE calcula o grau de informatividade dos extratos. ROUGE faz o cálculo baseando-se nos conjuntos de palavras em sequência (n-gramas) em comum entre os sumários de referência humanos e os Sumários Automáticos. Quanto maior o número de palavras em comum entre os sumários, maior será a nota dada ao sumário.

Para a validação dos resultados foram utilizados os testes estatísticos ANOVA de Friedman e coeficiente de concordância de Kendall que são empregados em conferências internacionais de avaliação de Sumários Automáticos como TAC (*Text Analysis Conference*), a mais relevante dentro da área de SA. Estes testes estatísticos verificam as diferenças de resultados existentes entre todas as amostras.

O teste ANOVA de Friedman é não paramétrico, seu o objetivo é gerar uma ordenação entre as linhas observando o *ranking* de cada amostra (coluna), avalia-se, desta forma, se existe diferença significativa entre todas as amostras presentes no experimento.

O coeficiente de concordância de Kendall é não paramétrico e é utilizado para normalizar o ANOVA de Friedman. Sua finalidade é gerar um grau de concordância com o *rank* estabelecido. Quanto mais próximo de 0, menor é a concordância e quanto mais próximo de 1, maior é a concordância.

4. Experimentos

O BLMSumm foi executado utilizando os cálculos de frequência (funções de avaliação) apresentadas na seção 2.1.2.1. Para cada cálculo de frequência foram aplicados os métodos abaixo com os seus respectivos nomes detalhes e referência nos experimentos:

- Guloso: o Guloso é representado pelas referências gu1 e gu2 nos experimentos: (os números: 1 e 2 identificam que cálculo de freqüência utilizados);
- Guloso com Corte: é uma variação do algoritmo guloso no qual as sentenças são ordenadas por seus respectivos valores e a cada iteração a sentença com maior valor heurístico é adicionada ao sumario sendo gerado. Ao encontrar uma sentença cuja quantidade de palavras adicionada às já pertencentes ao sumário sendo gerado ultrapasse a taxa de compressão estabelecida, o algoritmo é interrompido e o sumário é dado como gerado (excluindo a última sentença verificada). O **Guloso com Corte** é representado pelas referências guCorte1 e guCorte2 nos experimentos;
- *Simulated Annealing*: a agenda utilizada para possui 10000 posições tendo seus valores variando entre 0 e 373. O ***Simulated Annealing*** é representado pelas referências sa1 e sa2 nos experimentos;

4.1. Analise dos Resultados

A figura 4 apresenta os resultados produzidos pela avaliação baseando-se em sumários de referência humanos através do pacote ROUGE para compressão de 50%.

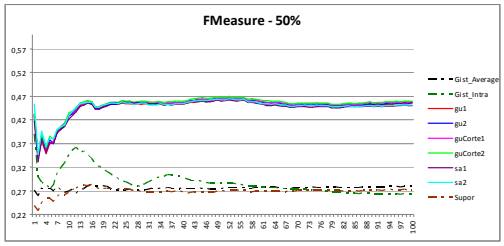


Figura 4: Resultados baseados na avaliação Rouge com compressão de 50%.

De acordo com a Figura 4, os métodos propostos neste trabalho obtiveram os melhores resultados. A pior avaliação foi do método Gist-Intrasentença. Os métodos da literatura, representados pelas linhas pontilhadas estão abaixo dos métodos propostos neste trabalho. Analisando a Figura 4, observa-se que não houve algoritmo com diferença significativa. Sendo assim, foram utilizados os testes estatísticos de ANOVA de Friedman e o coeficiente de concordância de Kendall para verificar a diferença entre as amostras.A Tabela 4 apresenta valores discordantes com a Figura 4 no método com o pior rendimento. Pelos testes estatísticos apresentados na Tabela 4 o pior desempenho foi atribuído ao método *Gist_Intrasentença* com 1,64 valor de ordem média. Conforme apresentado na Tabela 4 observa-se que os valores de ordem média estão a 65,5% dos 100% de concordância do coeficiente de Kendall do ranking estabelecido pelo teste estatístico ANOVA.

Tabela 4: Teste estatístico com Extratos Ideais.

Comparando amostras múltiplas relacionadas			
N	100	Graus de liberdade	8
qui-quadrado	524,7130376	p-nível	0
Coef. de concordância de Kendall	0,655891297	Ordem média	0,652415452
	Ordem média	Soma de ordens	Média
Gist_Average_Keyword	2,55	255	0,2807051
Gist_Intrasenteca	1,64	164	0,2623243
Guloso1	6,14	614	0,4551813
Guloso2	6,365	636,5	0,4567969
GulosoCorte1	6,94	694	0,4586951
GulosoCorte2	7,11	711	0,4598168
SimAnn1	5,93	593	0,450588
SimAnn2	6,175	617,5	0,45224
Supor	2,15	215	0,2721173

A Figura 5 apresenta os resultados produzidos pela avaliação baseando-se em sumários de referência humanos através do pacote ROUGE para compressão de 80%.

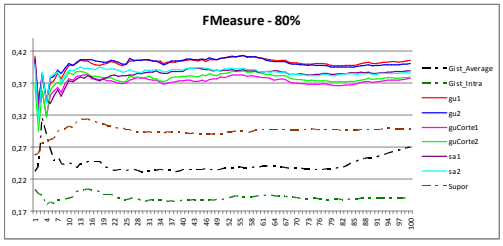


Figura 5: Resultados baseados na avaliação Rouge com compressão de 80%.

De acordo com a Figura 5, os métodos propostos neste trabalho obtiveram os novamente os melhores resultados.

Tabela 5: Teste estatístico com Extratos Ideais.

Comparando amostras múltiplas relacionadas			
N	100	Graus de liberdade	8
qui-quadrado	452,9488965	p-nível	0
Coef. de concordância de Kendall	0,566186121	Ordem média	0,561804162
	Ordem média	Soma de ordens	Média
Gist_Average_Keyword	2,845	284,5	0,2700962
Gist_Intrasenteca	1,125	112,5	0,1903291
Guloso1	7,24	724	0,405653
Guloso2	6,965	696,5	0,4008613
GulosoCorte1	5,415	541,5	0,377407
GulosoCorte2	5,51	551	0,3797337
SimAnn1	6,22	622	0,3895002
SimAnn2	6,11	611	0,3864529
Supor	3,57	357	0,2983482

Os métodos Gist Average,Intra e Supor, representados pelas linhas pontilhadas estão abaixo dos algoritmos propostos neste trabalho. Analisando a Figura 5, observa-se que não houve algoritmo com diferença significativa. Sendo assim, foram utilizados os testes estatísticos de ANOVA de Friedman e o coeficiente de concordância de Kendall para verificar a diferença entre as amostras.Conforme apresentado na Tabela 5 observa-se que os valores de ordem médios estão a 56,6% dos 100% de concordância do coeficiente de Kendall do *ranking* estabelecido pelo teste estatístico ANOVA. Estes valores justificam-se pela proximidade dos valores médios obtidos nos algoritmos propostos neste trabalho

5. Conclusão

Embora simples, o grafo bipartido mantém a estrutura de pontuação fora do escopo do método de solução empregado, isto permite testar diversos cálculos de freqüência na identificação de atributos. Valorar as sentenças e representar o problema de sumarização como um problema de otimização possibilitam a utilização de diversos métodos de busca presentes na literatura.

O BLMSumm trabalha com *stopwords* e para utilizá-lo em outro idioma, a lista de *stopwords* deste deve ser fornecida. A escolha do método de busca é feita pelo usuário. Nenhum algoritmo de treinamento é utilizado.

Apesar do não conhecimento do melhor sumário automático extrativo classificado pelo ROUGE em cada texto do corpus TeMário, constatou-se que o BLMSumm é competitivo com relação aos métodos presentes na literatura para este domínio, como visto nas Figuras 4 e 5.

5.1. Trabalhos Futuros

Utilizar o grafo bipartido para gerar diversas simulações com diferentes cálculos de freqüências, variar métodos de busca local, implementar métodos de busca populacional, variar domínios e idiomas, testar em diversas compressões.

Encontrar os melhores sumários, a partir do corpus Temário, classificados pelo ROUGE.

6. Referências Bibliográficas

Cormen, T. H., and Leiserson C. E. (2002) "Algoritmos - Teoria e Prática*." Segunda Edição.* Rio de Janeiro, RJ: Campus.

Dash, M., and Liu H. (1997) "Feature Selection for Classification." *Inteligence Data Analise*, no. 1 131-136.

Delgado, C. H., Vianna C. E., and Guelpeli, M. V. C.(2010) "Comparando sumários de referência humano com extratos ideais no processo de avaliação de sumários extrativos." *IADIS Ibero-Americana WWW/Internet*, 293-300.

Drozdek, A. (2001) "Data Structures and Algorithms in C++*", 2nd edition.* Pacific Grove, CA - USA: Brooks / Cole.

Guelpeli, M. V. C., Bernardini F. C., and Garcia A. C. B. (2008) "Todas as Palavras da Sentença como Métrica para um Sumarizador Automático.".

Hutchins, J. (1987) "Summarization: some problems and methods." The frontier of informatics, : 151-173.

Lin, C., and Hovy, E. H.(2003) "Automatic Evaluation of Summaries Using N-gram." Proccedings of the Language Technology Conference.

Liu, C. Y., and Yu L. (2005) "Toward Integrating Feature Selection Algorithms for Classification and Clustering." IEEE Transactions on Knowledge and Data Engineering.

Mark, I., and Maybury M. T. (1999) "Advances in automatic text summarization*."* Cambrige, USA: MIT Press.

Módolo, M. (2003) "SuPor: um Ambiente para a Exploração de Métodos Extrativos para a Sumarização Automática de Textos em Português." Dissertação de Mestrado - UFSCar.

Pardo, T. A. S. (2006) "Estrutura textual e multiplicidade de tópicos na sumarização automática: o caso do sistema GistiSumm." Relatório Técnico (NILC - ICMC - USP).

Pardo, T. A. S., and Rino, L. H. M. (2004) "TeMário: Um Corpus para Sumarização Automática de Textos." Relatório Técnico (NILC - ICMC - USP).

Raynor, W. (1999) "The International Dictionary of Artificial Intelligence*."* New York, NY - USA: Amacon.

Multi-Document Summarization Using Complex and Rich Features

Maria Lucía del Rosario Castro Jorge, Verônica Agostini, Thiago Alexandre Salgueiro Pardo

Núcleo Interinstitucional de Linguística Computacional (NILC)
Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo
Avenida Trabalhador são-carlense, 400. 13566-590 - São Carlos/SP, Brazil

{mluciacj,agostini,taspardo}@icmc.usp.br

Abstract. Multi-document summarization consists in automatically producing a unique informative summary from a collection of texts on the same topic. In this paper we model the multi-document summarization task as a problem of machine learning classification where sentences from the source texts have to be classified as belonging or not to the summary. For this aim, we combine superficial (e.g., sentence position in the text) and deep linguistic features (e.g. semantic relations across documents). In particular, the linguistic features are given by CST (Cross-document Structure Theory). We conduct our experiments on a CST-annotated corpus of news texts. Results show that linguistic features help to produce a better classification model, producing state-of-the-art results.

1. Introduction

Recently, new technologies have made available a large amount of textual information in digital format. Consequently, the capability to process all this information is reduced and for this reason Multi-Document Summarization (MDS) can be a useful resource. MDS is conceived as the automatic production of a summary given a group of texts on the same topic (McKeown and Radev, 1995; Radev and McKeown, 1998). The idea of this task was imagined as an extension of a previous area called single-document summarization or simply Automatic Summarization (AS), which consists in generating a summary from one text.

In both AS and MDS, the summary should ideally contain the most relevant information of the topic that is being discussed in the source texts. As an example, Figure 1 shows a summary extracted from the work of Jorge and Pardo (2010b). This summary was automatically produced from three texts telling the victory of Brazil at the volleyball world league. These texts are part of CSTNews corpus (Aleixo and Pardo, 2008; Maziero et al., 2010), which contains news texts written in Brazilian Portuguese. The summary was translated from Portuguese.

The Brazilian volleyball team won on Friday the seventh consecutive victory in the World League, defeating Finland by 3 sets to 0 - partials of 25/17, 25/22 and 25/21 - in a match in the Tampere city, Finland. The first set remained balanced until the middle, when André Heller went to serve. In the last part, Finland again paired the game with Brazil, but after a sequence of Brazilians points Finland failed to respond and lost by 25 to 21. The Brazilian team won five times the World League in 1993, 2001, 2003, 2004 and 2005.

Figure 1. Multi-document Summarization example (Jorge and Pardo, 2010b)

Moreover, MDS should not only focus on the extraction of relevant information, but also deal with challenges, such as redundancy, complementary and contradictory information, different writing styles and varied referential expressions. These are factors that are more likely to happen when there are various sources of information.

There are two ways of approaching MDS as described by Mani and Maybury (1999). The first one is the superficial approach in which statistical or little linguistic information is used to build the summary. Classical examples of this approach include methods based on word counting. The second approach is deep and is characterized by the usage of deep linguistic knowledge such as grammars, semantic or discourse information. The deep approach is said to produce summaries of higher quality in terms of information, coherence and cohesion, but it demands various high-cost resources. On the other hand, superficial approach requires low-cost processing, but summaries produced by models under this approach tend to have lower quality. In summarization, there are various researches for both approaches, varying from simple sentence selection based on word frequency (Luhn, 1958; Edmundson, 1969) to more complex methods that combine machine learning techniques and textual features (Kupiec et al., 1995), or methods that include more sophisticated linguistic knowledge such as CST (Cross-document Structure Theory) (Radev, 2000; Zhang et al. 2002; Otterbacher et al. 2002; Afantenos et al. 2004; Jorge and Pardo 2009, 2010a, 2010b).

According to Mani and Maybury, MDS should also take three phases: the first one is analysis, in which an intern representation of the texts is created; the second phase is called transformation, in which relevant content is selected given the representation in the previous phase; finally, the third phase is synthesis, where all the selected content is expressed in natural language. Given this context, this work is focused on the second MDS phase, where content selection is done. Particularly, we present a machine learning approach for content selection by representing this task as a supervised classification problem. For this aim, sentences are treated as instances for classification, in which classes indicate whether a sentence should be included in the summary or not.

We combine simple features, such as sentence position and sentence size, with more sophisticated linguistic features, such as semantic relations between sentences from different texts. In our case, these semantic relations are given by the CST model. CST is a multi-document model that provides a set of 24 semantic relations that aim to be applied across documents on the same topic. Our proposal is a novel approach for MDS, since it is the first work that treats MDS as a classification problem by including deep features. It is important to point out that, at the moment, CST information has been shown to be the most sophisticated information for MDS of texts written in Brazilian Portuguese (Jorge and Pardo, 2009; 2010b).

In order to manage the CST information, sentences and their relations across texts are organized as graphs, were sentences are represented as nodes and CST relations as edges. Having this representation, it is also easy to extract mathematical measures from the graph and treat them like features. A well studied way to extract these mathematical measures is through the concept of complex networks (Costa et al., 2007). In this work we will use the clustering coefficient, which has been previously used by Antiqueira and Nunes (2010) for single-document summarization and showed to be good for sentence extraction. The main hypothesis in this work is that deep linguistic knowledge features combined with simple and complex features will help to predict more accurately sentence classes.

Experiments are conducted over CSTNews corpus, which is a corpus of Brazilian Portuguese news texts already annotated with CST. The corpus also provides human summaries, which will determine the correspondent classes of the sentences in the original texts of the corpus, as well as be the basis for evaluation. Results show a good performance using traditional machine learning classifiers such as Decision Trees, Naïve Bayes, Support Vector Machines and One Rule, producing state-of-the-art results and demonstrating the importance of linguistic features.

This paper is organized as follows: in Section 2, we will make a review on the main related work for MDS and CST; in Section 3, the methodology used to drive the experiments will be described; in Section 4, the evaluation method will be described; in Section 5, results will be discussed and; in Section 6, some final remarks will be made.

2. Multi-Document Summarization

One of the first researches on MDS was the one proposed by McKeown and Radev (1995, 1998) where the authors represent in template format (like an attribute-value table) the information contained in texts. They relate those templates with semantic relations like the ones in CST, and then their system activates operators that combine information according to relations and attributes grades of importance to that information. The most important information is selected for the summary.

Mani and Bloedorn (1997) proposed a method for modeling the multi-document problem consisting in a word graph construction from a set of texts. The authors model, through the graph, similarities, differences and contradictions in texts, and then they walk on the graph to find relevant information according to the topic.

Another important research was introduced by Radev et al. (2000, 2001), who proposed the system MEAD, which creates a rank of sentences by giving a grade according to three basic features: lexical distance to the first sentence of the documents, lexical distance to the title of the documents and lexical distance to the centroid of the documents (which is the sentence or set of sentences that better represent the focus of the topic of the texts). Sentences with the best grades are candidates to compose the final summary.

Radev (2000), besides proposing CST model, also proposed a 4-stage summarization process: in the first stage, clustering of similar texts is done; in the second stage, texts may be internally structured (which can be a syntactic, semantic or discourse structure); in the third stage, CST relations are established among texts units; finally, in the forth stage, the content for the summary is selected by exploring CST relations.

After CST was proposed, some important works used the model for MDS. Zhang et al. (2002) proposed an improvement of MEAD system by re-ranking sentences according to the number of CST relations. Otterbacher et al. (2002) used CST relations to improve cohesion in summaries. Afantenos et al. (2004) proposed a new classification of CST relations considering two main categories: synchronic and diachronic relations. Synchronic relations describe the same event discussed in various sources, and diachronic relations describe the evolution of an event through time. The authors also proposed a summarization method based on pre-defined templates and ontologies. Jorge and Pardo (2009, 2010a, 2010b) explored CST knowledge by applying operators for content selection based on user preferences. Operators apply rules that map user preferences to CST relations in order to build the summary. Results of this work showed to be good for Brazilian Portuguese texts, in terms of informativity, redundancy reduction, cohesion and coherence. Maziero et al. (2010) proposed a refinement of CST relations by reducing the number of relations to 14. The authors also proposed a classification of these 14 relations, which we reproduce in Figure 2. This classification has two main categories: Content and Presentation/form. Content category involves all the relations that refer mainly to the information within two text units. On the other hand, the Presentation/Form category includes relations that refer mainly to the way text units are written. Content Category is divided in three subcategories: redundancy, complement and contradiction, while Presentation/Form category is divided in two subcategories: style and authorship.

For MDS there have not been significant researches based on machine learning using deep linguistic features. For single-document summarization there are some important works that use machine learning to create automatic models for summarization. One interesting work in this line is proposed by Kupiec et al. (1995), where the authors treat the content selection task as a problem of statistical classification. Given texts and their corresponding summaries, a

classification function will determine which sentences have more probability to be in the summary. The authors consider features such as sentence size, most frequent words, paragraph position, uppercase words, and fixed phrase attributes. A Naïve Bayes algorithm is used for classification. Results showed that fixed phrase and paragraph position attributes give a better performance rather than word frequency based attributes.

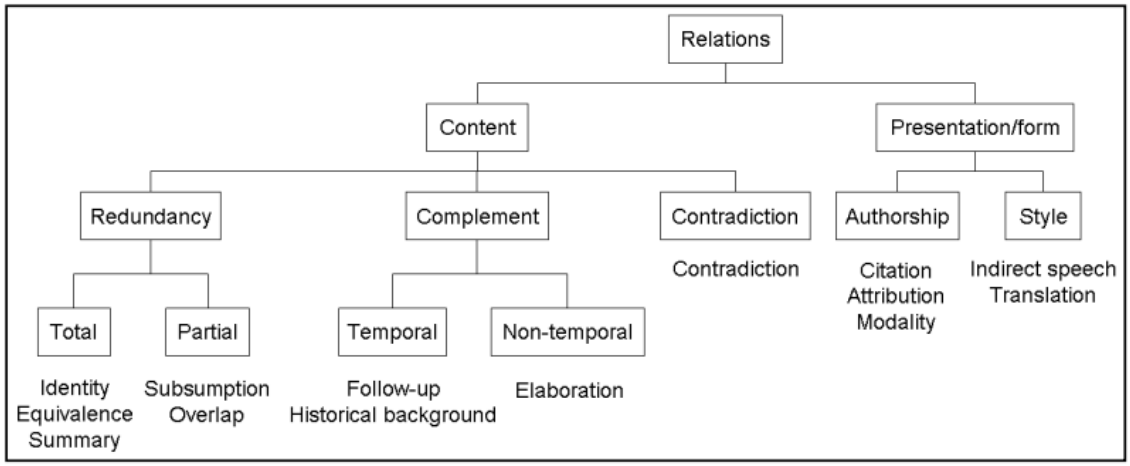


Figure 2. Maziero et al. (2010) classification of CST relations

Mani and Bloedorn (1998) proposed a learning method for generic and user-focused summaries. They used three types of features: location, thematic and cohesion. The authors use C4.5 (Quinlan, 1992) and AQ15c (Wnek et al., 1995) methods. The results showed that rules learned for user focused summaries use more keyword specific features and, for generic summaries, rules tend to use more location features. Chuang and Yang (2000) used two types of features: structured and non-structured features. Structured features correspond to rhetorical relations (Mann and Thompson, 1987) and non-structured features correspond to word frequencies, paragraph number, etc. The authors use decision trees, Naïve Bayes classifier and DistAl neural network (Yang et al., 1999). Results showed that the systems built with those learning algorithms outperformed the commercial Microsoft Word summarizer.

In the next section, we describe the methodology used in this work.

3. Methodology

As it has been mentioned before, the experiments of this work were conducted over the CSTNews corpus, which is composed of 50 news text clusters, and each cluster contains in average 3 documents discussing a common topic. Each cluster also provides the correspondent CST annotation of the sentences and a summary elaborated by humans. For the moment the annotation has been manually done. At the moment, an investigation for an automatic way of detecting CST relations is being developed (Maziero et al., 2010).

Our methodology consists of four stages: graph construction, feature selection, classification, and summary building, which will be described next.

3.1. Graph Construction

In this initial step we construct a graph containing all the CST information, where sentences are represented by nodes and relations are represented by edges. In Figure 3, we show a small example of a hypothetical graph containing CST information. This structure will help us to extract some important features as it will be detailed in Section 3.2.

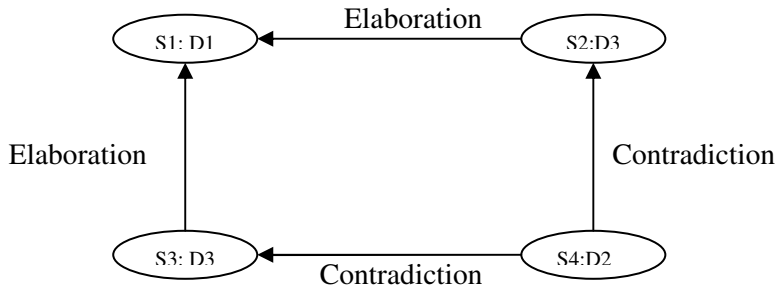


Figure 3. Illustration of a graph containing CST relations

In this hypothetical example, we see that sentence four from document two of the cluster, denoted as S4:D2, contradicts sentences two and three from document three (S2:D3 and S3:D3, respectively); and these two sentences elaborate sentence one from document one (S1:D1).

3.2. Feature Selection

We consider 9 features to describe the sentences of the corpus. We are considering sentences as the segments to be analyzed because the corpus CSTNews is already segmented and annotated at this level.

We divide features in two categories: superficial and deep features. We describe them below.

3.2.1. Superficial Features

This type of feature corresponds to the characteristics that refer to the structure and statistics rather than the semantics of the texts. We list and describe 2 superficial features: sentence size and sentence position:

- Sentence size: describes the size of a sentence in terms of the number of words it contains, excluding the stopwords. This attribute was normalized by diving all the size values of each sentence with the highest size value in the cluster. The final value of this feature results in a number between 0 and 1.

- Sentence position: refers to the position of the sentence in a particular text. This feature can assume three possible values: BEGIN, MIDDLE, and END. BEGIN value corresponds to the first sentence of any text; END value corresponds to the last sentence of any text; and MIDDLE value corresponds to the remaining sentences between BEGIN and END.

3.2.2. Deep Features

Deep features refer to semantic and discourse characteristics of the texts. In our case this characteristics are given by the information provided by CST. We consider seven deep features, three of them are metrics derived from the graph constructed in the first stage of the process; the other four features correspond to the type of relation according to Maziero et al. (2010) classification. All seven features are listed below:

- Centroid: this feature indicates whether a sentence is the most representative of the content of a text or not. According to this idea, this feature can assume two values: yes or no. To measure how representative a sentence can be, we calculate the grade of a node (sentence) in the graph constructed in stage 1. The grade is given by the number of edges (CST relations) that connects the given sentence (node). The sentence with the highest number of edges is considered the centroid.

- Similarity to centroid: once we know which sentence of the text is the centroid, we calculate how similar to the centroid the other sentences are. This similarity is measured by lexical distance, which is calculated by word overlap function. We consider this overlap function as the number of lexical matches between two sentences divided by the sum of the words in the two sentences, which results in a number between 0 and 1.

- Clustering coefficient (CC), which is a well known metric in the theory of complex networks that measures the level that nodes tend to cluster together. As an example, in Figure 4 it can be observed a graph that presents a highly connected group of nodes (nodes 1 to 5).

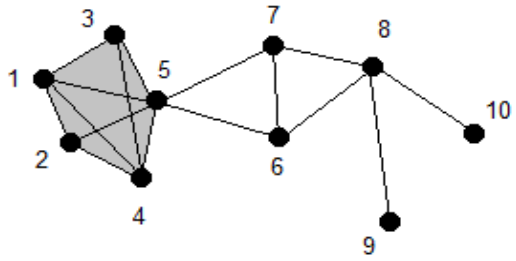


Figure 4. Example of cluster in a graph

In MDS, a graph that models CST information may have clusters in its topology and this can express highly elaborated topics in the texts, which we think are more likely to be in the summary. In other words, CC may be a good feature that can help to predict if a sentence should be or not in the final summary. We choose this measure based on the experiments made by Antiqueira and Nunes (2010) that show that CC is a good measure for AS using complex networks. CC is calculated using the next formula:

$$CC = \frac{3 \times \#triangles}{\#connected\ triples}$$

In this formula, triples refer to three nodes that are either connected by two edges or three edges. Particularly, triangles are defined as three nodes connected by three edges, while simple triples connect three nodes with only two edges. For example, considering the graph represented in Figure 4: it can be observed that nodes 8, 9, and 10 compose a simple triple, since 9 and 10 are not connected by any edge, but nodes 5, 6 and 7 compose a triangle because the three nodes are connected by an edge.

- Redundancy type: in this category are included all the relations that refer to redundant information, considering that redundancy can be partial or total. These relationships are: Identity, Equivalence, Overlap, Summary and Subsumption. Given a sentence in a CST graph, the value of the redundancy feature indicates the number of relations of this type that the sentence has.

- Complement type: in this category are included all the relations that refer to complementary information. This category includes: Historical background, Elaboration and Follow-up. Similarly to Redundancy feature, the value for complement feature indicates the number of relations of this type that the sentence has.

- Contradiction type: it only includes the contradiction relation. The value for this feature is calculated similarly to the redundancy and complement features.

- Authorship type: this category includes all relations that refer to the writing style and authorship information rather than the content of the text. The relations that are part of this

category are: Citation, Attribution, Modality, Indirect-speech and Translation. The value for this feature is calculated the same way as the other CST features.

The value for the last 4 features is normalized by dividing the feature with the total number of relations within the cluster. As an example of how these features are calculated, consider a sentence S that is connected by a Subsumption relation and an Attribute relation in the graph: this sentence will have 1 relation of the redundancy category and 1 of the authorship category. Supposing that these are the only relations in the cluster, the feature vector of sentence S is showed in Figure 5. In this example, features A, R, Comp and Contr correspond to Authorship/style, Redundancy, Complement and Contradiction categories, respectively.

Position	Size	Centroid	CC	Similarity to centroid	A	R	Comp	Contr
…	…	…	…	…	0.5	0.5	0	0

Figure 5. Example of feature vector of Sentence S

The idea behind these features is also that, after training, we may know which types of CST relations help to predict more accurately the classes of the sentences, i.e., which types of relations are more likely to include sentences in the final summary.

In this initial approach, we only considered these 9 features because our goal is to focus on how CST information, combined with superficial information, helps to predict sentence classes. Of course other features might be included, considering, for instance, syntactic, semantic or even ontology-based information, but these will be explored in future works.

3.3. Classification

Having the set of features, we need to determine the correspondent class of each sentence of our corpus. Two possible classes can be assigned: 0 or 1. Sentences classified as 1 represent the ones that should be included in the final summary and sentences classified as 0 represent the ones that should not be included in the final summary. Since for each cluster in the corpus we have the correspondent human summary, we can determine which sentences are in that summary. Our limitation here is that human summaries (abstracts) are not necessarily composed by the same exact sentences of the original texts. In order to solve this limitation, we consider as belonging to class 1 all the sentences that have at least 70% of lexical similarity with any sentence of the human summary; otherwise, class 0 is assumed. This is an initial approach, but for future works we plan to evaluate which threshold is the most suitable when comparing sentences versus summary sentences.

To train our classification models, we used 4 methods: Decision Trees (particularly, algorithm J48), Naïve Bayes, Support Vector Machine (SVM) and One Rule. We use WEKA (Witten and Frank, 2005) for running all these methods.

Since our database is unbalanced (because the number of sentences of class 1 is a lot lower than sentences of class 0) we balance it manually by duplicating instances from the minor class of the database. The data set is also divided in two subsets: training set and test set. The training set contains 80% of the instances of the data, while the testing set contains 20% of it.

3.4. Summary Building

After we create our models with each classification algorithm, we tested those models using the test set. For each model, we select the instances that were classified with class “1”. Before building the summary, instances of class 1 are ranked according to the probability of being actually from class 1. Sentences are then included in the summary according to the order in which they are ranked until the size of the summary reaches its maximum size considering the specified compression rate. In this work we consider a 70% compression rate in relation to the

longest text in the cluster (i.e., the summary may have up to 30% of the number of words of the longest text). We use this compression rate because is the same compression rate that was used to build the human summaries.

We also consider that sometimes there will not be enough instances classified as "1" to reach the specified summary size. In this case, we start including instances that were classified as "0" and had lower probability of being from that class (until de desired size is reached).

One important step that is taken into account is the redundancy treatment. As mentioned in Section 1, redundancy is something we have to deal with when we are working with several texts discussing the same topic. For this, we use the information provided by CST to eliminate redundancy from the final summary, by discarding sentences that have relations of the redundancy type. For example, if the relation between two sentences is Identity, one of them will be eliminated; if the relation is Equivalence, we will eliminate the longest sentence (considering the number of words of the sentence); if the relation is Subsumption, we eliminate the sentence that is subsumed.

One relation that is part of the redundancy type and is not treated here is the Overlap relation. We still do not include it in the process because eliminating redundancy in this case requires extra steps (sentence fusion, for instance) that we are not using at the moment, but certainly we will use in future works.

In the next section we will describe the evaluation methodology we use in this work.

4. Evaluation

Two types of evaluation were carried out in this work. The first evaluation corresponds to the traditional machine learning evaluation of accuracy: Precision, Recall, F-measure and ROC area. We used 10-fold cross-validation for this aim.

In the second type of evaluation, we wanted to measure how informative were the automatic summaries built from our models. For this, we used ROUGE (Lin and Hovy, 2003) which is an automatic measure that compares automatic and human summaries by considering the n-grams they have in common. There are various measures of ROUGE. In this work we will use two of them: ROUGE-1 and ROUGE-L. ROUGE-1 compares unigrams in both summaries; ROUGE-L compares the longest common subsequences between the summaries. Results for these measures are given in terms of Precision, Recall and F-measure. Note that these measures are different from the traditional machine learning ones. ROUGE considers Precision (P) and Recall (R) as shown in the formulas below:

$$P = \frac{\text{Number of common n-grams among human and automatic summaries}}{\text{Number of n-grams of the automatic summary}}$$

$$R = \frac{\text{Number of common n-grams among human and automatic summaries}}{\text{Number of n-grams of the human summary}}$$

In the next section the results will be discussed.

5. Experiments and Results

As mentioned before, we run our experiments considering 4 well known classification methods: Decision Trees (J48), Naïve Bayes (NB), SVM and One Rule (OneR). In Table 1 it is shown the results in terms of traditional Precision, Recall, F-Measure and ROC area for each class. In Table 2 it is shown more information from the confusion matrix for each class, such as True Positive rate (TP), False Positive Rate (FP), Correctly Classified Instances Rate (CCIR).

Table 1. Results in terms traditional Precision, Recall, F-measure and ROC area

	Class 1 (sentence should be in the summary)				**Class 0 (sentence ignored)**			
	Precision	**Recall**	**F-measure**	**ROC**	**Precision**	**Recall**	**F-measure**	**ROC**
J48	0.801	0.910	0.852	0.891	0.896	0.774	0.831	0.891
NB	0.712	0.306	0.428	0.687	0.558	0.877	0.682	0.687
SVM	0.694	0.538	0.606	0.651	0.623	0.764	0.686	0.651
OneR	0.690	0.749	0.718	0.706	0.726	0.664	0.693	0.706

Table 2. Confusion matrix information

	Class 1		**Class 0**		**CCIR**
	TP	**FP**	**TP**	**FP**	
J48	0.910	0.226	0.774	0.090	0.842
NB	0.306	0.123	0.877	0.694	0.591
SVM	0.538	0.236	0.764	0.462	0.650
OneR	0.749	0.336	0.664	0.251	0.706

It can be observed from the results of Table 1 and Table 2 that the best performance was achieved by algorithm J48, in terms of Precision, Recall, F-measure and ROC area. J48 has also the highest rate of correctly classified instances. We observed that, for this algorithm, deep features are considered good attributes for the tree division. Particularly, features of the type redundancy and complement are the ones that obtain more information gain values. To evaluate this, we applied an attribute evaluator from Weka, and the results showed that the three best attributes are: redundancy type, sentence position and complement type. This consists in important evidence that deep features such as relation type provide important information for the classification of sentences. This also corresponds to the intuition behind MDS that redundant information (that is repeated across texts) tends to be important. OneR also showed a good performance in terms of F-measure, ROC area and mean absolute error. This algorithm used the sentence size feature to build the rules.

We also wanted to see how a rule-based algorithm selects more than one feature to construct the rules. For this, we built a model using Prism algorithm (also part of WEKA), though we did not evaluate it. In this model, we observed that rules involving higher values for deep features require less feature combinations to determine the sentence class. On the other hand, rules involving lower values or superficial features require more feature combinations to complete the rule. For example, in Figure 6, the first learned rule expresses that, if a sentence has a complement feature value between 0.090 and 1, and a clustering coefficient value between 0.049 and 0.069, and a redundancy feature between 0.107 and 1, then the class for that sentence is 1. In other words, if the sentence has a considerable number of complement relations, and it is part of a well connected cluster in the graph, and the information of the sentence is repeated across various texts, then the sentence should be included in the summary. In the case of the second learned rule, the value of the complement feature is evaluated in the interval between 0 and 0.007, which is lower than 0.090 and 1. In other words, these rules express that if a sentence has higher values of deep features (like complement or redundancy), then the chance to be in class 1 is high, but if they have lower values for these features, the rule will have to include other conditions in order to determine the correct class.

SVM, which also has a good performance, gives higher weight to complement type and redundancy type features. For Naïve Bayes classifier, superficial features are more predictive, but this classifier has the lowest rate for correctly classified instances.

Learned rule 1:
If Complement feature = (0.090-1]
and Clustering coefficient = (0.049-0.069]
and Redundancy feature = (0.107-1] **then class 1**

Learned rule 2:
and Sentence size = (0-0.176]
and Similarity to centroid = (0-0.009]
and Redundancy feature = (0-0.007]
and Sentence position = MEDIO
and Centroid = NO
and Complement type = (0-0.007]
and Authorship type = (0-0.007] **then class 1**

Figure 6. Example of rules in Prism

Considering that J48 achieved the best performance, we used that model to build automatic summaries for 10 clusters of the corpus, which correspond to the 20% of the data that we selected as test set. These summaries have been evaluated using ROUGE, as described in Section 4. These results are compared with the ones obtained by Jorge and Pardo (2010b) and Radev et al. (2001) methods, with their generic CST operator and MEAD system, respectively. All these results are shown in Table 3. We show only ROUGE-1 and ROUGE-L results, which are among the most used measures.

Table 3. ROUGE results

	ROUGE-1			**ROUGE-L**		
	Precision	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
J48	0.537	0.550	0.542	0.509	0.521	0.514
Generic operator	0.572	0.523	0.543	0.538	0.492	0.512
MEAD	0.554	0.239	0.369	0.540	0.230	0.354

We can observe that, in terms of F-measure, model J48 and the generic CST operator proposed by Jorge and Pardo have a good performance when compared to MEAD. This confirms previous results (Jorge and Pardo, 2009, 2010b) that show that CST information helps to produce better summaries in terms of information quality. It is also interesting to see that some superficial features, like sentence size, also show to be important and to complement CST information.

To measure the confidence degree of these results, we used ANOVA test. The statistical test showed that the results for Precision, Recall and F-measure had a 95% confidence degree.

In future works we intend to investigate more deep features that may help to improve even more the performance of our model. These features may include syntactic and/or ontological information or even rhetorical information such as the proposed by Mann and Thompson (1987).

6. Final Remarks

This work presented a machine learning approach for multi-document summarization using deep features for the classification task. Results showed that these features are good predictors for sentence classes. At the moment the only deep features we are using are the ones that come from the information provided by CST model. For future work we intend to explore more linguistic features, such as syntactic, ontological or rhetorical information.

Acknowledgements

The authors are grateful to FAPESP, CAPES and CNPq for supporting this work.

References

Afantenos, S.D.; Doura, I.; Kapellou, E.; Karkaletsis, V. (2004). Exploiting Cross-Document Relations for Multi-document Evolving Summarization. In the *Proceedings of SETN*, pp. 410-419.

Aleixo, P. and Pardo, T.A.S. (2008). *CSTNews: Um Córpus de Textos Journalísticos Anotados segundo a Teoria Discursiva CST (Cross-Document Structure Theory).* Série de Relatórios Técnicos do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, no. 326. São Carlos-SP.

Antiqueira, L. and Nunes, M.G.V. (2010). Complex Networks and Extractive Summarization. In the *Extended Activities Proceedings of the 9th International Conference on Computational Processing of the Portuguese Language* – PROPOR. Porto Alegre/RS, Brazil.

Chuang W.T. and Yang J. (2000). Extracting sentence segments for text summarization: a machine learning approach. In the *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 152-159. Athens, Greece.

Costa, L.F.; Rodrigues, F.A.; Travieso, G.; Boas, P.R.V. (2007). Characterization of complex networks: A survey of measurements. *Advances in Physics*, Vol. 56, pp. 167-242.

Edmundson, H. P. (1969). New Methods in automatic extracting. *Journal of the ACM*, Vol. 16, pp. 264-285.

Jorge, M.L.C. and Pardo, T.A.S. (2009). Content Selection Operators for Multidocument Summarization based on Cross-document Structure Theory. In the *Proceedings of the 7th Brazilian Symposium in Information and Human Language Technology* – STIL, pp. 1-8. São Carlos/SP, Brazil.

Jorge, M.L.C. and Pardo, T.A.S. (2010a). Formalizing CST-based Content Selection Operations. In the *Proceedings of the International Conference on Computational Processing of Portuguese Language*- PROPOR. Porto Alegre/RS, Brazil.

Jorge, M.L.C. and Pardo, T.A.S. (2010b). Experiments with CST-based Multidocument Summarization. In the *Proceedings of the ACL Workshop TextGraphs-5: Graph-based Methods for Natural Language Processing,* pp. 74-82. Uppsala/Sweden.

Kupiec, J.; Pedersen, J.; Chen, F. (1995). A trainable document summarizer. In the *Proceedings of the 18th ACMSIGIR Conference on Research & Development in Information Retrieval*, pp. 68-73.

Lin, C.Y. and Hovy, E. (2003). Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In the *Proceedings of Language Technology Conference* – HLT-NAACL. Edmonton/Canada.

Luhn, H.P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development,* Vol. 2, pp. 159-165.

Mani, I. and Bloedorn, E. (1997). Multi-document summarization by graph search and matching. In the *Proceedings of the 14th National Conference on Artificial Intelligence* – AAAI, pp. 622-628.

Mani, I. and Bloedorn, E. (1998). Machine Learning of Generic and User-Focused Summarization. In the *Proceedings of the Fifteenth National Conference on Artificial Intelligence* – AAAI, pp. 821-826.

Mani, I. and Maybury, M.T. (1999). *Advances in automatic text summarization*. MIT Press, Cambridge, MA.

Mann, W.C. and Thompson, S.A. (1987). *Rhetorical Structure Theory: A Theory of Text Organization*. Technical Report ISI/RS-87-190.

Maziero, E.G.; Jorge, M.L.C.; Pardo, T.A.S. (2010). Identifying Multidocument Relations. In the *Proceedings of the 7th International Workshop on Natural Language Processing and Cognitive Science*- NLPCS, pp.60-69. Funchal/Madeira, Portugal.

McKeown, K. and Radev, D.R. (1995). Generating summaries of multiple news articles. In the *Proceedings of the 18th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 74-82. Seattle, WA.

Otterbacher, J.C.; Radev, D.R.; Luo, A. (2002). Revisions that improve cohesion in multi-document summaries: a preliminary study. In the *Proceedings of the Workshop on Automatic Summarization,* pp 27-36. Philadelphia.

Quinlan, J.R. (1992). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco/CA, USA.

Radev, D.R. (2000). A common theory of information fusion from multiple text sources, step one: Cross-document structure. In the *Proceedings of the 1st ACL SIGDIAL Workshop on Discourse and Dialogue*. Hong Kong.

Radev, D.R. and McKeown, K. (1998). Generating natural language summaries from multiple on-line sources. *Computational Linguistics*, Vol. 24, N. 3, pp. 469-500.

Radev, D.R.; Jing, H.; Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation and user studies. In the *Proceedings of the ANLP/NAACL Workshop*, pp. 21-29.

Radev, D.R.; Blair-Goldensohn, S.; Zhang, Z. (2001). Experiments in single and multi-document summarization using MEAD. In the *Proceedings of the First Document Understanding Conference*. New Orleans, LA.

Witten, I.H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wnek, J. (1995). *DIAV 2.0. User Manual: Specification and Guide through the Diagrammatic Visualization System*. Reports of the Machine Learning and Inference Laboratory, George Mason University.

Yang, J.; Parekh, R.; Honavar V. (1999). DistAl: An inter-pattern distance-based constructive learning algorithm. *Intelligent Data Analysis*, Vol. 3, pp. 55-73.

Zhang, Z.; Goldenshon, S.B.; Radev, D.R. 2002. Towards CST-Enhanced Sumarization. In the *Proceedings of the 18th National Conference on Artificial Intelligence*. Edmonton/Canada.

Generating Features from Textual Documents Through Association Rules

Rafael Geraldeli Rossi, Solange Oliveira Rezende

[1]Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo
São Carlos – SP – Brazil
{ragero,solange}@icmc.usp.br

Abstract. *The Text Mining techniques are used to organize, manage and extract knowledge from the huge amount of textual data available in digital format. In order to use these techniques, the textual documents need to be represented in an appropriate format. The common way to represent text collections is by using the bag-of-words approach, in which each document is represented by a vector. Each word in the document collection represents a dimension of the vector. This approach has well known problems as the high dimensionality, and sparsity of data. Besides, most of the concepts are described by a set of words, such as "text mining", "association rules", and "machine learning". The approaches, which generate features compounded by a set of words to solve this problem, suffer from other problems, such as the generation of features without meaning, and the need to analyze the high dimensionality of the bag-of-words in order to generate the features. An approach named bag-of-related-words is proposed to generate features compounded by a set of related words that avoids the problems as mentioned above. The features are generated from each textual document of a collection through association rules. Experiments were carried out using classification algorithms with different paradigms in order to evaluate the generated features. The obtained results demonstrated that the proposed approach is similar to the bag-of-words with much lower dimensionality and features which are easy to understand.*

1. Introduction

The information available in digital format at the world wide web has been increasing incessantly. Part of the data in the digital universe is in text format, such as e-mails, reports, papers, and web-pages contents. Due to the huge amount of information in textual format, Text Mining techniques are fundamental to manage and extract knowledge. The textual documents need to be represented in an appropriate way in order to use these techniques.

The Vector Space Model (VSM) [Salton 1989] is generally used to represent textual documents. In this model, each document is represented by a vector, and each position of this vector corresponds to features (dimensions) of a textual document collection. The common approach based on the VSM is the bag-of-words representation. In this representation, each word in a collection of documents becomes a dimension (feature) in the vector space. However, this approach presents some problems as the high dimensionality, a high sparsity data, and the relative positions of the words is lost.

One of the major problem presented by the bag-of-words is that concepts are usually represented by a set of words, as "data mining", "neural networks", "association

rules", and so on. These concepts can not be found in the bag-of-words, and they can help in the learning from textual document collection, as presented in [Liu and Hu 2007]. For example, single words as "learning", or "mining", can represent documents about machine learning or represent documents about teaching, or mineral extraction respectively.

To obtain set of words as features, representations were developed based on phrases [Carvalho et al. 2010, Carvalho and Cohen 2006, Fürnkranz 1998, Mladenic and Grobelnik 1998, Fagan 1989], and based on set of n words as features [Zhang and Zhu 2007, Tesar et al. 2006, Bekkerman and Allan 2004, Yang et al. 2003, Ahonen-Myka et al. 1999]. These types of representations usually add the features to the bag-of-words approach, which increases the already high dimensionality. Most of the researches analyze the entire collection to generate the features, which can have a high computational cost, due the dimensionality and features without meaning. Normally it is necessary labeled textual document collections to perform a process of supervised feature selection.

This paper proposes an approach that generates related words and uses them as features. The proposed approach, named *bag-of-related-words*, generates features from each document of a collection through the association rules to discover relations among items in a dataset. Another intention of using association rules is to reduce the dimensionality of the bag-of-words. This is possible because not all the words of a document are used as features, but only the words that occur with a frequency above a certain threshold and the words that cooccur in the document.

A mapping of the document into transactions is necessary to enable the generation of association rules from each document. This can be done by considering the sentences, paragraphs, or a sliding window as transactions. This way of mapping the document into transactions can produce more understandable features for the user, since the words must be related in specific contexts of the documents. The related words generated by our approach enable the pattern extraction with better results, and can be useful in situations as text categorization, text clustering, cluster labeling, and document identification.

After the feature generation, a representation is created based on the VSM. In this paper we analyze if the features generated by the proposed approach are feasible to be used as document descriptors or cluster labels according to their predictive power. The representations will be submitted to text categorization algorithms for evaluation and comparison with the bag-of-words.

This paper is organized as follows. Section 2 presents the related works about representation using features compounded by more than one word. Section 3 presents the bag-of-related-words. Section 4 shows the results obtained with the proposed approach and the bag-of-words for the categorization of textual documents. Finally, Section 5 presents the conclusions and future works.

2. Related Works

There are several works that extract set of words to use them as features. Usually this can be done using *n*-grams (or statistical phrases) or using set of words. The *n*-grams are sequences of words in the text collection.

In [Mladenic and Grobelnik 1998] the bag-of-words is increased with *n*-gram fea-

tures. The *n*-grams are obtained by *n* passes over the textual document collection, where the *i*-grams are generated based on the (*i* - 1)-grams from the previous pass. After this step, a process of feature selection to detect the best features is performed. The results demonstrated that features with n close to 3 are better than only single words.

In [Tan et al. 2002a], first the words with higher frequency are sorted in each document and only the most frequent words (U) are considered. Then, the words that appear in sequence with the words in U are used to generate the bigrams. After this a process of feature selection is carried out using the measures *tf-idf* and *Mutual Information*. The results were improved with the addition of the generated bigrams.

In [Carvalho et al. 2010], the automatic detection of *n*-grams is done and the impact of the addition of the n-grams in a system of information retrieval is investigated. The n in this work was set to 2. First, all the bigrams of the textual document collection are generated. Then a classification model using the Support Vector Machine method is built using a labeled collection of valid and invalid bigrams. For some domains, a domain specialist is necessary to create valid and invalid bigrams. The results using the *n*-grams improve the precision up to 50%.

Another way to use more than one word as features is by using a set of n words. The n words that compound the features can not appear together or close to each other in a textual document.

In [Yang et al. 2003] the "association features" is defined. This approach does not consider where the words occur in the documents. In order to generate the features, first the documents of a collection are represented by a bag-of-words, then a process of frequent itemset extraction is used to obtain the words that cooccur in the text document collection. The association features are added to the bag-of-words. The results demonstrated improvements using the Naïve Bayes classification algorithm.

In [Tesar et al. 2006], the use of bigrams and 2-itemsets is analyzed. 2-itemsets are sets compounded by two words. For the extraction of the 2-itemsets, a list of unique words of each textual document is created. Then, all the unique words are combined to form the 2-itemsets. As the number of 2-itemsets is huge, only the words with a frequency higher than a threshold are considered. Both the bigrams and the 2-itemsets are added to the bag-of-words representation. The results demonstrated better results with statistical significance using the Multinomial Naïve Bayes classification algorithm with the use of bigrams and 2-itemsets.

In the work of [Zhang and Zhu 2007], the "loose n-grams" were proposed. The loose n-grams are set of words that cooccur in a limited space, as sentences, or a sequence of words (window). The relative position of the words are not considered. Only the words that appear in a minimum number of documents are considered. The words that have a good discriminative power are separated. Then, sets of words compounded by words that cooccur in the collection are obtained, and these sets of words are filtered using the χ^2 method. The results demonstrated a little improvement in comparison to the bag-of-words.

Most of the related researches about the generation of features compounded by more than one word generated a number of features that is much higher than the bag-of-words approach and in most of the times a process of feature selection is necessary. How-

ever, due the high dimensionality of the collection, this process can be computationally extremely expensive, and sometimes impossible to be carried out. Besides, in unlabeled collections, which is very common, this process can be more difficult and might produce worse results.

Moreover, some approaches that analyze the entire collection may obtain features without meaning, as "*introduction_conclusion*" if we consider a document collection of scientific articles. Sometimes collections with few documents of some kind of vocabulary may not have their features compounded by more than one word. For example, suppose 5000 documents about chemistry and 20 about computer science. Suppose that some words of the documents about computer science cooccur only in 5 documents, the minimum cooccurrence must be 0.001. On the other hand a huge number of features about chemistry could be generated. Besides the cooccurrence analysis is executed in a bag-of-words representation, which can be computationally expensive.

The proposed approach treats all the problems as mentioned before i) it avoids the generation of a huge number of features, ii) it does not need labeled collections or collections with balanced topics, and iii) the features are more understandable.

3. Bag-of-related-words

The proposed approach was named *bag-of-related-words*. This approach generates features compounded by a set of related words from textual documents using association rules. An association rule is a rule of the type $A \Rightarrow B$, in which A and B are group of items, called itemsets, and $A \cap B = \emptyset$ [Agrawal and Srikant 1994]. The association rules extract the relations of the items in a dataset, in which $A \Rightarrow B$ means that when A (body) occurs, B (head) also tends to occur. Two classical measures to generate association rules are *support* and *confidence*. Support measures the joint probability of an itemset in a database, that is $sup(A \Rightarrow B) = n(A \cup B)/N$, in which $n(A \cup B)$ is the number of transactions that A and B occur together, and N is the total number of transactions. Confidence indicates the probability of A and B occur together given the occurrence of A, that is, $conf(A \Rightarrow B) = n(A \cup B)/n(A)$. Minimum values of support and confidence are defined to generate the rules.

The goal of our approach is to use related words that are repeated over the document in limited spaces as features. The four main steps to generate the features of the documents are:

1. Mapping the textual document into transactions;
2. Extracting association rules from the transactions;
3. Using the itemsets of the rules to compound the features;
4. Using the features to construct the *document-term* matrix;

The step of mapping of the textual document into transactions allows the extraction of association rules. In this step, Text Mining preprocessing techniques can be applied, as stopwords removal and stemming. The transactions can be obtained considering the sentences, paragraphs, or a sliding window as transactions. To illustrate this step, the text in Table 1 about Data Mining taken from Wikipedia[1] is considered.

[1] http://en.wikipedia.org/wiki/Data_mining (May 13, 2010).

Table 1. Example text.

Data mining is the process of extracting patterns from data. Data mining is becoming an increasingly important tool to transform this data into information. It is commonly used in a wide range of profiling practices, such as marketing, surveillance, fraud detection and scientific discovery. Data mining can be used to uncover patterns in data but is often carried out only on samples of data. The mining process will be ineffective if the samples are not a good representation of the larger body of data. Data mining cannot discover patterns that may be present in the larger body of data if those patterns are not present in the sample being "mined". Inability to find patterns may become a cause for some disputes between customers and service providers. Therefore data mining is not foolproof but may be useful if sufficiently representative data samples are collected. The discovery of a particular pattern in a particular set of data does not necessarily mean that a pattern is found elsewhere in the larger data from which that sample was drawn. An important part of the process is the verification and validation of patterns on other samples of data.

After the preprocessing steps, the text from Table 1 is mapped into transactions. Table 2 presents the transactions considering the sentences.

Table 2. Transactions considering the sentences of the preprocessed text from Table 1.

1. data mine process extract pattern data
2. data mine increasingli import tool transform data inform
3. commonli wide rang profil practic market surveil fraud detect scientif discoveri
4. data mine uncov pattern data carri sampl data
5. mine process ineffect sampl good represent larger bodi data
6. data mine discov pattern present larger bodi data pattern present sampl mine
7. inabl find pattern disput custom servic provid
8. data mine foolproof suffici repres data sampl collect
9. discoveri pattern set data necessarili pattern found larger data sampl drawn
10. import part process verif valid pattern sampl data

The next step consists of extracting association rules from transactions. The rules extracted from the text of Table 2 considering a minimum support of 30% and a minimum confidence of 75% are presented on the Table 3. The values on the right of the rules are the support and the confidence respectively.

Table 3. Rules obtained considering the transactions from Table 2.

$\emptyset \Rightarrow data$ (80.0, 80.0)	data $\Rightarrow$ mine (60.0, 75.0)	mine pattern $\Rightarrow$ data (30.0, 100.0)
process $\Rightarrow$ data (30.0, 100.0)	data $\Rightarrow$ sampl (60.0, 75.0)	larger data $\Rightarrow$ sampl (30.0, 100.0)
larger $\Rightarrow$ sampl (30.0, 100.0)	sampl $\Rightarrow$ data (60.0, 100.0)	pattern data $\Rightarrow$ sampl (40.0, 80.0)
larger $\Rightarrow$ data (30.0, 100.0)	mine $\Rightarrow$ data (60.0, 100.0)	mine sampl $\Rightarrow$ data (40.0, 100.0)
pattern $\Rightarrow$ data (50.0, 83.3)	larger sampl $\Rightarrow$ data (30.0, 100.0)	pattern sampl $\Rightarrow$ data (40.0, 100.0)

The proposed approach also considers features compounded by single words. In this case, the items of the rules with empty set[2] like $data \Rightarrow \emptyset$ will be used as features compounded by single words. In this case the feature "$data$" would be generated.

The third step consists of using the items of the obtained association rules to compound the features of a document. For example, the rule "$data \Rightarrow mine$" will generate the feature "*data_mine*". In order to avoid that rules with the same items generate different features, the items of the rule are sorted lexicographically or according to the order that they occur on the document. For example, "$association \Rightarrow rule$" and "$rule \Rightarrow association$" generate the feature "*association_rule*".

In this step, interest measures besides support and confidence can be used to generate different relations among the items, to rank and prune the association rules. The intention to use interest measures is to obtain more understandable features

[2]The empty set are only present on the body of the rule

and reduce even more the dimensionality of the obtained representation. In this approach, the used measures were *Confidence*, *Lift*, *Yules'Q*, *Linear Correlation Coefficient*, *Mutual Information*, *Gini Index*, *Kappa*, and *J-Measure* [Guillet and Hamilton 2007, Geng and Hamilton 2006, Blanchard et al. 2005, Tan et al. 2002b]. The interest measures are compared to verify which measure produces better results for the textual document classification task.

The fourth step consists of using the generated features of each document to build a representation in the vector space model.

The bag-of-related-words avoids several problems as discussed in Section 2. For instance, the approach does not analyze the whole dimensionality of the textual document collection, which can be extremely expensive. The proposed approach analyzes each document individually, in which the number of words can be orders of magnitude smaller than all the words in a text document collection.

Another characteristic of the bag-of-related-words is that it does not need to set the value of n, since the rules will be generated until the items have values of support and some interest measure higher than the thresholds informed by the user. To aid the user with the minimum support setting, we proposed the following equation.

$$AutSup = \frac{GeneralMeanFreq}{NumberTrans} \tag{1}$$

where $GeneralMeanFreq$ means the sum of the frequency of all the words, and $NumberTrans$ means the number of transactions. This formula is used for each document of the collection to the minimum support setting.

The definition of the thresholds of other interest measures can be also set automatically. In this paper we used the mean ($\overline{x}$) of all the rules extracted of a document as a threshold. According to the thresholds chosen by the user during the process, the dimensionality can be much lower than the bag-of-words, and the application of feature selection methods is not necessary. However, as the features are represented in the vector space model, these methods can be applied as an additional task.

Therefore, the proposed approach is independent on the i) natural language processing, which is normally computationally expensive ii) the interference of domain specialists, and iii) knowledge base.

4. Experiments and Results

To evaluate the predictive power of the generated features by the proposed approach with the different interest measures and compare them with the bag-of-words, 4 classification algorithms with different paradigms were used: (i) *Naïve Bayes* (probabilistic learning); (ii) J48[3] (symbolic learning); (iii) *SMO* (statistical learning); and (iv) *KNN* (instance based learning). The Weka tool [Witten and Frank 2005] with standard options was used to execute the classification algorithms.

Experiments were carried out using 8 textual document collections of proceedings

[3]J48 is an open source Java implementation of the C4.5

from the ACM digital library[4] and the Reuters-21578[5] collection. Each ACM collection has 5 classes and approximately 90 documents per class. For this collection, the 10-fold-cross validation was used to evaluate the error classification rate.

For the Reuters-21578 collection, only the content of the tags *TITLE* e *BODY* was used. Moreover, the documents with single topic of the 15 more frequent classes were considered. The *LEWISSPLIT* division (*"TRAIN"* or *"TEST"*) was considered to build and to evaluate the classification model respectively. 2537 documents for training and 952 documents for test were used.

The bag-of-related-words was obtained following the steps presented in section 2. In the step referring to the conversion of a document into transactions, the stopwords were removed and the words were stemmed using the Porter's algorithm [Porter 1980]. A sliding window of size 5 to map the document into transactions was considered, since it obtained the best results [Rossi and Rezende 2010]. The value of minimum support was calculated from each document of the collection based on the proposed Equation 1.

The measures mentioned in Section 3 were used to prune the association rules. For each measure, a set of thresholds was used. The values of the thresholds were obtained empirically based on the number of features. Also the threshold based on mean ($\overline{x}$) was used. Table 4 presents the used thresholds. As the Lift ranges from 0 to ∞, the standardization proposed in [McNicholas et al. 2008] was used.

Table 4. Thresholds of the interest measures used in the experiments.

Objective Measure	ACM	Reuters
Confidence	0.25; 0.50; $\overline{x}$	0.25; 0.50; $\overline{x}$
Lift	0.10; 0.20; $\overline{x}$	0.10; 0.20; $\overline{x}$
Yule's Q	0.50; 0.75; $\overline{x}$	0.50; 0.75; $\overline{x}$
Correlation	0.25; 0.50; $\overline{x}$	0.25; 0.50; $\overline{x}$
Mutual Information	0.005; 0.01; $\overline{x}$	0.01; 0.05; $\overline{x}$
Gini Index	0.005; 0.01; $\overline{x}$	0.01; 0.04; $\overline{x}$
Kappa	0.15; 0.20; $\overline{x}$	0.30; 0.50; $\overline{x}$
J-Measure	0.01; 0.02; $\overline{x}$	0.02; 0.05; $\overline{x}$

In order to compare the results, the Pretext tool [Soares et al. 2008] was used to generate the bag-of-words representation. The metric used to compare the results obtained with the representations was the error classification rate.

We verified if there were significant statistically difference among the obtained results with the thresholds used for each interest measure. The Friedman test was applied considering the results of the 8 ACM collections for each interest measure. The results obtained by this test demonstrated that although the number of features vary significantly for each threshold, the differences in general were not statistically significant. For Naïve Bayes, there were differences with the thresholds for the Mutual Information and Gini Index measures, in which the threshold 0.01 was better than 0.005; and Correlation, in which the threshold based on mean was statistically better than the other thresholds. For the KNN, there were statistical difference among the thresholds of the Kappa measure, in which the threshold based on mean was better than the other thresholds.

[4] `http://sites.labic.icmc.usp.br:8088/ragero/Acm-Collection`

[5] `http://www.daviddlewis.com/resources/testcollections/reuters21578`

To compare the representations obtained by different measures and the bag-of-words, the representations with threshold based on mean were chosen, since in most of the situations, they did not present statistical differences in relation to the other thresholds. Again, the Friedman test was applied with the results of the 8 ACM collections. For the Naïve Bayes, Correlation measure presented better results than the measures Confidence and Lift with significant statistical difference. For SMO, only the representation using the Gini Index measure presented worse results than the bag-of-words with significant statistical difference.

Tables 5, 6, 7, and 8 present the results of the representations that obtained the smallest error classification rates of the ACM collection using the Naïve Bayes, J48, SMO, and KNN classification algorithm respectively[6]. The lowest classification errors for each collection are highlighted in these tables. It can be noticed that our approach obtained the smallest error rate in 5 of the 8 collection using the Naïve Bayes, 2 of the 8 using the J48, 1 of the 8 using the SMO, and 5 of the 8 using the KNN. It must be highlighted that some representations, as [J-Measure 0.02], [Mutual Information 0.2], [Confidence 0.50], [Correlation 0.5], obtained the best results for some ACM collection using the classification algorithms Naïve Bayes, J48, SMO, and KNN respectively, with a reduction of the feature number in approximately 55%, 38%, 80%, and 76% respectively.

Table 5. Naïve Bayes - results of the representations with lowest error classification rates for the ACM collection.

Representation	ACM-1	ACM-2	ACM-3	ACM-4	ACM-5	ACM-6	ACM-7	ACM-8
Bag-of-words	***5.51** ± 0.14	7.80 ± 0.10	7.55 ± 0.14	11.42 ± 0.08	***2.76** ± 0.09	7.29 ± 0.10	7.89 ± 0.10	***5.86** ± 0.14
Correlation $\overline{x}$	9.27 ± 0.19	***3.66** ± 0.12	***3.13** ± 0.11	1.78 ± 0.08	3.40 ± 0.12	3.66 ± 0.12	4.48 ± 0.13	6.46 ± 0.16
Mutual Inf. 0.005	9.27 ± 0.19	3.90 ± 0.12	3.37 ± 0.12	***1.52** ± 0.08	3.61 ± 0.12	3.66 ± 0.12	5.33 ± 0.14	6.87 ± 0.17
Kappa 0.15	9.77 ± 0.20	4.39 ± 0.13	3.85 ± 0.12	***1.52** ± 0.08	3.61 ± 0.12	3.89 ± 0.12	4.69 ± 0.13	7.27 ± 0.17
Kappa $\overline{x}$	9.52 ± 0.20	4.15 ± 0.13	3.85 ± 0.12	1.78 ± 0.08	3.61 ± 0.12	3.66 ± 0.12	***4.05** ± 0.13	7.47 ± 0.17
J-Measure 0.02	9.77 ± 0.20	4.15 ± 0.12	3.85 ± 0.12	2.28 ± 0.10	3.61 ± 0.12	***3.43** ± 0.12	5.76 ± 0.15	7.47 ± 0.17

Table 6. J48 - results of the representations with lowest error classification rates for the ACM collection.

Representation	ACM-1	ACM-2	ACM-3	ACM-4	ACM-5	ACM-6	ACM-7	ACM-8
Bag-of-words	***11.53** ± 0.22	***7.07** ± 0.19	12.97 ± 0.22	***6.09** ± 0.13	***14.01** ± 0.22	11.62 ± 0.20	***11.30** ± 0.20	***10.30** ± 0.20
Lift 0.1	14.79 ± 0.24	13.66 ± 0.23	11.54 ± 0.21	9.39 ± 0.19	18.47 ± 0.26	***8.01** ± 0.18	18.55 ± 0.26	16.97 ± 0.25
Lift 0.2	15.04 ± 0.24	13.17 ± 0.22	***9.86** ± 0.19	8.12 ± 0.18	17.41 ± 0.26	10.53 ± 0.2	18.55 ± 0.26	19.39 ± 0.27
Yule's Q 0.5	14.04 ± 0.23	13.41 ± 0.23	11.06 ± 0.2	10.15 ± 0.2	17.41 ± 0.26	***8.01** ± 0.17	18.98 ± 0.26	18.59 ± 0.26
Yule's Q $\overline{x}$	14.54 ± 0.24	13.90 ± 0.23	10.10 ± 0.19	9.39 ± 0.18	17.41 ± 0.26	***8.01** ± 0.17	18.55 ± 0.26	18.38 ± 0.26

Table 9 presents the results of the representation with thresholds based on mean and the bag-of-words for the Reuters collection. For Naïve Bayes all the representations obtained an error classification rate lower than the bag-of-words. The lowest error rate was obtained by the representation [Kappa $\overline{x}$], with an error of 24.05%, which was 3.89% lower than the obtained by the bag-of-words. For the J48, again all the representations obtained an error classification rate lower than the bag-of-words. The lowest error rate

[6]The complete results are presented in http://sites.labic.icmc.usp.br:8088/ragero/ENIA2011/docint.pdf

Table 7. SMO - results of the representations with lowest error classification rates for the ACM collection.

Representation	ACM-1	ACM-2	ACM-3	ACM-4	ACM-5	ACM-6	ACM-7	ACM-8
Bag-of-words	***3.26** ± 0.32	***2.20** ± 0.32	***3.54** ± 0.32	***0.25** ± 0.32	***2.12** ± 0.32	***1.14** ± 0.32	3.84 ± 0.32	***4.04** ± 0.32
Confidence 0.25	5.26 ± 0.32	3.41 ± 0.32	4.09 ± 0.32	1.52 ± 0.32	3.40 ± 0.32	2.29 ± 0.32	***2.77** ± 0.32	8.08 ± 0.32
Confidence 0.5	5.01 ± 0.32	4.39 ± 0.32	4.33 ± 0.32	1.52 ± 0.32	3.18 ± 0.32	2.06 ± 0.32	***2.77** ± 0.32	7.68 ± 0.32
Yule's Q $\overline{x}$	5.26 ± 0.32	3.41 ± 0.32	4.57 ± 0.32	1.78 ± 0.32	2.76 ± 0.32	2.29 ± 0.32	***2.77** ± 0.32	7.07 ± 0.32
Correlation 0.50	5.26 ± 0.32	3.90 ± 0.32	4.33 ± 0.32	1.52 ± 0.32	3.18 ± 0.32	2.29 ± 0.32	***2.77** ± 0.32	7.68 ± 0.32
Mutual Inf. 0.005	5.26 ± 0.32	3.41 ± 0.32	4.33 ± 0.32	1.78 ± 0.32	2.76 ± 0.32	2.97 ± 0.32	***2.77** ± 0.32	6.67 ± 0.32

Table 8. KNN - results of the representations with lowest error classification rates for the ACM collection.

Representation	ACM-1	ACM-2	ACM-3	ACM-4	ACM-5	ACM-6	ACM-7	ACM-8
Bag-of-words	10.02 ± 0.06	7.32 ± 0.04	4.96 ± 0.03	2.04 ± 0.02	5.74 ± 0.03	***1.82** ± 0.03	***6.18** ± 0.03	***4.44** ± 0.03
Yule's Q 0.5	11.01 ± 0.06	8.78 ± 0.07	6.48 ± 0.04	0.76 ± 0.01	***3.18** ± 0.02	3.66 ± 0.03	6.61 ± 0.02	8.47 ± 0.04
Correlation 0.5	***8.78** ± 0.04	10.49 ± 0.04	5.49 ± 0.04	1.27 ± 0.01	4.65 ± 0.04	4.13 ± 0.03	7.25 ± 0.02	8.69 ± 0.04
Correlation $\overline{x}$	9.26 ± 0.05	***6.83** ± 0.04	5.55 ± 0.04	1.01 ± 0.02	4.04 ± 0.03	2.98 ± 0.02	6.40 ± 0.03	8.07 ± 0.05
J-Measure $\overline{x}$	11.52 ± 0.04	8.78 ± 0.04	***4.81** ± 0.03	***0.50** ± 0.02	4.02 ± 0.03	4.13 ± 0.03	7.46 ± 0.03	9.08 ± 0.02

was obtained by the representations [Mutual Inf. 0.05], [Gini Index 0.01], e [J-Measure 0.02], being 28.89%, which is 2.83% lower than the bag-of-words. For the SMO, 11 of the 24 representations obtained an error classification rate lower than bag-of-words. The lowest error classification rate was obtained by the representation [Kappa 0.50] with an error of 22.90%, which is 0.42% smaller than the bag-of-words. For KNN, all the error rates obtained by the proposed representation were higher than the bag-of-words. The best result obtained by our approach was obtained by the representation [Confidence 0.50%] with an error rate of 17.58%, which is 2.56% higher than the bag-of-words.

Table 9. Results for the Reuters collection using the Naïve Bayes (N.B.), J48, SMO, and KNN classification algorithms.

Representation	N.B.	J48	SMO	KNN
Bag-of-words	27.94	31.72	23.32	***15.02**
Lift $\overline{x}$	24.90	30.04	23.21	18.41
Confidence $\overline{x}$	25.00	30.36	24.05	18.41
Yule's Q $\overline{x}$	24.79	***29.52**	23.53	18.49
Correlation $\overline{x}$	24.26	29.62	23.63	18.09
Mutual Inf. $\overline{x}$	24.90	30.04	23.21	18.72
Gini Index$\overline{x}$	24.79	30.57	23.21	18.09
Kappa $\overline{x}$	***24.05**	29.73	***23.00**	18.84
J-Measure $\overline{x}$	25.00	29.83	***23.00**	18.41

For the Reuters collection, several representations based on the proposed approach for the classification algorithms Naïve Bayes, J48, SMO, obtained better results than the bag-of-words with a reduction of the feature number up to 20%.

The results using the Naïve Bayes classification algorithm demonstrated that the proposed approach is appropriate, since the best results were obtained for most of the collections, and with a feature number smaller than the bag-of-words

We observed that our approach obtained error rates higher than the bag-of-words for the most of the ACM collection using the algorithm J48. We noticed that this is related to the features of the collection as conference places, author names, and so on. For instance, some nodes of the decision tree obtained using the bag-of-words and our approach for the collection ACM-8 are presented on the Table 10.

Table 10. Example of rules generated by the J48 algorithm using the bag-of-words representation and the proposed approach.

Bag-of-words	Proposed Approach
"hawaii $>$ 0: Hypertext_Hypermedia"	"number_vertice $>$ 0: Hypertext_Hypermedia"
"jaschk $>$ 0: Hypertext_Hypermedia"	"system_time $>$ 0: Embedded_Systems"
"piezzo $>$ 0: Embedded_Systems"	"packet_process $>$ 0: Embedded_Systems"
"rajwar $>$ 0: Microarchtecture"	"architecture_ comput $>$ 0: Microarchitecture"

It can be noticed that the rules generated by the proposed approach are more understandable and bring more knowledge than the obtained rules using the bag-of-words representation. In situations in which there are no authors cited in several documents of a class, or there are no places in some documents of a class, our approach can obtain better results than the bag-of-words. Besides the fact that the bag-of-words obtained the lowest error classification rates for most of ACM collection, there were not significant statistical differences between the bag-of-words and the bag-of-related-words.

The results using the classification method SMO demonstrated that the proposed representation is appropriate, since there were no significant statistical differences between the bag-of-related-words and the bag-of-words. Again, features as author's names, places and conferences names might have influenced the results obtained by the bag-of-words, since the support vector might use vectors with these features as support vectors. For the Reuters-21578 collection, most of the representations of the proposed approach obtained better results with a number of features smaller than the bag-of-words.

For KNN, the smallest error classification rates were obtained for most of the ACM collections. However unlike the other classification algorithms, the proposed approach did not obtain the smallest error classification rate for the Reuters-21578 collection. Other experiments involving similarity measures demonstrated that the proposed approach led to good results when used in collections with long texts, and the results with short texts are not as good as the obtained results by the bag-of-words.

5. Conclusions and Future Work

The proposed approach demonstrated that it is capable of generating features that can predict the classes as good as the features of the bag-of-words for different classification algorithms. The initial dimensionality of our approach, even containing features composed by n words, in most cases was much smaller than the bag-of-words with better results. Analyzing the rules obtained by a symbolic algorithm, we can notice that the features are really easy to understand.

Compared to the previous paper that used only frequent itemsets as features of the textual documents [Rossi and Rezende 2010], the use of interest measures reduced the number of features and in most cases led to better results. The different interest measures used to generate the features demonstrated to be equivalent in most cases. Only for the

Naïve Bayes classification algorithm the Correlation measure presented better results than the Confidence and Lift measures, and for the classification algorithm SMO, the Lift measure presented worse results than the bag-of-words.

Another aspect that we are currently investigating is related to the determination of the interest measure threshold. This setting is not trivial and can be time consuming for the user. An automatically threshold setting was proposed in this paper. The results obtained with the automatically set thresholds were equivalent or better than the manually threshold setting.

The analysis of each document individually, besides avoids the entire dimensionality of the collection, allow the proposed approach to be used in dynamic contexts, in which the analysis of the entire document collection to generate the features is unfeasible for each new document.

For future research, we are going to evaluate the quality of the clustering used to build a topic taxonomy and a subjective evaluation with users browsing in big text document collections represented by the bag-of-related-words. Also we are going to evaluate the ensemble of the results of different interestingness measures, and the clustering of the rules to reduce the number of features.

Acknowledgments

The authors are grateful to the financial support of CNPq.

References

Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In *VLDB'94: International Conference on Very Large Data Bases*, pages 487–499. Morgan Kaufmann Publishers Inc.

Ahonen-Myka, H., Heinonen, O., Klemettinen, M., and Verkamo, A. I. (1999). Finding co-occurring text phrases by combining sequence and frequent set discovery. In *IJCAI-99: Workshop on Text Mining: Foundations, Techniques and Applications*, pages 1–9.

Bekkerman, R. and Allan, J. (2004). Using bigrams in text categorization. Technical Report IR-408, Center of Intelligent Information Retrieval, UMass Amherst.

Blanchard, J., Guillet, F., Gras, R., and Briand, H. (2005). Using information-theoretic measures to assess association rule interestingness. In *ICDM'05: Internation Conference on Data Mining*, pages 66–73.

Carvalho, A. L. C., Moura, E. S., and Calado, P. (2010). Using statistical features to find phrasal terms in text collections. *Journal of Information and Data Management*, 1(3):583–597.

Carvalho, V. R. and Cohen, W. W. (2006). Improving "email speech acts" analysis via n-gram selection. In *ACTS '09: Workshop on Analyzing Conversations in Text and Speech*, pages 35–41. Association for Computational Linguistics.

Fagan, J. (1989). The effectiveness of a nonsyntactic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science*, 40(2):115–132.

Fürnkranz, J. (1998). A study using n-gram features for text categorization. Technical Report OEFAI-TR-98-30, Austrian Research Institute for Artificial Intelligence.

Geng, L. and Hamilton, H. J. (2006). Interestingness measures for data mining: A survey. *ACM Computing Surveys*, 38(3):9.

Guillet, F. and Hamilton, H. J., editors (2007). *Quality Measures in Data Mining*, volume 43 of *Studies in Computational Intelligence*. Springer.

Liu, S. and Hu, H. (2007). Text classification using sentential frequent itemsets. *Journal of Computer Science and Technology*, 22(2):334–337.

McNicholas, P. D., Murphy, T. B., and O'Regan, M. (2008). Standardising the lift of an association rule. *Computational Statistics & Data Analysis*, 52(10):4712–4721.

Mladenic, D. and Grobelnik, M. (1998). Word sequences as features in text-learning. In *ERK'98: Electrotechnical and Computer Science Conference*, pages 145–148.

Porter, M. F. (1980). An algorithm for suffix stripping. *Readings in Information Retrieval*, 14(3):130–137.

Rossi, R. G. and Rezende, S. O. (2010). The use of frequent itemsets extracted from textual documents for the classification task. In *WTI 2010: International workshop on Web and Text Intelligence located on International Joint Conference (SBIA, SBRN, JRI)*, pages 1–10.

Salton, G. (1989). *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley Longman Publishing Co., Inc.

Soares, M. V. B., Prati, R. C., and Monard, M. C. (2008). PRETEXT II: Descrição da reestruturação da ferramenta de pré-processamento de textos. Technical Report 333, ICMC-USP.

Tan, C.-M., Wang, Y.-F., and Lee, C.-D. (2002a). The use of bigrams to enhance text categorization. *Information Processing and Management*, 38(4):529–546.

Tan, P.-N., Kumar, V., and Srivastava, J. (2002b). Selecting the right interestingness measure for association patterns. In *ACM SIGKDD'2002: International Conferenceon Knowledge Discovery and Data Mining*, pages 32–41. ACM.

Tesar, R., Strnad, V., Jezek, K., and Poesio, M. (2006). Extending the single words-based document model: a comparison of bigrams and 2-itemsets. In *DocEng'06: ACM Symposium on Document Engineering*, pages 138–146.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2 edition.

Yang, Z., Zhang, L., Yan, J., and Li, Z. (2003). Using association features to enhance the performance of naïve bayes text classifier. In *ICCIMA '03: International Conference on Computational Intelligence and Multimedia Applications*, page 336. IEEE Computer Society.

Zhang, X. and Zhu, X. (2007). A new type of feature - loose n-gram feature in text categorization. In *IbPRIA'07: Iberian Conference on Pattern Recognition and Image Analysis*, pages 378–385. Springer.

Construção Automática de Diretórios Web usando Agrupamento Incremental de Termos

Ricardo M. Marcacini[1], **Solange O. Rezende**[1]

[1]Instituto de Ciências Matemáticas e de Computação (ICMC)
Universidade de São Paulo (USP)
São Carlos - SP - Brasil

{rmm,solange}@icmc.usp.br

Abstract. *Hierarchical clustering methods are useful to support the construction of web directories in a unsupervised way. However, the traditional methods are ineffective in dynamic scenarios, with constant updating of knowledge. Moreover, these methods obtain a hierarchical structure that is difficult to be interpreted by users. In this paper, we propose an incremental term clustering approach that allows (1) the organization of document collections in dynamic scenarios and (2) obtain cluster descriptors to support the interpretation of results. An experimental evaluation was carried out through real data from a web directory, which presented good results.*

Resumo. *Métodos baseados em agrupamento hierárquico de documentos são úteis para apoiar a construção de diretórios web de maneira não supervisionada. No entanto, os métodos tradicionais são ineficientes em cenários dinâmicos, com constante atualização do conhecimento. Além disso, estes métodos obtêm uma estrutura hierárquica que é de difícil interpretação para os usuários. Neste trabalho, é proposta uma abordagem de agrupamento incremental de termos que permite (1) organizar coleções de documentos em cenários dinâmicos e (2) obter descritores ao agrupamento para apoiar a interpretação dos resultados. Uma avaliação experimental foi realizada em dados reais de um diretório web, apresentando bons resultados.*

1. Introdução

As plataformas online para publicação e armazenamento de conteúdo digital, como a Internet, têm contribuído de forma significativa para o surgimento de diversos repositórios textuais. Devido à necessidade de extrair conhecimento útil desses repositórios, abordagens para organização automática de documentos têm recebido grande atenção na literatura [Feldman and Sanger 2006, Premalatha and Natarajan 2010]. A organização do conteúdo da Internet nos chamados "diretórios web" é uma das abordagens mais populares, em que os documentos da web são organizados em diretórios e subdiretórios, e cada diretório contém documentos relacionados a um mesmo tema [Yang and Lee 2004, Kim 2006]. Desta forma, o usuário pode visualizar a informação de interesse em diversos níveis de granularidade e explorar interativamente grandes coleções de documentos.

Os diretórios web desempenham um papel importante na recuperação de informação da Internet, principalmente em tarefas de busca exploratória. Neste tipo de tarefa, o

usuário geralmente tem pouco domínio sobre o tema de interesse, dificultando expressar o objetivo diretamente por meio de palavras-chave [Marchionini 2006]. Assim, é necessário disponibilizar previamente algumas opções para guiar o processo de busca da informação. Para tal, cada diretório possui um conjunto de descritores que contextualizam e indicam o significado dos documentos ali agrupados. Esta organização está relacionada com a hipótese de que se um usuário está interessado em um documento específico pertencente a um determinado diretório deve também estar interessado em outros documentos desse diretório e de seus subdiretórios [Manning et al. 2008].

A maioria dos diretórios web é construída de maneira supervisionada, a exemplo do *Dmoz - Open Directory Project*[1] e *Yahoo! Directory*[2], editadas manualmente por especialistas humanos. Apesar da boa qualidade dos resultados obtidos, a construção manual desses diretórios é limitada pela grande quantidade de documentos disponíveis e pela alta frequência de atualização. Deste modo, métodos para automatizar a construção de diretórios web têm sido explorados, em especial com uso de algoritmos de agrupamento hierárquico [Yang and Lee 2004, Kim 2006].

Os algoritmos de agrupamento hierárquico organizam, de forma não supervisionada, um conjunto de objetos em grupos, no qual objetos de um mesmo grupo são altamente similares entre si, mas dissimilares em relação aos objetos de outros grupos [Xu and Wunsch 2008]. Ainda, a organização é disposta em grupos e subgrupos, formando-se uma estrutura hierárquica similar à dos diretórios web. Por outro lado, os algoritmos clássicos de agrupamento hierárquico possuem algumas limitações que desafiam a construção automática desses diretórios. Uma das limitações está relacionada com a interpretação dos resultados do agrupamento, pois apenas a organização em grupos e subgrupos geralmente não é suficiente para análise dos resultados [Moura and Rezende 2010]. Assim, é necessário também extrair características do agrupamento que descrevem o significado de cada grupo e apresentá-las ao usuário durante a busca exploratória. Outra limitação se deve ao ambiente dinâmico dos diretórios web, uma vez que novos documentos são publicados a todo momento na Internet. O agrupamento hierárquico geralmente obtém uma estrutura estática para organização dos objetos, o que inviabiliza incorporar diretamente novo conhecimento disponível. Repetir todo o processo de agrupamento sempre que ocorrer atualizações significativas é computacionalmente custoso em grandes bases de dados. Neste cenário, é desejável que o agrupamento seja realizado de maneira incremental [Fung et al. 2008].

Em vista desses desafios, neste trabalho é proposta e avaliada uma abordagem para construção automática de diretórios web. Para tal, foi adaptado um algoritmo de agrupamento incremental de termos que permite adicionar novo conhecimento sem reprocessamento redundante. Além disso, durante o processo de agrupamento são extraídos automaticamente descritores para cada grupo, visando auxiliar a interpretação e exploração da organização. Uma avaliação experimental realizada sobre um conjunto de dados reais, obtido do Diretório Web *Dmoz (Open Directory Project)*, indica que a abordagem proposta obtém resultados competitivos em comparação com uma abordagem similar da literatura. Uma visão geral das contribuições alcançadas durante o desenvolvido deste trabalho é descrita a seguir:

[1]Dmoz - Open Directory Project: http://www.dmoz.org/

[2]Yahoo Directory!: http://dir.yahoo.com/

- É apresentada a ideia de agrupamento de termos para a construção de diretórios web. Para isto, foi realizada uma adaptação do algoritmo IHTC (*Incremental Hierarchical Term Clustering*) [Marcacini and Rezende 2010a], que permite extrair grupos de termos (palavras e expressões) para identificação dos tópicos existentes nos textos. Os grupos de termos são associados a grupos de documentos, auxiliando a interpretação dos resultados e atividades de busca exploratória.
- Para lidar com o cenário dinâmico dos diretórios web, é explorada uma estratégia de agrupamento incremental visando obter uma representação condensada dos dados. Um algoritmo de agrupamento hierárquico é aplicado sobre esta representação condensada e, assim, é possível extrair uma organização hierárquica com custo computacional reduzido.
- Por fim, um módulo computacional para construção e análise de diretórios web a partir de textos foi desenvolvido e incorporado à ferramenta *Torch - Topic Hierarchies* [Marcacini and Rezende 2010b], que está disponível *online*[3] juntamente com o conjunto de dados utilizado na avaliação experimental.

O restante deste artigo está organizado da seguinte maneira. Na Seção 2 é apresentada uma breve revisão da literatura, com os conceitos principais sobre agrupamento de documentos e seu uso na organização não supervisionada de coleções textuais. A abordagem proposta neste trabalho é detalhada na Seção 3, em que é descrito o processo de agrupamento incremental de termos, a estratégia para obter uma representação condensada dos textos e a construção dos diretórios web. Na Seção 4 é apresentada a avaliação experimental da abordagem proposta bem como a discussão dos resultados. Por fim, as conclusões e as direções para trabalhos futuros são descritas na Seção 5.

2. Conceitos Básicos

A tarefa de construção automática de diretórios web pode ser vista como um problema de agrupamento de documentos [Yang and Lee 2004]. Para tal, é necessário definir um modelo para representação estruturada dos textos, uma medida de similaridade entre os documentos e uma estratégia para formação dos grupos [Feldman and Sanger 2006].

O modelo espaço-vetorial é uma das formas mais comuns para representação dos textos [Manning et al. 2008]. Neste modelo, cada documento é representado como um vetor $x = \{t_1, t_2, ..., t_m\}$ no espaço formado pelos termos da coleção textual. Os termos são palavras selecionadas para representar os documentos de forma concisa e estruturada. Cada termo t_i possui um valor associado que indica sua importância (peso) para determinado documento. Em particular, neste trabalho utiliza-se a frequência de ocorrência de t_i no documento d, pois é um critério computacionalmente eficiente e, ainda, é facilmente aplicável no processamento de documentos em cenários dinâmicos. Um grupo de documentos $G = \{x_1, x_2, ..., x_n\}$ também possui representação no modelo espaço-vetorial por meio de um centroide $C_G = (\sum_{i=0}^{n} x_i)/|G|$, o vetor médio de todos os documentos de G.

A similaridade entre dois documentos (ou grupos de documentos) representados no modelo espaço-vetorial é usualmente computada pela medida cosseno [Zhao et al. 2005]. Nesta medida, sejam x_i e x_j dois documentos, o ângulo cosseno $cos(x_i, x_j) = \frac{x_i \cdot x_j}{\|x_i\|\|x_j\|}$ tem valor 1 quando os dois documentos são idênticos e valor 0

[3]Torch - Topic Hierarchies: http://sites.labic.icmc.usp.br/marcacini/ihtc/

quando não possuem nenhum termo em comum (são ortogonais entre si). Em alguns casos, é útil adaptar a medida de similaridade cosseno para uma medida de dissimilaridade por meio da equação $dis(x_i, x_j) = 1 - cos(x_i, x_j)$.

As estratégias para formação dos grupos, com uso de algoritmos de agrupamento hierárquico, podem ser divididas em aglomerativas e divisivas [Xu and Wunsch 2008]. No agrupamento hierárquico aglomerativo, inicialmente cada documento é um grupo unitário e, em cada iteração, os pares de grupos mais próximos são unidos até formar um único grupo. Já no agrupamento hierárquico divisivo, inicia-se com um grupo contendo todos os documentos que é, então, dividido em grupos menores até restarem grupos unitários. Avaliações experimentais indicam que o algoritmo UPGMA (aglomerativo) e o Bisecting-kmeans (divisivo) são os que obtêm melhores resultados em dados textuais [Zhao and Karypis 2002]. No entanto, tanto as estratégias aglomerativas quanto as divisivas possuem, em geral, complexidade quadrática de tempo e espaço, o que limita sua aplicação em grandes bases de dados e em cenários dinâmicos que requerem frequentes atualizações no agrupamento.

Para superar esta limitação, pode-se utilizar um método de agrupamento baseado em condensação [Xu and Wunsch 2008]. Nesses métodos, o agrupamento hierárquico é aplicado sobre uma representação condensada dos dados ao invés do conjunto total dos dados. A representação condensada mantém um sumário com as características principais dos dados, possibilitando processar grandes bases de dados com redução significativa do custo computacional [Nassar et al. 2004, Xu and Wunsch 2008]. Para exemplificar, considere a base de dados ilustrada na Figura 1. Em um primeiro momento, o agrupamento hierárquico é obtido a partir do conjunto total dos dados (Figura 1a). Alternativamente, na Figura 1b, a base de dados é sumarizada em uma representação condensada em 14 grupos. O agrupamento hierárquico aplicado sobre esta representação obtém uma estrutura hierárquica com propriedades similares à anterior. No entanto, ao usar 14 centroides como representação condensada, há uma significativa redução do custo computacional.

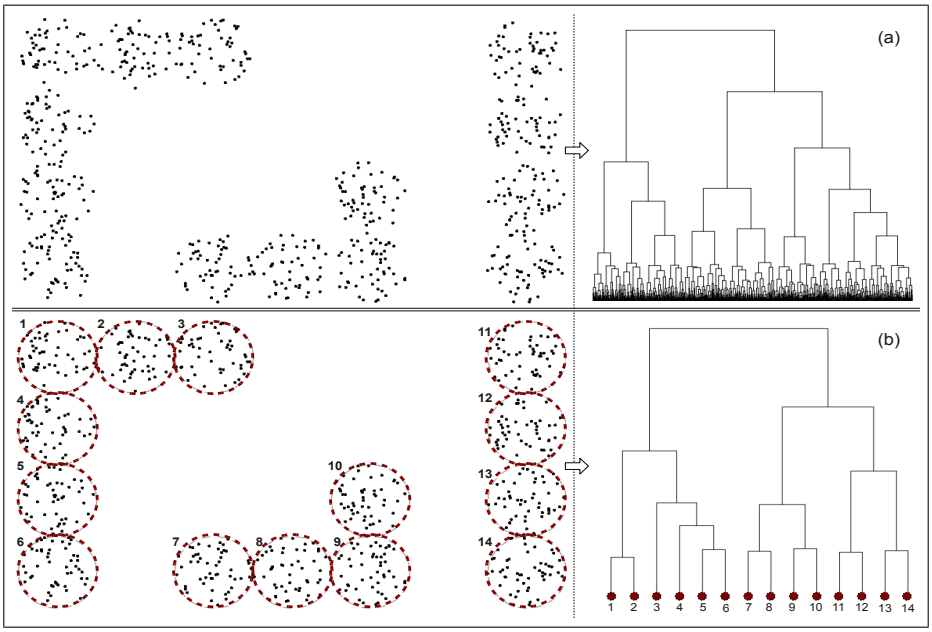


Figura 1. (a) Agrupamento hierárquico. (b) Agrupamento hierárquico resultante da representação condensada dos dados.

A técnica utilizada para obter a representação condensada dos dados, em cenários dinâmicos como os diretórios web, deve construir e atualizar esta representação maneira incremental. Para tal, são utilizados algoritmos de agrupamento incremental conhecidos como "*single-pass*", capazes de obter uma estrutura de agrupamento com poucas iterações sobre os dados, geralmente em tempo linear [Bradley et al. 1998, Farnstrom et al. 2000, Xu and Wunsch 2008]. Em um algoritmo de agrupamento "*single-pass*", cada novo objeto inserido é alocado ao grupo existente mais próximo, por exemplo, computando a similaridade entre o novo objeto e os centroides dos grupos existentes. Se este valor de similaridade não satisfizer um critério predeterminado, então um novo grupo é criado para alocar o objeto. O número máximo de grupos a ser criado define o nível de condensação da representação, e pode ser controlado de acordo com a memória disponível para a aplicação. O algoritmo Leader [Jain et al. 1999, Xu and Wunsch 2008] é um dos mais simples nesta categoria, e é utilizado em vários trabalhos que envolvem agrupamento incremental. No Leader, o agrupamento é obtido de maneira incremental com complexidade de tempo $O(n \cdot k)$ e de espaço $O(k)$, em que n é o número de objetos e k o número máximo de grupos. Ao usar o Leader para obter a representação condensada dos dados, a complexidade do agrupamento hierárquico a partir desta representação é de $O(k^2)$, em contraste com $O(n^2)$ quando aplicado em todo o conjunto de dados (em geral $n \gg k$).

A interpretação do agrupamento é uma dificuldade neste processo. Em geral, é necessário aplicar uma técnica após o agrupamento para seleção dos termos mais discriminantes e utilizá-los como descritores do agrupamento [Manning et al. 2008], por exemplo, pela seleção dos termos mais importantes em relação ao centroide de cada grupo [Moura and Rezende 2010]. A abordagem proposta neste trabalho obtém descritores juntamente com o processo de agrupamento incremental.

3. Construção Automática de Diretórios Web

Na seção anterior foi discutido um processo de agrupamento no qual similaridades entre documentos são o foco central na obtenção dos grupos. Neste trabalho, é explorada uma abordagem focada no agrupamento de termos da coleção textual. O objetivo é extrair grupos de termos correlacionados de acordo com suas distribuições nos documentos. Os grupos de termos identificam conceitos e tópicos implícitos nos textos, de forma que documentos relacionados a um mesmo tópico possam ser alocados em um mesmo grupo.

O método IHTC (*Incremental Hierarchical Term Clustering*), proposto inicialmente em [Marcacini and Rezende 2010a], permite organizar coleções textuais utilizando uma estratégia de agrupamento incremental de termos. A principal vantagem do IHTC é obter o agrupamento de documentos com descritores associados, permitindo a análise exploratória de coleções textuais.

Neste trabalho, o IHTC foi adaptado para obter um representação condensada dos dados e, assim, apoiar a construção automática de diretórios web. Uma visão geral da abordagem proposta é apresentada na Figura 2. A abordagem é dividida em quatro fases: (1) construção da rede de coocorrência de termos; (2) agrupamento de termos; (3) agrupamento de documentos e (4) extração da organização hierárquica. As três primeiras fases são responsáveis por obter uma representação condensada da coleção textual, de maneira incremental, que é utilizada na organização hierárquica dos textos na última fase.

Uma **rede coocorrência de termos** é definida como um grafo no qual os vértices

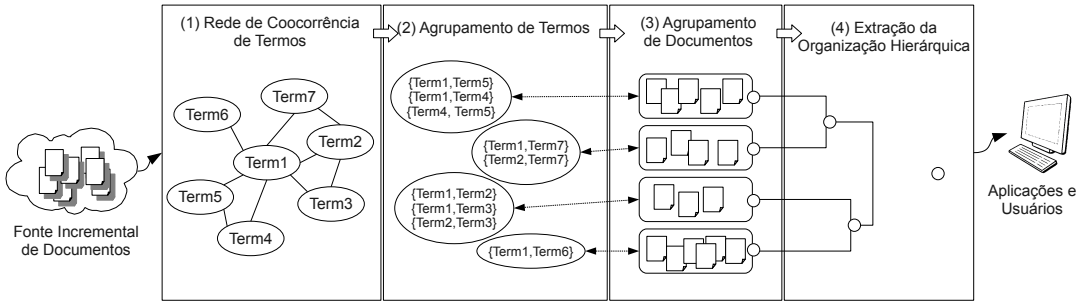


Figura 2. Visão geral da abordagem proposta

são os termos existentes na coleção textual, mais especificamente, termos selecionados para representação de cada documento no modelo espaço-vetorial. No entanto, nem todos os termos são utilizados como vértices. Para um termo ser utilizado, é necessário que o mesmo participe de uma relação de coocorrência com outro termo da coleção, ou seja, ocorrerem juntos em múltiplos documentos da coleção textual. A coocorrência entre dois termos também identifica as arestas do grafo. Assim, dois termos são conectados por uma aresta se existe frequência de coocorrência significativa entre eles, que é dependente da coleção textual. Em cada aresta, é associado um centroide que identifica relação entre dois termos t_i e t_j, por meio de uma função $w(e) = C(t_i \cap t_j)$, em que $C(t_i \cap t_j)$ é o centroide que representa o subconjunto de documentos com ambos os termos t_i e t_j. Desta forma, a rede de coocorrência de termos, conforme aplicada neste trabalho, pode ser vista como uma estrutura com duas características principais:

1. Capaz de identificar relações significativas entre os termos da coleção textual, baseada na frequência de coocorrência; e
2. Capaz de extrair subconjuntos de documentos (representados por centroides), em que os pares de termos (arestas) podem ser utilizados como descritores.

Algoritmo 1: Construção incremental da rede de coocorrência de termos

Parâmetros:
$X_{inc} = \{x_1, x_2, ...\}$: fonte incremental de documentos
r: número máximo de arestas para a rede de coocorrência de termos

```
1  M ← inicializar lista global com r contadores;
2  para cada novo documento x ∈ X_inc faça
3      E_x ← pares de termos obtidos do documento x;
4      para cada par de termos e = {t_i, t_j} ∈ E_x faça
5          se e = {t_i, t_j} está na lista de contadores M então
6              m_e ← contador de e na lista M;
7              m_e ← m_e + 1;
8              atualizar centroide da aresta e com o documento x;
9          senão
10             m_old ← menor contador da lista M;
11             m_new ← novo contador para e;
12             m_new ← m_old + 1;
13             substituir m_old por m_new na lista M;
14         fim
15     fim
16 fim
```

No IHTC, a rede de coocorrência de termos é construída e atualizada de forma

incremental, conforme apresentado no Algoritmo 1 [Marcacini and Rezende 2010a]. O algoritmo proposto é inspirado em uma técnica de detecção de elementos frequentes em "*data streams*" [Metwally et al. 2005]. Neste algoritmo, cada par de termos (aresta) extraído dos textos é monitorado por r contadores. Se um par de termos $e = \{t_i, t_j\}$ é frequente, então os termos t_i e t_j podem ser conectados na rede de coocorrência. Para tal, cada novo documento apresentado é processado individualmente. Os diferentes pares de termos do documento x são extraídos e armazenados em E_x (Linha 3). Para cada par de termos $e = \{t_i, t_j\} \in E_x$, é verificado se o mesmo já está sendo monitorado por algum contador. Se e já estiver sendo monitorado (Linha 5), significa que foi adicionado por um documento anterior, então o valor de frequência de coocorrência de e é incrementado (Linha 7) e a rede é atualizada (Linha 8). Caso contrário, busca-se o contador com menor valor de frequência m_{old} (Linha 10) e, em seguida, m_{old} é substituído pelo contador de e, identificado como m_{new} (Linha 13). Um detalhe desta substituição é que o valor para m_{new} é calculado como $m_{old} + 1$ (Linha 12), ou seja, o valor é superestimado a fim de garantir que cada elemento frequente seja monitorado. Desta forma, ao longo do tempo a lista de contadores mantém os pares de termos mais frequentes, enquanto os menos frequentes são gradativamente removidos. O número de contadores r é um parâmetro do algoritmo e define o tamanho máximo de arestas da rede de coocorrência de termos.

O objetivo da fase de **agrupamento de termos** é sumarizar as relações existentes na rede de coocorrência de termos. Para cada atualização ocorrida na rede, o respectivo centroide (aresta) é visto como um novo objeto por um algoritmo do tipo "*single-pass*". Assim, mantém-se o agrupamento de termos de forma incremental, no qual cada grupo possui pares de termos correlacionados que identificam tópicos existentes nos textos.

Na fase de **agrupamento de documentos**, os documentos da coleção são mapeados aos grupos de termos mais próximos. Assim, após o documento x passar pela fase de construção da rede de coocorrência e da fase de agrupamento de termos, calcula-se a similaridade cosseno entre x e o centróide de cada grupo de termos existente naquela iteração. Em seguida, o documento x é mapeado ao grupo de termos com maior valor de similaridade. Desta forma, o agrupamento final é formado por grupos do tipo $G_i = (G_i^d, G_i^t)$, ou seja, composto por um grupo de documentos G_i^d e um grupo de termos G_i^t; e os termos existentes em G_i^t são utilizados como descritores de G_i^d.

Por fim, a **extração da organização hierárquica** é feita com base na representação condensada dos dados obtida até o momento. Para tal, pode ser aplicado um algoritmo de agrupamento hierárquico sobre a representação condensada.

A complexidade de tempo para manter a representação condensada dos dados é linear em termo de números de documentos [Marcacini and Rezende 2010a], o que torna a abordagem útil em cenários dinâmicos. A organização hierárquica final possui grupos e descritores associados. Em diretórios web, cada grupo representa um diretório e os descritores do grupo são selecionados para indicar o conteúdo dos documentos ali agrupados.

4. Avaliação Experimental

A abordagem proposta neste trabalho para construção de diretórios web foi avaliada experimentalmente em uma base de dados real proveniente do projeto Dmoz ("*Open Directory Project*"). O Dmoz é um diretório público gerenciado manualmente por uma comunidade global de editores voluntários. Atualmente, várias ferramentas de busca na internet, como

o Google e AOL Search, incorporam o Dmoz em seus sistemas.

Em vista da relevância e da disponibilidade pública da sua base de dados, o Dmoz foi utilizado neste trabalho para um estudo exploratório sobre a construção automática de diretórios web. O objetivo é avaliar a abordagem proposta com o método IHTC e compará-la com uma estratégia similar da literatura baseada no algoritmo Leader [Jain et al. 1999, Xu and Wunsch 2008], analisando-se a qualidade do agrupamento hierárquico e dos descritores dos grupos.

4.1. Base de dados

Para a avaliação experimental, foi selecionado um subconjunto dos dados com 5 diretórios, provenientes do projeto Dmoz: "*Business*", "*Computers*", "*Health*", "*Science*" e "*Sports*". Cada diretório possui subdiretórios e documentos relacionados. Ao final, obteve-se uma base de dados com 566.234 documentos organizados em 43.817 diretórios e subdiretórios. Na Tabela 1 é apresentada uma visão geral da base de dados.

Tabela 1. Visão geral da base de dados selecionada a partir da base do Dmoz

Características da base de dados	
Documentos da Coleção	566234
Número de Termos	246280
Número de Termos com DF ≥ 2	84039
Número de (Sub)Diretórios	43817
Altura da Hierarquia	11
Média de Termos por Documento	12,41

O número total de termos (após remoção de *stopwords*[4] e aplicação de *stemming*) é de 246.280. Deste total, há 84.039 termos que ocorrem em dois ou mais documentos (DF ≥ 2) e que podem ser úteis para a tarefa de agrupamento. Os documentos associados aos diretórios da hierarquia representam uma página na web. Cada documento é composto por quatro campos: (1) o título da página, (2) a URL com o endereço de internet para a página, (3) uma breve descrição com 25 à 30 palavras sobre o conteúdo da página; e (4) o diretório da hierarquia na qual a página está alocada (removido para a realização dos experimentos). Por fim, foram selecionados 5 subconjuntos de dados, contendo diretórios de referência para serem utilizado na etapa de validação. Assim, é possível avaliar a eficácia da abordagem proposta por meio de critérios objetivos.

4.2. Critério de Avaliação

O índice FScore [Zhao and Karypis 2002] é uma medida que utiliza as idéias de *precisão* e *revocação*, da recuperação de informação, para avaliar a eficácia de recuperação em uma organização hierárquica de documentos. É empregada como critério externo de validação, pois utiliza o conhecimento prévio (informação externa) sobre categorias ou tópicos existentes. A ideia básica é verificar o quanto uma organização hierárquica de uma base de dados consegue recuperar a informação de categoria associada a cada documento. Para o cálculo do índice FScore, considere que

- H é um agrupamento hierárquico obtido por um determinado algoritmo;

[4]Os textos do DMOZ são em inglês, então foi utilizada uma lista de *stopwords* padrão para a língua inglesa (disponível em `http://sites.labic.icmc.usp.br/marcacini/ihtc`).

- L_r é uma determinada categoria/diretório (informação externa) representando um conjunto de documentos de um mesmo tópico; e
- G_i é um determinado grupo, e seu respectivo conjunto de documentos, pertencente ao agrupamento hierárquico H.

Assim, dada uma categoria L_r e um grupo G_i, calcula-se as medidas de precisão P e revocação R conforme a Equação 1 e Equação 2, respectivamente. Em seguida, é obtida a média harmônica F (Equação 3), um balanceamento entre a precisão e revocação.

$$P(L_r, G_i) = \frac{|L_r \cap G_i|}{|G_i|} \qquad (1) \qquad\qquad R(L_r, G_i) = \frac{|L_r \cap G_i|}{|L_i|} \qquad (2)$$

$$F(L_r, G_i) = \frac{2 * P(L_r, G_i) * R(L_r, G_i)}{P(L_r, G_i) + R(L_r, G_i)} \qquad (3)$$

A medida F selecionada para uma determinada categoria L_r é o maior valor obtido por algum grupo da hierarquia *H*, conforme a Equação 4. Finalmente, o valor FScore global de um agrupamento hierárquico com *n* documentos e *c* categorias, é o somatório do valor F das categorias ponderado pelo número de documentos (Equação 5).

$$F(L_r) = \max_{G_i \in H} F(L_r, G_i) \qquad (4) \qquad\qquad FScore = \sum_{r=1}^{c} \frac{|L_r|}{n} F(L_r) \qquad (5)$$

Conforme o agrupamento hierárquico consegue recuperar a informação das categorias predeterminadas de uma coleção, o valor de FScore se aproxima de 1. Caso contrário, a FScore tem valor 0.

O mesmo processo pode ser aplicado para avaliar os descritores selecionados para o agrupamento. Neste caso, o conjunto de documentos associado ao grupo G_i é substituído pelo conjunto de documentos T_i recuperados utilizando os descritores de G_i em uma expressão de busca. Assim, o índice FScore é utilizado tanto para medir a eficácia de recuperação do agrupamento hierárquico H quanto a eficácia de recuperação dos descritores selecionados para H.

4.3. Experimentos Realizados e Análise dos Resultados

Os experimentos foram realizados com auxílio da ferramenta *Torch - Topic Hierarchies* [Marcacini and Rezende 2010b], que atualmente disponibiliza vários algoritmos de agrupamento incremental de documentos, além de técnicas de pré-processamento de textos.

Foram realizados dois experimentos de construção automática de diretórios web. O primeiro com a abordagem proposta, representada por IHTC, e o segundo com base no algoritmo Leader. Para o IHTC, foi definido o parâmetro $r = 20000$ para o tamanho máximo da rede de coocorrência de termos. No caso do Leader, o parâmetro de dissimilaridade mínima foi definido como $\gamma = 0.8$. O agrupamento hierárquico final foi obtido por meio do *Bisecting-kmeans*, a partir de uma representação condensada em 3 mil grupos. A seleção de descritores para o agrupamento obtido com base no Leader é feita por um método baseado em centroides. Os parâmetros utilizados em ambos os algoritmos foram definidos de acordo com experimentos preliminares.

A seguir, é realizada uma análise dos resultados obtidos de acordo com dois critérios calculados com o índice FScore: (1) a eficácia de recuperação do agrupamento

hierárquico; e (2) a eficácia de recuperação dos descritores selecionados para o agrupamento. Para determinar o grau de confiança das comparações realizadas, foi aplicado o teste estatístico não paramétrico de Wilcoxon conforme descrito em [Demšar 2006].

Em relação ao critério de **eficácia da recuperação do agrupamento hierárquico**, o IHTC e o Leader obtiveram resultados similares. Na Figura 3a é ilustrada uma comparação entre a medida FScore em 5 subconjuntos de validação. O Leader apresentou resultados superiores em relação ao agrupamento hierárquico, entretanto, não foi possível indicar diferença estatisticamente significativa na comparação deste critério.

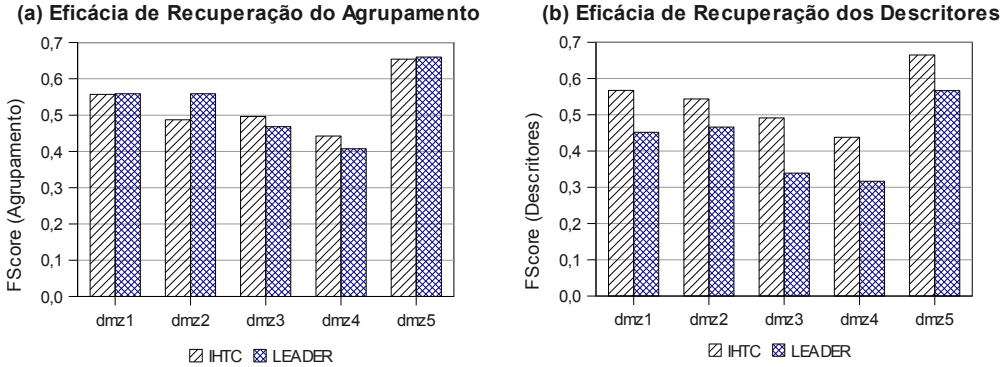


Figura 3. Eficácia de recuperação do agrupamento hierárquico e dos descritores entre o IHTC e Leader

Por outro lado, o IHTC obteve resultados superiores em relação ao critério de **eficácia da recuperação dos descritores**, com diferença estatisticamente significativa. Conforme descrito anteriormente, o IHTC permite fornecer grupos de termos como descritores para o agrupamento. Ao utilizar esses grupos de termos, os descritores selecionados para a hierarquia obtiveram maior eficácia de recuperação do que a estratégia baseada em centroide adotada no algoritmo Leader. Na Figura 3b é ilustrada a comparação da eficácia da recuperação dos descritores.

Este resultado está conformidade com a proposta do IHTC, que visa auxiliar a tarefa de interpretação do agrupamento. Para isto, o IHTC utiliza as relações existentes na rede de coocorrência de termos para identificação de possíveis tópicos na coleção textual. Na Tabela 2 são apresentados alguns dos diretórios de referência extraídos do Dmoz (construída por humanos). Para cada tópico de referência, também são apresentados os respectivos descritores obtidos pelos experimentos realizados com o IHTC e Leader.

É importante observar que o IHTC possui a vantagem de utilizar pares de termos (arestas da rede de coocorrências) como descritores dos grupos. Em geral, uma expressão ou conceito identificado por conjuntos de termos relacionados tem maior poder discriminativo do que termos simples. Mesmo em situações em que termos simples apresentam melhor eficácia de recuperação, o uso de conjunto de termos pode auxiliar mais os usuários na interpretação dos grupos, por exemplo, no caso de "*Cancers, Treatments*" e "*Cancers, Researchs*" em vez do termo simples "*Cancers*".

Os diretórios web completos obtidos com a realização deste trabalho estão disponíveis *online* no endereço http://sites.labic.icmc.usp.br/marcacini/ihtc, incluindo a base de dados utilizada, a ferramenta *Torch*, e um módulo computacional para apoiar a construção e publicação de diretórios web.

Tabela 2. Exemplo de diretórios obtidos automaticamente pelo IHTC e Leader em comparação com o diretório original da Dmoz

Origem	Diretório (Tópico)	Eficácia da Recuperação	
		Agrupamento	Descritores
DMOZ	Cancer	-	-
IHTC	{Cancers,Treatments}, {Cancers,Researchers}, {Cancers,Information}	**0,335**	0,499
Leader	Cancers, Breasts, Nci	0,332	**0,582**
DMOZ	Veterinarians	-	-
IHTC	{Animals,Services}, {Animals,Hospital}, {Hospital,Services}	0,706	**0,691**
Leader	Veterinary, Hours, Hospital	**0,782**	0,430
DMOZ	Translation	-	-
IHTC	{English,Translations}, {Germans,Translations}, {Frenchs,Translations}	**0,706**	**0,684**
Leader	Translations, Interpreting, English	0,691	0,481
DMOZ	Robotics	-	-
IHTC	{Researchers,Robotics}, {Mobile,Robotics}, {Controls,Robotics}	0,418	0,379
Leader	Robotics, Estimators, Autonomous	**0,524**	**0,486**
DMOZ	Money_Managers	-	-
IHTC	{Financial,Investment}, {Investment,Planning}, {Advisory,Investment}	**0,406**	**0,228**
Leader	Investment, Advisors, Advisory	0,260	0,041
DMOZ	Search_and_Rescue	-	-
IHTC	{Rescue,Searches}, {Searches,Volunteer}, {Searches,Teams}	0,428	**0,737**
Leader	Rescue, Trucks, Volunteer	**0,450**	0,184
DMOZ	Accounting	-	-
IHTC	{Services,Taxes}, {Accounting,Services}, {Accounting,Taxes}	0,593	0,611
Leader	Taxes, Accounting, Cpas	**0,655**	**0,653**

5. Considerações Finais e Trabalhos Futuros

Neste trabalho, foi descrita uma abordagem para construção automática de diretórios web que ataca dois desafios atuais: o agrupamento incremental e a interpretação dos resultados do agrupamento. A abordagem é baseada no método IHTC, desenvolvido para organizar coleções de textos em grupos de documentos com descritores associados.

As avaliações experimentais realizadas indicam que a abordagem proposta é competitiva. O diretório web construído com base no IHTC possui qualidade de agrupamento similar ao Leader. No entanto, o IHTC tem a capacidade de obter melhores descritores para os diretórios. Isto auxilia usuários em tarefas de busca exploratória, o que o torna especialmente útil na construção de diretórios web.

Como trabalho futuro, espera-se realizar uma avaliação mais profunda dos resultados, por meio da participação de usuários, utilização de um maior número de conjuntos de validação e emprego de outros critérios para análise do agrupamento. Ainda, explorar a rede de coocorrência de termos para extração de *n-gramas* na seleção de descritores.

Agradecimentos

Os autores gostariam de agradecer à FAPESP e ao CNPq pelo apoio financeiro.

Referências

Bradley, P. S., Fayyad, U. M., and Reina, C. (1998). Scaling Clustering Algorithms to Large Databases. In *Knowledge Discovery and Data Mining*, pages 9–15.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30.

Farnstrom, F., Lewis, J., and Elkan, C. (2000). Scalability for clustering algorithms revisited. *ACM SIGKDD Explorations Newsletter*, 2:51–57.

Feldman, R. and Sanger, J. (2006). *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press.

Fung, B. C. M., Wang, K., and Ester, M. (2008). *The Encyclopedia of Data Warehousing and Mining*, chapter Hierarchical Document Clustering, pages 970–975. Idea Group.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323.

Kim, H. J. (2006). On text mining algorithms for automated maintenance of hierarchical knowledge directory. In *Knowledge Science, Engineering and Management*, Lecture Notes in Computer Science, pages 202–214.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *An Introduction to Information Retrieval*. Cambridge University Press.

Marcacini, R. M. and Rezende, S. O. (2010a). Incremental construction of topic hierarchies using hierarchical term clustering. In *SEKE'2010: Proceedings of the 22nd International Conference on Software Engineering and Knowledge Engineering*, pages 553–558. KSI - Knowledge Systems Institute.

Marcacini, R. M. and Rezende, S. O. (2010b). Torch: a tool for building topic hierarchies from growing text collection. In *WFA'2010: IX Workshop de Ferramentas e Aplicações - XVI Webmedia*, pages 1–3.

Marchionini, G. (2006). Exploratory search: from finding to understanding. *Communications of ACM*, 49(4):41–46.

Metwally, A., Agrawal, D., and Abbadi, A. E. (2005). Efficient computation of frequent and top-k elements in data streams. In *ICDT'05: Proceedings of 10th International Conference on Database Theory*, pages 398–412.

Moura, M. F. and Rezende, S. O. (2010). A simple method for labeling hierarchical document clusters. In *IAI'10: Proceedings of the 10th International Conference on Artificial Intelligence and Applications*, pages 363–371, Acta Press, 2010.

Nassar, S., Sander, J., and Cheng, C. (2004). Incremental and effective data summarization for dynamic hierarchical clustering. In *SIGMOD'04: Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 467–478.

Premalatha, K. and Natarajan, A. (2010). A Literature Review on Document Clustering. *Information Technology Journal*, 9(5):993–1002.

Xu, R. and Wunsch, D. (2008). *Clustering*. Wiley-IEEE Press, IEEE Press Series on Computational Intelligence.

Yang, H. C. and Lee, C. H. (2004). A text mining approach on automatic generation of web directories and hierarchies. *Expert Systems with Applications*, 27(4):645–663.

Zhao, Y. and Karypis, G. (2002). Evaluation of hierarchical clustering algorithms for document datasets. In *CIKM '02: Proceedings of the 11th International Conference on Information and Knowledge Management*, pages 515–524.

Zhao, Y., Karypis, G., and Fayyad, U. (2005). Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, 10(2):141–168.

Detecção de Autores Duplicados Utilizando Estrutura de Comunidades em Redes de Cooperação Científica

Breno Júnio V. da Silva[1], **Robson Motta**[2], **Alneu de Andrade Lopes**[3]

[1]Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo (USP) – São Carlos – SP – Brazil

Abstract. *In collections of scientific papers is frequent to find different quotational names of a same author. For many applications, these duplicate records need to be identified. This is an instance of the problem known as identification of duplicates, for which good results have not been achieved yet. This study investigated the use of scientific cooperation networks and community detection techniques to deal with the problem of identifying duplicates. The results indicate that such strategy not only improves the accuracy of the identification of duplicates, but also reduces the computational cost associated with this task when compared with approaches in which a record is compared against all the others.*

Resumo. *Nas coleções de artigos científicos é frequente encontrar nomes diferentes de citação de um mesmo autor. Para muitas aplicações, estes registros duplicados devem ser identificados. Esta é uma instância do problema conhecido como a identificação de duplicados, para o qual bons resultados não foram alcançados ainda. Este estudo investigou a utilização de redes de cooperação científica e técnicas de detecção da comunidade para lidar com o problema de identificação de duplicados. Os resultados indicam que essa estratégia não só melhora a precisão da identificação de duplicatas, mas também reduz o custo computacional associado a esta tarefa quando comparado com as abordagens em que um registro é comparado com todos os outros.*

1. Introdução

A grande quantidade de dados armazenados em meio digital cresceu consideravelmente nas últimas décadas, dificultando a possibilidade de exploração de conhecimento específico e relevante. Desta forma, diversos trabalhos em diferentes áreas da ciência desenvolvem métodos automáticos para extrair conhecimento a partir de grandes volumes de dados [Fayyad et al. 1996].

Aprendizado de Máquina, subárea de Inteligência Artificial, é uma das áreas de pesquisa que estuda algoritmos e técnicas relacionadas a obtenção automática de conhecimento a partir de dados. O aprendizado de máquina é uma área multidisciplinar envolvendo conhecimentos de estatística, computação, matemática, biologia entre outros [Mitchell 1997]. Para o uso das técnicas de aprendizado de máquina é fundamental definir como os dados e o conhecimento serão representados, bem como o mecanismo de inferência apropriado para a tarefa em questão. A representação dos dados de entrada para a maioria dos algoritmos pode ser dividida em dois modelos: a representação proposicional, na qual os objetos são representados somente pelas suas características; e a

representação relacional, na qual considera as características individuais e as relações existentes entre os objetos [Raedt 2008].

Um grafo (ou rede) pode ser utilizado para representação dos dados de forma relacional. Redes são estruturas compostas por um conjunto de vértices e um conjunto de arestas que ligam os pares de vértices conforme as relações existentes no mundo real ou no sistema em questão [Newman 2003]. Redes Complexas refere-se a uma rede que apresenta uma estrutura de conexões não trivial, como por exemplo, a *World Wide Web* ou as redes sociais como *Orkut*[1], *Facebook*[2] e outras [Reka Albert and Barabasi 1999].

A representação relacional, neste trabalho, é usada para agregar informações sobre cooperação científica entre pesquisadores e usar essa informação para melhorar a detecção de registros duplicados em base de dados. Bases de dados podem conter registros que se referem à mesma entidade no mundo real, porém, não são sintaticamente idênticos. Variações na representação dos dados podem surgir de erros de digitação, erros ortográficos, abreviações, não padronização na entrada de dados e integração de múltiplas bases de dados. Dados extraídos de documentos não estruturados, semi-estruturados ou páginas da *web* tendem a propagar mais ruídos às bases de dados [Bilenko and Mooney 2003]. Neste trabalho, quaisquer pares de registros que sejam sintaticamente diferentes, mas representem o mesmo objeto no mundo real, serão identificados como duplicados.

O problema de registros duplicados se estende a bibliotecas digitais, onde a não utilização de padrões de citações e/ou união de diversas bibliotecas, fazem com que surjam registros duplicados de autores. É comum o mesmo autor aparecer em portais de periódicos com nomes de citação diferentes, por exemplo, “M. E. J. Newman”, “Mark Newman” e “Newman, Mark E.”. Esse problema é tradicionalmente tratado por uma tediosa análise manual dos dados e com técnicas automáticas e semi-automáticas que ainda não alcançam uma precisão desejável. Além disso, detectar um autor duplicado, grafado de formas distintas, é um problema computacionalmente caro. Uma implementação básica, em que são confrontados todos os autores de uma base de dados, seria no mínimo de complexidade computacional de ordem quadrática no número de registros.

Considerando o problema de autores duplicados , o objetivo deste trabalho é usar o formalismo de redes complexas de cooperação científica e de detecção de comunidades para identificar registros duplicados de autores em uma coleção de artigos científicos. A abordagem investigada detecta grupos de vértices fortemente conectados em uma rede de cooperação científica, e os utiliza para reduzir a complexidade do problema, minimizando a quantidade de pares de autores a serem comparados, e melhorar a precisão da detecção de duplicados.

O trabalho está organizado da seguinte forma. Na Seção 2, é apresentada uma revisão dos trabalhos relacionados ao problema. Na Seção 3, a modelagem proposta para simular o problema é apresentada. Na Seção 4, os resultados de cobertura e precisão da técnica são apresentados, bem como sua avaliação da complexidade de tempo. Por fim, na Seção 5 a conclusão e proposta de trabalhos futuros.

[1]http://www.orkut.com
[2]http://www.facebook.com

2. Detecção de Registros Duplicados

Em grandes bases de dados, são facilmente encontrados dados duplicados ou referências erradas. No mundo real, o problema se torna ainda mais complexo quando várias bases de dados são agrupadas. O problema se estende para bibliotecas digitas, pois, a falta de convenções para referências agrava o problema, dificultando obter medidas que caracterizam a base de dados. Medidas tradicionais de similaridade entre cadeias de caracteres podem ser usadas para identificar erros tipográficos e outros tipos de ruídos. No entanto, é difícil identificar quando referências sintaticamente distintas se referem ao mesmo objeto no mundo.

Bhattacharya e Getoor [Bhattacharya and Getoor 2004, Bhattacharya and Getoor 2007] utilizaram um algoritmo iterativo para eliminar registros duplicados em bases de dados a partir dos relacionamentos existentes entre as entidades. A modelagem utilizada procura por evidências que determinam se dois registros idênticos são na verdade distintos, por exemplo, dois registros de nome de pessoas idênticos em uma base de dados cadastrais de pessoas. Essas evidências são informações adicionais a um registro, as quais não são consideradas em abordagens tradicionais que utilizam medidas de aproximação de cadeia de caracteres. Como evidência adicional há, por exemplo, informação de filiação nos registros ou nome do cônjuge.

Algoritmos genéticos foram aplicados para descobrir uma função de similaridade que é capaz de dizer se dois registros são o mesmo objeto no mundo real [de Carvalho et al. 2006]. A motivação para usar tal técnica é a capacidade existente nos algoritmos genéticos de se adaptar aos dados, identificar padrões de informações que não são óbvias e diminuir a interferência do especialista na definição dos parâmetros de entrada dos modelos de detecção dos registros duplicados.

Dong e colegas [Dong et al. 2005] usaram um algoritmo que se baseia na construção de um grafo de dependência. As dependências são informações contidas no próprio trabalho científico, como instituição do autor, endereço eletrônico, grupo de pesquisa entre outros. Os vértices representam as referências e as arestas representam as dependências entre as referências. A construção do grafo visa o uso extensivo de informações de contexto para fornecer evidências necessárias para identificar registros duplicados. Por exemplo, dados duas pessoas, considera-se as relações de co-autoria e o endereço eletrônico para auxiliar na decisão de identificar se são a mesma pessoa.

No contexto da *web*, observou-se que um determinado registro A pode ser um registro duplicado de B, se A aparece frequentemente associado a um tipo de informação, em que B também aparece. Utilizando motores de busca e um conjunto de entidades fornecidas, o algoritmo proposto por Elmacioglu e colegas [Elmacioglu et al. 2007] utiliza uma função de densidade baseada nos resultados obtidos dos motores de busca para identificar os registros duplicados. Diferentemente dos outros modelos propostos, este por sua vez, utiliza informações da *web*, então os resultados variam no tempo conforme ocorrem mudanças na *web*.

Conforme a quantidade de registros aumenta, a natureza quadrática torna algumas abordagens proibitiva para o problema [Kim and Lee 2007]. Diante deste contexto, são propostas diversas formas de abordarem ao problema, considerando-se o relacionamento entre os registros da base de dados, com resultados consideráveis.

De maneira geral, as propostas de soluções para o problema consideram o relacionamento entre os registros, medida de similaridade e grupos ou classes de registros, porém, os métodos necessitam de interferência humana. A proposta de substituir os parâmetros estimados por especialista para métodos dinâmicos que alteram e adaptam os parâmetros aos dados em tempo de execução aumentam as chances de classificar corretamente um registro, porém, não ajuda no problema de custo computacional [Paskalev and Antonov 2006, Paskalev and Antonov 2007].

3. Detecção de Comunidades na Identificação de Duplicados

As bases de dados de artigos científicos são, de uma maneira geral, compostas por documentos não estruturados, ou seja, documentos nos quais não existem etiquetas separando as informações no texto. Para a construção das redes de cooperação científica é necessário extrair informações pertinentes dos documentos. Para isto, a ferramenta IESystem, desenvolvida no ICMC para extração de informações em artigos, se mostrou eficiente [Rossi et al. 2010]. A ferramenta IESystem foi desenvolvida especificamente para extrair metadados (título, autores, resumo, referências e etc) de artigos científicos em diferentes idiomas.

Utilizando um esquema de modelos proposto pelo autor, a ferramenta IESystem transforma o texto não estruturado em um modelo estruturado. Por exemplo, se utilizar o modelo contendo título, autores e referências, o texto é separado por *tags* em que o título é inserido entre as etiquetas <Título> e < \Título>, a lista de autores é inserida em <Autores> e < \Autores> e por último, as referências são inseridas entre <Referências> e < \Referências>.

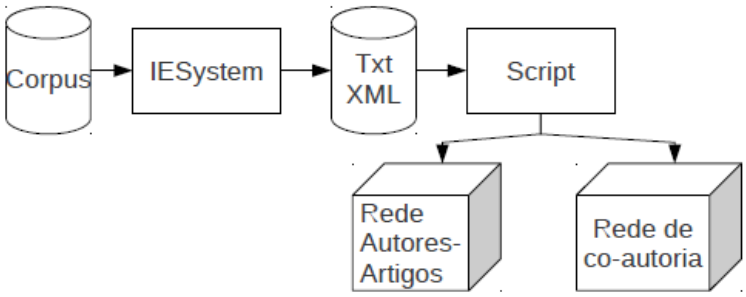


Figura 1. Sequência de atividades para extrair as informações dos artigos científicos e então construir a rede de cooperação científica.

Com as informações já identificadas pela ferramenta, uma nova base de dados parcialmente pré-processada e que mantêm as características de relacionamento original é construída. Para geração da rede de cooperação científica somente as informações referentes aos autores é necessária, sendo que neste trabalho somente os artigos completos estão sendo considerados, eliminando os autores das referências. A Figura 1 sintetiza as etapas do processo utilizado para construção das redes, utilizando *scripts* para processar os dados extraídos pela ferramenta IESystem e construir a rede de cooperação científica, conectando-se dois autores que estão presentes em um mesmo artigo.

A técnica proposta consiste em construir um sistema capaz de identificar autores duplicados em uma rede de cooperação científica, tendo como entrada a coleção de artigos

científicos em formato PDF. Para avaliar a qualidade da ferramenta na execução da tarefa foram utilizadas as medidas cobertura, precisão e *f-measure* [Witten and Frank 2005]. A ferramenta foi construída com a linguagem de programação *Python* [3].

O processo de detecção de autores duplicados é dinâmico e não dispensa a avaliação do especialista. A Figura 2 ilustra o modelo simplificado do funcionamento geral do sistema. Na entrada de dados, o sistema recebe uma rede de cooperação científica (ou co-autoria). Então processa essa rede e sugere ao especialista os possíveis autores duplicados. O especialista analisa os resultados e altera os parâmetros, de forma que se observe melhoria na qualidade dos resultados. Por fim, é possível atualizar a base de dados eliminando os itens duplicados.

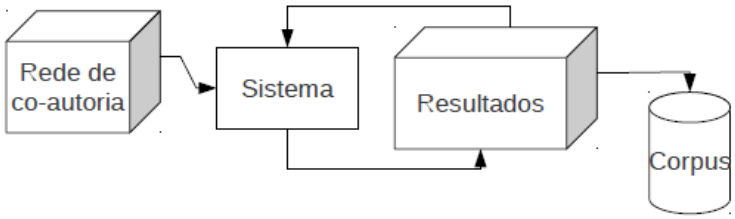


Figura 2. Sequência de atividades executadas pelo sistema para identificar autores duplicados na rede cooperação científica fornecida.

O sistema é dividido em quatro partes. Primeiro são as tarefas de pré-processamento da entrada de dados, em seguida detecção de comunidades na rede fornecida. Terceiro o processamento das comunidades com a medida de similaridade proposta e por fim o pós-processamento dos resultados.

A tarefa de pré-processamento da entrada de dados consiste em atribuir um número identificador (ID) para da cada elemento de entrada. Eliminar caracteres especiais como “pontos” (usados em abreviações) e espaços em branco entre nomes e sobrenomes. A eliminação desses caracteres formam as chaves de busca usadas na identificação dos duplicados, e o ID representa o vértice da rede.

Feito o pré-processamento da entrada de dados, inicia-se a fase de detecção de comunidades da rede. Estrutura de comunidades é uma característica comumente presente em redes complexas. As comunidades são formadas por vértices que são densamente conectados entre si e fracamente conectados a outros grupos [Newman 2004]. Normalmente, essas comunidades conectam vértices com alguma similaridade entre si.

Para detecção de comunidades exige-se como parâmetros uma rede e o número desejado de comunidades. Neste trabalho, utilizou-se o método de detecção de comunidades baseado na remoção de *betweenness* [Girvan and Newman 2002]. Neste método gera-se o caminho mínimo entre todos os pares de vértices na rede, e assume-se que arestas que fazem parte de muitos caminhos mínimos ligam grupos fortemente conectados, ou seja, as comunidades. A remoção iterativa destas arestas gera cada vez mais componentes, que são as comunidades identificadas na rede.

Após a detecção das comunidades, estas podem ser processadas individualmente. Comparando objeto por objeto a fim de identificar elementos duplicados usando a medida

[3]http://www.python.org/

de similaridade entre cadeia de caracteres (Medida de Levenshtein, também conhecida como Distância de Edição). A similaridade é definida em porcentagem sobre o comprimento da cadeia de caracteres. Por exemplo, dada uma cadeia de caracteres A de comprimento 10 e definido um limiar de diferença de 30% (ou limiar de similaridade em 70%), então diz-se que uma outra cadeia de caracteres B é similar a A se for necessário mudar 30% (ou manter 70%) dos caracteres de A para que esse se iguale a B.

Por fim, para cada comunidade é construída uma lista de itens duplicados, considerando o valor definido para o limiar de diferença utilizado na medida de similaridade. A fase de pós-processamento, exige que um especialista avalie os resultados, efetuando, se necessário, alterações no valor do limiar de diferença e no número de comunidades, refazendo o procedimento de identificação de duplicados. Um especialista pode também avaliar os itens duplicados, já os unificando na rede, permitindo, possivelmente, uma melhoria nos resultados futuros.

4. Resultados Obtidos

Para avaliação do modelo proposto, foi realizado um estudo de caso com um conjunto de artigos científicos. O conjunto de dados é formado por 992 artigos, sendo 654 documentos publicados entre 1994 e 2008 na área de *Case-Based Reasoning*, e 338 documentos publicados entre 1997 e 2008, na área de *Inductive Logic Programming*.

Para avaliar o modelo proposto a partir dos resultados obtidos foram utilizados 7 diferentes limiares de diferença para a medida de Levenshtein (20%, 25%, 30%, 35%, 40%, 45% e 50%), e o intervalo de 1 à 20 comunidades para cada limiar de diferença. A Tabela 1 contêm informações complementares da base de dados que se aplicam à rede de cooperação científica, no caso, cada autor é um vértice da rede e cada aresta é uma relação de co-autoria.

Base de Dados	Total de Artigos	Total de Autores
CBR-ILP	992	856

Tabela 1. Informações complementares da base de artigo científicos CBR-ILP.

Devido a dificuldade de saber exatamente quais são os duplicados existente na coleção, para a avaliação da metodologia, nesta base de dados foram inseridos 268 erros controlados, alterando então o número de objetos referentes ao mundo real de 856 para 1124. Os erros inseridos foram de alteração no formato de citação de um mesmo autor e erros de digitação. Erros de citação se referem a aproximadamente 70% dos erros inseridos, por exemplo, ao autor ”Mobyen Uddin Ahmed” são inseridas abreviações para o nome como “Mobyen U. A.” e “M. U. Ahmed”. Erros de digitação correspondem aos outros 30%, por exemplo, ao autor “José Ramon Méndez“, são inseridos erros de digitação ”José Ramom Méndez“ e ”José Ramon Méndes“. A avaliação foi feita com base na precisão e cobertura alcançada pela proposta, considerando o total de erros inseridos.

A primeira configuração para processar a base de dados CBR-ILP foi o limiar de diferença em 20% e quantidade de comunidades de 1 à 20. A média de objetos identificados como duplicados foi 170 e a média dos objetos identificados corretamente como duplicado foi 166, de um total de 268 erros inseridos.

Na Figura 3 observa-se o que no primeiro experimento a precisão é alta nos dados recuperados, porém, a cobertura dos objetos recuperados é baixa, e a variação na quantidade de comunidades não tem influência direta nos resultados. Tais características ocorrem também para os limiares de diferença em 25% e 30%, com uma pequena variação quando considerada uma comunidade ou valores acima de 15 comunidades, porém, conforme incrementa-se o limiar de diferença, consequentemente melhora-se a cobertura e prejudica-se a precisão.

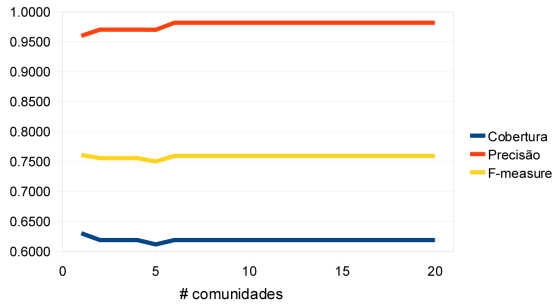


Figura 3. Cobertura, precisão e F-*measure* para o limiar de diferença em 20%.

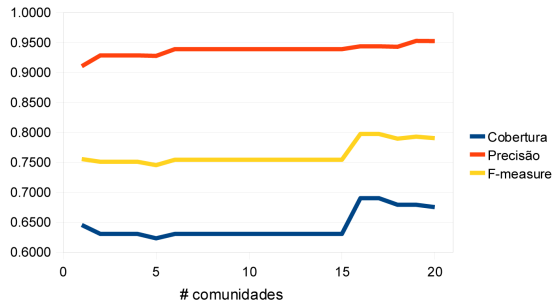


Figura 4. Cobertura, precisão e F-*measure* para o limiar de diferença em 25%.

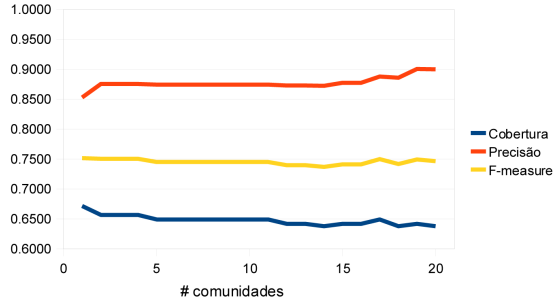


Figura 5. Cobertura, precisão e F-*measure* para o limiar de diferença em 30%.

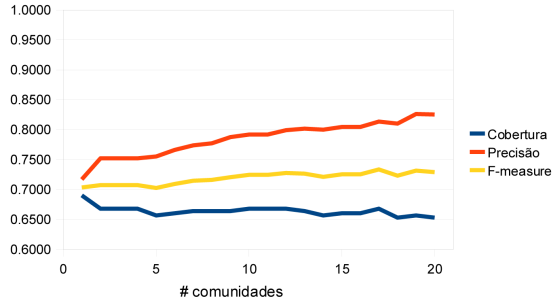


Figura 6. Cobertura, precisão e F-*measure* para o limiar de diferença em 35%.

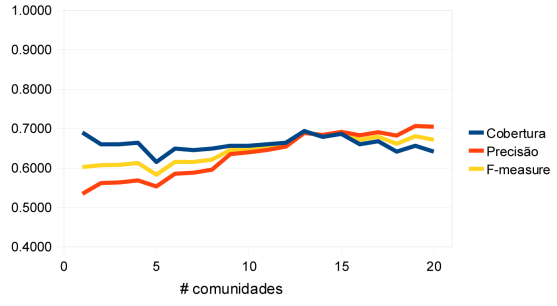


Figura 7. Cobertura, precisão e F-*measure* para o limiar de diferença em 40%.

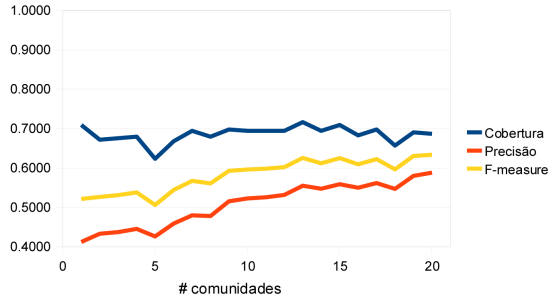


Figura 8. Cobertura, precisão e F-*measure* para o limiar de diferença em 45%.

Considerando os limiares de diferença 35% e 40%, observa-se que conforme aumenta-se o número de comunidades melhora-se a precisão e piora-se a cobertura, porém, a medida *f-measure* demonstra que a melhoria na precisão tem maior relevância

do que a perda da cobertura. Isso demonstra que a comparação somente entre objetos das comunidades apresenta uma melhoria comparada ao método que verifica todos os pares de objetos (equivalente a observar as Figuras 6 e 7 com 1 comunidade).

Mesma característica ocorre para os limiares de diferença 45% e 50%, com a cobertura atingindo altos valores, mas a precisão é prejudicada, ocorrendo muitos falsos positivos, i.é, registros, de fato distintos, são apresentados como possíveis duplicados. Ou seja, foram recuperados uma quantidade de duplicados superior a quantidade de erro inserido, resumida na Tabela 2.

Limiar de diferença	45%	50%
Média de objetos recuperados	366	553

Tabela 2. Objetos recuperados em média para limiares de diferença de 45% e 50%.

Portanto, foi possível observar que os resultados se dividiram em três grupos. O primeiro, contendo os valores para os limiares de diferença iguais a 20%, 25% e 30%, não apresenta ganho de cobertura e precisão no uso das comunidades, comparados com a verificação de todos os pares de objetos nos dados (ou seja, 1 comunidade), mas apresenta ganho no processamento, pois menos pares são comparados. O segundo grupo é para um limiar intermediário, no caso os valores de 35% e 40%, demonstrando um ganho nos resultados com o uso das comunidades. E, por fim, o terceiro grupo contém os valores dos limiares de diferença acima de 40%, o qual apresenta grande cobertura, mas contendo muitos falsos positivos.

Como já comentado, a complexidade computacional para determinar os autores duplicados no modelo tradicional é quadrática, sendo necessário comparar todos com todos, e considerando as comunidades encontradas este tempo é reduzido significativamente, comparando apenas elementos de uma mesma comunidade (Figura 9). Porém, para o método proposto há o custo computacional para identificação de comunidades. O método utilizado, baseado na remoção de *betweenness* [Girvan and Newman 2002], possui custo computacional quadrático no número de vértices, mas há outros métodos, que serão implementados e possuem custo computacional reduzido, por exemplo o método *FastGreedy* [Girvan and Newman 2002] com custo computacional $O(Nlog^2N)$, com N sendo o número de vértices.

Por fim, analisando os resultados obtidos, a capacidade de identificar elementos duplicados na base de dados está relacionada ao erro dominante na base, ou seja, se a base contém erros de digitação no qual geralmente se caracteriza por troca ou falta de poucos caracteres na referência ao autor, o limiar de diferença deve ser pequena. Combinando bases de dados, geralmente, o erro dominante está relacionado a modelos de citação diferentes, então o limiar de diferença deve ser maior.

5. Conclusões

A utilização de detecção de comunidades em redes de cooperação científica para reduzir a complexidade computacional se mostrou útil no estudo de caso realizado. O aumento no número de comunidades reduz o tempo necessário para identificar autores duplicados, porém, aumentar de maneira demasiada diminui a precisão na detecção de duplicados.

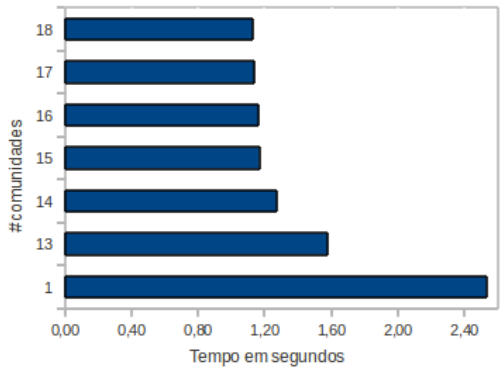


Figura 9. Tempo médio em segundos de execução dos métodos para identificar os autores duplicados na base de dados.

Assim sendo, as redes de cooperação científica tendem a agrupar autores da mesma linha de pesquisa na mesma comunidade. Então, os registros duplicados acompanham essa tendência. Aplicar a medida de similaridade nas comunidades se mostrou eficiente comparado ao método tradicional que compara todos pares de objetos, pois, mesmo conseguindo uma cobertura menor, a precisão é alta nos registros identificados.

As principais contribuições deste trabalho foram a abertura para uma nova abordagem ainda não explorada, até onde foi pesquisado em trabalhos relacionados, de uso de redes de cooperação científica e desenvolvimento do método de identificação de comunidades na detecção de duplicados. E, por fim, a aplicação da técnica de detecção de duplicados baseada nas comunidades, embora mantenha a complexidade computacional quadrática, ela passa a ser quadrática no número médio de elementos das comunidades.

Como proposta de trabalhos futuros, primeiramente torna-se necessário uma avaliação considerando uma grande quantidade de conjunto de dados. A utilização de outros modelos de detecção de comunidades [Clauset et al. 2004], principalmente para redução da complexidade do problema, permitindo escalabilidade do algoritmo. Além disso, verificou-se que é possível realizar uma comparação mais eficiente entre os pares de autores de uma mesma comunidade, considerando não somente a medida de similaridade utilizada, mas também comparando, por exemplo, as abreviações presentes no nome de um autor com os nomes completos de outro autor.

Além disso, as redes de cooperação científica foram construídas baseadas no relacionamento entre autores em trabalhos científicos. Outros modelos de rede podem ser utilizados, como modelos que consideram similaridade entre tópicos abordados nos artigos. Por fim, pode-se considerar aplicar processamento pararelo entre as comunidade, pois não existe relacionamento direto durante o processamento dos duplicados entre comunidades.

Referências

Bhattacharya, I. and Getoor, L. (2004). Iterative record linkage for cleaning and integration. In *DMKD '04: Proceedings of the 9th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 11 – 18, New York, NY, USA. ACM.

Bhattacharya, I. and Getoor, L. (2007). Collective entity resolution in relational data. *ACM Trans. Knowl. Discov. Data*, 1:5.

Bilenko, M. and Mooney, R. J. (2003). Adaptive duplicate detection using learnable string similarity measures. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '03, pages 39–48, New York, NY, USA. ACM.

Clauset, A., Newman, M., and Moore, C. (2004). Finding community structure in very large networks. *Physical Review E*, 70(1):066111.

de Carvalho, M. G., Gonçalves, M. A., Laender, A. H. F., and da Silva, A. S. (2006). Learning to deduplicate. In *JCDL '06: Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries*, pages 41–50, New York, NY, USA. ACM.

Dong, X., Halevy, A., and Madhavan (2005). Reference reconciliation in complex information spaces. In *SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 85–96, New York, NY, US. ACM.

Elmacioglu, E., Kan, M.-Y., Lee, D., and Zhang, Y. (2007). Web based linkage. In *WIDM '07: Proceedings of the 9th annual ACM international workshop on Web information and data management*, pages 121–128, New York, NY, USA. ACM.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). The kdd process for extracting useful knowledge from volumes of data. *Communication of the ACM*, 39(11):27–34.

Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *PNAS*, 99(12):7821–7826.

Kim, H.-s. and Lee, D. (2007). Parallel linkage. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, CIKM '07, pages 283–292, New York, NY, USA. ACM.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, 45:167–256.

Newman, M. E. J. (2004). Detecting community structure in networks. *The European Physical Journal B - Condensed Matter and Complex Systems*, 38:321–330.

Paskalev, P. and Antonov, A. (2006). Intelligent application for duplication detection.

Paskalev, P. and Antonov, A. (2007). Increasing the performance of an application for duplication detection. In *CompSysTech '07: Proceedings of the 2007 international conference on Computer systems and technologies*, pages 1–8, New York, NY, USA. ACM.

Raedt, L. D. (2008). *Logical and Relational Learning: From ILP to MRDM (Cognitive Technologies)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Reka Albert, H. J. and Barabasi, A. L. (1999). Diameter of the world-wide web. *Nature*, 401:130–131.

Rossi, R. G., Rezende, S. O., and de Andrade Lopes, A. (2010). Sistema para extração de informações de artigos científicos - iesystem. Technical Report 354, ICMC, São Carlos - SP.

Witten, I. H. and Frank, E. (2005). *Data Mining Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco, CA.

PNBA*: A Parallel Bidirectional Heuristic Search Algorithm

Luis Henrique Oliveira Rios[1] **, Luiz Chaimowicz**[1]

[1]Departamento de Ciência da Computação
Universidade Federal de Minas Gerais (UFMG) – Belo Horizonte, MG – Brasil

{lhrios,chaimo}@dcc.ufmg.br

Abstract. *A* (A-star) is a heuristic search algorithm used in various domains, such as robotics, digital games, DNA alignment, among others. In spite of its large use, A* can be a computationally expensive depending on the characteristics of the state space and heuristics used. Aiming at improving its performance, in this paper we propose a parallel implementation of a bidirectional version of A*. Named PNBA* (Parallel New Bidirectional A*) the proposed algorithm combines the benefits of bidirectional search and parallel execution in the development of an efficient A* based search algorithm. Experiments performed in different domains show that PNBA* is more efficient than the original A* and NBA*, the bidirectional version of A* it is based on.*

1. Introduction

Several problems in artificial intelligence can be solved by searching algorithms. In general, these algorithms explore large state spaces looking for a sequence of steps that will lead from an initial state to a goal state. One of the most studied algorithms in this area is A* [Hart et al. 1968], an *informed search algorithm* that assumes the existence of some problem-specific knowledge beyond the problem definition. This knowledge is represented by a heuristic function that guides the state space exploration. If this heuristic function respects some conditions, A* is proved to find the lowest cost solution.

By using a heuristic function, A* significantly reduces the number of states visited to find a solution without the sacrifice of the solution optimality. In spite of that, depending on the characteristics of the problem and heuristics used, its cost can still be prohibitive: in the worst case scenario, the number of states visited can be exponential in the length of the optimal solution. As discussed in [Rios and Chaimowicz 2010], several extensions and variations have been proposed to modify A*, improving its performance in different application scenarios.

Some of these extensions try to adapt A* for parallel execution. This possibility is gaining attention as multicore machines and computing clusters become more accessible. The main challenge faced by these implementations is how to reduce contention, *i.e.*, how to allow large periods of distributed computation without synchronization, still keeping the algorithm correctness.

The transformation of A* into a bidirectional heuristic search is another possible way to improve performance. On bidirectional search, the state space is simultaneously explored by two search process: one beginning at the start node and moving forward and the other from the goal and exploring the states backwards. A solution is found when the two exploration frontiers meet. The main motivation for the use of A* in a bidirectional setting is the possible reduction of the number of expanded nodes. In fact,

recent results show that the combination of A* algorithm with the bidirectional search is able to significantly reduce the search effort [Whangbo 2007, Klunder and Post 2006, Pijls and Post 2009b].

In this paper, we present PNBA* - *Parallel New Bidirectional A**, a parallel shared memory version of NBA* [Pijls and Post 2009b] - a recently proposed bidirectional search algorithm based on A*. Our main objective is to combine the benefits of bidirectional search and parallel execution in the implementation of an efficient A* search algorithm. In the paper, we discuss the parallel implementation of the algorithm and present some arguments regarding its admissibility. We perform a series of experiments in three different scenarios (fifteen puzzle and grid pathfinding on random generated mazes - with uniform and non-uniform costs) and show that PNBA* is more efficient than the original A* and also NBA*.

This paper is organized as follows: next section brings some background on heuristic search, reviewing some key properties of A* and presenting some related work on its parallel and bidirectional implementations. Section 3 discusses NBA*, the bidirectional implementation of A* proposed by [Pijls and Post 2009b]. In Section 4 we present our new algorithm, PNBA*, discussing its implementation and main properties. The experiments performed with the new algorithm and the obtained results are described in Section 5. Finally, Section 6 brings the conclusion and possibilities for future work.

2. Background

A search problem is generally defined by the following basic components [Russell and Norvig 2003, Nilsson 1998]: an initial state, a goal state, a set of operators that modifies/transforms the states (successor function) and an optional step cost function that provides the cost of producing one state from another by applying the successor function. The initial state and the successor function combined implicitly define the state space [Korf 1996]. Its size (number of states) is drastically affected by the problem characteristics. A problem instance is the set of operators together with the initial and the goal states.

The process of searching consists of, from the start state, systematically applying the operators until reaching a goal state. That is, the intention is to transform the initial state into the goal state by employing the available rules. Normally, we are interested on the sequence of steps necessary to reach the goal state or, alternatively, to check if it is possible or not to reach a goal state. The step cost function provides a metric for the solution. Therefore, it is possible to impose restrictions over the solution and to evaluate its quality. For example, the objective may be finding the less expensive sequence of steps.

It is suitable to view the state space as a graph, in which the states are nodes and an edge from state x to state y indicates that the application of a specific operator on x generates y (we say that y is a successor of x). Edges may be labeled with the cost of this operation. Therefore, finding an optimal solution (the lowest cost one) for a search problem is equivalent to compute the shortest path between the start node and the goal node - from now on, denoted as x_{start} and x_{goal} respectively.

2.1. A*

A* is one of the most well known search techniques in Artificial Intelligence. It is a best first heuristic search algorithm, so it uses a heuristic function to guide the node expansion. At each iteration, it selects the most promising node according to an evaluation function f, that includes the real cost of going to that node and an estimate of the cost from that node to the goal.

More formally, let $d(x, y)$ denote the weight/cost associated with edge (x, y). The cost of the shortest path from x to y (in this case, not necessarily two neighboring nodes) will be represented by $d^*(x, y)$. As mentioned, informed search algorithms uses a heuristic function that provides more information to the search process. For a node x, $h(x)$ expresses this heuristic, which is an estimate of $h^*(x) = d^*(x, x_{\text{goal}})$. In most cases, $h^*(x)$ is not available in advance. Function $g(x)$ (an estimate of $g^*(x) = d^*(x_{\text{start}}, x)$) is the cost of the shortest path found so far from x_{start} to x. It is not necessarily equal to $g^*(x)$ because more than one path from x_{start} to x may exist and the algorithm might not have considered all of them at that time. The cost of the shortest path that connects x_{start} to x_{goal} and goes through x is represented by $f^*(x) = g^*(x) + h^*(x)$. The function $f(x) = g(x) + h(x)$ is an estimate of this cost and is used by A* to decide which node should be expanded next.

In the context of A*, the heuristic function may have two important properties. It is said to be *admissible* if $h(x) \leq h^*(x)$ for all nodes x, *i.e.*, the heuristic function never overestimates the real cost. Also, the heuristic function is said to be *consistent* or *monotone* if $h(x) \leq d^*(x, y) + h(y)$ for any nodes x, y (or, alternatively, $h(x) \leq d(x, y) + h(y)$ for any edge (x, y)). This can be viewed as a kind of triangle inequality: each side of a triangle can not be larger than the sum of the two other. Every consistent heuristic function is also admissible [Russell and Norvig 2003]. The importance of this distinction will be explained below.

A* is complete, *i.e*, it always terminates with a solution wherever one exists, if the following conditions are satisfied [Pearl 1984]: (i) nodes must have a finite number of successors and (ii) the weight associated with the edges must be positive. Also, it is admissible (returns the lowest cost solution) if one more condition is satisfied: heuristic admissibility. Imposing a more strict restriction (consistency) on the heuristic function allows a simpler and more efficient version of A*, in which a node needs to be expanded at most once. From now on, the heuristic function is assumed to be consistent.

Two data structures are generally used during the search to manage node expansions: *open* and *closed* lists. The *open* list contains all nodes that are in the frontier of the search (candidates to be expanded). The *closed* list stores the nodes that have already been expanded. The *open* list helps the algorithm during the selection of the most promising node to expand and is commonly implemented as binary heap. The *closed* list structure is used by A* to avoid the expansion of the same state multiple times and is frequently represented as a hash table.

Before starting the node expansion loop, A* initializes $g(x_{\text{start}})$ with 0 and then inserts this node in the *open* list. The loop is executed while there are any nodes in this list and x_{goal} has not been removed from it yet. At each iteration, the node with the lowest f-value is removed from *open* list for expansion. Its successors are generated and those

that have not been expanded yet (are not in *closed* list) are inserted in the *open* list. It is possible that one or more of these nodes are already in the *open* list. In this case, a different path from x_{start} to this node has been found. If the new cost is smaller, the g-value of this node will be updated. Finally, the expanded node is inserted into the *closed* list.

There are basically two termination conditions for the algorithm. If there are no more nodes to be expanded (the *open* list is empty), there is no solution for the problem. On the other hand, if the goal state is selected for expansion from the *open* list, a solution has been found (an optimal solution, considering that the heuristic is admissible or consistent). A common way to retrieve the nodes that compose the final path is to keep a pointer from each node to its parent - the node that has been expanded to achieve it with the lowest cost. Hence, A* normally maintains an explicit search tree with all the generated nodes, a (hopefully very small) subset of the search space.

One of the most important consequences of the adoption of a consistent heuristic is that a node x is only expanded if the lowest cost path from x_{start} to x has already been found [Nilsson 1998]. This guarantees the second termination condition just mentioned. Another important consequence is that the sequence of f-values expanded is monotonically non-decreasing.

As mentioned, the main drawback of A* is that the number of nodes expanded can be exponential in the length of the optimal path. More formally, let b denote the node branching factor - the average number of successors over the search tree. Note that b is deeply related with the problem's successor function. Let d be the length - number of operators/edges employed in the transformation of x_{start} into x_{goal} - of the optimal solution. The worst case time and space complexity is $O(b^d)$ [Korf 1985].

2.2. Parallel A*

There are several reports on the adoption of parallel heuristic search algorithms based on A* to solve search problems. Grama and Kumar [Grama and Kumar 1993] provide a survey of these algorithms and presents a discussion of the challenges related with A* parallelization with feasible solutions for them. It is possible to classify these methods according to the memory architecture they assume available: shared memory or distributed memory, but most algorithms were designed for distributed memory architectures.

One example is Parallel Local A* (PLA*) [Dutt and Mahapatra 1993]. Basically, each processor has its local *open* and *closed* lists and the processors interact to inform the best solution found, to redistribute the work, to send or receive cost updates and to detect the algorithm termination. A new parallel start up phase and distribution strategy are proposed. The former has shorter execution time compared with previous works. The latter combines anticipatory work request, quantitative and qualitative load balance with a duplicate pruning mechanism to improve scalability. Different versions of this algorithm have been evaluated on a distributed memory environment to solve the Traveling Salesman Problem (TSP), showing a notable speed up in an environment with 256 processors.

Another A* based algorithm [Kishimoto et al. 2009] exploits distributed memory computing clusters using a hash function to distribute and schedule work among processors. Named Hash Distributed A* (HDA*), this algorithm maintains separated *open*/*closed* lists for each processor. The communication is performed through a message

exchange protocol. Each processor executes the following steps while the computation has not finished. It process the received states using the *closed* list to avoid duplications and the *open* list to store the nodes it will expand. If there are no more nodes to be processed, it selects the node with the lowest f-value from its *open* list, expands and sends asynchronously those that do not belong to it to the proper processors. When a goal is found, a message is broadcast and the computation continues until all processors prove that is not possible to improve the solution quality. Although the authors concentrated the effort on parallelization for distribute memory architectures, an evaluation on a single, multicore machine has also shown good results.

Parallel Best-NBlock-First (PBNF) algorithm [Burns et al. 2009] is an example of a competitive shared memory parallel heuristic search algorithm. Its implementation employs a technique called Parallel Structured Duplicate Detection (PSDD) to deal with the synchronization. The idea is to create an abstract function (specific for each kind of problem) that maps several nodes of the original graph to a unique node of the abstract graph (denoted as nblock). PBNF maintains an *open* list and a *closed* list for each nblock. The abstract graph is used to help the selection of nblocks that can be explored in parallel without synchronization. As there is no restriction in the order nodes are explored, it will continue the search while there are nodes with f-values smaller than the cost of the current solution.

2.3. Bidirectional A*

As mentioned, bidirectional search executes two searches simultaneously: one forward from the initial state and the other backward from the goal [Russell and Norvig 2003].

Some of the early works to deal with bidirectional heuristic search did not show good results. [Pohl 1971], for example, argued that bidirectional heuristic search was not faster than its unidirectional version. A kind of consensus that the search frontiers pass each other (without meeting in the middle) has been established. In these cases, the number of expansions and the time of execution would be larger if compared with same values of unidirectional versions. A metaphor of two missiles separately directed at each other base desiring that they collide has been created to illustrate this problem [Pohl 1969]. The belief on this motivated the creation of front-to-front algorithms [Politowski and Pohl 1984, Davis et al. 1984] where each search aims at the opposite search tree frontier instead of its root. However, this approach did not solve the problem, resulting in computationally expensive algorithms or in algorithms that did not guarantee solution quality, as discussed in [Kaindl and Kainz 1997].

The conjecture of searching frontiers passing each other has been shown to be wrong [Kaindl and Kainz 1997] because the major computational effort is done in the post processing phase, after the two search frontiers meet. The authors have also presented evidences that bidirectional heuristic search is more appropriate for solving certain problems than the corresponding unidirectional search versions. These results have revived the interest on bidirectional heuristic.

More recently, bidirectional heuristic search algorithms that execute faster than their corresponding unidirectional version have been proposed. Whangbo presents an efficient modified bidirectional A* [Whangbo 2007] that improves previous versions in two aspects: by avoiding repetitive searches and through the use of a new stop

condition that does preserve the admissibility. Other relevant work [Ikeda et al. 1994] shows ways of combining A* with bidirectional version of Dijkstra's algorithm without losing the guarantee of finding the optimal solution. This result has been employed with some improvements in an empirical evaluation of various shortest path algorithms [Klunder and Post 2006]. It concludes that bidirectional heuristic search is the most efficient alternative in the context of large real road networks.

One of the latest bidirectional heuristic search algorithms is NBA*, proposed in 2009 [Pijls and Post 2009b]. This algorithm is the base for our parallel implementation and will be discussed in the next section.

3. New Bidirectional A* (NBA*)

New Bidirectional A* (NBA*) [Pijls and Post 2009b] is a bidirectional heuristic search algorithm based on A*. Unlike the methods that inspired the creation of a preliminary version [Pijls and Post 2009a], NBA* just imposes one constraint on the heuristic function: consistency.

As any bidirectional search algorithm, two search process are managed simultaneously but not necessary in parallel. A common technique to accomplish the simultaneous execution in this context is to alternate the execution of each search process. This approach has been employed by NBA* authors. The search process number 1 operates in the original graph and process number 2 handles the reverse graph where every original edge (x, y) is replaced by an edge (y, x) with the same weight. Moreover, on the second process, the original start node (x_{start}) becomes the goal node and the original goal (x_{goal}) changes into start node. We will use the notation s_1, t_1 and s_2, t_2 for the start and goal nodes of each search process.

Before we start explaining the NBA* algorithm, the notation that has been employed so far should be enhanced. Let $d_p(x, y)$ and $d_p^*(x, y)$ denote, respectively, the same as $d(x, y)$ and $d^*(x, y)$ but for search process $p \in \{1, 2\}$. The subscript will also be added on the other elements resulting in $g_p(x)$, $h_p(x)$ and $f_p(x) = g_p(x)+h_p(x)$ estimates of $g_p^*(x) = d_p^*(s_p, x)$, $h_p^*(x) = d_p^*(x, t_p)$ and $f_p^*(x) = g_p^*(x) + h_p^*(x)$ respectively. Hence, the semantic is equivalent but now it is restricted to the search process expressed by p.

Like A*, NBA* uses a structure to keep control of node expansions. Using the original notation, $\mathcal{M}$ contains the nodes that are in the "middle", *i.e.*, between the two searches. Initially, all nodes are inside it. The nodes in the search frontiers are those that belongs to $\mathcal{M}$ and have been *labeled*. A node x is *labeled* if $g_1(x)$ or $g_2(x)$ is finite and the node has not been expanded. Besides $\mathcal{M}$, $\mathcal{L}$ is a shared variable that is read and written by both search process. It contains the cost of the best solution found by the algorithm so far and is initialized with an infinity value ($\mathcal{L} = \infty$). $\mathcal{L}$ is employed in the pruning criteria together with F_p, the lowest f_p-value on the frontier of search process p. Variables f_p, g_p and F_p are only written by one side but read on both.

The operations executed by each process will now be detailed. The search process number 1 has been chosen as reference but the operations are analogous for the other side. Initially, the g_1-value of all graph nodes is set to ∞. Then, the algorithm makes $g_1(s_1) = 0$ and $F_1 = f_1(s_1)$. At each iteration, the node $x \in \mathcal{M}$ with the smallest finite f_1-value is selected for expansion. It is removed from $\mathcal{M}$ and pruned (not expanded) if

$f_1(x) \geq \mathcal{L}$ [1] or $g_1(x) + F_2 - h_2(x) \geq \mathcal{L}$. Otherwise, all its successors y are generated. In the first case, it is classified as *rejected* and in the other situation as *stabilized* because $g_1(x)$ will not be changed anymore. For each y, $g_1(y)$ and $\mathcal{L}$ are updated, respectively, by the following statements values: $\min(g_1(y), g_1(x) + d_1(x, y))$ and $\min(\mathcal{L}, g_1(y) + g_2(y))$. At the end of the iteration, F_1 is updated with the lowest f_1-value inside the frontier. The execution stops when there are no more candidates to be expanded in one of the search sides. At this moment, $\mathcal{L}$ holds the cost of the optimal solution or an infinity value if no solution exits.

4. Parallel New Bidirectional A* (PNBA*)

The Parallel New Bidirectional A* (PNBA*), proposed in this work, is a parallel implementation of NBA*. It is similar to NBA* but tailored for parallel execution on shared memory architectures. The main idea is to run both search process in parallel instead of alternating their execution. As the majority of the variables involved in the algorithm are written by only one side of the search, the necessity of mutual exclusion sections is small, which tends to yield a good performance.

The code executed by one of the search processes is shown in Algorithm 1. The other process is analogous and can be obtained by properly replacing the subscripts (from 1 to 2 and vice-versa). The algorithm is very similar to the version presented on NBA* paper [Pijls and Post 2009b] but is more focused on implementation. We opt to explicitly represent the priority queue, denoting it by $open_p$ and adopting f_p as the sort criterion. Each side has its own priority queue that is able to execute the following operations: *insert*, *peek*, *pop*, *remove* and *size*. The variable *finished* indicates when the computation must stop and is common for both sides. The structure $\mathcal{M}$ is also shared and need to cope with concurrent access. Some initializations are made before each process start: on each side, $g_p(s_p)$ is set to 0 and s_p is inserted into $open_p$. Therefore, at the beginning, all nodes are on $\mathcal{M}$ and have their g_p-values equal ∞, except for s_1 and s_2 as $g_1(s_1) = 0$ and $g_2(s_2) = 0$.

Besides the inclusion of the mutual exclusion sections, one important change was made in the original algorithm: line 17, originally placed after line 3, was moved to its current position. This line is responsible for removing node x from $\mathcal{M}$. Without this change, PNBA* could possibly ignore some nodes depending on the execution sequence of some parallel statements. For example, suppose there is a graph with only two nodes a and b that are the start and goal respectively. Considering that there is an edge connecting a to b and line 17 is just after line 3 (like in NBA*), search process 1 removes a from its *open* list, then from $\mathcal{M}$ and the execution of this process is suspended. Search process 2 does the same steps with b, expands b generating a but is not able to update $g_2(a)$ and $\mathcal{L}$ as $a \notin \mathcal{M}$. Search process 1 continues its execution, expands a generating b but is not able to refresh $g_1(b)$ and $\mathcal{L}$ as $b \notin \mathcal{M}$ as well. The displacement of statement $\mathcal{M} \leftarrow \mathcal{M} - \{x\}$ is sufficient to overcome this problem with the drawback of possible unnecessary expansions that do not violate the solution optimality as will be discussed below. Another difference in the algorithm regards the update of $\mathcal{L}$ (lines 12-16). First, a verification is done outside the mutual exclusion section. Then, if there is a chance of a

[1] The original condition should be $f_1(x) - h_1(t_1) \geq \mathcal{L}$. However, since in most heuristics (including those used in this work) $h_1(t_1) = h_2(t_2) = 0$, this term was simplified to $f_1(x) \geq \mathcal{L}$.

better value to $\mathcal{L}$, the verification and update are done inside the mutual exclusion section to ensure the operation will be atomic.

Variables F_p, $\mathcal{L}$ and $\mathcal{M}$ are checked by the algorithm to decide if parts of the search tree may be pruned. As they are updated by both processes, their values may be outdated at the time they are read. In spite of that, we can argue that this does not affect PNBA* admissibility. It is important to notice that F_p and $\mathcal{L}$ are respectively monotonically non-decreasing and monotonically non-increasing. The former is guaranteed by the characteristics of the heuristic that is also monotonic and the latter by the lines that update $\mathcal{L}$ (lines 14-15). The g_p-values are also monotonically non-increasing (lines 6-7). All these conditions assure that the algorithm will not prune a node that should not be pruned, guaranteeing admissibility. It is worth mentioning that the opposite can happen: a node that should be pruned will be not. For example, a node that should not be expanded (because it is not more on $\mathcal{M}$) might be expanded due to the use of outdated information on line 3. But this will not affect the solution optimality as the algorithm verifies if the new values are smaller than the old values before updating $\mathcal{L}$ or g_p-values. It will only cause unnecessary computation that is compensated by the parallel execution, as shown in our experimental evaluation.

The g_p-values of a node can also be outdated on $\mathcal{L}$ actualization. However, after a write on $g_p(x)$ there is always an update on $\mathcal{L}$. Therefore, if the $g_1(x)$ change is followed by an update of $\mathcal{L}$ using $g_2(x)$ that is just about to be refreshed (with a lower value), search process number 2 will store the proper value on $\mathcal{L}$ because it will have the newest values of $g_1(x)$ and $g_2(x)$ available.

Algorithm 1 PNBA*, Parallel New Bidirectional A*

```
1:  while ¬finished do
2:    x ← open_1.pop()
3:    if x ∈ M then
4:      if (f_1(x) < L) ∧ (g_1(x) + F_2 − h_2(x) < L) then
5:        for all edges (x, y) of the graph being explored do
6:          if (y ∈ M) ∧ (g_1(y) > g_1(x) + d_1(x, y)) then
7:            g_1(y) ← g_1(x) + d_1(x, y)
8:            f_1(y) ← g_1(y) + h_1(y)
9:            if y ∈ open_1 then
10:             open_1.remove(y)
11:           open_1.insert(y)
12:           if g_1(y) + g_2(y) < L then
13:             lock
14:             if g_1(y) + g_2(y) < L then
15:               L ← g_1(y) + g_2(y)
16:             unlock
17:     M ← M − {x}
18:   if open_1.size() > 0 then
19:     F_1 ← f_1(open_1.peek())
20:   else
21:     finished ← true
```

5. Experiments

In this section, we present the experimental evaluation of the proposed algorithm. PNBA* is compared with A* and NBA* in three different scenarios: fifteen puzzle and grid pathfinding on random generated mazes with uniform and non-uniform costs. The former is a type of problem whose state space does not fit entirely on memory, thus it is not represented explicitly. During the computation, only the part needed is generated. The latter is assumed to fit entirely on memory and has an explicit graph associated with it. This distinction is important because it directly impacts implementation.

For both problems, a flag is kept to indicate wherever a node is on $\mathcal{M}$ (NBA* and PNBA*) or on the *closed* list (A*). As the state space is not explicit for fifteen-puzzle, a hash table is employed to guarantee that during the execution there will be an unique representation per state. For PNBA*, this structure implements atomically the operations *has* and *insertIfAbsent*. The first queries about the existence of an item and gets it while the second inserts an item if it is not there already. As the hash table implementation is based on buckets, just the affected one needs to be locked during these operations. For the other algorithms, the same implementation is used but without the overhead associated with the synchronization mechanisms demanded by PNBA*. A binary heap is used to implement the priority queue in the algorithms.

The implementations were done using the C++ programming language (g++ 4.4.3 compiler) with the *pthreads* library. The experiments were performed on an Intel(R) Core(TM) i5 3.20GHz CPU and with 4 gigabytes of memory, running Ubuntu with Linux kernel version 2.6.32. To evaluate the algorithms on fifteen-puzzle, an instance database was created. In this database, the instances are separated by the length (number of moves) of their optimal solution. The algorithms were executed on instances which solution length varied from 46 to 58 on increments of two and, for each length, we computed the arithmetic mean of 25 executions on different instances. On grid pathfinding, a series of square mazes were generated randomly. On these mazes, a cell was blocked with a probability of 33% and start and finish cells were positioned on the top left and bottom right corner cells respectively. The costs of moving between cells on non-uniform maze were randomly selected from 1 to 8. The maze size was varied from 1000 to 4000 in steps of 300. The data collected for each size was the result of 25 executions - mazes without solution are ignored until this number is achieved. The Manhattan distance was used as the heuristic on all domains. The ties (nodes with the same f-value) were broken in favor of those with the largest g-values. In all experimental scenarios the number of expanded nodes and the clock time spent were collected.

Figure 1 shows two graphs with the execution time and the number of node expansions as the fifteen-puzzle instances get harder. NBA* clearly beats A* on both criteria. It is able to expand fewer nodes because it has two search trees that together are smaller than A* search tree. NBA* pruning mechanism is important to achieve this result. The number of expansions of PNBA* and NBA* are very similar. However, PNBA* has a shorter execution time since it explores both search trees in parallel.

The results for the uniform grid pathfinding scenario can be seen on the graphs on Figure 2. The results are very similar, although PNBA* surprisingly expands a little bit less nodes than NBA* especially for large instances. A possible explanation is the non-synchronized manner through which the search space is explored. This can randomly

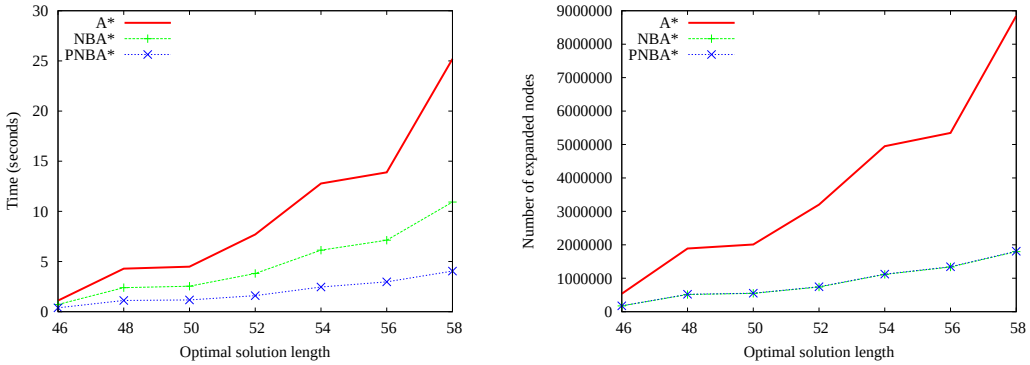


Figure 1. Results on fifteen-puzzle domain.

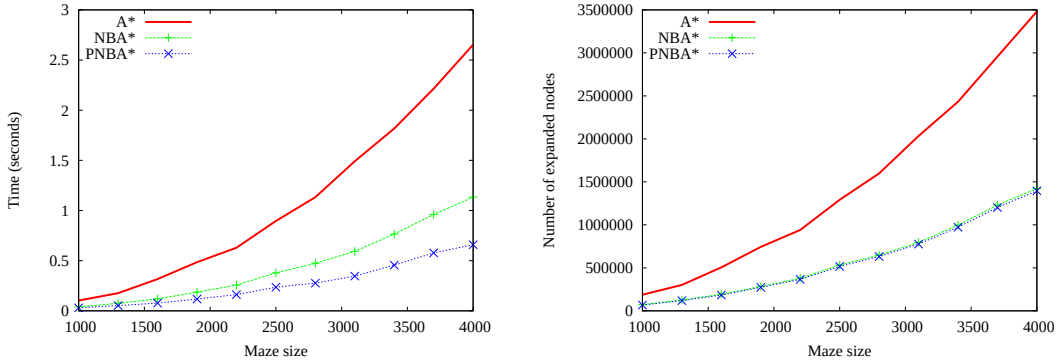


Figure 2. Results on uniform cost grid pathfinding domain.

favor the computation of a good initial first solution (very close to the optimal solution), enabling the pruning of more nodes during the rest of the computation.

Finally, the results for the non-uniform cost grid pathfinding are shown in Figure 3. The difference between figures 2 and 3 (uniform and non-uniform respectively) is remarkable. The problem with non-uniform cost is clearly harder as the heuristic estimates the lowest cost considering that all cells have the smallest weight: 1. This is done in order to preserve the admissibility property, since it is always possible that one path from start to goal passing on cells which cost is 1 exists. The consequence is that the heuristic may underestimate the real cost by a large factor, mainly when compared to the uniform cost domain. Therefore, the number of expanded nodes for the three algorithms increased, but as in the other scenarios, PNBA* and NBA* have similar values. In terms of execution time, the difference between A* and NBA* reduced significantly, but PNBA* was again the faster among the compared algorithms. Based on this experiment, it seems that bidirectional heuristic search is more affected by a poor heuristic than unidirectional versions, but a further investigation is necessary in order to evaluate this.

These results clearly demonstrate the advantages of PNBA*, the parallel implementation of NBA*. It leverages the benefits of a bidirectional heuristic search (reduction in the search space) by exploring both searches in parallel, reducing the total execution time.

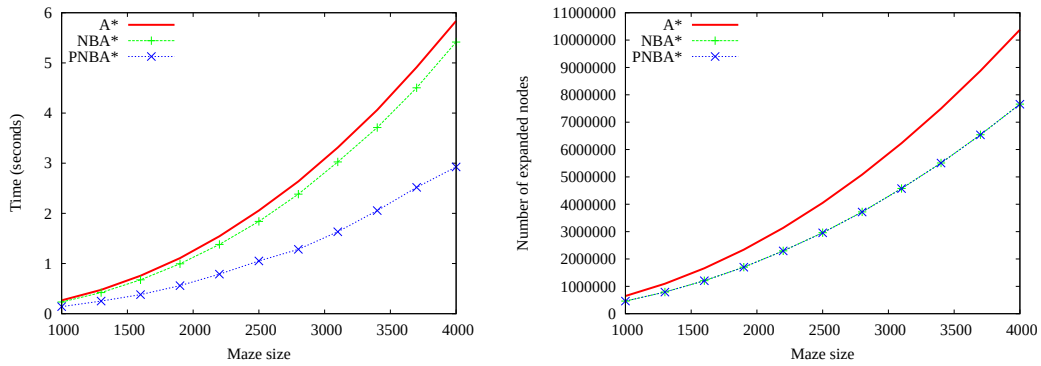


Figure 3. Results on non-uniform cost grid pathfinding domain.

6. Conclusion

This paper presented PNBA*, a parallel bidirectional heuristic search algorithm. Based on NBA*, a novel bidirectional version of A*, PNBA* improves A* performance by combining the benefits of bidirectional search and parallel execution. The algorithm was empirically evaluated in three different scenarios: fifteen puzzle and grid pathfinding on random generated mazes - with uniform and non-uniform costs and the results showed a significant reduction of execution time without the sacrifice of solution optimality. We also discussed the main characteristics of the algorithm and presented arguments that the parallel implementation does not violate its admissibility properties.

Continuing this work, we intend to extend the comparisons considering different application scenarios - maybe from the classical planning instances of the IPC (International Planning Competition) benchmark. They will also contemplate more parallel heuristic search algorithms, including bidirectional based parallelizations as [Sohn et al. 1994]. Additionally, together with a deeper analysis of these results, the development of formal proofs of PNBA* admissibility will be an important contribution. Finally, in order to better explore the current available multicore machines, we want to adapt PNBA* to include the ability of employing more than 2 two processors in its execution, allowing a greater performance improvement.

Acknowledgments

This work is partially supported by Fapemig and CNPq.

References

Burns, E., Lemons, S., Zhou, R., and Ruml, W. (2009). Best-first heuristic search for multi-core machines. In *Proceedings of IJCAI*, pages 449–455.

Davis, H. W., Pollack, R. B., and Sudkamp, T. (1984). Towards a better understanding of bidirectional search. In *AAAI*, pages 68–72.

Dutt, S. and Mahapatra, N. R. (1993). Parallel a* algorithms and their performance on hypercube multiprocessors. In *IPPS*, pages 797–803.

Grama, A. Y. and Kumar, V. (1993). A survey of parallel search algorithms for discrete optimization problems. *ORSA Journal on Computing*, 7.

Hart, P. E., Nilsson, N. J., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *Systems Science and Cybernetics, IEEE Transactions on*, 4(2):100–107.

Ikeda, T., Hsu, M.-Y., Imai, H., Nishimura, S., Shimoura, H., Hashimoto, T., Tenmoku, K., and Mitoh, K. (1994). A fast algorithm for finding better routes by ai search techniques. In *VNIS*, pages 291–296.

Kaindl, H. and Kainz, G. (1997). Bidirectional heuristic search reconsidered. *J. Artif. Intell. Res. (JAIR)*, 7:283–317.

Kishimoto, A., Fukunaga, A. S., and Botea, A. (2009). Scalable, parallel best-first search for optimal sequential planning. In *ICAPS*. AAAI.

Klunder, G. A. and Post, H. N. (2006). The shortest path problem on large-scale real-road networks. *Networks*, 48(4):182–194.

Korf, R. E. (1985). Depth-first iterative-deepening: An optimal admissible tree search. *Artificial Intelligence*, 27(1):97–109.

Korf, R. E. (1996). Artificial intelligence search algorithms. Technical report, University of California.

Nilsson, N. J. (1998). *Artificial Intelligence: A New Synthesis*. Morgan Kaufmann Publishers, San Francisco.

Pearl, J. (1984). *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley.

Pijls, W. and Post, H. (2009a). A new bidirectional search algorithm with shortened postprocessing. *European Journal of Operational Research*, 198(2):363–369.

Pijls, W. and Post, H. (2009b). Yet another bidirectional algorithm for shortest paths. Technical Report EI 2009-10, Erasmus University Rotterdam, Econometric Institute.

Pohl, I. (1969). Bi-directional and heuristic search in path problems. Technical Report 104, SLAC (Stanford Linear Accelerator Center), Stanford, California.

Pohl, I. (1971). Bi-directional search. In Meltzer, B. and Michie, D., editors, *Machine Intelligence*, volume 6, pages 124–140, Edinburgh. Edinburgh University Press.

Politowski, G. and Pohl, I. (1984). D-node retargeting in bidirectional heuristic search. In *AAAI*, pages 274–277.

Rios, L. and Chaimowicz, L. (2010). A survey and classification of a* based best-first heuristic search algorithms. In *SBIA 2010*, volume 6404, pages 253–262. Springer.

Russell, S. J. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Pearson Education.

Sohn, A., Sato, M., Sakai, S., Kodama, Y., and Yamaguchi, Y. (1994). Parallel bidirectional heuristic search on the em-4 multiprocessor. In *Proceedings of the 6th Symposium on Parallel and Distributed Processing*, pages 100–109.

Whangbo, T. K. (2007). Efficient modified bidirectional a* algorithm for optimal route-finding. In *IEA/AIE*, volume 4570, pages 344–353. Springer.

Agent motion planning with pull and push moves

Tadeu Zubaran, Marcus Ritt

Departamento de Informática Teorica, Instituto de Informática
Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil

Abstract. *Agent motion planning is a common task in artificial intelligence. One of the simplest scenarios considers the delivery of boxes to storage locations on a regular grid with obstacles. When the agent can only push boxes, we obtain the well-known, PSPACE-hard Sokoban puzzle [Culberson 1997]. In this paper we propose an exact solver for the scenario called* Pukoban *where the agent can push and pull boxes. The solver is, to the best of our knowledge, the first one proposed for Pukoban. It is based on the A* search algorithm with several problem-specific improvements. We evaluate its efficiency on* 100 *instances from the literature. Our algorithm is able to solve* 30 *instances exactly.*

1. Introduction

Pukoban is a game on an integer grid, where an agent (the warehouse keeper or robot) has to move boxes to designated storage locations. Each grid cell can contain a box, or an unmovable obstacle (or wall), which neither the agent nor a box can occupy, or a storage (or target) location, where a box must be placed to solve the game. Cells that do not contain a wall nor a box form the *free space*. The agent can push or pull a box one cell horizontally or vertically if the destination cell is free and he has enough space to do so. When the agent succeeds in placing every box at a storage locations the puzzle is solved. This simple set of rules can be deceiving as even a relatively small puzzle can be very hard to solve for both computers and humans alike, requiring that we use clever strategies to solve them. The goal of these puzzles is to study the solvability of abstract versions of typical motion planning situations, e.g., in robotics [Dor and Zwick 1999].

The reader can find an instance of a Pukoban puzzle in Figure 1. It shows one of the easiest and smallest test cases for our solver.

1.1. Other Versions of the Game

A slightly different and much more famous puzzle where the worker can only push the boxes is known as Sokoban. Schaeffer and Junghanns present a study of techniques to solve Sokoban puzzles [Schaeffer and Junghanns 2000] . When we attempt to solve a

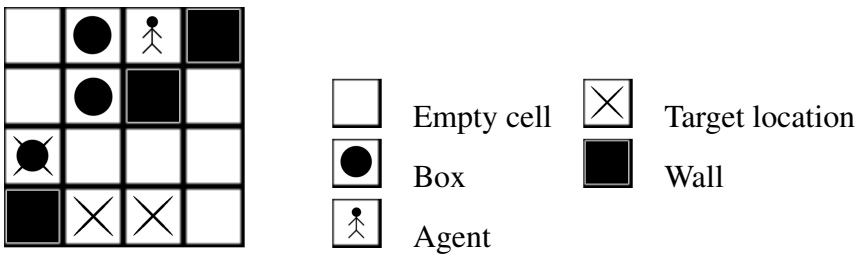


Figure 1. Example instance of Pukoban. The optimal solution needs 48 **box movements.**

Pukoban game either with a computer or by hand the successful strategies tend to be somewhat different from those successful in Sokoban, in spite of the striking similarities in their rule set.

There are many other versions of Sokoban-like games. Such versions include, for example, movable obstacles, allow the keeper to push up to k boxes, or even an unlimited number, require boxes to slide until hitting the next obstacle, or include boxes which occupy more than one cell. Most of these version have been proved to be NP-hard or PSPACE-complete [Demaine et al. 2003]. The NP-hardness of the Pukoban puzzle is, to the best of our knowledge, open.

While there is all this plethora of modifications of Sokoban there are other similar puzzles that have being explored in the literature and provided some insight on how to solve Pukoban puzzles such as Atomix [Hüffner et al. 2001].

2. Exact solution of Pukoban

Solving Pukoban is equivalent to finding the shortest path from the initial state to some solution state in the *state graph*. The vertices of this graph are the possible states of the game. Two states are joined by an arc, if there is a move transforming the first into the second one. In this work we define the number of box movements as the distance metric. One might be interested in minimising the number of movements of the worker itself, which is not considered by our approach.

2.1. Comparison with Sokoban

The rules for a Pukoban are quite similar to the Sokoban puzzle however they differ in key characteristics making the process of automatically solving a Pukoban game quite different from a Sokoban game.

First, and perhaps most importantly, a Sokoban search space graph is directed, making deadlocks possible. This is particularly important in man-made maps where most of the opening moves put the game in deadlock. Junghanns cleverly exploited this characteristic using a deadlock table that drastically reduces the branching factor – the number of possible moves in a given state, equal to the (out-)degree of the state in the state graph – of his search tree [Schaeffer and Junghanns 2000]. In Pukoban every move is reversible, and thus our search space graph is undirected and it is impossible to form a deadlock.

Solving Sokoban can be accelerated by using so-called tunnel macros. In Sokoban once the agent starts to push the box through a tunnel of width one, no target cells are in the tunnel, and there is no possibility to get to the other side of the tunnel, the agent must push the box all the way through the tunnel. In Pukoban there is no such limitation and the tunnel can conceivably be used as storage place for the box. Another key difference is that, even without considering deadlocks, the branching factor of Pukoban is usually bigger than that of Sokoban since there are more possibilities for the worker, leading to considerably larger search trees, and thus making the puzzle harder to solve.

2.2. Pukoban-specific Strategies

Our approach is based on the A* algorithm [Hart et al. 1968] applied to the state graph of the game to find a path of minimum cost from an origin state to a target state, along with

Algorithm 1 Pseudo code for the A* algorithm with *closed set*

```
open set contains only start node
closed set is empty
while open set is not empty do
    current node is the node with the cheapest total cost in open set
    remove current node from open set and put it in closed set
    if current node is target node then
        shortest path has been found: puzzle is solved
    else
        for each neighbour of current node do
            if current neighbour is not in open set nor in closed set then
                put neighbour in open set
            end if
        end for
    end if
end while
target node is unreachable from the start node
```

several improvements. A* is a modification of Dijkstra's shortest path algorithm using a heuristic which estimates the distance to the target to guide the exploration of the search space. A* has the same worst case complexity $O(m + n \log n)$ as Dijkstra's algorithm (when implemented using a Fibonacci heap), where n and m are the number of vertices and edges of the search graph. Observe that in our case the search graph has up to $\binom{rc}{b}$ vertices, for a map of dimension $r \times c$ and b boxes, which is exponential in the size of the input. Algorithm 1 shows the pseudo-code for the A* algorithm.

We call *current cost* of a node the cost to get from the start to the current node, *heuristic cost* the cost predicted by the heuristic to get from the current node to the target node, and *total cost* the sum of the current cost and the heuristic cost.

When we reach the target state, the current solution is guaranteed to be optimal for *admissible* heuristics. To be admissible the estimated distance to the target can never exceed the true distance. It is desirable that the heuristic is also *monotone*. A heuristic is monotone if for every pari of states x and y we have $h(x) \leq d(x, y) + h(y)$ where $h(x)$ is the estimated distance of state x to the target state, $d(x, y)$ is the actual distance to go from state x to state y. This means that it is impossible to decrease the *total cost* of a path between two nodes by adding another node in the path. As it will be seen in the following sections our total cost never decreases so all our heuristics are monotone. This helps A* because once A* visits a node (i.e. the node is the node with the lowest *total cost*) we can guarantee there will be no cheaper path to it so we can use a *closed set*, the set of nodes that do not need to be explored again improving the algorithm efficiency.

2.2.1. Distance Heuristic

We need a heuristic to guide the search of the A* algorithm. The quality of this heuristic usually has a significant effect on the overall performance of the algorithm. A simplistic yet computationally efficient approach is to compute the Manhattan distance of each box

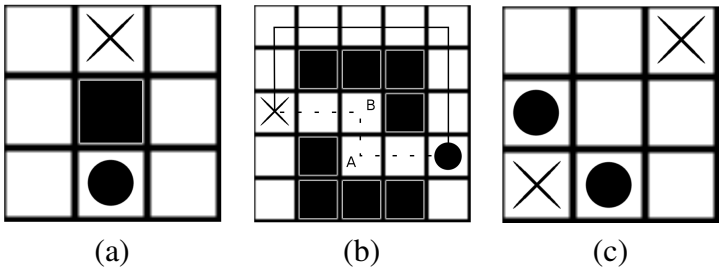


Figure 2. (a) The Manhattan distance underestimates the cost. (b) A path without enough space for the worker. (c) Multiple boxes concur for the same destination.

to the closest target cell. The sum of these values for all the boxes will give us a monotone, admissible heuristic. It is monotone because we do not decrease the total cost, if we are able to follow the path predicted by the heuristic. Otherwise the total cost will increase. The same argument applies to the following heuristics. If we compute the distance of each non-wall cell to its nearest target in a preprocessing step, we can compute the heuristic in $O(N)$, where N is the number of boxes.

We can improve the heuristic by taking into account the actual geometry of the map and use the distance to the closest target in free space. In Figure 2(a) the Manhattan distance to the target is two, yet the box can not be moved over the wall so a cost of four would be much more accurate. This technique gives us a more accurate yet still admissible and monotone heuristic. We introduce a small overhead in the preprocessing, but the cost of dynamically computing the heuristic remains $O(N)$.

Another improvement is to take into account that the box has to be either pushed or pulled by the worker so we can only take paths where there is space for the worker itself. In Figure 2(b) our previous heuristic would give us a cost of five for moving the box along the dashed line. But this is an impossible route because there is no space for the worker to move the box from A to B. Taking this into account we obtain a heuristic cost of nine for the path indicated by a continuous line. Similar to our previous improvement this introduces only a small overhead in the preprocessing but has no effect in the dynamic computing of the heuristic cost. We call this the *shortest distance* heuristic. Notice that it remains admissible and monotone, making it strictly better than the previous two.

We can further improve the heuristic taking into account that each target can be occupied by at most one box. In Figure 2(c) our current heuristic is two, but obviously we need at least four moves to bring both boxes to target positions. Thus the heuristic can be improved by considering only matchings of boxes and target positions. We can not allow an arbitrary matching since our heuristic can never overestimate the cost to guarantee its admissibility. Therefore we compute the distance of each of the non-wall cells to each of the targets. With that information we can compute the minimum matching of the distances of the boxes to the targets. The sum of the costs of this matching can be safely used as our heuristic keeping it monotone and admissible. We call this the *matching heuristic*.

A minimum matching in a bipartite graph (in our case the graph consisting of current and target positions) can be found in polynomial time using, for example, the Hungarian (or Kuhn-Munkres) algorithm [Kuhn 1955]. Like our previous improvements this introduces a small overhead in the preprocessing since we now have to compute the distance of

each cell to each of the targets, but unlike our other improvements this also makes the dynamic computing of the heuristic more expensive because the minimum matching has to be calculated for each computed node of the search space.

2.2.2. Inertia

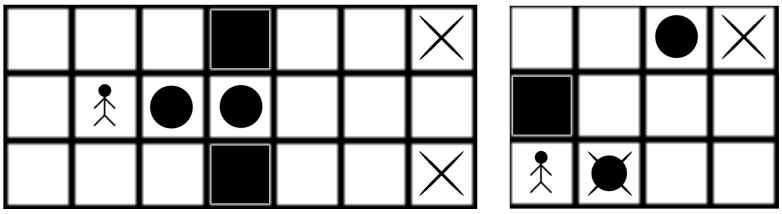
When a human solves an open map (either in a Sokoban or a Pukoban game) it is normal to see the same box being moved several times in a row. We use this characteristic to break ties between states of the same total cost, giving preference to states that keep the agent moving the same box. This was discussed by [Schaeffer and Junghanns 2000] when solving Sokoban puzzles and translates quite well to our case.

2.2.3. Choke Points

Several of the Sokoban specific maps have a room (a connected region of cells) where all the targets are [Schaeffer and Junghanns 2000]. In such a situation further optimisations are possible. We define a *choke point* as a cells that separates targets from boxes, i.e., when a choke point is blocked, no box can reach any target. If we find multiple choke points we choose the one that minimises the size of the room containing the targets. We call the choke point and all the cells beyond this point in the direction of the targets *inside* the room and the other cells *outside*. Choke points can be found in a preprocessing step by applying a direct test of the definition above. The following improvements are only possible when we find a choke point.

Prefer Box Closest To Target An improvement based on the strategy that humans tend to use when solving open maps is to attempt to move boxes that are already near a target first. When there are nodes tied in cost, we prefer to explore first those at positions inside the target room. If there is no such box accessible, we prefer the box closest to the choke point. Both inertia and this improvement usually point to the same node to be explored first. When they differ the box closest to the target has priority over inertia, since this may help to remove obstacles for the later boxes.

Dynamic Computation of Minimum Matchings If we have a choke point sometimes the minimum matching can be updated using the matching of the previous state. If we move a box that was outside the target room at position c in state s to position c' in a state s' that still is outside, then the new heuristic is given by $h(s') = h(s) + d(c') - d(c)$, where $h(x)$ is the heuristic cost of the state x and $d(y)$ is the distance (as defined when we calculate the heuristic) of the position y to the choke point. To show this we point out that the heuristic cost of any box outside with regard to any of the targets $t \in T$ is given by $d_t(a) = d_c(a) + d_t(c)$, where $d_x(y)$ is the heuristic distance from y to x and T the set of all targets. By hypothesis the moved box was outside, so only $d_c(a)$ changes, meaning the distance of the box to all the targets changes by the same amount, so the minimum matching of the previous state is also the minimum matching of the new state, but with a cost increased by $d(c') - d(c)$. Since most of the search is performed moving outside boxes and we can compute the heuristic in constant time in such cases, this improves the performance considerably.



(a) (b)

Figure 3. (a) Example of a clog: The worker has to move a box away from the targets to be able to solve the puzzle. (b) Example of an ineffective clog: The agent can not move when the condition of the clog is satisfied.

Clogs The state graph of a Pukoban game is bidirectional, therefore we do not have deadlocks but during the process of solving the game sometimes we form some structures that cannot be undone without increasing the total cost. We call these structures *clogs*. Clogs can be formed by two or more boxes. Figure 3(a) gives an example of a two-box clog.

Most clogs depend on the position of the worker as well as the position of the boxes that form the clog. In the example above we would not have a clog (meaning we would not underestimate the heuristic with certainty) if the worker was inside the room with the targets. A clog is completely defined by the position of the boxes that form the clog and a set of positions where the worker can be. We can calculate clogs in the preprocessing stage by attempting to solve the Pukoban puzzle with a limited number of boxes without increasing the heuristic.

Observe that since the matching of a subset of the boxes to the targets in the optimal solution is unknown, we can not just compute the minimal matching of the boxes to the targets to decide if they form a clog. However, in the common situation that there exists a choke point, and the candidate boxes are outside the target room, the matching of boxes to targets is irrelevant, and we can safely identify clogs. We further limit the preprocessing to computing clogs with a small number of boxes since computing all clogs with the any number of boxes would mean solving the original Pukoban problem (along with several others), rendering this process useless.

We can increase the heuristic cost when we detect a clog in some state. We choose a conservative approach of increasing the heuristic by the minimal possible increase of two, if we find at least one clog. The process of using clogs to improve the heuristic introduces an overhead to find the clogs when preprocessing and during the execution of the A*, because we need to test if any clogs are present at each new state. This is true even if we use the choke point optimisation where we can use the previous heuristic to compute the new heuristic faster since any move of a box may form or destroy a clog.

We chose to look for two-box clogs only. This is already a costly preprocessing and usually yields between ten and 500 clogs. We further trim the types of clogs we will be looking for by eliminating those that either limit the position of the worker too much (i.e. the worker can not move more than a few cells) or consist of very distant boxes, since such patterns are very uncommon. Figure 3(b) shows an example of the former situation. Since the agent can not move if we satisfy the condition of the clog, we can remove it without increasing the cost of the search.

3. Notes on the Implementation

A small memory footprint of the nodes is important for a good performance of the A* search. We store the static geometry of the maze (i.e. position of walls, targets and free spaces) only once in a two-dimensional array. The worker's position is stored as a pair of 16-bit integers, while the position of the boxes and the targets is stored as a set of such pairs. The program also computes and stores the cells reachable by the agent without moving a box in a bitmap. Using this representation a node in the search space occupies only $2(n+1) + rc/8$ bytes, for n boxes and a map of size $r \times c$.

If the map contains a choke point, we store another bitmap indicating the cells inside the target room and position of the choke point. The clogs found during preprocessing are stored in a list. Each element of the list contains the set of the boxes forming the clog and a bitmap of the agent's positions that make the clog effective. If we use the heuristic without the minimum matching we store the distance of each position to the closest target in a two-dimensional array; if we use the minimum matching then we store a three-dimensional array with the distance between every cell and every target.

The open and the closed set are two key data structures of A* that must be implemented efficiently (see Algorithm 1). The closed set has to support addition of a new element and a test if some element is already contained. We chose a hash table to represent the open set, since it can execute both operations in amortised constant time provided we have a collision-free hash function. Due to the large number of states, computing such a function can be very expensive so we made a compromise between the number of hash conflicts and the cost to compute the hash function. The function is given by $H(s) = \sum_{1 \leq i \leq n} rn2^i c_i + ir_i$ for n boxes at positions (r_i, c_i), $1 \leq i \leq n$. The open set can be implemented as a priority queue, which supports the removal of the element with the lowest total cost and the update of the distance of an element already contained in the queue, when we have found a shorter path to the node in question. Our implementation combines the priority queue with an additional hash table to efficiently find the states already stored in the queue. This way an update has cost logarithmic in the number of open states, the extraction of the smallest element has constant amortized time.

4. Experimental Results

We have implemented our algorithm in C++. (A preliminary version of the solver implemented by our group is available at [Jurkovski 2010].) It has been compiled using the GNU C++ 4.4.1 compiler with the "-O3" option. We used a PC with an Intel Core i7 930 processor with 12 GB of main memory for the experiments. The time limit for each test was 3600 seconds. All times are wall clock time.

We tested our algorithm on two quite distinct sets of test cases. The first set consists of small, densely populated maps designed specifically for the Pukoban game [Clercq]. Humans tend to have difficulty solving these maps, while computers can solve them with relative ease since the search space tends to be small. The second set of maps consists of much bigger maps, which were designed for Sokoban and tend to have much more free space between the boxes. Humans typically solve Pukoban games of these maps with relative ease close to optimality. The search space grows rapidly with the size of the map, and the free space leads to lots of equivalent moves, so these are challenging maps for a solver.

We tested the shortest distance heuristic and the matching heuristic with three groups of improvements: a *basic* set consisting only of box ordering, a *medium* set adding inertia, choke points and giving preference to boxes closer to the destination, and a *complete* set further adding clogs. Due to space limitations, we present only five of the 12 experiments. Table 1 presents the results for the Pukoban instances with the shortest distance heuristic with basic improvements, and the matching heuristic with complete improvements, respectively. Tables 2, 3, and 4 present the results for the XSokoban instances using the matching heuristic and basic, medium, and complete improvements. In all tables column "M" reports the number of moves if an optimal solution has been found or "-" otherwise, column "T" reports the execution time in seconds, column "V" reports the number of visited nodes, column "Nodes" reports the total number of nodes, and column "BF" reports the branching factor.

The branching factor depends on the characteristics of the instance (e.g. the number and placement of obstacles) and varies between 3 and 30. We can observe small fluctuations for different heuristics, which can explained by the different order in which states are explored. The larger instances have an average branching factor of about 12.3, which is larger than the typical branching factor of 10 for Sokoban and 7 for Atomix [Hüffner et al. 2001], showing that we have to explore more states to find an optimal solution.

All ten of the smaller Pukoban instances can solved in less than five seconds even with the simplest heuristic. Actually, the better heuristic almost doubles the execution time, since it is not able to amortize its higher computational cost by pruning reducing the already small search space. Looking at the XSokoban instances, we can see positive effects of better heuristics. In our experiments, the simple distance heuristics could not solve any instance. Using the matching heuristic, but only box ordering, we are able to solve 15 of the 90 XSokoban instances. Adding inertia and choke points the solution time for the already solved puzzles drops significantly, but we are able to solve only two more puzzles.

Finally, when adding clogs, the number of solved instances increases to 20. Of all the improvements, clogs have the most significant impact on the solution of the puzzle, except the quality of the heuristic. When introducing clogs, the number of visited nodes and the execution time drops by 10% for puzzles already solved with medium improvements. To evaluate the quality of the our solver, we can compare it to Sokoban where the currently best implementation is able to solve 57 of 90 instances [Schaeffer and Junghanns 2000].

5. Conclusions and Future Work

We have proposed an exact solver for Pukoban puzzles. It is able to solve 30 of 100 challenging instances. The most important characteristics which make this possible are a precise distance heuristic, which can be computed efficiently, and the detection of patterns of boxes which increase the estimated distance. We are currently evaluating the quality of the proposed algorithm as an approximate solver, since the heuristic usually is very close to the optimal solution, and most of the time is spent in proving optimality of a solution found early in the search.

Since the implementation has a low memory footprint there was no need to use more sophisticated algorithms to save memory such as IDA* [Korf 1985]. We intend to study the behaviour of IDA* to solve the larger instances. Other future improvements are more effi-

Table 1. Results for Pukoban instances. Left: Shortest distance heuristic. Right: Matching heuristic, box ordering, inertia, choke points and clogs.

Map	M	T	V	Nodes	BF	Map	M	T	V	Nodes	BF
1	48	0	433	1255	2.9	1	48	0.01	431	1247	2.9
2	51	0.03	1056	3464	3.3	2	51	0.03	1014	3367	3.3
3	51	0.02	1028	3332	3.2	3	51	0.04	879	2822	3.2
4	62	0.01	1176	3473	3.0	4	62	0.03	1127	3352	3.0
5	59	0.02	850	2742	3.2	5	59	0.02	802	2608	3.3
6	80	0.27	11808	37431	3.2	6	80	0.59	11642	37040	3.2
7	114	0.17	8865	26544	3.0	7	114	0.38	8792	26338	3.0
8	85	0.14	6645	21200	3.2	8	85	0.37	6637	21188	3.2
9	94	1.22	31632	99005	3.1	9	94	2.63	31610	98961	3.1
10	173	1.82	41797	130804	3.1	10	173	4.03	41746	130721	3.1

cient data structures, such as the data structure proposed by Hüffner [Hüffner et al. 2001] for a better priority queue management, and the evaluation of other search space pruning techniques. Pukoban has a lot of potential for new discoveries applying domain knowledge for finding and exploring patterns such as clogs, which can improve the distance estimate to the solution in specific situations. As an example, we may be able to improve the heuristic in the presence of multiple clogs.

References

Clercq, D. D. Pukoban. `http://puzzles.net23.net/pukoban.htm`.

Culberson, J. C. (1997). Sokoban is PSPACE-complete. Technical report, University of Alberta, 97-02.

Demaine, E. D., Demaine, M. L., Hoffmann, M., and O'Rourke, J. (2003). Pushing blocks is hard. *Computational Geometry*, 26:21–36.

Dor, D. and Zwick, U. (1999). Sokoban and other motion planning problems. *Computational Geometry*, 13:215–228.

Hart, P. E., Nilsson, N. J., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107.

Hüffner, F., Edelkamp, S., Fernau, H., and Niedermeier, R. (2001). Finding optimal solutions to Atomix. In *Advances in artifical intelligence*, volume LNCS 2174, pages 229–243.

Jurkovski, B. (2010). Pukoban solver. Available: `https://github.com/bjurkovski/PukobanSolver`.

Korf, R. E. (1985). Depth-first iterative-deepening: An optimal admissible tree search. *Artificial Intelligence*, 27:97–109.

Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:83–97.

Schaeffer, J. and Junghanns, A. (2000). Sokoban: Enhanceing general single-agent search methods using domain knowledge. *Artificial Intelligence*, 129(1–2):219–251.

Table 2. Results for Xsokoban instances, using the matching heuristic and inertia.

Map	M	T	V	Nodes	BF	Map	M	T	V	Nodes	BF
1	87	0	125	1053	8.4	46	-	+	324987	6272285	19.3
2	117	2	6900	44840	6.5	47	-	+	413292	4143255	10.0
3	128	148	161348	1558977	9.7	48	-	+	245623	3317753	13.5
4	-	+	747745	4650435	6.2	49	-	+	420263	4485298	10.7
5	131	821	411147	4047201	9.8	50	-	+	452012	5359271	11.9
6	-	+	753529	5951821	7.9	51	-	+	564492	9591362	17.0
7	78	0	97	1328	13.7	52	-	+	404374	4507530	11.1
8	210	1	296	6242	21.1	53	-	+	677418	10060146	14.9
9	-	+	794601	8924617	11.2	54	-	+	434991	5077483	11.7
10	-	+	171838	2536990	14.8	55	-	+	677916	7932238	11.7
11	-	+	558184	7100514	12.7	56	179	45	40727	460398	11.3
12	-	+	247054	6862772	27.8	57	-	+	305656	6121244	20.0
13	-	+	438962	4910516	11.2	58	-	+	901254	6503799	7.2
14	-	+	306240	4010911	13.1	59	-	+	490509	5963677	12.2
15	-	+	427873	5460235	12.8	60	-	+	369231	6853194	18.6
16	-	+	580914	5332977	9.2	61	-	+	399899	5121082	12.8
17	-	+	1498085	11911426	8.0	62	-	+	310049	4681194	15.1
18	-	+	447757	4940984	11.0	63	-	+	447377	7489930	16.7
19	-	+	624368	8875012	14.2	64	-	+	390168	4629855	11.9
20	-	+	406655	5086489	12.5	65	-	+	486078	5495624	11.3
21	-	+	589015	7340251	12.5	66	-	+	797247	7643583	9.6
22	-	+	274599	2905952	10.6	67	-	+	870596	8129591	9.3
23	-	+	779940	6421782	8.2	68	-	+	503990	6613114	13.1
24	-	+	130118	2042980	15.7	69	-	+	731776	7651692	10.5
25	-	+	288138	4554366	15.8	70	-	+	578488	7525758	13.0
26	-	+	598874	5528350	9.2	71	-	+	587682	6803117	11.6
27	-	+	292889	3129733	10.7	72	-	+	600192	6305527	10.5
28	-	+	380271	5047256	13.3	73	-	+	1130097	13838485	12.2
29	-	+	1248308	5169734	4.1	74	-	+	758917	8324322	11.0
30	-	+	688126	4320733	6.3	75	-	+	463823	4273020	9.2
31	-	+	413349	3839136	9.3	76	-	+	364254	4775723	13.1
32	-	+	303835	4152244	13.7	77	-	+	645304	5506194	8.5
33	-	+	715774	5966822	8.3	78	126	0	131	2233	17.0
34	-	+	410182	7671479	18.7	79	164	0	166	2994	18.0
35	-	+	293196	3789752	12.9	80	-	+	941829	7539384	8.0
36	345	3311	968816	9929790	10.2	81	167	0	179	2967	16.6
37	-	+	462132	3888028	8.4	82	133	28	33199	361712	10.9
38	29	0	361	3631	10.1	83	194	3578	1379790	13222972	9.6
39	-	+	318771	4015157	12.6	84	141	0	201	3901	19.4
40	-	+	538140	8134070	15.1	85	-	+	270608	5128451	19.0
41	-	+	1148548	6977961	6.1	86	-	+	681512	8624364	12.7
42	-	+	207050	3971964	19.2	87	-	+	704518	7947805	11.3
43	-	+	844737	9124619	10.8	88	-	+	244461	3510997	14.4
44	-	+	1003579	9559526	9.5	89	-	+	371842	4316562	11.6
45	-	+	473525	5203336	11.0	90	-	+	611637	4573388	7.5

M : Moves. A "-" indicates that the optimal solution has not been found.
T: Time in seconds. A "+" indicates a time exceeding 3600.
V: Number of visited nodes.
BF: Branching factor.

Table 3. Results for Xsokoban instances, using the matching heuristic, box ordering, inertia and choke points.

Map	M	T	V	Nodes	BF	Map	M	T	V	Nodes	BF
1	87	0	125	1101	8.8	46	-	+	325529	6281460	19.3
2	117	0	2166	25571	11.8	47	-	+	414447	4153623	10.0
3	-	+	984518	10759268	10.9	48	-	+	245533	3316770	13.5
4	-	+	738609	10178233	13.8	49	-	+	423930	4526595	10.7
5	131	2	14042	80189	5.7	50	-	+	449481	5324368	11.8
6	-	+	901610	8641494	9.6	51	-	+	565276	9602237	17.0
7	78	0	97	1328	13.7	52	-	+	679712	9532711	14.0
8	210	1	296	6242	21.1	53	-	+	679179	10090817	14.9
9	-	+	998126	11646397	11.7	54	-	+	436405	5094118	11.7
10	-	+	177113	2631903	14.9	55	-	+	678934	7942215	11.7
11	191	1	4726	48610	10.3	56	179	43	40727	460398	11.3
12	-	+	267024	7448642	27.9	57	-	+	312557	6246182	20.0
13	-	+	455884	5074505	11.1	58	-	+	928279	6684635	7.2
14	-	+	323762	4229232	13.1	59	-	+	520410	6418169	12.3
15	-	+	470328	6042569	12.8	60	-	+	373725	6941073	18.6
16	-	+	584752	5360521	9.2	61	-	+	412022	5232583	12.7
17	-	+	1508163	11996694	8.0	62	235	0	242	3691	15.3
18	-	+	450441	4978555	11.1	63	-	+	456934	7697593	16.8
19	-	+	634292	8983881	14.2	64	-	+	387411	5628970	14.5
20	-	+	413107	5166390	12.5	65	-	+	494145	5589309	11.3
21	-	+	586851	7310533	12.5	66	-	+	815513	7818883	9.6
22	-	+	278067	2949167	10.6	67	-	+	898367	8327960	9.3
23	272	3	8264	71927	8.7	68	-	+	517425	6768557	13.1
24	-	+	129292	2031646	15.7	69	-	+	743286	7775347	10.5
25	-	+	224291	2915515	13.0	70	-	+	585291	7622824	13.0
26	-	+	683708	6254966	9.1	71	-	+	596014	6919532	11.6
27	-	+	653057	4767663	7.3	72	-	+	605611	6365504	10.5
28	-	+	396800	5182789	13.1	73	-	+	1156948	14141886	12.2
29	-	+	1238633	5130540	4.1	74	-	+	770812	8442975	11.0
30	-	+	685263	4302399	6.3	75	-	+	457082	4201340	9.2
31	-	+	414745	3854533	9.3	76	-	+	350637	4581203	13.1
32	-	+	308531	4236095	13.7	77	-	+	633572	5374093	8.5
33	-	+	718333	5990458	8.3	78	126	0	131	2233	17.0
34	-	+	413940	7734000	18.7	79	164	0	166	2994	18.0
35	-	+	294234	3807764	12.9	80	-	+	1278263	9673825	7.6
36	345	3288	968816	9929790	10.2	81	167	0	179	2967	16.6
37	-	+	464060	3906289	8.4	82	133	3	12149	114172	9.4
38	29	0	361	3631	10.1	83	194	1145	648937	5599484	8.6
39	-	+	447187	5768782	12.9	84	141	0	201	3901	19.4
40	-	+	419124	5144530	12.3	85	-	+	268697	5088022	18.9
41	-	+	1036220	11304848	10.9	86	-	+	767727	10232581	13.3
42	-	+	207027	3971594	19.2	87	-	+	709049	7995999	11.3
43	-	+	845884	9136733	10.8	88	-	+	242607	3484913	14.4
44	-	+	1005276	9580484	9.5	89	-	+	369655	4292151	11.6
45	-	+	591382	6713599	11.4	90	-	+	616302	4602935	7.5

M : Moves. A "-" indicates that the optimal solution has not been found.

T: Time in seconds. A "+" indicates a time exceeding 3600.

V: Number of visited nodes.

BF: Branching factor.

Table 4. Results for Xsokoban instances, using the matching heuristic, box ordering, inertia, choke points and clogs.

Map	M	T	V	Nodes	BF	Map	M	T	V	Nodes	BF
1	87	0	88	849	9.6	46	-	+	511667	7546028	14.7
2	117	1	2089	25493	12.2	47	-	+	370019	3446207	9.3
3	128	1	12072	58747	4.9	48	-	+	223040	3136806	14.1
4	327	0	328	9823	29.9	49	-	+	405296	4513823	11.1
5	131	2	13728	78143	5.7	50	-	+	417800	5075133	12.1
6	-	+	557734	6802452	12.2	51	-	+	559787	9527692	17.0
7	78	0	97	1328	13.7	52	-	+	664429	9578048	14.4
8	210	1	296	6242	21.1	53	-	+	685347	10207725	14.9
9	-	+	533730	5579917	10.5	54	-	+	430857	5035762	11.7
10	-	+	172697	2549974	14.8	55	-	+	676478	7918272	11.7
11	191	1	4555	47490	10.4	56	179	43	40727	460398	11.3
12	-	+	258335	7186237	27.8	57	-	+	311050	6210172	20.0
13	-	+	427501	4803970	11.2	58	-	+	904296	6527903	7.2
14	-	+	308726	4037505	13.0	59	-	+	484496	5882935	12.1
15	-	+	441718	5663094	12.8	60	-	+	362076	6716824	18.6
16	-	+	562795	5180993	9.2	61	-	+	344446	4495913	13.1
17	-	+	1502845	11944752	7.9	62	235	0	236	3662	15.5
18	-	+	452349	5004612	11.1	63	-	+	440801	7333880	16.6
19	-	+	613026	8772684	14.3	64	-	+	377204	5486005	14.5
20	-	+	414637	5184949	12.5	65	-	+	491234	5552317	11.3
21	-	+	602391	7550315	12.5	66	-	+	814817	7811358	9.6
22	-	+	271109	2868611	10.6	67	-	+	898984	8331686	9.3
23	272	3	7739	68856	8.9	68	-	+	518070	6777196	13.1
24	-	+	131117	2057483	15.7	69	-	+	745022	7789062	10.5
25	-	+	247855	3235373	13.1	70	-	+	585147	7620891	13.0
26	-	+	666969	7893205	11.8	71	-	+	593326	6880524	11.6
27	-	+	443317	3781240	8.5	72	-	+	604484	6349325	10.5
28	-	+	403603	5241978	13.0	73	-	+	1153156	14099720	12.2
29	-	+	1258655	5210841	4.1	74	-	+	769904	8433590	11.0
30	-	+	713702	4482867	6.3	75	-	+	465790	4295976	9.2
31	-	+	412396	4044442	9.8	76	-	+	365590	4796145	13.1
32	-	+	306827	4204944	13.7	77	-	+	652735	5585089	8.6
33	-	+	696681	5845747	8.4	78	126	0	131	2233	17.0
34	-	+	404778	7582671	18.7	79	164	0	166	2994	18.0
35	-	+	270892	3281111	12.1	80	219	3278	936075	7492699	8.0
36	345	3105	897979	9493586	10.6	81	167	0	178	2963	16.6
37	-	+	459437	3921612	8.5	82	133	1	4585	41826	9.1
38	29	0	359	3623	10.1	83	194	16	74091	583068	7.9
39	-	+	450074	5802580	12.9	84	141	0	157	3508	22.3
40	-	+	415672	5097355	12.3	85	-	+	277516	5266711	19.0
41	-	+	1042130	11366048	10.9	86	-	+	740512	10172732	13.7
42	-	+	222883	4736412	21.3	87	-	+	717532	8108724	11.3
43	-	+	782197	9579875	12.2	88	-	+	252170	3628606	14.4
44	-	+	921674	9321561	10.1	89	-	+	380273	4399666	11.6
45	-	+	335943	3580918	10.7	90	-	+	636812	4729089	7.4

M : Moves. A "-" indicates that the optimal solution has not been found.
T: Time in seconds. A "+" indicates a time exceeding 3600.
V: Number of visited nodes.
BF: Branching factor.

GRASP Reativo e Iterated Local Search aplicado ao problema de programação de tarefas em máquinas paralelas com tempos de preparação dependentes da sequência e de recursos

Edmar Hell Kampke[1], José Elias Cláudio Arroyo[2], André Gustavo dos Santos[2]

[1]Departamento de Engenharia Rural – Universidade Federal do Espírito Santo – UFES
Alegre – ES

[2]Departamento de Informática – Universidade Federal de Viçosa – UFV
Viçosa – MG

mazinhok@yahoo.com.br, jarroyo@dpi.ufv.br, andre@dpi.ufv.br

Abstract. *This work deals with the parallel machine scheduling problem with resource-assignable sequence dependent setup times. The goal of the problem is to minimize the total completion time and the total assignes resources. Due to the combinatorial complexity of this problem, it is proposed algorithms based on metaheuristics Reactive GRASP and Iterated Local Search (ILS). The obtained results for each metaheuristic are compared with the best results available in literature. The results show the good performance of the proposed algorithms.*

Resumo. *Este trabalho aborda o problema de sequenciamento de tarefas em máquinas paralelas, com tempos de preparação das máquinas dependente da sequência e do número de recursos utilizados. O objetivo do problema é minimizar o tempo total de conclusão das tarefas e número total de recursos utilizados na preparação das máquinas. Dada a complexidade combinatória do problema, propõem-se algoritmos baseados nas metaheurísticas GRASP Reativo e Iterated Local Search (ILS). Os resultados obtidos em cada metaheurística são comparados entre si e com os melhores resultados disponíveis na literatura. Os resultados apresentam o bom desempenho dos algoritmos propostos.*

1. Introdução

As modernas indústrias de manufatura têm sido estimuladas a tornar seus processos de produção mais eficientes, principalmente por causa da competitividade crescente imposta pelas transformações que têm afetado a ordem econômica mundial. As principais finalidades da programação da produção são: administrar os recursos de uma empresa industrial, sequenciar de forma eficiente os pedidos nos centros de trabalho e atender aos prazos de entrega dos produtos vendidos. Os problemas de programação de tarefas em máquinas encontram-se intimamente ligados ao planejamento e programação da produção, e são muito estudados na literatura científica.

Neste trabalho é abordado um problema de Programação de Tarefas em Máquinas Paralelas (PTMP), no qual são considerados tempos de preparação (*setup times*) de máquinas, que dependem de recursos disponíveis (por exemplo, mão-de-obra). Um dos primeiros trabalhos propostos para o problema PTMP foi apresentado por Marsh e Montgomery (1973). Eles estudaram o problema considerando a minimização de *setups times*. Guinet (1991) modelou matematicamente o problema e aplicou métodos heurísticos para minimizar outros critérios, tais como, média dos tempos de conclusão (*completions times*) das tarefas, média dos atrasos das tarefas e o tempo máximo de conclusão (*makespan*).

Nos trabalhos mais recentes sobre problemas de PTMP, nota-se uma tendência de aplicação de metaheurísticas, tais como *Simulated Annealing* (Kim *et al.,*2002), GRASP (Laguna e González-Velarde, 1991; Armentano e França Filho, 2007) e *Iterated Local Search* (Tang e Luo, 2006).

2. Descrição do Problema

O problema considerado neste trabalho consiste em determinar o melhor sequenciamento de n tarefas em um conjunto de m máquinas paralelas diferentes (uma tarefa é processada em uma única máquina). Cada tarefa j possui um tempo de processamento na máquina i (p_{ij}) e se a tarefa k é processada logo após a tarefa j na máquina i, existe um tempo de preparação S_{ijk} cuja duração depende da quantidade de recurso disponível R_{ijk}. Cada tempo S_{ijk} pode variar entre dois valores S_{ijk}^- (*setup time* mínimo) e S_{ijk}^+ (*setup time* máximo). Similarmente, R_{ijk} pode variar entre R_{ijk}^- (recurso mínimo) e R_{ijk}^+ (recurso máximo). Os *setup times* e os recursos são relacionados de forma linear: se é utilizado um número mínimo (máximo) de recursos então a duração do *setup time* será o maior (menor) possível. Dessa forma, o problema de PTMP abordado neste trabalho, relaciona *setup time* e recursos, e é aqui denotado por Problema de Programação de Tarefas em Máquinas Paralelas com Setup time e Recursos (PPTMPSR), e foi abordado pela primeira vez por Ruiz e Andrés (2007).

O objetivo do problema é minimizar simultaneamente dois critérios: tempo total de conclusão das tarefas e número de recursos utilizados na preparação das máquinas. O tempo de conclusão (*completion time*) da tarefa j na máquina i é denotado por C_{ij}. Conforme foi proposto por Ruiz e Andrés (2007), a função objetivo deste problema é definida como:

$$Z = \lambda \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{\substack{k=1 \\ k \neq j}}^{n} R_{ijk} + \delta \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij}$$

onde, λ e δ representam as importâncias ou pesos atribuídos aos critérios e $R_{ijk} = 0$ se a tarefa j não for precedente da tarefa k na máquina i. Similarmente, se a tarefa j não for processada na máquina i, então $C_{ij} = 0$.

O PPTMPSR descrito acima é provado ser NP-Difícil (Ruiz e Andrés, 2007). Esta prova consiste em simplesmente reduzir o problema a um clássico problema de PTMP com *setup times* dependentes da sequência, que é NP-Difícil (Webster, 1997).

O PPTMPSR é de suma importância para as indústrias de manufatura. Este problema ocorre, por exemplo, na fabricação de cerâmicas, onde existem diferentes

máquinas de polimento. Cada máquina inicia o polimento de um tipo de cerâmica, e ao final a máquina precisa ser limpa e preparada para realizar o polimento de outro tipo de cerâmica. Este tempo de limpeza e preparação varia de acordo com a máquina, com o tipo da cerâmica polida anteriormente, com o tipo de cerâmica que será polida em seguida e com o número de recursos utilizados, que neste caso, pode ser, por exemplo, o número de pessoas que realizarão a limpeza e ajuste da máquina.

A Figura 1 ilustra um exemplo de solução para um caso do problema, com $n = 4$ tarefas e $m = 2$ máquinas. As tarefas 4 e 2 são processadas, nesta ordem, na máquina 1, finalizando nos tempos $C_{14} = 43$ e $C_{12} = 155$, respectivamente. O *setup time* entre estas tarefas é $S_{142} = 61$, sendo utilizados $R_{142} = 3$ unidades de recursos. Da mesma forma, as tarefas 3 e 1 são processadas na máquina 2 e são finalizadas nos tempos $C_{23} = 27$ e $C_{21} = 100$, respectivamente. O *setup time* entre estas tarefas é $S_{231} = 28$ com utilização de $R_{231} = 4$ unidades de recursos.

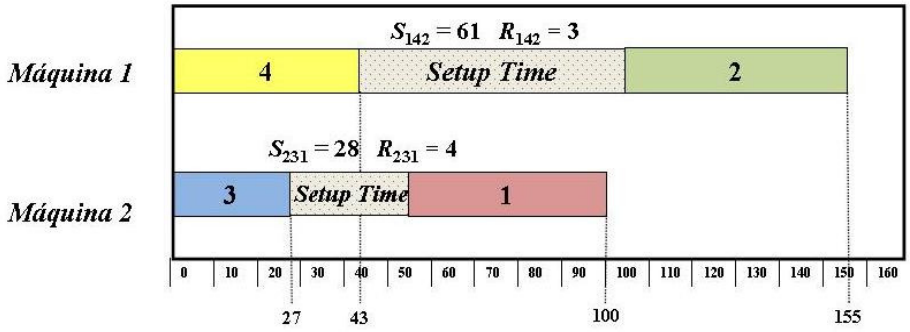


Figura 1. Exemplo de solução.

Neste trabalho, uma solução do problema é representada por um arranjo de tamanho $n+m-1$ contendo as n tarefas e $m-1$ elementos (-1) utilizados para dividir as tarefas em m grupos, um para cada máquina. A solução mostrada na Figura 1 é representada pelo arranjo [4, 2, -1, 3, 1]. Note que os grupos de tarefas [4,2] e [3,1] são processadas respectivamente nas máquinas 1 e 2.

Ruiz e Andrés (2007) propuseram três heurísticas construtivas para resolver o PPTMPSR. A diferença entre as heurísticas está no critério de ordenação das tarefas, utilizado na construção de uma solução. As heurísticas são denominadas *Shortest Processing Time with Setups Resource Assignment* (SPTSA), *Shortest Processing and Setup Time with Setups Resource Assignment* (SPSTSA) e *Dynamic Job Assignment with Setups Resource Assignment* (DJASA). Esta última constrói a solução de forma dinâmica, ordenando crescentemente as tarefas que ainda não foram incluídas na solução parcial, pelo incremento que elas proporcionarão na função objetivo se forem incluídas naquele momento. Ruiz e Andrés (2007) verificaram que a heurística DJASA apresenta os melhores resultados.

Neste artigo são propostos dois algoritmos, um deles utiliza a metaheurística *Greedy Randomized Adaptive Search Procedure* (GRASP) acrescida de um mecanismo reativo para a definição do parâmetro de aleatoriedade utilizado na construção da solução inicial. O outro é baseado na metaheurística ILS (*Iterated Local Search*). Conforme dito anteriormente, há uma tendência em utilizar tais metaheurísticas para a solução de problemas de PTMP, pois em grande parte dos casos, apresentam resultados

satisfatórios. Para testar os algoritmos, utiliza-se 720 problemas disponíveis na literatura. Os resultados obtidos em cada metaheurística são comparados entre si e com os melhores resultados encontrados pelas heurísticas de Ruiz e Andrés (2007).

3. GRASP Reativo

Para a resolução do PPTMPSR, é proposto um algoritmo que utiliza a metaheurística *Greedy Randomized Adaptive Search Procedure* (GRASP), proposta inicialmente por Feo e Resende (1989), acrescida de um mecanismo reativo (Prais e Ribeiro, 2000) para a definição do parâmetro de aleatoriedade utilizado na construção da solução inicial.

```
Procedimento GRASP (α, CritérioParada)
1     f_min ← +∞;
2     enquanto não CritérioParada faça
3        s_1 ← Construção_Solução(α);
4        s_2 ← Busca_Local(s_1);
5        se f(s_2) < f_min então
6           s ← s_2;
7           f_min ← f(s_2);
8        fim-se
9     fim-enquanto;
10    retorne s;
fim GRASP;
```

Figura 2. Pseudocódigo Genérico da Metaheurística GRASP.

A metaheurística GRASP é um método iterative de múltiplas partidas, na qual cada iteração consiste de duas fases: uma fase de construção de uma solução viável e uma fase de busca local, na qual se procura melhorar a qualidade da solução construída na fase anterior. A melhor solução encontrada em todas as iterações da metaheurística é retornada como resultado. Os únicos parâmetros a serem definidos na metaheurística GRASP são o parâmetro de aleatoriedade α e o critério de parada, que geralmente é o número de iterações da metaheurística. Na Figura 2, apresenta-se um pseudocódigo genérico da metaheurística GRASP. Para valores pequenos de α, as soluções são construídas com um maior grau de "gulosidade" e para valores maiores de α, as soluções são construídas de forma mais aleatória. Da Figura 2, nota-se que a metaheurística GRASP iterativamente constrói uma solução s_1 (passo 3) e esta solução é melhorada por um procedimento de busca local (passo 4). Sempre é armazenada a melhor solução encontrada até o momento (passos 5 a 7).

3.1. Construção de soluções

Na fase de construção, uma solução (sequência de tarefas) para o PPTMPSR é gerada iterativamente. Iniciando com uma sequência vazia, a cada iteração, é adicionada uma única tarefa à sequência parcial. Para escolher a tarefa a ser adicionada, é definida uma Lista de Candidatos (LC) com todas as tarefas ainda não sequenciadas. Para cada tarefa dessa lista são feitas simulações de inclusão da tarefa nas máquinas da sequencia parcial. As tarefas desta lista são então ordenadas de forma crescente pelo incremento

que elas proporcionarão ao valor da função objetivo. Esta forma de ordenação das tarefas também é usada na heurística construtiva DJASA (Ruiz e Andrés, 2007). Na heurística DJASA sempre é adicionada a primeira tarefa, ou seja, a tarefa que apresenta o menor incremento ao valor da função objetivo. No algoritmo de construção implementado neste trabalho, ao invés de escolher a primeira tarefa, é escolhida aleatoriamente uma tarefa, dentre as η primeiras tarefas da lista LC. A tarefa escolhida é adicionada na máquina da sequência parcial que proporciona o menor incremento na função objetivo. As η primeiras tarefas da LC formam uma Lista Restrita de Candidatas (LRC) cujo tamanho depende do parâmetro α. O valor de η é definida como MAX(1,α×|LC|). A Figura 3 mostra o algoritmo de construção de soluções. O algoritmo finaliza quando a sequencia contiver todas as *n* tarefas.

```
Procedimento Construção_Solução (α)
1     LC ← {t1,...,tn};        // Lista com todas as tarefas
2     Sequência ← { };
3     enquanto ( | Sequência | < n ) faça
4        Ordena_LC( );
5        η ← MAX(1, α × |LC|);
6        i ← Random(1, η);
7        Sequência ← Sequência ∪ {ti};
8        LC ← LC - {ti};
9     fim-enquanto;
10    retorne Sequência;
fim Construção_Solução;
```

Figura 3. Algoritmo de Construção de Soluções.

3.2. Busca Local

A Busca Local também é um procedimento iterativo que consiste em melhorar uma solução s_1 procurando novas soluções vizinhas dela. Estas soluções vizinhas são obtidas realizando algumas alterações (movimentos) na estrutura da solução atual s_1. Escolhe-se um vizinho s dentre todos os vizinhos de s_1. Se o vizinho escolhido s é melhor que s_1, a busca continua a partir de s (ou seja, $s_1 \leftarrow s$). O procedimento finaliza quando não é possível melhorar a solução atual s_1, ou seja, quando s_1 é um ótimo local. Neste trabalho foram utilizados movimentos de inserção para a obtenção de soluções vizinhas. Um vizinho de s_1 é gerado inserindo uma tarefa que está na posição i da sequência em outra posição j, tal que $1 \leq i, j \leq n$ e $i \neq j$. Por exemplo, para a sequência [4, 2, -1, 3, 1] da Figura 1, as sequências vizinhas [4, -1, 3, 2, 1] e [2, 4, -1, 3, 1] são geradas através da inserção da tarefa 2 após a tarefa 3 e antes da tarefa 4, respectivamente. Realizando este tipo de movimento é possível gerar $k(k\text{-}1)$ soluções vizinhas, onde $k = n+m\text{-}1$ (tamanho da sequência).

3.3. Ajuste do Parâmetro α

A metaheurística GRASP Reativo (Prais e Ribeiro, 2000), denotada neste trabalho por GR, tem como principal característica o auto-ajuste do parâmetro de aleatoriedade α,

utilizado na fase de construção, de acordo com a qualidade das soluções encontradas nas iterações anteriores.

```
Procedimento GR (CritérioParada)
1    f_min ← +∞;
2    iter ← 1;
3    Defina A = {α_1, α_2,..., α_v}
4    para k ← 1 até v faça
5      count[k] ← 0; score[k] ← 0; p_k ← 1/v;
6    fim-para;
7    enquanto não CritérioParada faça
8      α ← Selecione α_k ∈A com probabilidade de escolha p_k;
9      s_1 ← Construção_Solução(α);
10     s_2 ← Busca_Local(s_1);
11     se f(s_2) < f_min então
12       s ← s_2;
13       f_min ← f(s_2);
14     fim-se
15     count[k] ← count[k] + 1; score[k] ← score[k] + f(s_2);
16     se iter mod γ = 0 então
17       avg[k] ← score[k]/count[k] para todo k ∈{1,2,...,v};
18       σ ← Σ (f_min/avg[k])^Θ para todo k ∈{1,2,...,v};
19       p_k ← (f_min/avg[k])^Θ/σ para todo k ∈{1,2,...,v};
20     fim-se
21     iter ← iter + 1;
22   fim-enquanto;
23   retorne s;
fim GR;
```

Figura 4. Estrutura Genérica da Metaheurística GR.

A metaheurística GR funciona da seguinte maneira: define-se um conjunto A de v possíveis valores para o parâmetro α, $A = \{\alpha_1, \alpha_2, ..., \alpha_v\}$. Inicialmente, todos os valores $\alpha_k \in A$ tem a mesma probabilidade de serem escolhidos, ou seja, $p_k = 1/v$. A cada iteração escolhe-se um valor $\alpha_k \in A$ (com probabilidade p_k) para o parâmetro α. As probabilidades p_k são recalculadas a cada γ iterações. Os valores de α_k que produzem melhores soluções terão maior probabilidade p_k, consequentemente, nas próximas iterações terão maior chance de serem escolhidos.

De acordo com a estrutura da metaheurística GR, a quantidade de soluções construídas utilizando α_k é armazenada num arranjo *count* e a soma dos valores da função objetivo dessas soluções é armazenada num arranjo *score*. Considera-se f_{min} o menor valor da função objetivo encontrada até o momento. Os valores de p_k serão

atualizados, a cada γ iterações, utilizando *count*, *score*, f_{min} e um parâmetro de amplificação θ.

Na Figura 4 é apresentado o pseudocódigo da metaheurística GR. Nota-se que nos passos 2 a 6, são inicializadas as variáveis e estruturas usadas no ajuste do parâmetro α. No passo 8 seleciona-se aleatoriamente, com probabilidade p_k, um $\alpha_k \in A$ que será utilizado na iteração atual. No passo 15 é incrementado o número de soluções construídas com α_k e é acumulado o valor da função objetivo da solução encontrada nessa iteração. Os passos 17 a 19, correspondentes à atualização das probabilidades p_k, são executados a cada γ iterações.

Na implementação da heurística GR foi considerado o seguinte conjunto de valores para o parâmetro α: $A = \{0,1;0,2;...;0,9\}$. Os valores dos parâmetros γ e θ foram calibrados e os melhores resultados foram obtidos para $\gamma = 20$ e $\theta = 10$.

3.4. Critério de Parada

Neste trabalho adota-se como critério de parada o tempo computacional, baseado no número de tarefas (n) e número de máquinas (m) de cada problema. O tempo computacional utilizado como critério de parada é de $(n \times m)/2$ segundos. Note que, quanto maior os valores de n e m, maior será o tempo de execução do algoritmo.

4. Iterated Local Search (ILS)

Iterated Local Search (ILS) é um algoritmo metaheurístico, proposto por Lourenço *et al.,* (2002), baseado na idéia de que um procedimento de busca local pode ser melhorado, gerando-se novas soluções de partida, as quais são obtidas por meio de perturbações numa solução ótima local. A perturbação deve permitir que a busca local explore diferentes soluções e, além disso, deve evitar um reinício aleatório. O método ILS é, portanto, um método de busca local que procura focar a busca não no espaço completo de soluções, mas em um pequeno subespaço, definido por soluções que são ótimas locais (Lourenço *et al.,* 2002).

```
Procedimento ILS (CritérioParada, d, T)
1    s1 ← Construção_Solução_Inicial;
2    s ← s1;
3    enquanto não CritérioParada faça
4       s2 ← Perturbação(s1, d);
5       s2 ← Busca_Local(s2);
6       Criterio_Aceitação(s, s1, s2, T);
7    fim-enquanto;
8    retorne s;
fim ILS;
```

Figura 5. Pseudocódigo Genérico da Metaheurística ILS.

Na Figura 5, apresenta-se um pseudocódigo genérico da metaheurística ILS, que possui 4 etapas principais: Obtenção de uma solução ótima local inicial s_1 (passo 1), perturbação da solução s_1 obtendo uma solução s_2 (passo 4), melhoria da solução s_2 (busca local - passo 5) e um critério de aceitação da solução atual (passo 6). As três

últimas etapas são executadas iterativamente enquanto o critério de parada não seja atendido. O algoritmo retorna a melhor solução obtida durante toda sua a execução. A metaheurística ILS é fortemente dependente da solução inicial ótima local. Na etapa de construção de soluções é utilizada a heurística gulosa construtiva DJASA, já detalhada em seções anteriores, seguida do procedimento de busca local.

Neste algoritmo, o procedimento de busca local e o critério de parada serão os mesmos já descritos para o algoritmo anterior. Os parâmetros a serem definidos na metaheurística ILS, além do critério de parada, são a quantidade d de elementos perturbados e o pârametro T utilizado no critério de aceitação.

4.1. Perturbação

O procedimento *Perturbação* modifica a solução corrente, por meio de movimentos aleatórios de remoção de tarefas na solução corrente. Esses movimentos são responsáveis por alterar a solução corrente, guiando-a para uma solução intermediária. Dessa forma, o mecanismo de perturbação deve ser forte o suficiente para permitir escapar do ótimo local corrente e permitir também a exploração de diferentes regiões do espaço de soluções. Ao mesmo tempo, a perturbação precisa ser fraca o suficiente para guardar características do ótimo local corrente, evitando o reinício aleatório.

Neste trabalho, o procedimento *Perturbação* é composto de 2 etapas: *Destruição* e *Construção*. Na primeira são escolhidas, de forma aleatória e sem repetição, d tarefas que serão removidas da solução corrente s_1. O movimento de remoção das d tarefas cria duas subsequências, a primeira de tamanho d, denotada por s_R, contém as tarefas removidas. A segunda subsequência, de tamanho $k - d$, onde $k = n+m$-1 (tamanho da sequência), é a sequência original s_1 sem as tarefas removidas.

```
Procedimento Perturbação(s1, d)
1    sR ← ∅
2    enquanto ( |sR| < d ) faça
3        ti ← Escolhe aleatoriamente uma tarefa de s1;
4        sR ← sR ∪ {ti};              //Insere a tarefa selecionada em sR
5        s1 ← s1 - {ti};              //Remove a tarefa selecionada de s1
6    fim-enquanto;
7    enquanto ( |sR| > 0) faça
8        s1 ← s1 ∪ sR[1];      //Insere a primeira tarefa de sR na melhor posição de s1
9        sR ← sR - sR[1];      //Remove a primeira tarefa de sR
10   fim-enquanto;
11   retorne s1;
fim Perturbação;
```

Figura 6. Pseudocódigo do procedimento de Perturbação.

Após a criação das duas subsequências, é aplicada a etapa de *Construção*. Nesta etapa, iterativamente, todas as tarefas de s_R são reinseridas na solução original s_1. A cada iteração, a primeira tarefa de s_R é removida e inserida em todas as possíveis posições de s_1. Cada sequência obtida é avaliada, e no final a que possui o menor valor da função

objetivo é considerada para a próxima iteração. O procedimento encerra-se quando todas as tarefas de s_R forem reinseridas em s_1.

Na Figura 6 apresenta-se um pseudocódigo genérico do procedimento de *Perturbação*. No passo 1 a sequência s_R é inicializada vazia. A etapa iterativa de *Destruição* é apresentada nos passos 3 a 5. Os passos 8 e 9 mostram a etapa iterativa de *Construção*, onde em cada iteração a primeira tarefa de s_R é removida e inserida em s_1, na posição em que apresentará o menor valor da função objetivo.

4.2. Critério de Aceitação

O *Critério de Aceitação* define se a solução ótima local retornada pela busca local será aceita ou descartada. Uma solução é aceita se ela melhorar a melhor solução atual. No algoritmo ILS, soluções que pioram a melhor solução atual podem também ser aceitas com uma pequena probabilidade. Dessa forma, evita-se a utilização constante da melhor solução atual e, consequentemente, uma rápida estagnação das soluções avaliadas. Neste trabalho utiliza-se um critério de aceitação similar ao usado pela metaheurística *Simulated Annealing*. No entanto, o valor do parâmetro *Temperatura*, neste caso é constante:

$$Temperatur\ a = T \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} p_{ij}}{n \times m \times 10}$$

O valor da *Temperatura* depende dos tempos de processamento (p_{ij}) das tarefas, do número total de tarefas n, do número de máquinas m e do parâmetro T. Esse critério de aceitação é usado nos trabalhos de Osman e Potts (1989) e Ruiz e Stutzle (2008). Neste critério de aceitação a solução s_2 obtida após a busca local, é comparada com a solução atual s_1 que foi submetida à *Pertubação*. Caso s_2 seja melhor que s_1, a solução s_2 é aceita ($s_1 \leftarrow s_2$) para ser perturbada nas próximas iterações. Se s_2 melhorar a melhor solução atual s, então a solução s também é atualizada ($s \leftarrow s_2$). No caso de s_2 ser pior que s_1, s_2 poderá ser aceita para ser perturbada nas iterações seguintes, se:

$$r < exp\{-(f(s_2) - f(s_1))/Temperatura\}$$

onde r é um número aleatório no intervalo [0, 1] e *exp* é a função exponencial.

Neste trabalho, os parâmetros d e T, utilizados no procedimento de perturbação e no critério de aceitação, respectivamente, foram calibrados e os melhores resultados foram obtidos para $d = 4$ ($6 \leq n \leq 10$), $d = 10$ ($n > 10$) e $T = 0{,}5$.

5. Resultados Computacionais

Neste artigo, testa-se o desempenho das metaheurísticas GRASP Reativo(GR) e *Iterated Local Search* (ILS), na resolução do PPTMPSR. Para tal são utilizados os problemas teste gerados por Ruiz e Andrés (2007), que são disponibilizados em *http://www.upv.es/gio/rruiz*. Eles geraram um total de 720 problemas, divididos em dois grupos: *small* e *large* (cada um com 360 problemas). O grupo *small* contém problemas com número de tarefas $n \in \{6,8,10\}$ e número de máquinas $m \in \{3,4,5\}$. Já no grupo de problemas *large* os valores para o número de tarefas (n) e máquinas (m) pertencem respectivamente aos seguintes conjuntos: {50,75,100} e {10,15,20}.

Em todos os problemas, os parâmetros λ e δ, que representam os pesos dos critérios otimizados, foram definidos como: λ=50 e δ=1 (Ruiz e Andrés, 2007).

As metaheurísticas GR e ILS foram implementadas na linguagem de programação Java, com a versão 1.6, e executadas usando o compilador JDK 6.0. Os problemas teste foram resolvidos utilizando um computador com processador Intel® Core™ Quad com 2.4GHz e 3GB de memória RAM.

5.1. Comparação dos Resultados Obtidos

Os resultados obtidos pelas metaheurísticas são comparados com os melhores resultados disponibilizados na literatura (Ruiz e Andrés, 2007). As soluções disponibilizadas para o grupo de problemas *small* foram obtidas pelo software CPLEX (versão 9.1) através da resolução do Modelo Matemático de Programação Inteira (MMPI) (Ruiz e Andrés, 2007). A execução do CPLEX foi limitada a 300 segundos para problemas com $n = 6$ tarefas e 3600 segundos para problemas com $n = 8$ e $n = 10$ tarefas. Para todos os problemas com $n = 6$ e $n = 8$, o CPLEX determinou a solução ótima. Nos problemas com $n = 10$ somente foram obtidas soluções aproximadas, devido ao limite do tempo de execução do CPLEX.

As soluções disponibilizadas para o grupo de problemas *large* são os melhores resultados obtidos pelas heurísticas construtivas propostas por Ruiz e Andrés (2007). Na Tabela 1, as colunas "*Média GR*" e "*Média ILS*" exibem, respectivamente, as médias dos resultados encontrados pelas metaheurísticas GR e ILS, para a função objetivo (*Z*), nos problemas que possuem o mesmo número de tarefas (*n*) e o mesmo número de máquinas (*m*). As colunas "CPLEX" e "*Heurísticas Construtivas*" exibem as médias dos resultados disponibilizados na literatura para os problemas dos grupos *small* e *large*, respectivamente.

Tabela 1. Comparação de resultados das heurísticas GR e ILS com os melhores resultados da literatura (Ruiz e Andrés, 2007).

Problemas *small*				Problemas large			
n*×*m	**Média do CPLEX**	**Média GR**	**Média ILS**	***n*×*m***	**Média dos Melhores Resultados**	**Média GR**	**Média ILS**
6 × 3	**738**	739	**738**	50 × 10	12.828	11.721	**11.674**
6 × 4	474	474	474	50 × 15	8.656	7.982	**7.901**
6 × 5	**287**	288	**287**	50 × 20	6.195	5.741	**5.689**
8 × 3	**1.353**	**1.353**	1.354	75 × 10	26.099	23.489	**23.236**
8 × 4	**943**	**943**	949	75 × 15	18.290	16.945	**16.826**
8 × 5	**683**	**683**	684	75 × 20	14.014	13.016	**12.855**
10 × 3	2.116	**2.076**	2.080	100 × 10	41.517	37.962	**37.488**
10 × 4	1.503	**1.477**	1.484	100 × 15	30.316	28.008	**27.707**
10 × 5	1.117	**1.077**	1.079	100 × 20	23.708	22.287	**22.063**
Média	1.023,7	**1.012,2**	1.014,3	**Média**	20.180,3	18.572,3	**18.382,1**

Para os problemas do grupo *small*, os resultados obtidos pelas metaheurísticas GR e ILS foram bastante satisfatórios. Nos 120 problemas com $n = 6$ tarefas, as metaheurísticas GR e ILS encontraram a solução ótima em 114 e 120 problemas, respectivamente. Em todos os 120 problemas com $n = 8$ tarefas, a metaheurística GR encontrou a solução ótima, enquanto que a metaheurística ILS encontrou a solução ótima em 102 problemas. Nos 120 problemas restantes, com $n = 10$ tarefas, as metaheurísticas, GR e ILS, quando comparadas ao CPLEX, encontraram soluções superiores em 90 e 86 problemas e soluções idênticas em 30 e 34 problemas, respectivamente. Para o grupo *small* de problemas, a porcentagem média de melhoria das soluções obtidas pelas metaheurísticas GR e ILS, em relação aos resultados do CPLEX, foi de 1,12% e 0,92%, respectivamente.

Para os 360 problemas do grupo *large*, as metaheurísticas GR e ILS obtiveram, em todos os casos, resultados superiores em relação aos melhores resultados heurísticos disponibilizados na literatura. Observa-se pela Tabela 1 que dos 9 grupos de problemas *large* que possuem as mesmas dimensões $n \times m$, em todos a metaheurística ILS superou os resultados obtidos pela metaheurística GR. Comparando os resultados obtidos pelas metaheurísticas GR e ILS, para cada um dos 360 problemas do grupo *large*, foi observado que em 297 problemas o resultado da metaheurística ILS superou a heurística GR. Em 4 problemas os resultados obtidos por ambas as heurísticas foram idênticos e a metaheurística GR mostrou resultado superior, com relação a metaheurística ILS, nos 59 problemas restantes. No grupo *large* a porcentagem média de melhoria das soluções obtidas pelas metaheurísticas GR e ILS, em relação aos melhores resultados disponibilizados, foram de 7,97% e 8,91%, respectivamente.

6. Conclusões

Neste trabalho foram propostas duas metaheurísticas: GRASP Reativo (GR) e *Iterated Local Search* (ILS) para resolver o problema de programação de tarefas em máquinas paralelas, denominado PPTMPSR. O PPTMPSR foi recentemente formulado na literatura e até o momento não existem trabalhos de aplicação e comparação de resultados com diferentes metaheurísticas. O desempenho das metaheurísticas propostas foram testados em 720 problemas, de porte variado. As soluções obtidas pelas metaheurísticas foram comparadas com as melhores soluções encontradas na literatura. Para os problemas de grande porte, a metaheurística ILS obteve uma melhoria média de 8,91% com relação às melhores soluções heurísticas disponibilizadas na literatura, o que demonstra a eficácia desta metaheurística. A metaheurística GR, em média, obteve desempenho superior à metaheurística ILS para problemas de pequeno porte. Como trabalho futuro sugere-se o desenvolvimento de métodos híbridos que combinem as metaheurísticas GRASP Reativo com *Iterated Local Search* (ILS), ou até mesmo com outras metaheurísticas, como exemplo: *Simulated Annealing* e Algoritmos Genéticos.

Agradecimentos: Este trabalho foi financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq e pela Fundação de Amparo à Pesquisa do Estado de Minas Gerais – FAPEMIG.

References

Armentano, V.A. e França Filho, M.F. (2007), "Minimizing total tardiness in parallel machine scheduling with setup times: An adaptive memory based GRASP approach". European Journal of Operational Research, 183, 100–114.

Feo, T.A. e Resende, M. (1989), "A probabilistic heuristic for a computationally difficult set covering problem". Operations Research Letters, 8, 67–71.

Guinet, A. (1991), "Textile production systems: a sucession of non-identical parallel processor shops". Journal of the Operational Research Society, 42(8): 655-671.

Kim, D-W., Kim, K-H, Jang, W. e Chen, F.F.(2002), "Unrelated parallel machine scheduling with setup times using simulated annealing". Robotics and Computer Integrated Manufacturing, 18: 223-231.

Laguna, M. e González-Velarde, J.L. (1991), "A search heuristic for just-in-time scheduling in parallel machines". Journal of Intelligent Manufacturing., 2, 253–260.

Lourenço, H.R.; Martin, O. e Stutzle, T. (2002), "Iterated Local Search". In F. Glover and G. Kochenberger, editors, Handbook of Metaheuristics, pages 321–353. Kluwer Academic Publishers, Norwell, MA.

Marsh, J.D. e Montgomery, D.C. (1973), "Optimal procedures for scheduling Jobs with sequence-dependent changeover times on parallel processors". AIIE Technical Papers, 279-286.

Osman, I. e Potts, C. (1989), "Simulated annealing for permutation flow-shop scheduling". Omega, 17(6), 551–557.

Prais, M. e Ribeiro, C.C. (2000), "Reactive grasp: An application to a matrix decomposition problem in tdma traffic assignment". INFORMS Journal on Computing, 12(3), 164–176.

Ruiz, R. e Andrés, C. (2007), "Unrelated parallel machines scheduling with resource-assignable sequence dependent setup times". Em Baptiste, P., Kendall, G., Munier-Kordon, A., Sourd, F., eds.: Proceedings of the 3rd Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA), Paris, France, 439–446.

Ruiz, R. e Stutzle, T. (2008), "An iterated greedy heuristic for the sequence dependent setup times flowshop problem with makespan and weighted tardiness objectives". European Journal of Operational Research, 187(3), 1143–1159.

Tang, L.X. e Luo, J.X. (2006), "A new ILS algorithm for parallel machine scheduling problems". Journal of Intelligent Manufacturing., 17(5), 609–619.

Webster, S.T. (1997), "The complexity of scheduling job families about a common date". Operations Research Letters, 20(2), 65–74.

Um algoritmo de indução de árvore de decisão baseado em agrupamento

Rodrigo C. Barros[1], **Márcio P. Basgalupp**[2], **André C.P.L.F. de Carvalho**[1], **Marcos G. Quiles**[2]

[1]Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo (USP)

[2]Instituto de Ciência e Tecnologia - Universidade Federal de São Paulo (UNIFESP)

{rcbarros,andre}@icmc.usp.br, {basgalupp,quiles}@unifesp.br

Abstract. *Decision tree induction algorithms are well known techniques for assigning objects to predefined categories in a transparent fashion. Most decision tree induction algorithms rely on a greedy top-down recursive strategy for growing the tree, and pruning techniques to avoid overfitting. Even though such a strategy has been quite successful in many problems, it falls short in several others. For instance, there are cases in which the hyper-rectangular surfaces generated by these algorithms can only map the problem description after several sub-sequential partitions, which results in a large and incomprehensible tree. Hence, we propose a new decision tree induction algorithm based on clustering which seeks to provide more accurate models and/or shorter descriptions, which are comprehensible for the end-user. We do not base our performance analysis solely on the straightforward comparison of our proposed algorithm to baseline methods. Instead, we propose a data-dependent analysis in order to look for evidences which may explain in which situations our algorithm outperforms a well-known decision tree induction algorithm.*

1. Introdução

A classificação é um tarefa de aprendizado supervisionado cujo objetivo geral é construir um modelo conciso de distribuição dos valores do atributo classe em função dos valores dos demais atributos, designados preditivos. Nas últimas décadas, pesquisadores na área de Aprendizado de Máquina (AM), estatística e reconhecimento de padrões têm investigado novos algoritmos de classificação tanto para problemas genéricos como para domínios específicos de aplicação. Dentre as diferentes formas de representação do conhecimento adquirido, destacam-se as árvores de decisão (ADs), uma técnica poderosa e amplamente utilizada em processos de tomadas de decisão inteligentes. Seu sucesso pode ser explicado por diversos fatores, tais como [Tan et al. 2005]: (i) interpretabilidade do conhecimento adquirido - uma árvore de decisão usa uma representação gráfica e pode ser facilmente convertida em regras; (ii) robustez na presença de ruídos; (iii) algoritmos de indução de árvores de decisão não são computacionalmente custosos, mesmo para grandes conjuntos de dados; e (iv) capacidade de lidar com atributos redundantes e irrelevantes, os quais, se não tratados adequadamente, podem prejudicar o desempenho do classificador. Alguns algoritmos bem conhecidos para indução de árvores de decisão são o ID3 [Quinlan 1986], o C4.5 [Quinlan 1993] e o CART (*Classification and Regression Trees*) [Breiman et al. 1984].

Embora a maioria dos algoritmos de indução de árvores de decisão empregue uma estratégia gulosa, *top down* e com particionamento recursivo, trabalhos mais recentes propõem outras estratégias [Basgalupp et al. 2009] [Barros et al. 2011]. Por exemplo, um tópico crescente na comunidade de AM e *Data Mining* (DM) é a utilização de um conjunto (*Ensemble*) de árvores de decisão. Um *Ensemble* é a composição de várias árvores de decisão construídas a partir dos exemplos de treino, e o resultado final da classificação é, geralmente, calculado por um esquema de votação [Hastie et al. 2009, Seni and Elder 2010]. Geralmente, um *Ensemble* de classificadores melhora a taxa de acerto quando comparado com um único classificador. No entanto, essa melhora se dá às custas da interpretabilidade, principal virtude das árvores de decisão. Um único

classificador pode ser interpretado por um especialista; porém, na prática, dificilmente um especialista consegue interpretar uma combinação de classificadores, mesmo que eles sejam, isoladamente, fáceis de serem compreendidos. Além de tal análise ser tediosa e demorada, os modelos de classificação que compõem um *Ensemble* estão sujeitos a inconsistências, as quais são necessárias para o aumento de sua acurácia preditiva, dificultando ainda mais a interpretabilidade. Uma vez que cada modelo pode ser considerado uma hipótese para explicar os padrões preditivos, isso significa que um *Ensemble* não representa uma única hipótese coerente sobre os dados, mas sim um conjunto grande de hipóteses inconsistentes, as quais podem confundir o especialista [Freitas et al. 2010]. Por essa razão, a utilização de *Ensembles* não é recomendada em domínios de aplicação em que a interpretabilidade é essencial.

Outra abordagem que tem sido constantemente utilizada é a indução de árvores de decisão por meio de Algoritmos Evolutivos (AEs), cuja principal vantagem está na capacidade de evitar convergência prematura para ótimos locais. Isso porque os AEs realizam uma busca robusta e global no espaço de soluções candidatas, sendo menos suscetíveis à convergência para ótimos locais. Adicionalmente, como resultado dessa busca global, os AEs tendem a lidar melhor que os métodos gulosos em relação à interação entre atributos, pois os relacionamentos complexos entre os atributos não detectados durante uma avaliação gulosa podem ser descobertos por um AE [Freitas 2008]. No entanto, os AEs também apresentam desvantagens, especialmente relacionadas ao tempo de execução. Os AEs constituem uma heurística sólida porém custosa computacionalmente. Além disso, um AE utiliza um grande número de parâmetros a serem definidos (e sintonizados) para otimizar o resultado de sua execução, o que também pode consumir um tempo considerável.

Como em qualquer algoritmo de AM desenvolvido até hoje, há prós e contras em escolher uma ou outra estratégia para indução de ADs. Além disso, algoritmos que apresentam bons resultados em determinados conjuntos de dados podem apresentar resultados ruins em outros. Isso é explicado pelo teorema NFL [Wolpert and Macready 1997]. Embora muitos estudos estejam constantemente propondo novos algoritmos de indução de classificadores, muitos deles neglicenciam o fato de que o desempenho do classificador induzido é altamente dependente dos dados. Além disso, esses estudos normalmente apresentam análises superficiais que não indicam claramente os casos em que o classificador proposto apresentará ganho significativo em relação a outros classificadores.

Com o objetivo de melhorar a acurácia em relação aos algoritmos tradicionais para indução de árvores de decisão e ao mesmo tempo tentar preservar a interpretabilidade do modelo, este trabalho propõe o Clus-DTI (*Clustering for improving Decision Tree Induction*), um novo algoritmo de indução de AD baseado em agrupamento. A ideia chave deste trabalho é que um problema difícil pode ser decomposto em subproblemas mais simples. A hipótese é que é possível utilizar agrupamento para melhorar a classificação com a premissa de que, em vez de solucionar diretamente um problema difícil, é possível melhorar o desempenho da classificação ao solucionar de forma independente subproblemas mais fáceis.

A contribuição deste trabalho pode ser dividida em duas linhas: (i) é proposto um novo algoritmo de indução de AD baseado em agrupamento, e uma análise empírica é realizada com diversos conjuntos de dados públicos; (ii) são investigados cenários nos quais o algoritmo proposto é a melhor opção em relação ao C4.5, um algoritmo tradicional para indução de árvores de decisão. Essa investigação é baseada na análise de dependência de dados, isto é, da forma com que as características estruturais dos conjuntos de dados influenciam no desempenho do algoritmo proposto.

O restante deste trabalho é organizado como segue. Na Seção 2 é apresentado em detalhes o algoritmo Clus-DTI. A Seção 3 apresenta uma comparação entre o algoritmo proposto, Clus-DTI, e o tradicional algoritmo de indução de árvores de decisão, C4.5 [Quinlan 1993] (mais especificamente a sua versão em Java, denominada J48). Na Seção 4 é realizada a análise de dependência de dados, em que são apresentados os possíveis

cenários que sugerem a utilização do Clus-DTI. A Seção 5 apresenta os trabalhos relacionados. As conclusões e sugestões de trabalhos futuros são apresentados na Seção 6.

2. Algoritmo Clus-DTI

Neste trabalho, foi desenvolvido um novo algoritmo chamado Clus-DTI (*Clustering for improving decision tree induction*), cujo objetivo é utilizar agrupamento do conjunto de dados original para melhorar o desempenho das árvores de decisão geradas. Este trabalho é baseado na seguinte premissa: alguns problemas de classificação são difíceis de serem solucionados; assim, em vez de construir uma única árvore de decisão para solucionar um problema de classificação potencialmente difícil, talvez dividir o conjunto de dados em pequenos subproblemas pode tornar menos complexa a solução do problema completo. Em outras palavras, o algoritmo proposto aproxima a função alvo em regiões específicas do espaço de busca. Essas regiões não são aleatoriamente escolhidas, mas sim definidas por um algoritmo de agrupamento disseminado na área. Logo, é razoável supor que exemplos (objetos) pertencentes à mesma distribuição de classes possam ser previamente agrupados e, assim, facilitar a classificação final. No entanto, cabe destacar que o objetivo não é encontrar um mapeamento perfeito entre grupos (*clusters*) e classes. A hipótese deste trabalho é que solucionar subproblemas independentemente em vez de solucionar diretamente o problema completo pode gerar melhores resultados de forma geral. Essa ideia não é nova e tem motivado diversas estratégias na área da ciência da computação, como, por exemplo, a estratégia dividir-para-conquistar.

Assim, Clus-DTI combina dois algoritmos de Aprendizado de Máquina para gerar os modelos de classificação. Seu funcionamento é apresentado como segue.

1. Dado um conjunto de treino $\mathbf{X}$, Clus-DTI divide $\mathbf{X}$ em p partições $\mathbf{P_i}$ de $i+1$ conjuntos (sem sobreposição) de $\mathbf{X}$, tal que a primeira partição tenha dois grupos, a segunda tenha três grupos, e assim por diante. Por exemplo, partição $\mathbf{P_1} = \{C_1, C_2\}$, e na forma geral $\mathbf{P_i} = \{C_1, ..., C_{i+1}\}$, para $1 \leq i \leq p$. Adicionalmente, $\mathbf{C_j} \cap \mathbf{C_k} = \emptyset$ para $j \neq k$ e $\mathbf{C_1} \cup ... \cup \mathbf{C_{i+1}} = \mathbf{X}$. Neste passo, Clus-DTI usa o algoritmo bem conhecido *Expectation Maximization* (EM) [Dempster et al. 1977, Hastie et al. 2009]. Uma vez que o EM é um método estatístico de agrupamento em que todos exemplos pertencem a todos os grupos com uma dada probabilidade, ele atribuirá cada exemplo ao grupo mais provável. Obviamente, Clus-DTI ignora o atributo classe na etapa de agrupamento.
2. Após cada exemplo ser atribuído a um único grupo, Clus-DTI cria uma árvore de decisão para cada grupo utilizando seus respectivos dados (exemplos) como conjunto de treino. Nessa etapa, é utilizado o algoritmo C4.5 [Quinlan 1993]. Assim, a partição $\mathbf{P_i}$ terá um conjunto de $i+1$ árvores de decisão distintas, $D = \{Dt_1, ..., Dt_{i+1}\}$, treinadas de acordo com os dados pertencentes a cada grupo.
3. Para decidir qual partição será selecionada dentre as p disponíveis, Clus-DTI emprega duas opções. A primeira é avaliar cada partição utilizando um critério bem conhecido para validação de agrupamento, o *Simplified Silhouette Width Criterion* (SSWC) [Vendramin et al. 2010]. A partição selecionada é aquela que maximiza SSWC, ou seja, minimiza a dissimilaridade intra-grupo e maximiza a dissimilaridade inter-grupo. A medida de dissimilaridade utilizada foi a distância de Mahalanobis [Mahalanobis 1936], já que os grupos produzidos por essa versão rígida do EM terão um formato elipsoidal. A segunda opção é mais simples e direta: escolher o número de grupos da partição que maximiza a acurácia de treino. Uma vez que o algoritmo está sendo proposto para problemas de classificação, talvez a melhor opção para escolher o número ideal de grupos seja, de fato, um critério relacionado à própria tarefa de classificação. Por ser a mais utilizada, a acurácia foi escolhida como medida de avaliação.
4. Após a partição $\mathbf{P_i}$ ser escolhida, cada exemplo do conjunto de teste $\mathbf{Z}$ é atribuído a um dos $i+1$ grupos, e então a árvore de decisão correspondente $\mathbf{Dt_k}$, $1 \leq k \leq i+1$ é utilizada para classificar esse exemplo. Esse procedimento é repetido para todos os exemplos do conjunto de teste $\mathbf{Z}$.

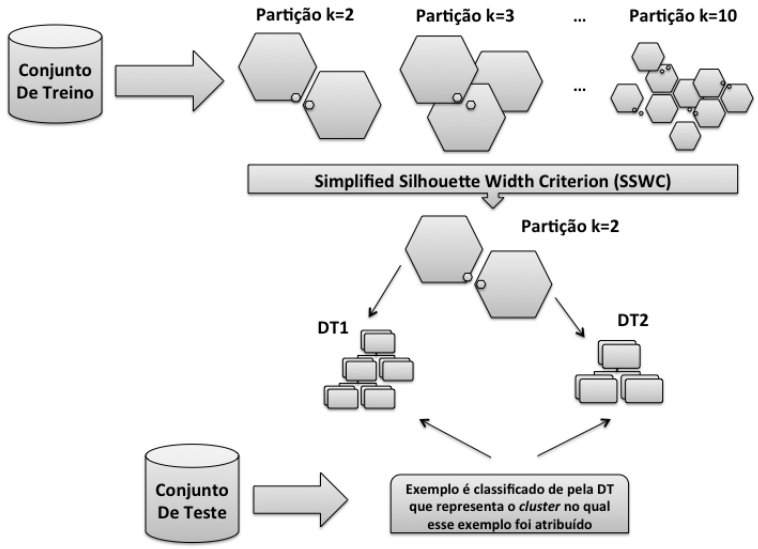


Figura 1. Exemplo de execução do Clus-DTI. Neste cenário, SSWC seleciona a partição com dois grupos como melhor opção dentre as 9 possíveis.

A Figura 1 ilustra esses passos executados pelo Clus-DTI. Nesse cenário fictício, a partição $\mathbf{P_2}$ é aquela que maximiza SSWC (considerando $p = 9$) e, portanto, as duas árvores de decisão Dt_1 e Dt_2 são construídas a partir dos exemplos de treino pertencentes a cada um dos dois grupos. A seguir, cada exemplo do conjunto de teste é atribuído ao grupo com o menor valor da distância de Mahanalobis. A árvore de decisão correspondente ao grupo em questão é então utilizada para classificar esse exemplo em particular.

Novamente, é importante destacar que não se espera um mapeamento total entre grupos e classes. Mesmo que o agrupamento dos dados possa gerar grupos contendo exemplos de uma mesma classe, o que significaria dizer que EM foi capaz de descobrir os parâmetros corretos de distribuição de probabilidades de tal classe, este dificilmente será o caso para a grande maioria dos conjuntos de dados. O objetivo do algoritmo Clus-DTI é encontrar subconjuntos de dados que tornam o problema mais fácil de ser resolvido do que o original. Em outras palavras, o objetivo é identificar situações em que a soma dos desempenhos das árvores de decisão criadas a partir dos subconjuntos apresente melhor resultado geral do que uma simples árvore considerando o conjunto de dados original. Embora esse conceito seja similar ao conceito de *Ensemble*, há uma diferença importante: não é utilizado nenhum esquema de votação para classificar os exemplos. A ausência desse esquema de votação implica em uma grande vantagem de Clus-DTI em relação aos *Ensembles*: é possível acompanhar a árvore de decisão usada para classificar um exemplo em particular em vez de apenas receber a média dos resultados fornecidos por um conjunto de árvores de decisão inconsistente.

3. Análise Comparativa

Para avaliar o desempenho do algoritmo Clus-DTI, foram utilizados 29 conjuntos de dados obtidos da *UCI Machine Learning Repository* [Frank and Asuncion 2010]. Os resultados obtidos foram comparados com o J48 (implementação em Java do algoritmo C4.5, disponível na ferramenta Weka [Witten and Frank 1999]). Os conjuntos de dados utilizados foram: *anneal, audiology, autos, balance-scale, breast-cancer, breast-w, colic, credit-a, credit-g, diabetes, glass, heart-c, heart-h, heart-statlog, hepatitis, ionosphere, iris, kr-vs-kp, labor, letter, lymph, mushroom, primary-tumor, segment, sick, sonar, soybean, vote* e *waveform-5000*. Os conjuntos de dados com k classes, sendo $k > 2$, foram transformados em k conjuntos de duas classes (uma classe contra todas as outras).

Tabela 1. Comparação entre Clus-DTI e J48.

	J48	Clus-DTI
# Wins	34	39
Greatest Difference	8.88%	10%
Average Difference	1.82%	1.84%
Smallest Difference	0.8%	0.01%
Difference $\geq 1\%$	14	19
Difference $\geq 3\%$	10	5
Difference $\geq 5\%$	6	2
Difference $\geq 7\%$	2	1
Difference $\geq 10\%$	0	1

Assim, todos os problemas de classificação utilizados neste trabalho são binários. Essa restrição de lidar apenas com problemas binários é devido às medidas de complexidade [Ho and Basu 2002, Ho et al. 2006] utilizadas na análise de dependência de dados, as quais, em sua maioria, são aplicáveis apenas a problemas de duas classes. Com essas transformações, os 29 conjuntos de dados originais se transformaram em 156 novos conjuntos, os quais foram divididos como segue: 70% para treino[1] e 30% para teste.

A Tabela 1 apresenta os resultados da comparação entre Clus-DTI e J48. *# Wins* indica o número de conjuntos de dados em que cada método superou o outro em termos de acurácia. *Greatest (Average/Smallest) Difference* é a maior (média/menor) diferença em termos de acurácia entre os métodos quando um deles supera o outro. *Difference* $\geq x\%$ é o número de conjuntos de dados em que um método obteve uma diferença maior ou igual a $x\%$ em termos de acurácia.

Em termos de número de vezes em que um algoritmo supera o outro, é possível notar que os dois métodos apresentam resultados similares. Sem considerar os 83 casos em que ocorreu empate, o Clus-DTI superou mais vezes o J48 do que o contrário. Isso é um indicativo de que é vantajoso realizar o agrupamento como etapa prévia à classificação em muitos conjuntos de dados. Entretanto, o teste estatístico não-paramétrico de *Wilcoxon Signed Rank* [Wilcoxon 1945] não apontou nenhuma diferença significativa entre os resultados obtidos pelos dois algoritmos considerando todos os conjuntos de dados. Uma vez que o Clus-DTI é mais custoso computacionalmente que o J48, não seria prudente sugerir o uso de Clus-DTI para quaisquer bases de dados.

Assim, considerou-se de extrema importância investigar os casos em que Clus-DTI supera J48 para, então, tentar encontrar uma relação desses resultados com as características (metadados e medidas de complexidade) desses conjuntos de dados. Isso possibilitaria, então, sugerir a utilização ou não do Clus-DTI de acordo com as características do conjunto de dados a ser utilizado. Essa análise é apresentada na próxima seção.

4. Análise de Dependência dos Dados

Costuma-se dizer que o desempenho de um classificador é fortemente dependente dos dados. Não obstante, trabalhos que propõem novos classificadores normalmente negligenciam essa dependência durante a análise de seus desempenhos. Normalmente, a estratégia adotada é a de fornecer alguns casos em que o classificador proposto supera algum classificador tradicional (comparativo) conforme alguma medida de avaliação dos resultados. Da mesma forma, os estudos teóricos que se propõem a analisar o comportamento dos classificadores também tendem a negligenciar essa dependência. Recentemente, alguns esforços têm sido feitos na busca pela associação das características dos dados com o desempenho de diferentes classificadores com o objetivo de construir sistemas de recomendação. Os algoritmos conhecidos como meta-aprendizado são

[1]Recomenda-se a leitura de [Sánchez et al. 2003] para os interessados em saber como obter conjuntos de treino de alta qualidade para classificação.

uma tentativa de compreender os dados *a priori* da execução de um algoritmo de aprendizado. Dados que descrevem as características dos conjuntos de dados e algoritmos de aprendizado são chamados de metadados, e um novo algoritmo de aprendizado é responsável por interpretar esses metadados e sugerir um modelo de aprendizado em particular [Smith-Miles 2009].

Meta-aprendizado para a seleção de algoritmos geralmente se baseia em medidas estatísticas ou informações teóricas obtidas dos conjunto de dados. Embora essas descrições possam ser suficientes para a recomendação de algoritmos, elas não explicam às características geométricas das distribuições dos exemplos em suas classes, ou seja, negligenciam a maneira pela qual as classes são separadas ou intercaladas, o que representa um fator crítico para determinar a precisão da classificação. Dessa forma, medidas geométricas foram propostas em [Ho and Basu 2002, Ho et al. 2006] para caracterizar a complexidade geométrica de problemas de classificação. O estudo dessas medidas foi um primeiro esforço para compreender melhor a dependência entre conjuntos de dados e classificadores. Além disso, ao estabelecer de forma quantitativa a dificuldade de um problema de classificação, diversos aspectos do processo de classificação podem ser estabelecidos, como recomendação de algoritmos, pré-processamento guiado dos dados e desenvolvimento de algoritmos de classificação para domínios específicos.

Nesta seção, são utilizados os metadados citados e também as medidas propostas em [Ho and Basu 2002, Ho et al. 2006] com o objetivo de compreender as características e a complexidade geométrica de diversos conjuntos de dados para classificação. Primeiro, um resumo dos metadados e das medidas geométricas utilizadas é apresentado. Deve-se observar que as medidas geométricas visam encontrar a complexidade geométrica aparente dos dados, considerando que a probabilidade de distribuição real de cada classe é desconhecida.

4.1. Metadados

A primeira tentativa de caracterizar um conjunto de dados para avaliação de desempenho de algoritmos de aprendizado foi realizada por [Rendell et al. 1987]. A abordagem adotada teve como objetivo prever o tempo de execução dos algoritmos de classificação utilizando exemplos bastante simples de metacaracterísticas, como número de atributos e número de exemplos. Um avanço dessa abordagem é o projeto STATLOG [Michie et al. 1994], o qual investigou o desempenho de vários algoritmos de aprendizado utilizando mais de 20 conjuntos de dados. Outros autores aprofundaram essa abordagem utilizando o mesmo conjunto de metacaracterísticas para caracterização dos dados [Brazdil et al. 1994, Gama and Brazdil 1995]. Esse conjunto pode ser dividido em três categorias: (i) simples; (ii) estatístico; e (iii) baseado na teoria da informação. Um conjunto melhorado de metacaracterísticas foi posteriormente apresentado em [Kalousis 2002], e, dessas, as seguintes são adotadas neste trabalho: número de exemplos (N), número de atributos (n), número de atributos contínuos (con), número de atributos nominais (nom), número de atributos binários (bin), número de classes (cl), percentual de valores ausentes ($\%mv$), entropia de classe ($H(Y)$), entropia média dos atributos (MAE), média do atributo Gini (MAG), informação mútua média entre classes e atributos (MMI) e coeficiente de incerteza (UC). Além dessas medidas apresentadas em [Kalousis 2002], a razão entre o número de exemplos das classes com menor e maior quantidade de exemplos também foi adotada. Essa medida indica o nível de balanceamento do conjunto de dados. Quanto menor o valor (entre zero e um), maior o desbalanceamento.

Em seguida, são apresentadas medidas que procuram explicar como os dados são estruturados geometricamente a fim de avaliar a dificuldade de um problema de classificação.

4.2. Medidas de Complexidade Geométricas

Em [Ho and Basu 2002, Ho et al. 2006], um conjunto de medidas é apresentado para caracterização do conjunto de dados conforme sua estrutura geométrica. Essas medidas são dividas em três categorias: (i) medidas de sobreposição no espaço de atributos; (ii)

medidas de separabilidade de classe; e (iii) medidas de geometria, topologia e densidade. A Tabela 2 apresenta um sumário dessas medidas.

Tabela 2. Sumarização das medidas de complexidade.

Medida	Categoria	Descrição
F1	Sobreposição no Espaço de Atributos	Computa a taxa de discriminação de Fisher. Um valor alto para F1 indica que existe um vetor de transformação que pode separar os exemplos pertencentes a diferentes classes após a projeção dos exemplos neste novo espaço de atributos.
F2	Sobreposição no Espaço de Atributos	Computa a sobreposição dos limites definidos pelos exemplos de cada classe. Um valor baixo da métrica representa que os atributos podem discriminar os exemplos de classes distintas.
F3	Sobreposição no Espaço de Atributos	Computa o poder discriminativo de atributos individuais e retorna o valor do atributo que é capaz de discriminar o maior número de exemplos de treino. Um problema é dito fácil se existe um atributo para o qual os intervalos dos valores gerados por cada classe não se sobrepõem.
F4	Sobreposição no Espaço de Atributos	Segue a mesma ideia utilizada em F3, porém considera o poder discriminativo de todos os atributos (coletivamente).
N1	Separabilidade de Classes	Fornece uma estimativa do comprimento dos limites da classe. Valores altos desta medida sugerem que a maioria dos exemplos está situada nas fronteiras de classes, indicando que o problema em questão é de solução complexa.
N2	Separabilidade de Classes	Compara a distância dos dados de uma classe com a distância aos vizinhos mais próximos pertencentes às outras classes. Valores menores de N2 sugerem que os exemplos pertencentes à mesma classe estão situados em locais próximos no espaço de atributos, e são, portanto, mais facilmente classificados.
N3	Separabilidade de Classes	Denota quão próximos estão os exemplos pertencentes a classes distintas. Retorna o erro *leave-one-out* do vizinho mais próximo (classificador kNN assumindo k=1). Valores baixos de N3 indicam que existe uma grande margem entre classes.
L1	Separabilidade de Classes	Avalia a separabilidade linear do conjunto de treino. Para tal, a soma das diferenças entre o rótulo obtido por um classificador linear e o rótulo real dos exemplos é computada. L1 retorna o valor zero quando o problema é linearmente separável.
L2	Separabilidade de Classes	Fornece informações sobre a separabilidade linear do conjunto de treino. Um classificador linear, como definido em L1, é utilizado e L2 retorna o erro de treinamento.
L3	Geometria, Topologia e Densidade	Implementa a medida de não-linearidade proposta em [Hoekstra and Duin 1996]. A partir do conjunto de treino, o método define um conjunto de testes por meio de uma interpolação linear com coeficientes aleatórios entre os pares de exemplos de uma mesma classe selecionados aleatoriamente.
N4	Geometria, Topologia e Densidade	Define um conjunto de testes conforme apresentado na métrica L3 e retorna o erro utilizando um classificador 1NN.
T1	Geometria, Topologia e Densidade	Descreve a forma das variedades das classes em relação ao subconjunto de adesão. Em suma, o subconjunto de adesão é definido por uma esfera centrada em um exemplo, e tal esfera cresce tanto quanto for possível antes de tocar um exemplo pertencente à outra classe. Assim, um subconjunto de adesão contém um conjunto de exemplos de uma mesma classe de tal forma que é impossível da mesma crescer sem que exemplos de uma outra classe sejam incluídos.
T2	Geometria, Topologia e Densidade	Retorna o razão entre o número de exemplos do conjunto de dados pelo número de atributos. Se apresenta como um indicador da dispersão dos dados.

4.3. Resultados

Foram calculadas 13 medidas de complexidade e 13 metacaracterísticas para os 156 conjuntos de dados. Após, um conjunto de treino foi criado da seguinte forma: cada exemplo correspondendo a um dos 156 conjuntos de dados e cada atributo correspondendo a uma das 26 medidas. Foram também incluídos outros dois atributos, um correspondendo ao número k de grupos (*clusters*) e outro atributo categórico indicando qual método (Clus-DTI ou J48) apresentou o melhor desempenho, em termos de acurácia, para o conjunto de dados em questão. Esse processo resultou em um conjunto de treino com 156 linhas (exemplos) e 28 colunas (atributos).

A ideia de obter esse conjunto de treino teve o objetvo de realizar uma análise descritiva dos resultados e, assim, entender quais aspectos dos dados têm maior influência em determinar o desempenho dos algoritmos. Em particular, foram investigadas evidências que poderiam ajudar o usuário a escolher *a priori* um dos algoritmos de indução. Para isso, foi criada uma árvore de decisão, a partir do conjunto de treino gerado, com o intuito de interpretar essa árvore e entender em quais circunstâncias um algoritmo supera o outro. A Figura 2 ilustra essa árvore de decisão, a qual explica o comportamento de aproximadamente 85% do conjunto de treino.

Ao analisar a árvore de decisão, é possível concluir que Clus-DTI é mais robusto em relação a valores ausentes do que o J48. Uma hipótese que pode explicar esse comportamento é o fato de que o número de valores ausentes tende a ser amortizado entre

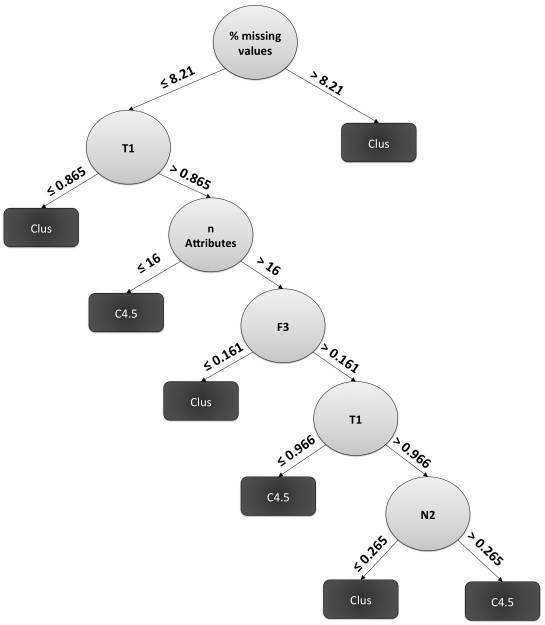


Figura 2. Árvore de decisão que descreve o relacionamento entre as medidas e o melhor algoritmo a ser utilizado.

os grupos, e então cada árvore criada tem um número menor desses valores para tratar. Outra conclusão que pode ser feita é em relação aos conjuntos de dados com poucos valores ausentes (aqueles em que a percentagem de valores ausentes é menor que 8.21%) e a medida T1. Só para relembrar, a medida T1 introduz o conceito de subconjuntos aderentes. Ela projeta hiperesferas que incluem exemplos da mesma classe e normaliza o número de hiperesferas pelo número total de exemplos. Quanto maior é o valor de T1, menor é a tendência dos exemplos serem agrupados em hiperesferas monoclasse. A árvore de decisão indica que os casos em que os exemplos têm uma tendência alta de serem agrupados em hiperesferas monoclasse ($T1 \leq 0.865$) são melhor classificados com o Clus-DTI. Essa parece ser uma conclusão coerente, pois o Clus-DTI agrupa os conjuntos de dados em subconjuntos, e quanto mais puros são esses subconjuntos, mais fáceis são de serem classificados. Um problema potencial nessa análise é que o Clus-DTI não gera grupos limitados por hiperesferas, mas sim por hiperelipses. Mesmo assim, essa descoberta parece ser válida embora toda suposição seja baseada em hiperesferas. Para trabalhos futuros, tem-se a intenção de modificar T1 para que ela possa projetar hiperelipses.

Para os casos restantes (baixa percentagem de valores ausentes e altos valores de T1), a árvore de decisão apresenta outras relações relevantes. Por exemplo, ela testa o número de dimensões e também o F3. A Figura 3 ilustra a análise bivariada de F3 e o número de atributos. O retângulo horizontal indica o primeiro teste sobre o número de atributos separando aqueles exemplos com menos de 16 dimensões. Os círculos que estão dentro desse retângulo são os que já foram filtrados pela árvore de decisão durante os testes sobre a baixa percentagem de valores ausentes e também sobre o valor de T1. Portanto, o terceiro teste sobre o número de atributos de fato separa os casos em que o J48 supera o Clus-DTI. O retângulo vertical, por sua vez, representa o resultado do quarto teste, sobre os valores de F3 (com a restrição adicional de ter mais que 16 dimensões, pois o quarto teste é uma consequência do terceiro).

Uma análise semelhante pode ser feita com os atributos restantes da árvore de decisão. O próximo passo é tentar entender o comportamento dos casos extremos, ou seja, casos em que a diferença de desempenho entre Clus-DTI e J48 é bem alta. As Tabelas 3 and 4 apresentam esses casos, bem como algumas medidas como: (i) balanceamento (*Balance*), a taxa de frequência da classe menos frequente sobre a classe mais frequente,

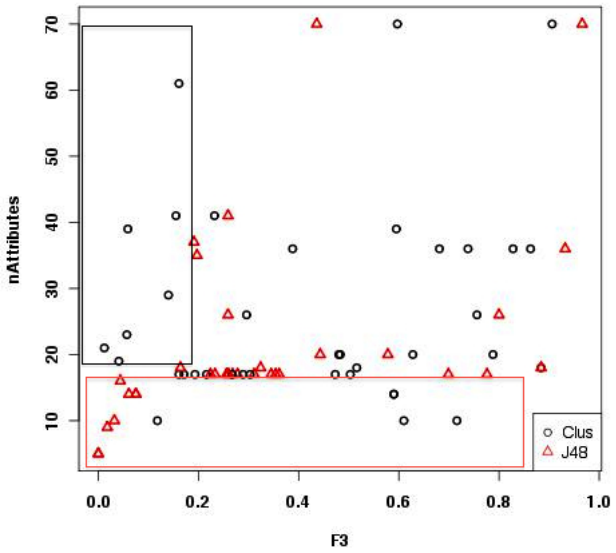


Figura 3. Análise bivariada entre F3 e o número de atributos. Os círculos indicam os conjuntos de dados em que o Clus-DTI supera o J48. Os triângulos indicam os conjuntos de dados em que o J48 supera o Clus-DTI.

Tabela 3. Conjuntos de dados em que o Clus-DTI supera o J48 com uma margem alta.

Conjunto de Dados	Balance	Original TS	Clus Average TS	K	Accuracy J48	Accuracy Clus
Audiology C_2	0.3393	11	8.14	2	96%	100%
Lymph C_2	0.68	15	7.28	3	70%	80%
Primary Tumor C_{17}	0.09	1	2.43	6	91%	94%
Soybean C_{12}	0.03	1	9.53	2	96%	100%

(ii) tamanho da árvore gerada pelo J48 considerando o conjunto de dados completo (*Original TS*), (iii) média ponderada dos tamanhos das árvores geradas pelos grupos formados por Clus-DTI (*Clus Average TS*), (iv) acurácia obtida pelo J48, e (v) acurácia obtida pelo Clus-DTI.

Uma análise cuidadosa indica que o Clus-DTI apresenta melhores resultados que o J48 para conjuntos de dados não balanceados (*primary tumor* e *soybean*). Isso porque o J48 gera apenas um nodo folha representando a classe mais frequente, enquanto o Clus-DTI é apto a explorar as classes menos frequentes nos seus grupos. Em *soybean*, por exemplo, o Clus-DTI classifica corretamente todos os exemplos de uma classe rara.

Outro fator interessante é que o Clus-DTI gera árvores pequenas para todos os conjuntos de dados (excluindo os conjuntos de dados não balanceados, em que o J48 gera apenas um nodo). De fato, isso é previsível porque o conjunto de dados está sendo reduzido e, em alguns casos, estão sendo eliminadas diversas arestas que poderiam ser construídas devido às regiões em que as classes se sobrepõem com frequência. Árvores menores são mais fáceis de interpretar, além de serem mais robustas a *overfitting*.

Tabela 4. Conjuntos de dados em que o J48 supera o Clus-DTI com uma margem alta.

Conjunto de Dados	Balance	Original TS	Clus Average TS	K	Accuracy J48	Accuracy Clus
Autos C_3	0.47	33	6.44	2	79%	73%
Heart-c C_1	0.83	28	11.94	4	86%	80%
Heart-Statlog	0.8	37	16.98	3	82%	74%
Ionosphere	0.56	15	8.13	2	93%	87%

Analisando o conjunto de dados *hepatitis*, por exemplo (não apresentados nas Tabelas 3 e 4), embora a diferença entre J48 e Clus-DTI seja de 3.85% a favor do J48, Clus-DTI gerou duas árvores distintas com 5 nodos cada, enquanto a árvore gerada pelo J48 tem 17 nodos. Dependendo do cenário em questão, é possível que o usuário prefira duas árvores pequenas, as quais são mais fáceis de analisar, às custas de 3.85% de acurácia. Esse tipo de situação deixa claro que o número de vitórias de Clus-DTI poderia ser ajustado ao levar-se em conta o custo/benefício entre acurácia e interpretabilidade (em termos de tamanho da árvore).

Por fim, a última recomendação dessa análise é para os usuários que realmente precisam de modelos simples e interpretáveis: se nenhum dos cenários apontados anteriormente sugerem que Clus-DTI é a melhor opção em relação ao J48, mesmo assim ainda pode ser útil utilizá-lo devido ao tamanho das árvores geradas ser menor. No entanto, essa vantagem desaparece em casos que o número de grupos (*clusters*) sugeridos pelo Clus-DTI multiplicado pela média dos tamanhos das árvores desses grupos excede o tamanho da árvore do J48.

5. Trabalhos Relacionados

[Gaddam et al. 2007] propuseram um algoritmo denominado *cascading* K-Médias [Lloyd 1982] e ID3 [Quinlan 1986] com objetivo de melhorar a detecção de anomalias nos dados. Nesses modelos, o conjunto de treino é dividido em k grupos disjuntos (k é empiricamente definido) pelo algoritmo de agrupamento K-Médias. Um exemplo de teste é primeiramente associado a um dos grupos gerados, e seu rótulo é definido conforme a maioria dos exemplos rotulados dentro do grupo selecionado. Uma árvore de decisão é construída com o algoritmo ID3 a partir dos exemplos do conjunto de treino associado a cada grupo (uma árvore por grupo). Essas árvores também são utilizadas para rotular os exemplos de teste associados aos seus respectivos grupos. O modelo também mantém a informação dos f grupos mais próximos ao exemplo de teste e utiliza um esquema combinado para verificar se o rótulo sugerido pelo ID3 coincide com a do K-Médias. Observando a semelhança entre o modelo de Gaddam et al. com o algoritmo proposto neste trabalho, as principais diferenças entre ambos são destacadas a seguir: (i) Clus-DTI faz uso de dois algoritmos de AM mais robutos do que os utilizados em [Gaddam et al. 2007]; (ii) Clus-DTI não possui o esquema de rotulação combinada dos exemplos de teste utilizado em [Gaddam et al. 2007]. Dessa forma, o modelo proposto permite, de maneira compreensiva, verificar quais foram as regras utilizadas no processo de validação, apresentando, assim, uma vantagem sobre o modelo proposto em [Gaddam et al. 2007] e outros que também adotam um sistema de rotulação combinada ou baseada em votos; (iii) Clus-DTI não está restrito a aplicações de detecção de anomalias; (iv) Clus-DTI define automaticamente o número "ideal' de grupos segundo critérios de validação dos grupos gerados; (v) Clus-DTI é conceitualmente mais simples e de mais fácil implementação que o K-Médias+ID3, mesmo considerando que algoritmos mais robustos são utilizados.

6. Conclusões e Trabalhos Futuros

As principais conclusões desse trabalho são apresentadas abaixo:

- Clus-DTI se apresenta como um algoritmo mais robusto a valores ausentes, pois superou o J48 em todos os conjuntos de dados cujos percentuais de valores ausentes superaram um dado limiar (8%).
- A aderência ao conceito de subconjunto é uma importante variável na seleção de algoritmos. Todos os casos em que os exemplos possuem uma maior tendência de serem agrupados em um único grupo hiperesférico (isto é, $T1 \leq 0.865$) são melhor tratados pelo Clus-DTI. Esse fato se mostra coerente, pois o agrupamento é realizado *a priori* ao processo de classificação. No entanto, é importante salientar que os grupos gerados pelo EM possuem forma geométrica de hiperelipses e não hiperesperas. Assim, a reformulação de T1 para o crescimento de hiperelipses pode gerar resultados ainda superiores.

- Outros fatores importantes na seleção de um algoritmo estão relacionados ao número de atributos, à medida F3 e à medida N2. Nesse último caso, é intuitivo pensar que valores menores de N2 (classes bem separadas de acordo com a distância euclidiana) aumentam as chances do Clus-DTI produzir melhores resultados, assumindo que este primeiramente realiza o agrupamento dos exemplos, tornando o processo de classificação mais simples.
- Clus-DTI é robusto a classes desbalanceadas. Ele fornece resultados melhores do que J48 em todos os casos em que a árvore resultante apresenta um único nodo. Clus-DTI reduz o impacto do desbalanceamento das classes quando essas são agrupadas em grupos distintos, possibilitando, inclusive, a produção de regras que descrevem classes pouco representadas.
- J48 apresentou resultados superiores em 7 dos 8 conjuntos de dados formados por classes mais balanceadas.

Este trabalho se caracteriza como um primeiro esforço no estudo dos potenciais benefícios do agrupamento de dados na classificação. Ele apresenta um novo algoritmo de classificação que pode ser visto como um arcabouço mais elegante do que simplesmente pré-processamento de dados por meio de técnicas de agrupamento. Ele abriu diversas linhas de trabalhos futuros. Pretende-se implementar vários métodos alternativos para escolher o melhor valor de k. Em particular, duas estratégias: (i) utilizar a acurácia de um conjunto de validação para escolher o melhor valor de k, e (ii) implementar a abordagem proposta por [Kothari and Pitts 1999] para determinar o valor de k. Além disso, também pretende-se testar outros algoritmos de agrupamento, bem como outros algoritmos de classificação a fim de avaliar se as mesmas conclusões podem ser generalizadas para qualquer par de algoritmos. Finalmente, tem-se a intenção de testar Clus-DTI em conjuntos de dados artificiais, com a finalidade de garantir sua eficácia em diferentes cenários, tais como conjuntos de dados linearmente separáveis, não-linearmente separáveis, desbalanceados, com alta dimensionalidade e esparsos.

Referências

Barros, R. C., Ruiz, D. D., and Basgalupp, M. P. (2011). Evolutionary model trees for handling continuous classes in machine learning. *Information Sciences*, 181:954–971.

Basgalupp, M., Barros, R. C., de Carvalho, A., Freitas, A., and Ruiz, D. (2009). Legal-tree: a lexicographic multi-objective genetic algorithm for decision tree induction. In Shin, S., Ossowski, S., Martins, P., Menezes, R., Virol, M., Hong, J., Shin, D., Palakal, M., Fritzke, U., Crosby, M., and Haddad, H., editors, *Proceedings of the 2009 ACM Symposium on Applied Computing*, pages 1085–1090. ACM Press.

Brazdil, P., Gama, J. a., and Henery, B. (1994). Characterizing the applicability of classification algorithms using meta-level learning. In *Proceedings of the European conference on machine learning on Machine Learning*, pages 83–102, Secaucus, NJ, USA. Springer-Verlag New York, Inc.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees*. Wadsworth.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository.

Freitas, A. A. (2008). *Soft Computing for Knowledge Discovery and Data Mining*, chapter A Review of evolutionary Algorithms for Data Mining, pages 79–111. Springer US.

Freitas, A. A., Wieser, D. C., and Apweiler, R. (2010). On the importance of comprehensible classification models for protein function prediction. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 7:172–182.

Gaddam, S. R., Phoha, V. V., and Balagani, K. S. (2007). K-means+id3: A novel method for supervised anomaly detection by cascading k-means clustering and id3 decision

tree learning methods. *IEEE Transactions on Knowledge and Data Engineering*, 19:345–354.

Gama, J. and Brazdil, P. (1995). Characterization of classification algorithms. In *Proceedings of the 7th Portuguese Conference on Artificial Intelligence: Progress in Artificial Intelligence*, pages 189–200, London, UK. Springer-Verlag.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. Springer Series in Statistics. Springer, 2nd ed. 2009. corr. 3rd printing edition.

Ho, T., Basu, M., and Law, M. (2006). Measures of geometrical complexity in classification problems. In Jain, L., Wu, X., Basu, M., and Ho, T. K., editors, *Data Complexity in Pattern Recognition*, Advanced Information and Knowledge Processing, pages 1–23. Springer London.

Ho, T. K. and Basu, M. (2002). Complexity measures of supervised classification problems. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(3):289–300.

Hoekstra, A. and Duin, R. P. (1996). On the nonlinearity of pattern classifiers. In *In Proceedings of the 13th ICPR*, pages 271–275.

Kalousis, A. (2002). *Algorithm Selection via Meta-Learning*. PhD thesis, Université de Genève, Centre Universitaire d'Informatique.

Kothari, R. and Pitts, D. (1999). On finding the number of clusters. *Pattern Recognition Letters*, 20(4):405 – 416.

Lloyd, S. (1982). Least squares quantization in pcm. *Information Theory, IEEE Transactions on*, 28(2):129 – 137.

Mahalanobis, P. C. (1936). On the generalised distance in statistics. In *Proceedings National Institute of Science, India*, volume 2, pages 49–55.

Michie, D., Spiegelhalter, D. J., Taylor, C. C., and Campbell, J., editors (1994). *Machine learning, neural and statistical classification*. Ellis Horwood, Upper Saddle River, NJ, USA.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Rendell, L., Sheshu, R., and Tcheng, D. (1987). Layered concept-learning and dynamically variable bias management. In *Proceedings of the 10th international joint conference on Artificial intelligence - Volume 1*, pages 308–314, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Sánchez, J. S., Barandela, R., Marqués, A. I., Alejo, R., and Badenas, J. (2003). Analysis of new techniques to obtain quality training sets. *Pattern Recogn. Lett.*, 24:1015–1022.

Seni, G. and Elder, J. (2010). *Ensemble Methods in Data Mining: Improving Accuracy Through Combining Predictions*. Morgan and Claypool Publishers.

Smith-Miles, K. A. (2009). Cross-disciplinary perspectives on meta-learning for algorithm selection. *ACM Comput. Surv.*, 41:6:1–6:25.

Tan, P.-N., Steinbach, M., and Kumar, V. (2005). *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Vendramin, L., Campello, R. J. G. B., and Hruschka, E. R. (2010). Relative clustering validity criteria: A comparative overview. *Stat. Anal. Data Min.*, 3:209–235.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics*, 1:80–83.

Witten, I. H. and Frank, E. (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann.

Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82.

Uma Proposta de Melhoria do Algoritmo Guloso de Estimação de Mistura de Gaussianas

Andre Paim Lemos[1], **Antonio Pádua Braga**[1]

[1]Universidade Federal de Minas Gerais - UFMG
Departamento de Engenharia Eletrônica
Av. Antônio Carlos, 6627 - Belo Horizonte - MG - Brasil

{andrepl,apbraga}@cpdee.ufmg.br

Abstract. *Este trabalho propõe modificações no critério de parada do algoritmo guloso de estimação de Misturas de Gaussianas, com o objetivo de melhorar sua acurácia na busca pelo número de componentes ótimo. Neste trabalho o critério de parada desse algoritmo é modificado para utilizar um teste de normalidade multivariado amostral, de forma que o algoritmo para quando todas as componentes da mistura passem nesse teste. O algoritmo modificado é comparado com o algoritmo original, que utiliza critérios de parcimônia como critério de parada. Resultados de simulações numéricas sugerem a melhoria na acurácia quando o criterio de parada proposto neste trabalho é utilizado.*

Resumo. *This work proposes modifications on the stop criterion of the greedy algorithm for Gaussian Mixtures, in order to increase the accuracy in the search for the optimum number of mixture components. In this work, the stop criterion is modified in order to use a sampling multivariate normality test. The algorithm stops when all mixture components pass on the proposed test. The modified algorithm is compared with the original one, that uses parsimony criterion as stop criterion. Numerical simulation results suggest the accuracy improvement when the stop criterion proposed in this work is used.*

1. Introdução

Modelos de Mistura de Gaussianas são usualmente estimados por meio de métodos iterativos de Maximização da Expectativa (*Expectation Maximization, EM*) [Dempster et al. 1977]. Embora esta abordagem para a maximização da função de verossimilhança seja eficiente em muitas situações, são comuns os problemas de convergência para ótimos locais, além de haver a necessidade de se prover, a priori, o número de componentes da mistura [Ververidis and Kotropoulos 2008].

Diversas abordagens podem ser encontradas na literatura para tratar as deficiências desse algoritmo. Em geral, as técnicas mais comuns tratam o problema através de buscas heurísticas pelo número ótimo de componentes [Ververidis and Kotropoulos 2008] ou através de metodologias de inicialização dos parâmetros do algoritmo visando a evitar a convergência para ótimos locais [Figueiredo et al. 2000].

Em [Verbeek et al. 2003] um algoritmo guloso de estimação de Misturas de Gaussianas é proposto. Esse algoritmo utiliza uma busca incremental pelo número de

componentes da mistura, iniciando de apenas uma, e inserindo componentes no modelo através da divisão das componentes já existentes. Como a componente inicial é a única, esta pode ser estimada a partir de todos os dados via estimador de máxima verossimilhança, resolvendo o problema de inicialização dos parâmetros. Além disso, a busca incremental evita que seja necessário o conhecimento do número de componentes a priori, lidando assim com as duas deficiências do algoritmo EM. Porém, o critério de parada desse algoritmo é baseado em critérios de parcimônia de modelos, como AIC [Ververidis and Kotropoulos 2005] e MDL [Biernacki et al. 1999], o que, segundo [Li and Ma 2008], em muitos casos, não garantem a busca pelo resultado ótimo, relativo ao número de componentes final da mistura.

Assim, esse trabalho propõe uma modificação no critério desse algoritmo, para melhorar sua acurácia na busca pelo número de componentes ótimo do modelo. Para isso, o critério de parada desse algoritmo é modificado para utilizar um teste de normalidade multivariado, baseado na *Distância de Mahalanobis*, proposto por [Ververidis and Kotropoulos 2008], de forma que o algoritmo só acrescenta componentes no modelo, ou seja, uma nova iteração é iniciada, se pelo menos alguma das componentes presentes rejeitar a hipótese nula do teste, dado um nível de significância.

Esse trabalho está dividido da seguinte maneira: a seção 2 descreve o modelo de Mistura de Gaussianas, em seguida, a seção 3 descreve o Algoritmo EM e apresenta suas principais deficiências. A seção 4 descreve algumas das abordagens descritas na literatura para tratar as deficiências desse algoritmo. Em seguida, a seção 5 descreve a metodologia proposta nesse trabalho baseada na modificação do algoritmo guloso, proposto em [Verbeek et al. 2003]. A seção 6 descreve experimentos avaliando a modificação proposta. Finalmente, a seção 7 apresenta as conclusões finais.

2. Mistura de Gaussianas

A função de densidade de probabilidade (*probability density function, pdf*) Gaussiana é uma função parametrizada pelo vetor médio μ e a matriz de covariância Σ em um espaço paramétrico de dimensão D, descrita pela seguinte equação:

$$N(x;\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} exp\left[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right] \quad (1)$$

Os parâmetros desse modelo podem ser estimados a partir de um conjunto de amostras, utilizando-se o estimador de máxima verossimilhança [Duda et al. 2001]. Para isso, dado um conjunto de amostras de tamanho N, $x = x_1, \cdots, x_N$, define-se a função de verossimilhança como:

$$L(x;\Theta) = \prod_{n=1}^{N} p(x_n;\Theta) \quad (2)$$

esta função descreve a verossimilhança das amostras em função dos parâmetros. Assim, para encontrar os parâmetros que melhor descrevem o conjunto de amostras, deve-se encontrar o valor máximo dessa função:

$$\hat{\Theta} = \arg\max_{\Theta} L(x;\Theta) \tag{3}$$

Geralmente, essa função não é maximizada diretamente, mas sim seu logaritmo:

$$\ell(x;\Theta) = lnL(x;\Theta) = \sum_{n=1}^{N} lnp(x_n;\Theta) \tag{4}$$

definida como o logaritmo da verossimilhança, sendo geralmente mais fácil de ser manipulada.

Assim, para encontrar os parâmetros da pdf Gaussiana, $\Theta = [\mu \ \Sigma]$, deriva-se o logaritmo da verossimilhança e iguala-se o resultado a zero. Realizada essa manipulação, encontram-se os estimadores de máxima verossimilhança para o vetor médio e a matriz de covariância.

Modelos de mistura de gaussianas (*Gaussian Mixture Models, GMM*) são modelos formados por uma mistura de várias distribuições Gaussianas que podem ser utilizados para modelar dados multimodais [Duda et al. 2001]. A pdf desse modelo é definida como:

$$p(x;\Theta) = \sum_{c=1}^{C} \alpha_c N(x;\mu_c,\Sigma_c) \tag{5}$$

sendo α_c o peso de cada componente c do modelo, de forma que $0 < \alpha_c < 1$ e $\sum_{c=1}^{C} \alpha_c = 1$.

O conjunto de parâmetros:

$$\Theta = [\alpha_1, \mu_1, \Sigma_1, \cdots, \alpha_C, \mu_C, \Sigma_C] \tag{6}$$

definem o modelo.

Os parâmetros desse modelo não podem ser estimados a partir de um conjunto de amostras, de forma análoga a estimação dos parâmetros da pdf Gaussiana, uma vez que não se tem a informação de qual componente gerou cada uma das amostras. Caso essa informação estivesse disponível, a estimação dos parâmetros poderia ser feita de forma trivial e independente para cada componente. Porém, como essa informação não é disponível, os parâmetros são estimados via o Algoritmo de Maximização da Expectativa.

3. Algoritmo EM para Mistura de Gaussianas

O Algoritmo de Maximização da Expectativa (EM) é um algoritmo iterativo utilizado para estimar parâmetros de um modelo, baseado na função de verossimilhança, quando uma solução analítica é infactível ou quando conjunto de amostras possui dados incompletos.

Para isso, esse algoritmo assume que o conjunto de amostras utilizado na estimação dos parâmetros é formado por algumas características conhecidas e outras ocultas. Denota-se por x as características conhecidas, e y as ocultas. O algoritmo é formado então por dois passos. O primeiro passo, (*E-Step*), resolve a seguinte equação:

$$Q(\Theta; \Theta^i) = E_y[\ell(x, y; \Theta)|x; \Theta^i] \tag{7}$$

em que Θ^i é a estimativa da iteração anterior dos parâmetros e Θ é a nova estimativa. Este passo calcula o valor esperado da verossimilhança dos dados, incluindo os dados ocultos marginalizados em função da estimativa corrente dos parâmetros, Θ^i.

O segundo passo do algoritmo, (*M-Step*), maximiza $Q(\Theta, \Theta^i)$ com respeito ao parâmetro Θ, gerando uma nova estimativa para os parâmetros:

$$\Theta^{i+1} \leftarrow \arg\max_{\Theta} Q(\Theta; \Theta^i) \tag{8}$$

Esses passos se repetem até que uma condição seja atingida, como por exemplo:

$$Q(\Theta^{i+1}; \Theta^i) - Q(\Theta^i; \Theta^{i-1}) \leq T \tag{9}$$

sendo T um valor definido como parâmetro do algoritmo.

O Algoritmo EM começa a partir de uma estimativa inicial Θ^0 para os parâmetros e garante que o logaritmo da função de verossimilhança aumenta a cada iteração, até que ocorra a convergência [Dempster et al. 1977].

Esse algoritmo pode ser utilizado para estimar os parâmetros de uma Mistura de Gaussianas, assumindo que o conjunto de amostras possui dados ocultos. Esses dados ocultos são definidos como o conhecimento de qual componente do modelo gerou cada amostra. Assim, define-se um vetor de variáveis ocultas $y = \{y_i\}_{i=1}^N$, sendo que para cada amostra x_n, existe uma variável y_n, tal que $y_n \in 1, \cdots, C$, e $y_n = c$, se a amostra x_n foi gerada pela componente c [Bilmes 1998]. O logaritmo da função de verossimilhança completa (dados ocultos e conhecidos) é dada por:

$$\ell(x, y; \Theta) = \sum_{n=1}^{N} ln(\alpha_{y_n} p(x_n|\Theta_{y_n})) \tag{10}$$

O passo *E-Step* deve calcular o valor esperado do logaritmo da verossimilhança condicionado aos dados conhecidos, x, e da estimativa corrente dos parâmetros Θ^i. Como o valor do vetor y não é conhecido, assume-se que y é um vetor aleatório, e a expressão de cada um dos seus componentes pode ser estimada utilizando-se a regra de *Bayes*:

$$p(y_n = k|x_n, \Theta^i) = \frac{\alpha_k^i p(x_n|\Theta_k^i)}{\sum_{c=1}^{C} \alpha_c p(x_n|\Theta_c)}) \tag{11}$$

E a expressão final de y é definida como:

$$p(y|x, \Theta^i) = \prod_{n=1}^{N} p(y_n|x_n, \Theta^i) \tag{12}$$

Assim, a expressão do valor esperado do logaritmo da verossimilhança é dado por [Bilmes 1998]:

$$Q(\Theta; \Theta^i) = \sum_{y \in \Upsilon} \ell(x, y; \Theta) p(y|x, \Theta^i) \tag{13}$$

sendo Υ os possíveis valores que y pode assumir. Essa expressão pode ser maximizada, derivando-se e igualando a zero, gerando assim as estimativas dos parâmetros correspondentes ao passo *M-Step* do Algoritmo EM.

Assim, os passos do Algoritmo EM para a mistura de gaussianas são definidos como:

E-Step: A probabilidade de que cada amostra foi gerada por cada componente é calculada pela equação (11) e uma nova estimativa da verossimilhança é gerada ($Q(\Theta; \Theta^i)$).

M-Step: Os parâmetros do modelo são atualizados, dada uma nova estimativa da verossimilhança, pelos seguintes estimadores:

$$\alpha_c^{i+1} = \frac{1}{N} \sum_{n=1}^{N} p(y_n = c|x_n, \Theta^i) \tag{14}$$

$$\mu_c^{i+1} = \frac{\sum_{n=1}^{N} x_n p(y_n = c|x_n, \Theta^i)}{\sum_{n=1}^{N} p(y_n = c|x_n, \Theta^i)} \tag{15}$$

$$\Sigma_c^{i+1} = \frac{\sum_{n=1}^{N} p(y_n = c|x_n, \Theta^i)(x_n - \mu_c^{i+1})(x_n - \mu_c^{i+1})^T}{\sum_{n=1}^{N} p(y_n = c|x_n, \Theta^i)} \tag{16}$$

O Algoritmo EM descrito possui as seguintes deficiências:

- O número de componentes C da mistura, deve ser definida a priori.
- O valor inicial dos parâmetros, Θ^0, afeta no resultado final, uma vez que o algoritmo pode atingir um máximo local.

4. Estado da Arte

Diversas técnicas podem ser encontradas na literatura para lidar com os problemas do Algoritmo EM, quando utilizado para estimação dos parâmetros de uma Mistura de Gaussianas. Em geral, essas técnicas podem ser divididas em três níveis, baseadas na parte do algoritmo EM em que são aplicadas [Ververidis and Kotropoulos 2008]. O terceiro nível corresponde a técnicas que utilizam heurísticas para estimar o número de componentes da mistura. Já o segundo nível, corresponde a técnicas que realizam uma modificação dos passos do algoritmo para evitar ótimos locais. Finalmente, o primeiro nível corresponde a técnicas que abordam o problema de inicialização dos parâmetros de cada componente.

O número de componentes da mistura pode ser estimado por critérios de parcimônia ou por operações de divisão e agregação (*split and merge*), aplicadas às componentes do modelo.

Métodos baseados em critérios de parcimônia, otimizam uma função que relaciona o logaritmo da verossimilhança com o número de parâmetros do modelo, a fim

de evitar o sobre ajuste (*overfitting*). Esses métodos, em geral podem ser divididos em: métodos baseados em buscas incrementais ou buscas decrementais, ou seja, incrementais iniciam a busca a partir de um modelo formado por apenas uma componente e adicionam componentes até que um critério seja atingido, enquanto métodos decrementais, iniciam a busca com um número alto de componentes e removem componentes até atingir o critério. O critério de parcimônia utilizado pode ser: o Critério de Informação de Akaike (AIC) [Ververidis and Kotropoulos 2005], o Tamanho Mínimo de Descrição (MDL) [Biernacki et al. 1999], entre outros.

Métodos baseados em operações de divisão e agregação de componentes utilizam diferentes critérios para verificar se cada componente do modelo deve ser dividida ou se duas componentes do modelo devem ser agregadas. Métodos de divisão, geralmente, são baseados em informações de *kurtosis* multivariável, pois um valor de *kurtosis* baixo ou alto é uma indicação que o componente não se ajusta adequadamente aos dados e deve ser dividida [Ververidis and Kotropoulos 2008]. Já métodos de agregação de componentes são baseados no produto interno entre os pesos de duas componentes [Ververidis and Kotropoulos 2008].

O método denominado *Deterministic Annealing EM* (DAEM) [Ueda and Nakano 1998] faz parte do segundo nível, ou seja, métodos que realizam alterações nos passos do Algoritmo EM. Esse método modifica o passo *E-Step* adicionando um parâmetro $1/\beta \in [1, \infty)$, denominado temperatura:

$$p(y_n = k|x_n, \Theta^i) = \frac{(\alpha_k^i p(x_n|\Theta_k^i))^\beta}{(\sum_{c=1}^{C} \alpha_c p(x_n|\Theta_c))^\beta} \tag{17}$$

A medida que $1/\beta$ aumenta, $p(y_n = k|x_n, \Theta^i) \rightarrow 1/C$, ou seja, uma determinada amostra tende a pertencer a todas as componentes com a mesma probabilidade. Assim, as componentes tornam-se similares e a chance de se escapar de um mínimo local é alta. Geralmente, inicia-se $\beta = 0.9$ e roda o algoritmo até sua convergência, em seguida o valor de β é acrescido de 0.05 e o algoritmo é executado novamente, esses passos são repetidos até que $\beta = 1$.

Finalmente, existem diversos métodos que tratam o problema de inicialização dos parâmetros das componentes. Esses podem ser estimados via algoritmos de agrupamento, como o k-médias [Ueda and Nakano 1998], através de técnicas de reamostragem, como bootstrap [Mclachlan 1987], ou aleatoriamente [Figueiredo et al. 2000], em que todas as componentes são inicializadas com pesos iguais a $1/C$, centros iguais a amostras escolhidas aleatoriamente e matrizes de variância iguais a $\sigma^2 I$, sendo I uma matriz identidade de dimensão $D \times D$ e σ^2 é proporcional à covariância amostral de todas as amostras [Figueiredo et al. 2000].

5. Metodologia Proposta

A metodologia proposta nesse trabalho baseia-se em uma modificação do algoritmo gulososo, ou ganancioso, proposto por [Verbeek et al. 2003] para aprendizado de Misturas de Gaussianas. Esse algoritmo utiliza uma busca incremental, realizando operações de divisão de componentes, para estimar o número de componentes do modelo e seus respectivos parâmetros, e com isso resolve o problema de inicialização dos componentes do

modelo, assim como evita a necessidade do conhecimento a priori do número de componentes.

O algoritmo proposto por [Verbeek et al. 2003] pode ser sumarizado pelos seguintes passos:

1. Calcula-se a mistura ótima formada por um componente f_1, setando $k = 1$.
2. Procura-se por uma nova componente $p(x_n; \Theta^*)$ e seu respectivo peso α^* que maximizem:

$$\{\Theta^*, \alpha^*\} = \arg\max_{\Theta,\alpha} \sum_{n=1}^{N} ln\left[(1-\alpha) f_k(x_n) + \alpha p(x_n; \Theta)\right] \tag{18}$$

3. Defini-se $f_{k+1} = (1-\alpha^*) f_k(x) + \alpha^* p(x_n; \Theta^*)$ e $k = k + 1$.
4. Atualiza-se f_k utilizando o algoritmo EM.
5. Se o critério de parada é atingido, sai, senão vai para passo 2.

A mistura ótima, formada por apenas um componente f_1 é estimada via estimador de máxima verossimilhança, utilizando todo o conjunto amostral (passo 1).

A procura pelo novo componente ótimo (passo 2) é realizada através de uma técnica de divisão das componentes já existentes no modelo. Inicialmente, o conjunto de amostras é particionado em k partições disjuntas definidas como $A_c = \{x_n \in x : c = \arg\max_j \{p(y_n = c|x_n; \Theta_c)\}\}$, ou seja, as amostras são relacionadas à componente com maior a posteriori relacionada, utilizando-se a regra de Bayes. Em seguida, para cada partição A_c $(c = 1, \cdots, k)$, m componentes candidatas são construídas. Para isso, duas amostras são escolhidas aleatoriamente na partição, denominadas x_l e x_r, e a partição é reparticionada em dois subconjuntos disjuntos A_{cl} e A_{cr}, sendo que A_{cl} representa as amostras da partição mais próximas de x_l e A_{cr} mais próximas de x_r. Esses subconjuntos são utilizados como componentes candidatas, com média e covariância estimados a partir dos dados e peso igual a $\alpha_c/2$, ou seja, metade do valor do peso da componente original. Essa operação é repetida $m/2$ vezes até que m componentes candidatas sejam construídas. Uma vez geradas as candidatas, para cada uma, é executado o Algoritmo EM parcial, ou seja, executa-se o Algoritmo EM fixando os parâmetros de f_k e otimizando apenas os parâmetros da componente. Assim, a componente candidata que apresentar o maior valor de verossimilhança final, após a execução do Algoritmo EM parcial, é inserida no modelo.

Uma vez que uma nova componente é inserida no modelo, atualiza-se o modelo f_k utilizando o algoritmo EM (passo 3).

O critério de parada (passo 4) é definido como um número máximo, pré-definido, de componentes ou baseado em um critério de parcimônia de modelos, tais como AIC ou MDL. Segundo [Li and Ma 2008], esses critérios tradicionais geralmente resultam em um número errado de componentes. Assim, esse trabalho propõe uma modificação no critério de parada deste algoritmo para utilizar um teste de normalidade multivariado, proposto por [Ververidis and Kotropoulos 2008], de forma que novas componentes são adicionadas no modelo somente se alguma das componentes do presentes rejeitarem a hipótese nula do teste, dado um nível de significância.

5.1. Teste de Normalidade Multivariado

O teste de normalidade multivariado proposto por [Ververidis and Kotropoulos 2008] é baseado na *Distância de Mahalanobis*. Assim, para estabelecer uma hipótese de que um

conjunto de amostras, x, é distribuído de acordo com uma pdf Gaussiana multivariada, inicialmente calcula-se essa distância para todas as amostras:

$$r_n = (x_n - \hat{x})^T S^{-1} (x_n - \hat{x}) \tag{19}$$

onde $\hat{x}$ é a média amostral e S a covariância amostral do conjunto de amostras.

Em seguida, constrói-se a função de distribuição cumulativa (*Cumulative Distribution Function, cdf*) amostral dessa distância baseando-se nas amostras r_n. A cdf amostral da distância, definida como $\hat{F}(r_n)$, é estimada via função de massa, isto é, ordenando os valores de $\{r_n\}_{n=1}^{N}$ em ordem crescente e definindo $\hat{F}(r_n) = n/N$. Define-se também a cdf teórica da distância $F(r_n)$, dado a média $\hat{x}$ e a matriz de covariância S amostrais, assumindo que a distância tem distribuição Beta [Ververidis and Kotropoulos 2008]. Caso N_{r_n} seja definido como o número de amostras dentro de uma elipse com valores equiprováveis de r_n, então N_{r_n} é descrito como uma variável aleatória com distribuição Binomial com parâmetros N e $F(r_n)$:

$$P(N_{r_n} = k) = \left(\begin{array}{c} N \\ k \end{array} \right) F(r_n)^k (1 - F(r_n))^{N-k} \tag{20}$$

dado que $F(r_n)$ é também a probabilidade de se encontrar uma amostra dentro da elipse com *Distância de Mahalanobis* igual a r_n.

Deve-se então definir um intervalo de confiança, denominado $[k^l_{n,\lambda} k^h_{n,\lambda}]$, para N_{r_n}, dado um nível de significância λ, de forma que esse intervalo deve satisfazer:

$$\sum_{k=k^h_{n,\lambda}}^{N} \left(\begin{array}{c} N \\ k \end{array} \right) F(r_n)^k (1 - F(r_n))^{N-k} = \frac{\lambda}{2}$$

$$\sum_{k=0}^{k^l_{n,\lambda}} \left(\begin{array}{c} N \\ k \end{array} \right) F(r_n)^k (1 - F(r_n))^{N-k} = \frac{\lambda}{2}$$

sendo que $\lambda \in \{0.10, 0.05, 0.01\}$.

Caso N seja suficientemente grande e $F(r_n)$ assuma valores próximos de 0 ou 1, ou seja:

$$NF(r_n)(1 - F(r_n)) \gg 1 \tag{21}$$

segundo o Teorema de *Demoivre-Laplace* a distribuição Binomial pode ser aproximada por uma distribuição Gaussiana com média $NF(r_n)$ e variância $NF(r_n)(1 - F(r_n))$. Um valor típico para assumir essa aproximação seria $NF(r_n)(1 - F(r_n)) > 25$ [Papoulis 1984]. Assim, o intervalo é calculado analiticamente como:

$$k^l_{i,\lambda} = \left[NF(r_n) - z_{1-\lambda} \sqrt{2NF(r_n)(1 - F(r_n))} \right]$$

$$k_{i,\lambda}^{h} = \left[NF(r_n) + z_{1-\lambda}\sqrt{2NF(r_n)(1-F(r_n))} \right] \quad (22)$$

em que o operador [] é definido como o inteiro mais próximo do número entre colchetes.

Caso a condição (21) seja violada, o intervalo pode ser estimado numericamente:

$$\begin{aligned} k_{i,\lambda}^{l} &= \arg\min_{k_1} \left| \sum_{k=0}^{k_1} \left(\begin{array}{c} N \\ k \end{array} \right) F(r_n)^k (1-F(r_n))^{N-k} - \frac{\lambda}{2} \right| \\ k_{i,\lambda}^{h} &= \arg\min_{k_2} \left| \sum_{k=k_2}^{N} \left(\begin{array}{c} N \\ k \end{array} \right) F(r_n)^k (1-F(r_n))^{N-k} - \frac{\lambda}{2} \right| \end{aligned} \quad (23)$$

sendo que k_1 e k_2 podem assumir valores no intervalo $[0, 1, \cdots, N]$.

A hipótese nula do teste deve ser validada caso:

$$\begin{aligned} N_{r_n} \in (k_{n,\lambda}^{l},\ k_{n,\lambda}^{h}) \Rightarrow \frac{N_{r_n}}{N} &\in \left(\frac{k_{n,\lambda}^{l}}{N}\ , \frac{k_{n,\lambda}^{h}}{N} \right) \\ \Rightarrow \hat{F}(r_n) &\in \left(\frac{k_{n,\lambda}^{l}}{N}\ , \frac{k_{n,\lambda}^{h}}{N} \right) \end{aligned}$$

$\forall\ n = 1, 2, \cdots, N$.

Assim a hipótese nula H_0 do teste, de que os dados são gerados por uma distribuição Gaussiana multivariada é aceita, dado um nível de significância λ, se:

$$\hat{F}(r_n) \in \left(\frac{k_{n,\lambda}^{l}}{N}\ , \frac{k_{n,\lambda}^{h}}{N} \right) \quad (24)$$

para pelo menos $(1-\lambda)N$ das N amostras.

A figura 1 ilustra os valores teóricos ($F(r_n)$), amostrais ($\hat{F}(r_n)$) e o intervalo de confiança analítico da cdf dos valores de r_n ($N = 200$ e $\lambda = 0.01$) para amostras geradas por um modelo de Mistura de Gaussianas. A figura 1a ilustra o caso de um modelo contendo apenas uma componente, enquanto a figura 1b ilustra um modelo contendo duas componentes com parâmetros distintos. Analisando a figura 1a percebe-se que os valores de $\hat{F}(r_n)$ estão dentro do intervalo de confiança, indicando a normalidade dos dados. Já analisando a figura 1b percebe-se que o número de valores de $\hat{F}(r_n)$ fora dos intervalos de confiança é maior que o limiar, definido como $(1-\lambda)N) = 2$, de forma que a hipótese nula é rejeitada, dado o nível de significância $\lambda = 0.01$.

Em [Ververidis and Kotropoulos 2008] sugere-se os seguintes valores para o nível de significância, dado o tamanho amostral:

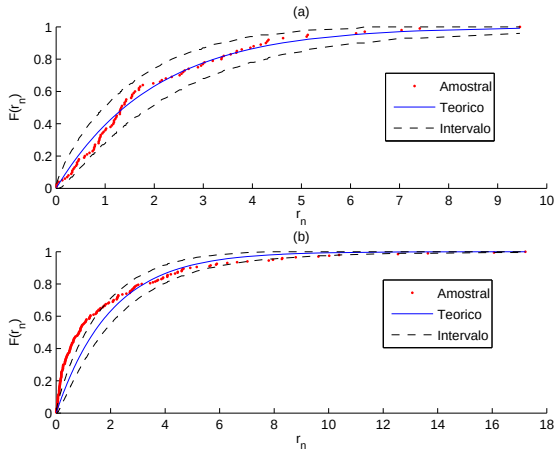


Figura 1. Critério de normalidade multivariado aplicado a Misturas de Gaussianas, contendo uma componente (a) e duas componentes com parâmetros distintos (b)

$$\lambda = \begin{cases} 0.99, & \text{se } N \geq 100 \\ 0.95, & \text{se } 20 \leq N < 100 \\ 0.90, & \text{se } 10 \leq N < 20 \end{cases} \tag{25}$$

Assim, para utilizar esse teste como critério de parada do algoritmo proposto por [Verbeek et al. 2003], inicialmente as amostras são particionadas em partições disjuntas utilizando a metodologia proposta no passo 2 do algoritmo guloso e o teste de normalidade multivariado é aplicado a cada uma das partições. Caso a hipótese nula seja validada para todas as partições, o critério de parada do algoritmo é atingido.

O teste de normalidade proposto possui ordem de complexidade temporal $O(NlogN)$, uma vez que a distância de Mahalanobis deve ser computada para cada amostra e o vetor resultante deve ser ordenado para estimar a cdf amostral.

6. Experimentos

Experimentos foram realizados para verificar se o critério de parada proposto neste trabalho resulta em um ganho na acurácia da seleção do número ótimo de componentes do modelo. Para isso, a metodologia proposta é comparada com critérios de parada baseados em critérios de parcimônia.

Foram utilizados os critérios de parcimônia AIC e MDL:

$$AIC = -\ell(x;\Theta^*) + 2\upsilon \tag{26}$$

$$MDL = -\ell(x;\Theta^*) + \frac{\upsilon}{2} ln(N) \tag{27}$$

sendo $-\ell(x;\Theta^*)$ o valor máximo do logaritmo da verossimilhança e υ é o número de parâmetros livres do modelo, dado por [Ververidis and Kotropoulos 2005]:

$$v = k\left(1 + D + 2D + \frac{D}{2}(1 + D)\right) \tag{28}$$

sendo k o número de componentes e D a dimensão das amostras.

Para os critérios de parcimônia, utilizou-se o critério de parada proposto por [Ververidis and Kotropoulos 2005], de forma que o algoritmo só executa uma nova iteração, caso o valor do critério de parcimônia da iteração corrente seja menor que o valor da iteração anterior, ou seja, para o AIC, o algoritmo finaliza caso:

$$AIC^k - AIC^{k-1} > 0 \tag{29}$$

Foram realizados 200 experimentos, sendo que em cada um gerou-se uma mistura contendo de 2 a 10 Gaussianas de dimensão $D = 2$ e parâmetros distintos e o número de pontos em cada mistura foi ajustado para 15 vezes o número de componentes da mistura. Para cada experimento, o algoritmo foi executado com os três critérios de parada: proposto, AIC e MDL. Para todas as três variações, o número de componentes candidatas por componente do modelo foi ajustado para $m = 6$ e o número máximo de componentes foi definido como 20, de forma que, caso o critério de parada não seja atingido e o modelo já possua 20 componentes, o algoritmo finaliza.

A tabela 1 ilustra os resultados obtidos em relação à acurácia de cada um dos critérios de parada. Nessa tabela são apresentados o número de experimentos em que o número de correto de componentes foi estimado, e a diferença média (média $\pm$ desvio padrão) entre o número correto e o número estimado, para cada um dos critérios de parada.

Tabela 1. Resultados Obtidos nos Experimentos

Método	Experimentos Corretos	Diferença Média
Proposto	55	0.07 ± 1.95
AIC	21	2.88 ± 3.01
MDL	20	2.87 ± 3.02

Analisando a tabela 1, percebe-se que o critério de parada proposto neste trabalho melhora a acurácia do algoritmo, em relação ao número ótimo de componentes, quando comparado com critérios baseados em critérios de parcimônia. O critério proposto é capaz de estimar o número de componentes corretamente cerca de 2.5 vezes mais que os métodos baseados no AIC e MDL. Além disso, a diferença média entre o número de componentes real e estimado para o algoritmo proposto é significativamente menor que a diferença média calculada para os outros métodos.

7. Conclusões

Este trabalho propôs modificações no algoritmo guloso de estimação de Misturas de Gaussianas proposto por [Verbeek et al. 2003], para melhorar sua acurácia, com relação ao número ótimo de componentes do modelo. Para isso, foi utilizado um teste de normalidade multivariável, proposto por [Ververidis and Kotropoulos 2008], para testar as componentes resultantes do modelo no final de cada iteração, de forma que, caso a hipótese

nula do teste seja validada para todas as componentes, o critério de parada do algoritmo é atingido. Os experimentos realizados sugerem uma melhoria na acurácia do algoritmo, quando o critério de parada proposto nesse trabalho é utilizado.

Como sugestões de trabalhos futuros, sugere-se comparar a metodologia proposta com outras metodologias de estimação do número ótimo de componentes. Além disso, o teste de normalidade descrito na seção 5.1 pode ser utilizado também para reduzir o custo computacional do algoritmo guloso, de forma a gerar componentes candidatas apenas a partir de componentes do modelo que rejeitem a hipótese nula do teste.

Agradecimentos

Esse trabalho contou com o apoio financeiro do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e a Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG).

Referências

Biernacki, C., Celeux, G., and Govaert, G. (1999). An improvement of the nec criterion for assessing the number of clusters in a mixture model. *Non-Linear Anal.*, 20(3):267–272.

Bilmes, J. (1998). A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Technical report.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Duda, R., Hart, P., and Stork, D. (2001). *Pattern classification*. John Wiley & Sons, Inc., New York, NY, USA.

Figueiredo, M. A. T., Figueiredo, M. A. T., and Jain, A. K. (2000). Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:381–396.

Li, L. and Ma, J. (2008). A BYY scale-incremental EM algorithm for Gaussian mixture learning. *Applied Mathematics and Computation*, 205(2, Sp. Iss. SI):832–840.

Mclachlan, G. J. (1987). On bootstrapping the likelihood ratio test stastistic for the number of components in a normal mixture. *Applied Statistics*, 36(3):318–324.

Papoulis, A. (1984). *Probability, Random Variables, and Stochastic Processes*. Mc-Graw Hill.

Ueda, N. and Nakano, R. (1998). Deterministic annealing em algorithm. *Neural Netw.*, 11(2):271–282.

Verbeek, J. J., Vlassis, N., and Kröse, B. (2003). Efficient greedy learning of gaussian mixture models. *Neural Comput.*, 15(2):469–485.

Ververidis, D. and Kotropoulos, C. (2005). Emotional speech classification using gaussian mixture models and the sequential floating forward selection algorithm. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 1500–1503.

Ververidis, D. and Kotropoulos, C. (2008). Gaussian mixture modeling by exploiting the mahalanobis distance. *Signal Processing, IEEE Transactions on*, 56(7):2797–2811.

Semi-Supervised Learning in Complex Networks

Thiago C. Silva[1] **and Liang Zhao**[1]

[1]Department of Computer Sciences, Institute of Mathematics
and Computer Science (ICMC), University of São Paulo (USP)

{thiagoch, zhao}@icmc.usp.br

Abstract. *Semi-Supervised Learning (SSL) is a machine learning scheme which is able to employ both labeled and unlabeled samples in the training process. In this paper, we propose a semi-supervised classification model based on a combined random-deterministic walk of particles in the network (graph) constructed from the input data set. The particles of the same class cooperate among them, while the particles of different classes compete with each other to propagate class labels to the whole network. A rigorous definition of the model is provided. An interesting feature of the proposed model is that each particle only visits a portion of nodes potentially belonging to it due to the competition mechanism. Thus, many long range, apparently meaningless visits are avoided. As a result, the proposed model can achieve a good classification rate while exhibiting low computational complexity order in comparison to other network-based semi-supervised algorithms. Computer simulations carried out for synthetic and real-world data sets show good performance of the model.*

1. Introduction

Nowadays, information reaches us at a remarkable speed and the amount of data it brings is unprecedented. In many situations only a small subset of data items can be effectively labeled. This is because the labeling process is often expensive, time consuming, and requires intensive human involvement. As a result, partially labeled data sets become more frequently encountered. Traditional classifiers are constructed by supervised learning, where only labeled data are considered in the training process and unlabeled examples are simply ignored. On the other hand, there is no mechanism for unsupervised learning, such as data clustering, to treat label information. In order to get a better characterization of partially labeled data sets, semi-supervised classifiers are designed to learn from both labeled and unlabeled data. It turned out to be a new topic of machine learning research that has received increasing attention in the past years [Chapelle et al. 2006].

Semi-supervised methods include generative models [Fujino et al. 2005], cluster-and-label techniques [Dara et al. 2002], co-training and tri-training techniques [Mitchell 1999], low-density separation models, like Transductive Support Vector Machines (TSVM) [Vapnik 2008], and graph-based methods. Many SSL techniques, such as TSVM, can identify data classes of well-defined forms, but usually fail to identify classes of irregular forms. Thus, assumptions on class distributions have to be made. Unfortunately, such information is usually unknown a priori. In order to overcome this problem, graph based methods have been developed over the last years, like Local and Global Consistency [Zhou et al. 2004], Local Learning Regularization [Wu and Schölkopf 2007], Local and Global Regularization [Wang et al. 2008]. The main advantage of graph-based

methods is the ability of identifying classes of arbitrary distributions. However, most of the graph-based methods share the regularization framework, differing only in the particular choice of the loss function and the regularizer [Belkin et al. 2005], and most of them have cubic order of computational complexity ($\mathrm{O}(n^3)$). This factor makes their applicability limited to small or middle-sized data sets [Zhu 2005]. As data sets get larger and larger, the development of efficient semi-supervised learning methods is still necessary.

Competition is a natural process observed in nature and in social systems sharing limited resources. Competitive learning is an important category of machine learning and is widely implemented in artificial neural networks to realize unsupervised learning [Kohonen 1990, Jain et al. 2010]. Without a doubt, competitive learning neural networks represent one of the main successes of neural network development. However, at least two problems remain: 1) The constructed network is usually small. So competition occurs among a small number of neurons. Consequently, the model may not exhibit high robustness in data processing. 2) There is not a direct connection between the input data and the trained competitive learning neural networks. When a large data set is mapped to a network of a small number of neurons, it becomes hard to see the correspondence between the original data and the trained neural networks. This is one of the reasons why neural networks sometimes are considered as "black box" systems.

In order to heritage the interesting features and at the same time overcome the problems of competitive learning neural networks, Quiles et. al. [Quiles et al. 2008] proposed a particle walking model to realize the competitive learning mechanism. In this paper, we propose a network-based semi-supervised learning model based on particle competition and cooperation in the network constructed from the input data set. This model not only maintains the competition mechanism presented in Ref. [Quiles et al. 2008], but also introduces a cooperative mechanism. In this way, particles of the same class proceed in the network in a cooperative manner to propagate their labels, while particles of different classes compete with each other to determine the class borders. Another interesting feature of the proposed technique is that it has a local label spreading fashion, i.e., at each time step, each particle spreads its label to a neighbor node chosen by the combined random-deterministic rule. Due to the competition mechanism, each particle only visits a portion of nodes potentially belonging to the current particle or its teammates, while it is not allowed to visit those nodes definitely occupied by other teams of particles. It can be roughly understood that our method has a "divide-and-conquer" effect embedded in the competition-cooperation scheme. In this way, many long-range redundant operations are avoided. As a result, the proposed method has a lower computational complexity order. Since the underlying network is constructed directly from the input data set, the correspondence between the input data and the processing result (the final network) is maintained. As a result, the "black box" effect can be avoided at a large extent.

The remainder of the paper is organized as follows. The proposed model definition is described in Sect. 2. In Sect. 3, computer simulations are performed to show how the proposed model solves data classification by using artificial and real data sets. Finally, Sect. 4 concludes the paper.

2. Model Description

Consider that we are given a graph $G = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges. In the competitive learning model, a set of particles $\mathcal{K} = \{1, ..., K\}$ is inserted into the vertices of the network in a random manner. Each particle can be thought of carrying a flag and its objective is to conquer new territories - represented here by the vertices - while defending its owned territories. In this case, a competition process will naturally take place amongst the particles. When a particle visits an arbitrary vertex, it strengthens its own domination level on the vertex and simultaneously weakens the domination levels of all other rival particles on the same vertex. It is expected to this model, in a broad horizon of time, to end up uncovering the classes in the network in such a way that each particle (or a team of particle through cooperation) dominates a class. In the next sections, we will give detailed explanation on how to derive the competitive dynamical system with the above-mentioned characteristics.

2.1. The Competitive Transition Matrix

Regarding the movement policy of each particle $j \in \mathcal{K}$, it is basically composed of two distinct types: 1) a random movement term, modeled by the matrix $\mathbb{P}^{(j)}_{\text{rand}}$, which permits the particle to adventure through the network, without accounting for the defense of the previously dominated vertices; 2) a deterministic movement term, modeled by the matrix $\mathbb{P}^{(j)}_{\text{det}}$, which is responsible for inducing the particle to reinforce the vertices that are owned by itself, i.e., the particle will prefer visiting its dominated vertices, instead of a randomly selected one. In order to model such dynamics, consider that $p(t) = [p^{(1)}(t), p^{(2)}(t), ..., p^{(K)}(t)]$ denotes the localization of the set of K particles presented to the network, where the jth-entry, $p^{(j)}(t)$, indicates the location of the particle j in the network at time t, i.e., $p^{(j)}(t) \in \mathcal{V}, \forall j \in \mathcal{K}$. It is desired to find a transition matrix that governs the probability distribution of the particles' movement to the immediate future state, $p(t+1) = [p^{(1)}(t+1), p^{(2)}(t+1), ..., p^{(K)}(t+1)]$.

With only those two types of movement behavior, it is possible that the owned territory of each particle to be swapped between them. As there is no force that compels the particles to regress to their owned territory time to time, the particles can be considered free to travel anywhere in the network with no penalties; so, the aforementioned scenario could happen a substantial number of times until the system comes to a stationary state. On account of that, the number of steps that the system would require to converge is expected to be usually high with only these two movement behaviors. In order to overcome that, we introduce energy levels for all particles and, with the aid of this measure, we apply some restrictions on all particles in the following manner: if a particle visits a vertex that is being dominated by itself, then the corresponding energy of that particle increases. Likewise, if a particle visits a vertex that is being dominated by a rival particle, then the corresponding energy of that particle is drained. If the actual energy of a specific particle reaches a certain minimum threshold, then it is said that the particle has died at that step. In the subsequent step, that particle is automatically resurrected in a vertex that belongs to it in a random manner. With this behavior, we expect that the particles will no longer wander free in the network, possibly swapping territories with other particles several times. Thus, this characteristic is expected to restrain the particles' effective acting region. In the semi-supervised version, which we will further detail in this section, it is guaranteed

that there will be at least one vertex that is being dominated by each particle, so that the behavior of teleporting back to its territory can always be applied.

With the intent of modeling such dynamics, we introduce the following random quantity $S^{(t)} = [S^{(1)}(t), \ldots, S^{(K)}(t)]$, where the jth-entry, $S^{(j)}(t) \in \{0, 1\}$, indicates whether the particle j is dead or alive at time t. Specifically, if $S^{(j)}(t) = 1$, then particle j is said to be dead. Likewise, when $S^{(j)}(t) = 0$, the particle is said to be alive. Thus, if $S^{(j)}(t) = 0$, the particle navigates in the network according to a combined behavior of randomness and determinism. However, if $S^{(j)}(t) = 1$, the particle switches its movement policy to a new transition matrix, here entitled $\mathbb{P}^{(j)}_{\text{res}}(t)$, which is responsible for taking the particle back to its owned territory ("safe ground"). In brief terms, $S(t)$ acts as a switch that determines the movement policy of all particles at time t. With all this information in mind, we are able to define the transition matrix associated to the particle j as:

$$\mathbb{P}^{(j)}_{\text{transition}}(t) \triangleq (1 - S^{(j)}(t)) \left[\lambda \mathbb{P}^{(j)}_{\text{det}}(t) + (1 - \lambda)\mathbb{P}^{(j)}_{\text{rand}}(t)\right] + S^{(j)}(t)\mathbb{P}^{(j)}_{\text{res}}(t) \qquad (1)$$

where $\lambda \in [0, 1]$ indicates the desired fraction of deterministic movement that all particles in the network will perform, $\mathbb{P}^{(j)}_{\text{det}}(t)$ portrays the transition matrix with a probability distribution according to the deterministic behavior described above and, likewise, $\mathbb{P}^{(j)}_{\text{rand}}(t)$ describes the random behavior, $S^{(j)}(t)$ indicates whether particle j is alive or dead, and $\mathbb{P}^{(j)}_{\text{res}}(t)$ is responsible for the particle resurrection behavior. It is worth noting that Eq. (1) is a convex combination of two transition matrices (the first term is itself a combination of two transition matrices, too), since the sum of the coefficients is unitary, therefore, the resulting matrix is guaranteed to be another transition matrix. Now we proceed to define each matrix that appears in Eq. (1) in a detailed manner.

The derivation of the random movement matrix is straightforward, since this matrix is only dependent on the adjacency matrix of the graph, which is previously known. Then, each entry $(i, k) \in \mathcal{V} \times \mathcal{V}$ of the matrix $\mathbb{P}^{(j)}_{\text{rand}}(t)$ is given by:

$$\mathbb{P}^{(j)}_{\text{rand}}(i, k) \triangleq \frac{a_{i,k}}{\sum_{u=1}^{V} a_{i,u}} \qquad (2)$$

where $a_{i,k}$ denotes the (i, k)th-entry of the adjacency matrix A of the graph. Note that Eq. (2) resembles the traditional Markovian matrix for a single random walker, here symbolized as a particle. Also note that matrix $\mathbb{P}^{(j)}_{\text{rand}}(t)$ is time-invariant and is the same for every particle in the network, therefore, we will drop the independent variable t and the superscript j whenever the situation makes it clear. In short terms, the probability of an adjacent neighbor to be visited using only the random movement behavior is proportional to the edge weight linking the vertex that a specific particle is visiting and that neighbor vertex.

In order to assist in the calculation of the matrix associated to the deterministic movement term, $\mathbb{P}^{(j)}_{\text{det}}(t)$, for a given particle $j \in \mathcal{K}$, we introduce the following random quantity: $N_i(t) \triangleq [N_i^{(1)}(t), N_i^{(2)}(t), ..., N_i^{(K)}(t)]$, where $\dim(N_i(t)) = 1 \times K$ and $N_i(t)$

stands for the absolute number of visits that vertex i has received until the time t, including t, by all the particles scattered throughout the network. Specifically, the jth-entry, $N_i^{(j)}(t)$, indicates the number of visits made by the particle j to vertex i up to time t. We now simply extend this notation to all vertices in the network, defining the global matrix that maintains the number of visits made by every particle in the network to all the vertices as: $N(t) \triangleq [N_1, N_2, \ldots, N_V]^T$, where $\dim(N(t)) = V \times K$.

Let us also formally define the domination level vector of vertex i, $\bar{N}_i(t)$, according to the following random variable: $\bar{N}_i(t) \triangleq [\bar{N}_i^{(1)}(t), \bar{N}_i^{(2)}(t), ..., \bar{N}_i^{(K)}(t)]$, where $\dim(\bar{N}_i(t)) = 1 \times K$ and $\bar{N}_i(t)$ denotes the relative frequency of visits of all particles in the network to vertex i until the time t, including t. Particularly, the jth-entry, $\bar{N}_i^{(j)}(t)$, indicates the relative frequency of visits performed by particle j to vertex i up to time t. Similarly to the previous case, we extend this notion to all vertices in the network, defining the domination level matrix that sustains all the domination levels imposed by every particle in the network to all the vertices as: $\bar{N}(t) \triangleq [\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_V]^T$, where $\dim(N(t)) = V \times K$. Mathematically, we define each entry of $\bar{N}_i^{(j)}(t)$ as:

$$\bar{N}_i^{(j)}(t) \triangleq \frac{N_i^{(j)}(t)}{\sum_{p=1}^{K} N_i^{(p)}(t)} \tag{3}$$

In view of that, we can define the (i,k)th-entry of the matrix responsible for the deterministic movement behavior of a single particle $j \in \mathcal{K}$, denoted here by $\mathbb{P}_{\text{det}}^{(j)}(t)$, at time t, as following:

$$\mathbb{P}_{\text{det}}^{(j)}(i,k,t) \triangleq \frac{a_{i,k}\bar{N}_k^{(j)}(t)}{\sum_{u=1}^{V} a_{i,u}\bar{N}_u^{(j)}(t)} \tag{4}$$

Clearly, from Eq. (4), it can be observed that each particle has a different transition matrix associated to its deterministic movement and that, unlike the matrix associated to the random movement, this matrix is time-variant with dependence on the domination levels of all the vertices ($\bar{N}(t)$) in the network at the time t. It is worth remarking that the approach taken here to characterize the deterministic movement of the particles is the visiting frequency of each particle to a specific vertex, in such a way that as more visits are performed from a specific particle to an arbitrary vertex, the higher the chance will be of the same particle to come back visiting the same vertex again and again. Also it is important to emphasize that Eq. (4) produces two distinct features presented by a natural competition model: 1) the strengthening of the domination level of the visiting particle to a vertex; 2) the consequent weakening of the domination levels of all other particles on that same vertex. This behavior is naturally represented by the model due to the frequency approach taken.

Now we define each entry of $\mathbb{P}_{\text{res}}^{(j)}(t)$ that is accounted for teleporting a dead particle $j \in \mathcal{K}$ back to its owned territory as:

$$\mathbb{P}_{\text{res}}^{(j)}(i,k,t) \triangleq \frac{\mathbb{1}_{\left\{\max\limits_{m\in\mathcal{K}}\left(\bar{N}^{(m)}_{p^{(j)}(t)}(t)\right)=j\right\}}\bar{N}_k^{(j)}(t)}{\sum_{u=0}^{V}\mathbb{1}_{\left\{\max\limits_{m\in\mathcal{K}}\left(\bar{N}_u^{(m)}(t)\right)=j\right\}}\bar{N}_u^{(j)}(t)} \tag{5}$$

where $\max\limits_{m\in\mathcal{K}}(.)$ returns the index m which maximizes the argument and $\mathbb{1}_{\{.\}}$ is the indicator functions that yields 1 if the argument is logically true and 0, otherwise. Indeed, a careful analysis of the expression in Eq. (5) shows that the denominator term sums over all dominations levels of all the vertices that are being dominated by particle j, i.e., vertices that are dominated by rival particles do not have their values considered. With that in mind, Eq. (5) only results in non-zero transition probabilities for vertices k that are being dominated by particle j, regardless of the existence of a connection between i and k in the adjacency matrix. In essence, once the particle is dead, the switch is enabled, which in turn compels the particle j to return to its previously owned territory, no matter if there is a physical connection or not in the adjacency matrix.

Now we proceed to the development of the particle's energy update rule. Firstly, it is useful to introduce the random variable $E(t) = [E^{(1)}(t), \ldots, E^{(K)}(t)]$, where the jth-entry, $E^{(j)}(t) \in [\omega_{\min}, \omega_{\max}]$, $\omega_{\max} \geq \omega_{\min}$, denotes the energy level of particle j at time t, whose update rule is given by:

$$E^{(j)}(t) = \begin{cases} \min(\omega_{\max}, E^{(j)}(t-1) + \Delta), & \text{if } \text{owner}(j,t) \\ \max(\omega_{\min}, E^{(j)}(t-1) - \Delta), & \text{if } \lnot\, \text{owner}(j,t) \end{cases} \tag{6}$$

where $\text{owner(j, t)} = \left(\max\limits_{m\in\mathcal{K}}\left(\bar{N}^{(m)}_{p^{(j)}(t)}(t)\right) = j\right)$ is a logical expression that essentially yields true if the vertex that particle j visits at time t (vertex $p^{(j)}(t)$) is being dominated by the visiting particle, and false otherwise; $\dim(E(t)) = 1 \times K$; $\Delta > 0$ symbolizes the increment or decrement of energy that each particle will receive at time t. Indeed, the first expression in Eq. (6) represents the increment of the particle's energy and occurs when the particle j visits a vertex $p^{(j)}(t)$ which is dominated by itself, i.e., $\max\limits_{m\in\mathcal{K}}\left(\bar{N}^{(m)}_{p^{(j)}(t)}(t)\right) = j$. Similarly, the second expression in Eq. (6) portrays the decrement of the particle's energy and occurs when particle j visits a vertex $p^{(j)}(t)$ which is not dominated by itself, i.e., there is a domination level on that vertex that is higher than the one imposed by particle j. Hence, in this model, particles will be given a penalty if they are wandering in rival territory, so as to minimize aimless navigation of the particles in the network which would only reduce the speed of convergence of the dynamical system. By the same reasons, we expect this behavior to improve the final classification rate of the algorithm.

Now we advance to the update rule that governs $S(t)$, which is responsible for determining the movement policy of each particle. As we have stated, an arbitrary particle j will be transported back to its domain only if its energy drops to a threshold $\omega_{\min}$. With that in mind, it is natural that each entry of $S^{(j)}(t)$ has to monitor the current energy value of its corresponding particle, i.e., if it ever drop to the given threshold, the switch must

be enabled; analogously, if the particle still has an energy value greater than this lower threshold, then the switch should be disabled. Mathematically, the jth-entry of $S(t)$ can be precisely written as:

$$S^{(j)}(t) = \mathbb{1}_{\{E^{(j)}(t)=\omega_{\min}\}} \tag{7}$$

where $\dim(S(t)) = 1 \times K$. Specifically, $S^{(j)}(t) = 1$ if $E^{(j)}(t) = \omega_{\min}$ and 0, otherwise. As there is an upper limit for the random variable $E^{(j)}(t)$, it is clear that if particle j frequently visits vertices owned by rival particles, its energy will decrease in such a way that it could reach the minimum energy $\omega_{\min}$ and, hence, die. The upper limit, $\omega_{\max}$, was established to prevent any particle in the network to keep increasing its energy to an undesirably high value (by constantly visiting vertices in its territory), and, once this energy is high enough, it could go far away from its territory and visit a substantial number of vertices belonging to rival particles before dying, thus, considerably decreasing the convergence time and classification rate of the dynamical system.

Having defined each matrix associated to each particle, we couple all these matrices into a representative transition matrix $\mathbb{P}_{\text{transition}}(t)$ using the following fact: when a specific particle is alive, the movement of each particle is independent of all the other rival particles, because at each time step the next vertex to be visited by a specific particle depends solely on a convex combination of randomness - directly proportional to the edge weight of the neighbor vertices, see Eq. (2) - and determinism - in this case, is a function of the domination levels of all the vertices in the neighborhood, see Eq. (4). Due to that, the location of the other particles causes no influence over the action of choosing the next vertex to be visited by a given particle. The same idea can be applied when a specific particle is dead. In virtue of this property, we can describe the full matrix associated to the transition $p(t)$ to $p(t+1)$, i.e., for all particles in the network, as:

$$\mathbb{P}_{\text{transition}}(t) = \mathbb{P}^{(1)}_{\text{transition}}(t) \otimes \ldots \otimes \mathbb{P}^{(K)}_{\text{transition}}(t) \tag{8}$$

where $\otimes$ denotes the Kronecker tensor product operator. In this way, Eq. (8) completely specifies the transition distribution matrix for *all* the particles in the network and not only one particle as is expressed in Eq. (1).

Essentially, $p(t+1)$ can be seen as a discrete stochastic process whose probability distribution is given by the row indicated by the scalar form of $p(t)$ (to be defined) of matrix $\mathbb{P}_{\text{transition}}(t)$, which strictly depends on $N(t)$ for its construction. Upon doing so, we need to enumerate the states of the particle localization $p(t)$ such as to be feasible the usage of the transition matrix, because, for $K \geq 2$, $p(t)$ will be a vector and we would no longer be able to conventionally define row $p(t)$ of $\mathbb{P}_{\text{transition}}(t)$. For that, we simply enumerate the state $p(t) = [p^{(1)}(t), p^{(2)}(t), ..., p^{(K)}(t)]$ to a scalar form, respecting the natural ordering of the tuples, i.e., $p(t) = [1, 1, ..., 1, 1]$ (all particles at vertex 1) denotes the first state; $p(t) = [1, 1, ..., 1, 2]$ (all particles at vertex 1, except the last particle, which is at vertex 2) is the second state; and so on, up to the scalar state V^K.

2.2. The Semi-Supervised Competitive Learning Model

In light of all we have obtained in the previous section, we are ready to enunciate the proposed dynamical system which models the competition of particles in a given network. The internal state of the dynamical system has been chosen to be: $X(t) = [N(t)\ \ p(t)\ \ E(t)\ \ S(t)]^T$ and the proposed competitive dynamical system is given by:

$$\phi : \begin{cases} N_i^{(j)}(t+1) & = N_i^{(j)}(t) + \mathbb{1}_{\{p^{(j)}(t+1)=i\}} \\ E^{(j)}(t+1) & = \begin{cases} \min(\omega_{\max}, E^{(j)}(t) + \Delta), \text{if owner}(j,t) \\ \max(\omega_{\min}, E^{(j)}(t) - \Delta), \text{if } \lnot \text{ owner}(j,t) \end{cases} \\ S^{(j)}(t+1) & = \mathbb{1}_{\{E^{(j)}(t+1)=\omega_{\min}\}} \end{cases} \tag{9}$$

where, by the considerations that we have previously stated, $\dim(N(t)) = V \times K$, $\dim(p(t)) = 1 \times K$, $\dim(E(t)) = 1 \times K$, and $\dim(S(t)) = 1 \times K$, resulting that $\dim(X(t)) = (V+3) \times K$, with $N_i^{(j)}(t) \in [1, \infty), (i,j) \in \mathbb{S}$, where $\mathbb{S}$ is the space spawned by $\mathcal{V} \times \mathcal{K}$. Observe that $p(t+1)$ has no closed form because it is qualified as a distribution with dependence on $p(t)$ and $N(t)$, therefore its acquisition is merely by random number generation. Succinctly, the internal state of system ϕ, $X(t)$, carries the current total number of visits made by each particle to each vertex in the network, the current localization of all particles in the network, the current energy that each particle holds, and the information about each particle whether it is current alive or dead. The first equation of system ϕ is responsible for updating the number of visits at vertex i by particle j up to time t; the second equation is used to maintain the current energy levels of all the particles inserted in the network; and the third equation is used to trigger the particle dead or alive, depending on its actual energy level. It is valuable to emphasize that the first expression of system ϕ must be used for every $(i,j) \in \mathbb{S}$ and the second and third expressions must be performed for every $j \in \mathcal{K}$ with the intention of one properly derive the full state $X(t)$ of the system ϕ.

2.3. The Initial Conditions of the System

In order to run system ϕ, we need a set of initial conditions. Firstly, the initial position $p(0)$ of the particles is user-controllable. Additionally, consider a set of classes $\mathcal{C}$ and a set of pre-labeled examples $\mathcal{V}_{\text{L}} \subset \mathcal{V}$. Let $\mathcal{L}$ denote the set that indicates the pair of a labeled node and its corresponding class, i.e., $\mathcal{L} = \{(v_1, c_1), ..., (v_{|\mathcal{V}_{\text{L}}|}, c_{|\mathcal{V}_{\text{L}}|})\}$, where $v_i \in \mathcal{V}_{\text{L}}$, and $c_i \in \mathcal{C}$, $0 \leq i \leq |\mathcal{L}| = |\mathcal{V}_{\text{L}}|$. Then, we have that each entry of $N(0)$ is given by:

$$N_i^{(j)}(0) = \begin{cases} \infty, & \text{if particle j represents node i} \\ 1 + \mathbb{1}_{\{p^{(j)}(0)=i\}}, & \text{otherwise} \end{cases} \tag{10}$$

where we apply Eq. (10) to every $(i,j) \in \mathbb{S}$. Note that the scalar 1 was used in the second expression of Eq. (10) in order to unlabeled and not visited vertices at time 0 to have their calculation well-defined, according to Eq. (3). Regarding the initial condition of $E(0)$, we desire a fair competition amongst the particles, so we place isonomy in their initial energy values, i.e., all particles $j \in \mathcal{K}$ start out with the same energy level given by:

$$E^{(j)}(0) = \omega_{\min} + \left(\frac{\omega_{\max} - \omega_{\min}}{K} \right) \tag{11}$$

Lastly, the variable that accounts for indicating whether the particle j is dead or alive at the initial step, $S^{(j)}(0)$, $\forall j \in \mathcal{K}$, is given by $S^{(j)}(0) = 0$, i.e., we deliberately set alive all particles in the network in the beginning of the process.

3. Computer Simulations

In this section, we present simulation results in order to show the effectiveness of the proposed competitive model. Specifically, in Subsect. 3.1 we analysis the behavior of the dynamical system ϕ by using a simple artificial network; in Subsect. 3.2, we provide results of semi-supervised learning on real data sets as well as a comparison against several well-known semi-supervised learning techniques.

3.1. Simulations on Synthetic Data Sets

In order to facilitate the understanding of how the proposed technique works, we design a synthetic data set of two simple classes, each of which with 50 vertices. We have inserted into the network 2 particles, each one representing a class. With this simple data set, we are able to closely observe the behavior of our proposed algorithm working on synthetic data. Figure 1a shows the initial configuration of the network, where the colored dots symbolize labeled samples. The black dots denote unlabeled data. The ownership of any vertex is given by the particle which has the highest domination level over that vertex. According to Eq. (10), the vertices that are initially labeled have their ownership fixed to its corresponding representative particle. For this simulation, we fix $\lambda = 0.6$, $\Delta = 0.05$, $\omega_{\min} = 0$, and $\omega_{\max} = 1$. As the dynamical system evolves, the particles will visit the vertices of the network in agreement with the probability distribution given by the matrix $\mathbb{P}_{\text{transition}}(t)$. Figure 1b shows the ownership of each vertex after 300 iterations, Fig. 1c depicts the ownership state after 600 iterations, and Fig. 1d reveals the stationary ownership state of the algorithm $\bar{N}(t)$, which is reached after 3000 iterations. We now take a closer look at the evolutional behavior of the average domination level of the vertices that belong to the same class. Figure 2a provides the average domination level imposed by the particle representing the initially blue labeled vertex on the vertices 1 to 50 (blue class) and 51 to 100 (red class), whereas Fig. 2b displays the same information for the particle representing the initially red labeled vertex. Clearly, as time progresses, one can see that the classes are unmistakeably separated by the competitive system.

3.2. Simulations on Benchmark Data Sets

In order to measure the performance of the proposed method, we have applied it to 7 standard semi-supervised data sets. For a detailed description of the data set, refer to [Chapelle et al. 2006]. For each data set in the benchmark, it is provided 12 benchmark partitions,

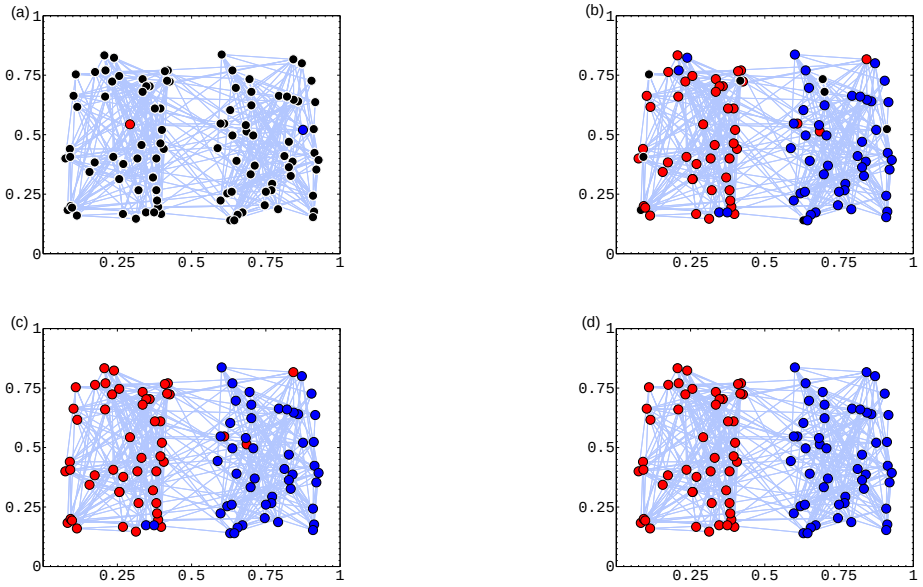


Figure 1. Illustration of an artificial classification process through competitive particle walking. The total number of nodes is $V = 100$. (a) A snapshot of the initial configuration: there are two previously labeled nodes (red and blue nodes) and $K = 2$ particles, each one representing a labeled node. The black dots represent unlabeled data. The particles have been spawned at their representative labeled node. (b) A snapshot at iteration 300. (c) A snapshot at iteration 600. (d) A snapshot at iteration 3000.

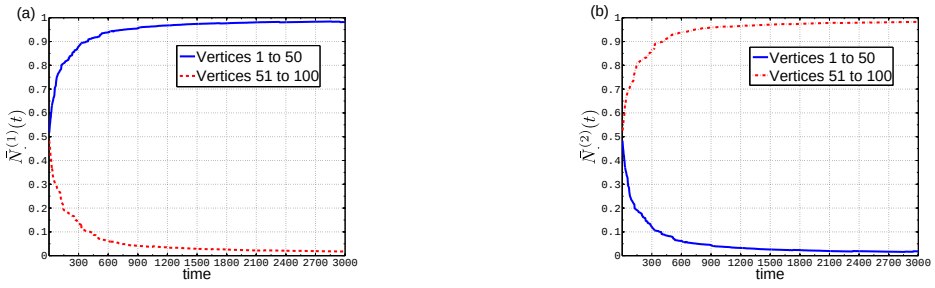


Figure 2. Evolutional behavior of the average class domination level imposed by the particles in the network. (a) Average class domination level imposed by particle 1. (b) Average class domination level imposed by particle 2.

each of which having 10 distinct non-biased labeled vertices. In this partition composition, it is ensured that exists at least 1 labeled node for each class in the database. For comparison matters, we have conducted experiments against a selected set of semi-supervised techniques, whose simulations results were performed under optimal conditions and were integrally taken from [Chapelle et al. 2006]. For comparison purposes, we have used the kNN graph formation technique with optimized k in the interval $1 \leq k < 30$ and have also optimized the algorithm parameters in the intervals: $0 \leq \lambda \leq 1$, $0 < \Delta < 1$, $0 < \omega_{\max} < 5$ with $\omega_{\min} = 0$ fixed. The optimization is realized by using the genetic algorithm available in the Global Optimization Toolbox of MATLAB with its default pa-

Table 1. Test errors (%) with 10 labeled training points and the corresponding average rank of each technique.

Technique	g241c	g241d	Digit1	USPS	COIL	BCI	Text	Avg. Rank
1-NN	47.88	46.72	13.65	16.66	63.36	49.00	38.12	**9.86**
Support Vector Machines	47.32	46.66	30.60	20.03	68.36	49.85	45.37	**14.14**
Maximum Variance Unfolding	47.15	45.56	14.42	23.34	62.62	47.95	45.32	**9.86**
Laplacian Eigenmaps	44.05	43.22	23.47	19.82	65.91	48.74	39.44	**10.00**
Quadratic Criterion and Class Mass Regularization	39.96	46.55	9.80	13.61	59.63	50.36	40.79	**7.86**
Discrete Regularization	49.59	49.05	12.64	16.07	63.38	49.51	40.37	**10.86**
Transductive Support Vector Machines	24.71	50.08	17.77	25.20	67.50	49.15	31.21	**10.86**
Spectral Graph Transducer	22.76	18.64	8.92	25.36	-	49.59	29.02	**6.50**
Cluster Kernels	48.28	42.05	18.73	19.41	67.32	48.31	42.72	**10.86**
Data-Dependent Regularization	41.25	45.89	12.49	17.96	63.65	50.21	-	**9.83**
Low-Density Separation	28.85	50.63	15.63	17.57	61.90	49.27	27.15	**8.43**
Laplacian Regularized Least Squares	43.95	45.68	5.44	18.99	54.54	48.97	33.68	**6.14**
Conditional Harmonic Mixing	39.03	43.01	14.86	20.53	-	46.90	-	**7.20**
Local and Global Consistency	45.82	44.09	9.89	9.03	63.45	47.09	45.50	**7.29**
Label Propagation	42.61	41.93	11.31	14.83	55.82	46.37	49.53	**5.57**
Linear Neighborhood Propagation	47.82	46.24	8.58	17.87	55.50	47.65	41.06	**7.43**
Proposed Method	43.89	46.47	8.10	15.69	54.18	48.00	34.84	**5.29**

rameters. The values obtained for the proposed method are averaged by 100 realizations on each of the 12 subsets. The results obtained from these techniques against the aforementioned databases are reported in Table 1. From the same table, we can conclude that our technique had a satisfactory result in comparison to the other techniques. With only a few nodes, our technique was able to correctly spread the labels to the vicinity of nodes with the aid of the competitive mechanism provided by the algorithm. This is an attractive characteristic, since the task of vertex labeling is often expensive and cumbersome, which generally involves the work of a human expert. In order to statistically verify whether the algorithms presented in Table 1 have significant difference to each other, we apply the Friedman Test with a significance level of 0.05. See [Demšar 2006] for details. We get that the critical statistical descriptor is lower than the quantity provided by the data itself, therefore, the null hypothesis is rejected and we confirm that the algorithms under analysis present significant difference. Specifically, we can see that our algorithm is superior to the others for the set of databases that we have conducted our tests.

4. Conclusions

This paper proposes a nonlinear and stochastic mathematical model for competitive learning in complex networks, biologically inspired by the competition process taking place in many nature and social systems. In this model, several particles, each of which representing a class, navigate in the network to explore their territory and, at the same time, attempt to defend its dominion against rival particles. If several particles propagate the same class label, then a team is formed, and a cooperation process amongst these particles occurs. A confinement mechanism was proposed in order to prevent the particles to

wander free in the network, possibly reducing the overall classification rate of the algorithm and its speed of convergence. Consequently, the proposed technique spreads labels in a local fashion, instead of the traditional semi-supervised techniques which propagates labels in a global fashion. Simulations were carried out with the purpose of quantifying the robustness of the proposed technique on synthetic and real-world data sets for the task of data classification and reasonable results have been obtained. More importantly, this work is an attempt to provide an alternative way to the study of competitive learning.

References

Belkin, M., P., N., and Sindhwani, V. (2005). On manifold regularization. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTAT 2005)*, pages 17–24, New Jersey. Society for Artificial Intelligence and Statistics.

Chapelle, O., Schölkopf, B., and Zien, A., editors (2006). *Semi-supervised Learning*. Adaptive computation and machine learning. MIT Press, Cambridge, MA, USA.

Dara, R., Kremer, S., and Stacey, D. (2002). Clustering unlabeled data with SOMs improves classification of labeled real-world data. In *Proceedings of the World Congress on Computational Intelligence (WCCI)*, pages 2237–2242.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30.

Fujino, A., Ueda, N., and Saito, K. (2005). A hybrid generative/discriminative approach to semi-supervised classifier design. In *AAAI-05, Proceedings of the Twentieth National Conference on Artificial Intelligence*, pages 764–769.

Jain, L. C., Lazzerini, B., and (eds.), U. H. (2010). *Innovations in ART Neural Networks (Studies in Fuzziness and Soft Computing)*. Physica-Verlag, Heidelberg.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480.

Mitchell, T. M. (1999). The role of unlabeled data in supervised learning. In *Proceedings of the Sixth International Colloquium on Cognitive Science*.

Quiles, M. G., Zhao, L., Alonso, R. L., and Romero, R. A. F. (2008). Particle competition for complex network community detection. *Chaos*, 18(3):033107.

Vapnik, V. N. (2008). *Statistical Learning Theory*. Wiley-Interscience, New York.

Wang, F., Li, T., Wang, G., and Zhang, C. (2008). Semi-supervised classification using local and global regularization. In *AAAI'08: Proceedings of the 23rd national conference on Artificial intelligence*, pages 726–731. AAAI Press.

Wu, M. and Schölkopf, B. (2007). Transductive classification via local learning regularization. In *11th International Conference on Artificial Intelligence and Statistics*, pages 628–635. Microtome.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. (2004). Learning with local and global consistency. In *Advances in Neural Information Processing Systems*, volume 16, pages 321–328. MIT Press.

Zhu, X. (2005). Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison.

Agrupamento de Dados Baseado em Dinâmica de Troca de Energia

Roberto Alves Gueleri[1], **Zhao Liang**[1]

[1]Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo
São Carlos – SP – Brasil

{gueleri,zhao}@icmc.usp.br

Abstract. *We present a data clustering method based on dynamics of energy exchange. Initially, each object is assigned an energy state. So the process consists of the progressive exchange of energy among objects. When two of them reach close enough energy states, they are grouped together. The distance between each pair of objects defines its degree of interaction. Thus, it tends to gradually cluster the regions of higher density of objects. The natural and dynamic property of the proposed method shows interesting results. Simulations with artificial and real data show its potentialities and limitations.*

Resumo. *Apresentamos um método de agrupamento de dados baseado em dinâmica de troca de energia. No início, a cada objeto é atribuído um estado de energia. O processo de agrupamento consiste então na gradativa troca de energia entre os objetos. Quando dois deles atingem estados de energia suficientemente semelhantes, eles são agrupados. A distância que separa cada par de objetos define seu grau de interação. Assim, esse método tende a agrupar gradualmente as regiões com maior densidade de objetos. O caráter natural e dinâmico desse método mostra resultados interessantes. Simulações com dados artificiais e reais procuram mostrar as potencialidades e limitações desse método.*

1. Introdução

A tarefa do agrupamento de dados (*clustering*) pode ser entendida como: dado um conjunto não-classificado de objetos, encontrar a melhor partição desse conjunto, de modo que objetos de um mesmo grupo sejam mais semelhantes que objetos de grupos distintos [Mitchell 1997, Jain et al. 1999, Alpaydin 2004, Xu and Wunsch 2005, Zhao et al. 2005, Bishop 2006, Hastie et al. 2009]. Os métodos de agrupamento de dados costumam ser classificados em *hierárquicos* e *particionais* [Jain et al. 1999, Xu and Wunsch 2005]. Um método hierárquico agrupa os objetos segundo uma sequência de partições, desde aquela que atribui um *cluster* distinto para cada objeto até aquela que confina todos os objetos num único *cluster*. Já um método particional divide diretamente o conjunto de objetos num número pré-especificado de *clusters*. Os métodos de agrupamento hierárquico são, por sua vez, classificados em *aglomerativos* e *divisivos* [Jain et al. 1999, Xu and Wunsch 2005]. A essencial

diferença entre eles é a ordem da sequência de partições. Enquanto os métodos aglomerativos iniciam com uma partição que atribui um *cluster* distinto para cada objeto, os métodos divisivos seguem o caminho contrário, iniciando com uma partição que confina todos os objetos num único *cluster*.

Tipicamente, o agrupamento de dados é um problema combinatoriamente difícil [Jain et al. 1999, Xu and Wunsch 2005, Abraham et al. 2007], sendo comumente conhecido como um problema de otimização. A inspiração na natureza, além da sua intrínseca elegância, tem sido uma prática valiosa para lidar com problemas computacionais complexos, combinatoriamente difíceis [Castro 2007]. Blatt et al. (1997) desenvolveram um método de agrupamento de dados baseado nas propriedades físicas de um ferromagneto heterogêneo. Um *spin* é atribuído a cada objeto, de modo que cada um desses *spins* será influenciado pelos *spins* dos objetos vizinhos. Ao final, a correlação entre os *spins* dirá quais objetos pertencem a quais *clusters*. Abraham et al. (2007), em seu trabalho intitulado *Swarm Intelligence Algorithms for Data Clustering*, mostram aplicações bem-sucedidas de técnicas bioinspiradas em problemas de agrupamento de dados. Mais especificamente, o trabalho trata da otimização por colônia de formigas e da otimização por nuvem de partículas. Ambas as abordagens fundamentam-se na inteligência que emerge da interação de muitos indivíduos relativamente simples. Oliveira (2008) também aplica com sucesso a técnica de otimização por nuvem de partículas a problemas de agrupamento de dados. Zhao et al. (2005) propuseram um método de agrupamento dinâmico e auto-organizável baseado na atração de objetos similares. A evolução do processo de agrupamento consiste em gradativamente concentrar os grupos de objetos em pontos, que são considerados os centros de cada *cluster*.

O presente trabalho propõe um método de agrupamento de dados baseado em dinâmica de troca de energia. Trata-se de um método hierárquico e aglomerativo. No início, a cada objeto é atribuído um estado de energia. O processo de agrupamento consiste então na gradativa troca de energia entre os objetos. Quando dois deles atingem estados de energia suficientemente semelhantes, eles são agrupados. Não pretende-se, contudo, modelar fielmente as leis físicas que regem os processos naturais de troca de energia. Pelo contrário, essas "leis" são justamente parâmetros a serem modificados. Assim, este trabalho objetiva investigar o comportamento dinâmico do método proposto, observando, para diferentes parâmetros e diferentes conjuntos de dados, como os *clusters* são formados durante o processo de agrupamento. A Seção 2 descreve esse método. A Seção 3 mostra algumas simulações realizadas.

2. O método de agrupamento de dados baseado em dinâmica de troca de energia

Duas são as entidades envolvidas no método aqui descrito: *objetos* e *clusters*. Os objetos são os elementos a serem agrupados, enquanto os *clusters* são justamente os grupos de objetos. Cada *cluster* é caracterizado por um *estado de energia*, significando que cada um de seus objetos compartilham esse estado de energia. O processo de agrupamento é então caracterizado pela progressiva troca de energia entre os *clusters*, agrupando-os conforme atingem estados de energia suficientemente semelhantes.

Seja *natt* a quantidade de atributos que caracterizam cada objeto. Assim, cada objeto $\mathbf{o}_i \in \mathbb{R}^{natt}$ é definido como

$$\mathbf{o}_i = \Big(o_{i,1}, o_{i,2}, o_{i,3}, \ldots, o_{i,natt}\Big) \tag{1}$$

A evolução do processo de agrupamento dá-se através de sucessivas iterações, denotadas pela variável $t \in \mathbb{N}$, tal que $t = 0$ refere-se ao estado inicial do sistema. A cada iteração, os *clusters* (representando seus objetos) trocam certa quantidade de energia entre si. Quando dois ou mais atingem estados de energia suficientemente semelhantes, eles combinam-se, originando um *cluster* maior. A essa combinação dá-se o nome de *fusão*. Assim, cada iteração t é dividida em dois sub-estados: *pré-fusão* e *pós-fusão*.

Seja *nobj* a quantidade de objetos envolvidos no problema. Denota-se por $\mathbf{c}'_i(t)$ e $\mathbf{c}_i(t) \in \mathbb{R}^{nobj}$ os *clusters* na pré-fusão e na pós-fusão de t, respectivamente.

$$\mathbf{c}'_i(t) = \Big(e'_{i,1}(t), e'_{i,2}(t), e'_{i,3}(t), \ldots, e'_{i,nobj}(t)\Big) \tag{2}$$

$$\mathbf{c}_i(t) = \Big(e_{i,1}(t), e_{i,2}(t), e_{i,3}(t), \ldots, e_{i,nobj}(t)\Big) \tag{3}$$

Aqui, as componentes $e'_{i,j}(t)$ e $e_{i,j}(t)$ representam o estado de energia do *cluster*. Denotam-se também por $C'(t)$ e $C(t)$ os conjuntos de todos os *clusters* existentes na pré-fusão e na pós-fusão de t, respectivamente.

Todo *cluster* está associado a seus objetos. Assim, definem-se *cluster*$'$, *cluster* e *obj* como sendo as relações entre os *clusters* e seus respectivos objetos. Tais relações são definidas reciprocamente como

$$cluster'(\mathbf{o}_i, t) = \mathbf{c}'_j(t) \iff \mathbf{o}_i \in obj(\mathbf{c}'_j(t)) \tag{4}$$

$$cluster(\mathbf{o}_i, t) = \mathbf{c}_j(t) \iff \mathbf{o}_i \in obj(\mathbf{c}_j(t)) \tag{5}$$

Ou seja, *cluster*$'$ e *cluster* resultam no *cluster* ao qual pertence um certo objeto. Já *obj* representa a relação inversa, resultando nos objetos que compõem um certo *cluster*.

Inicialmente, na pré-fusão de $t = 0$, atribui-se um *cluster* distinto a cada objeto. Têm-se pois *nobj clusters*. A cada um desses *clusters* é então atribuído um estado inicial de energia:

$$e'_{i,j}(0) = \begin{cases} 0 & \text{para } i \neq j \\ 1 & \text{para } i = j \end{cases} \tag{6}$$

A disposição simétrica assim concebida garante que nenhum dos objetos seja eventualmente privilegiado. Disso decorre a necessidade de tantas componentes de energia quantos são os objetos envolvidos no problema.

Descreve-se agora o processo de fusão. Essencialmente, há dois tipos de fusão: um para $t = 0$ e outro para $t > 0$. Após a disposição inicial dos *clusters* em $t = 0$, é conveniente já agrupar objetos situados muito perto um do outro. Denomina-se essa proximidade por *omin*, tal que objetos cuja distância euclidiana seja menor ou igual

a *omin* são agrupados. Essa é a fusão para $t = 0$. Já para $t > 0$, procura-se, a cada iteração, por *clusters* cujos estados de energia sejam suficientemente semelhantes. Denomina-se essa semelhança por *emin*, de modo que *clusters* cujos estados de energia tenham distância euclidiana menor ou igual a *emin* são agrupados.

A fim de realizar a fusão, é necessário criar uma partição do conjunto $C'(t)$, tal que os *clusters* contidos numa mesma parte sejam todos agrupados entre si. A partição $P(t)$ procurada é definida como

$$P(t) = \Big\{ M_m(t) \subseteq C'(t) \Big\}, \quad m = 1 \ldots \text{tamanho da partição} \tag{7}$$

onde $M_m(t)$ representa cada parte da partição. Todo *cluster* estará em uma e somente uma parte. Havendo dois *clusters* tal que a distância entre seus respectivos objetos seja menor ou igual a *omin* ($t = 0$), ou a distância entre seus estados de energia seja menor ou igual a *emin* ($t > 0$), esses *clusters* estarão na mesma parte e serão agrupados.

Define-se $merge(M_m(t))$ como sendo uma operação capaz de tomar uma parte $M_m(t)$ de alguma partição e transformá-la num novo *cluster*, resultado da fusão dos *clusters* de $M_m(t)$:

$$merge(M_m(t)) = \begin{cases} \dfrac{\displaystyle\sum_{\mathbf{c}'_i(t) \in M_m(t)} \mathbf{c}'_i(t)}{|M_m(t)|} & \text{para } t = 0 \\ \dfrac{\displaystyle\sum_{\mathbf{c}'_i(t) \in M_m(t)} \Big[\mathbf{c}'_i(t) \cdot |obj(\mathbf{c}'_i(t))| \Big]}{\displaystyle\sum_{\mathbf{c}'_i(t) \in M_m(t)} |obj(\mathbf{c}'_i(t))|} & \text{para } t > 0 \end{cases} \tag{8}$$

onde $|M_m(t)|$ e $|obj(\mathbf{c}'_i(t))|$ referem-se, respectivamente, ao tamanho da parte $M_m(t)$ e ao tamanho do conjunto $obj(\mathbf{c}'_i(t))$ de objetos do *cluster* $\mathbf{c}'_i(t)$. Nota-se que, para $t = 0$, o *cluster* resultante possui estado de energia igual à média dos estados de energia dos *clusters* que entraram em fusão. Para $t > 0$, o estado de energia resultante é igual à média ponderada pelo tamanho dos *clusters*, ou seja, quanto mais objetos um *cluster* possui, mais influente ele é. Na verdade, a primeira expressão, para $t = 0$, é um caso especial da segunda, visto que em $t = 0$ todo *cluster* tem tamanho igual a 1. Então, o novo conjunto $C(t)$ resultante da fusão dos *clusters* em $C'(t)$, agora organizados na partição $P(t)$, é

$$C(t) = \Big\{ merge(M_m(t)) \Big\}, \quad m = 1 \ldots \text{tamanho da partição} \tag{9}$$

Passa-se agora ao processo de troca de energia. Define-se $neighborhood(\mathbf{o}_i, t)$ como sendo o conjunto dos *nneighbors* $\mathbf{o}_j$ mais próximos de $\mathbf{o}_i$, que não façam parte do mesmo *cluster* de $\mathbf{o}_i$ na pós-fusão da iteração t. O parâmetro *nneighbors* simplesmente limita a quantidade de vizinhos considerados na troca de energia. $neighborhood(\mathbf{o}_i, t)$ pode resultar num conjunto menor que *nneighbors*, caso a quantidade de objetos não seja mais suficiente. Observar que não faz sentido a interação

entre objetos contidos num mesmo *cluster*, uma vez que já compartilham o estado de energia.

Então, define-se uma função chamada *resultant* que informa a variação do estado de energia que cada *cluster* "deseja" ter entre os instantes t e $t+1$. Mais precisamente, trata-se de um par de funções, uma para os *clusters*:

$$resultant(\mathbf{c}_i(t)) = \Big(resultant(e_{i,1}(t)), \ldots, resultant(e_{i,nobj}(t))\Big) \tag{10}$$

e outra para as componentes dos *clusters*:

$$resultant(e_{i,k}(t)) = \sum_{\substack{\mathbf{o}_m \in obj(\mathbf{c}_i(t)) \\ \mathbf{o}_n \in neighborhood(\mathbf{o}_m, t)}} \left[\frac{e_{j,k}(t) - e_{i,k}(t)}{|obj(\mathbf{c}_i(t))|} \cdot \right.$$
$$\left. \cdot\ exchange(\mathbf{c}_i(t), \mathbf{c}_j(t), k)\ \cdot\ falloff(\|\mathbf{o}_m - \mathbf{o}_n\|) \right] \tag{11}$$

onde $\mathbf{c}_j(t) = cluster(\mathbf{o}_n, t)$, $e_{j,k}(t)$ refere-se às componentes de $\mathbf{c}_j(t)$ e $\|\mathbf{o}_m - \mathbf{o}_n\|$ é a distância entre $\mathbf{o}_m$ e $\mathbf{o}_n$. Observam-se, na definição de *resultant*, duas outras funções: *exchange* e *falloff* (*fall off*).

$$exchange : \mathbb{R}^{nobj} \times \mathbb{R}^{nobj} \times \mathbb{N} \longmapsto \mathbb{R} \tag{12}$$
$$falloff : \mathbb{R} \longmapsto \mathbb{R} \tag{13}$$

Essas outras duas funções fornecem fatores para o cálculo de *resultant* e constituem parâmetros do presente método de agrupamento. Informalmente, *exchange* diz qual a taxa de troca de energia entre dois *clusters* em função da diferença entre seus estados de energia. Já *falloff* diz o quanto diminui a influência entre dois objetos em função da distância que os separa.

Foi dito que *resultant* refere-se à variação do estado de energia "desejada" por cada *cluster*. No entanto, quando os estados de energia de dois (ou mais) *clusters* caminham um em direção ao outro, convergindo ao equilíbrio, deve-se evitar que quando cheguem razoavelmente perto, uma próxima iteração lance-os com demasiada violência a lados opostos, tornando-os mais distantes do que estavam. A Figura 1(a) ilustra essa situação indesejada. Introduz-se então a função $normalization(t)$, cujo objetivo é gerar um fator capaz de normalizar a variação de energia de todos os *clusters*, de modo que a maior das variações seja igual à constante *emin*:

$$normalization(t) = \frac{emin}{max\Big(\Big\{\|resultant(\mathbf{c}_i(t))\|\Big\}\Big)}, \quad i = 1 \ldots |C(t)| \tag{14}$$

onde $\|resultant(\mathbf{c}_i(t))\|$ e $|C(t)|$ denotam, respectivamente, a magnitude do vetor $resultant(\mathbf{c}_i(t))$ e o tamanho do conjunto $C(t)$. A função max resulta, consequentemente, na maior das variações do estado de energia. A Figura 1(b) ilustra a variação dos estados de energia sendo limitada por *emin*. Essa figura ilustra o caso onde a variação é a máxima possível, ou seja, igual a *emin*. Vale lembrar que quando a

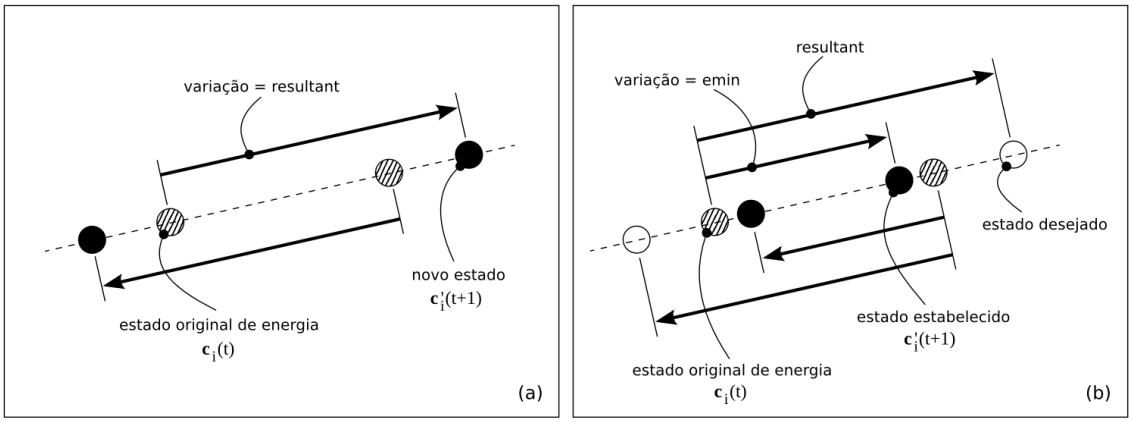


Figura 1. Dois estados de energia num movimento simétrico de aproximação. Em (a), a situação indesejada, onde a distância entre eles após a variação se torna maior que a distância antes da variação. Em (b), a variação sendo limitada por $emin$**.**

Tabela 1. Os parâmetros do método baseado em dinâmica de troca de energia.

Parâmetro	Descrição
omin	Objetos cuja distância seja menor ou igual a *omin* são agrupados já em $t = 0$.
emin	*Clusters* cuja diferença entre seus estados de energia seja menor ou igual a *emin* são agrupados. Valores grandes de *emin* aceleram a convergência, mas reduzem a precisão do processo de agrupamento.
nneighbors	Número de vizinhos considerados. Cada objeto, a cada iteração, só troca energia com os *nneighbors* vizinhos mais próximos que não pertencem ao mesmo *cluster*.
$exchange(\mathbf{c}_i(t), \mathbf{c}_j(t), k)$	Define a taxa de troca de energia entre dois *clusters* em função da diferença entre seus estados de energia.
$falloff(\|\mathbf{o}_m - \mathbf{o}_n\|)$	Define o quanto diminui a influência entre dois objetos em função da distância que os separa.

distância entre os estados de energia de dois *clusters* torna-se menor ou igual a *emin*, esses *clusters* são definitivamente agrupados. Importante notar também que, caso o maior dos valores para *resultant* seja menor que a constante *emin*, *normalization* aumenta todas as variações de energia, evitando que a convergência do processo de agrupamento seja muito lenta. Assim, valores grandes de *emin* aumentam a velocidade de convergência, mas reduzem a precisão do processo de agrupamento.

Finalmente, a variação efetiva do estado de energia de cada *cluster*:

$$e'_{i,j}(t+1) = e_{i,j}(t) + resultant(e_{i,j}(t)) \cdot normalization(t) \tag{15}$$

Importante notar que todo o processo de agrupamento consiste na intercalação de dois sub-processos: fusão dos *clusters* e troca de energia entre os *clusters*. A fusão leva de um conjunto $C'(t)$ de *clusters* a um novo conjunto $C(t)$, eventualmente menor que $C'(t)$. A troca de energia leva o estado de energia de cada *cluster*

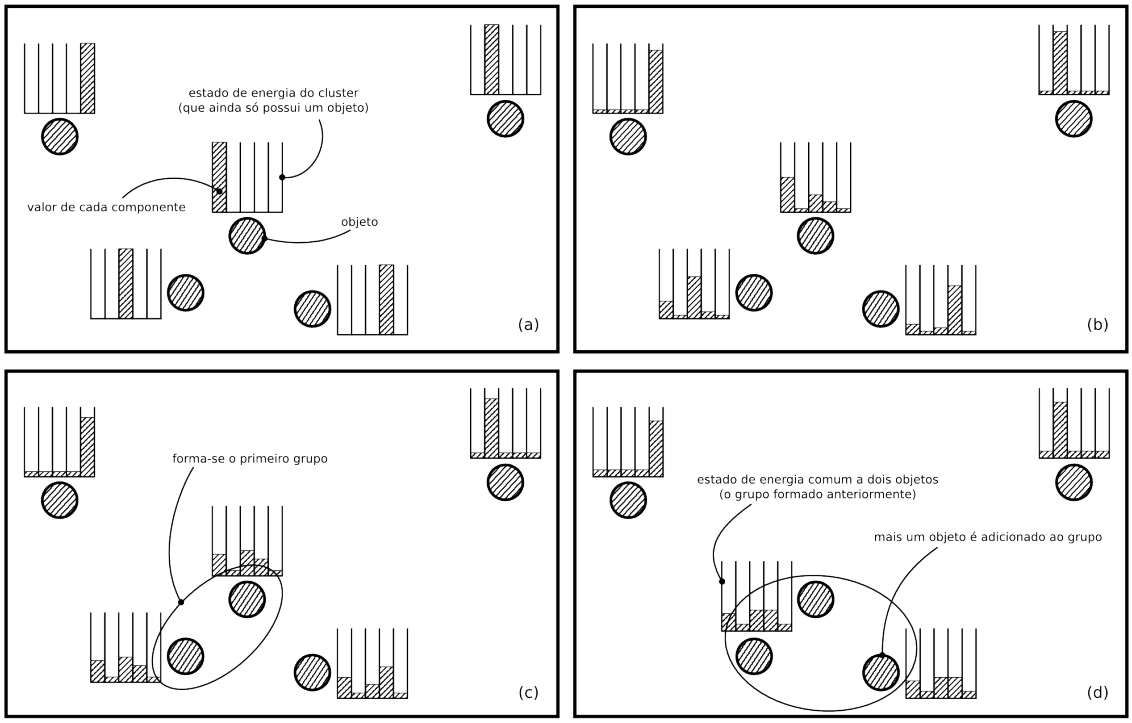


Figura 2. Ilustração do processo de agrupamento. (a): o estado inicial dos objetos. (b): algum estado após a evolução do processo. (c): dois dos objetos atingem estados de energia suficientemente semelhantes e são agrupados. (d): um terceiro objeto é adicionado ao grupo.

$\mathbf{c}_i(t)$ a um novo valor em $\mathbf{c}'_i(t+1)$. A Tabela 1 sumariza os parâmetros do método descrito.

A fim de ilustrar o processo de agrupamento, a Figura 2 mostra um conjunto contendo cinco objetos. A cada objeto é atribuído um estado inicial de energia, conforme a Equação (6). Então esses objetos começam a interagir entre si e os grupos vão se formando. Quanto maior a distância separando dois objetos, menor o grau de interação entre eles. Assim, os objetos mais próximos são os primeiros a atingirem estados de energia semelhantes, portanto são os primeiros a serem agrupados.

3. Simulações realizadas

Simulações foram realizadas com o método de agrupamento de dados baseado em dinâmica de troca de energia proposto neste trabalho. As Seções 3.1 e 3.2 trazem, respectivamente, os resultados de simulações sobre conjuntos de dados artificiais e reais.

Todas as simulações empregaram os seguintes parâmetros: $omin = 0,1$, $emin = 0,02$, $nneighbors = 20$ e $exchange(\mathbf{c}_i(t), \mathbf{c}_j(t), k) = 1$. Esse valor para $exchange$, quando aplicado na Equação (11), faz a intensidade da troca de energia entre dois objetos ser diretamente proporcional à diferença entre seus valores de energia. O parâmetro variado foi a função *falloff*, que para todos os conjuntos testados, assumiu: $falloff = 1/\|\mathbf{o}_m - \mathbf{o}_n\|$ e $falloff = 1/\|\mathbf{o}_m - \mathbf{o}_n\|^5$, lembrando que $\|\mathbf{o}_m - \mathbf{o}_n\|$ representa a distância entre os objetos.

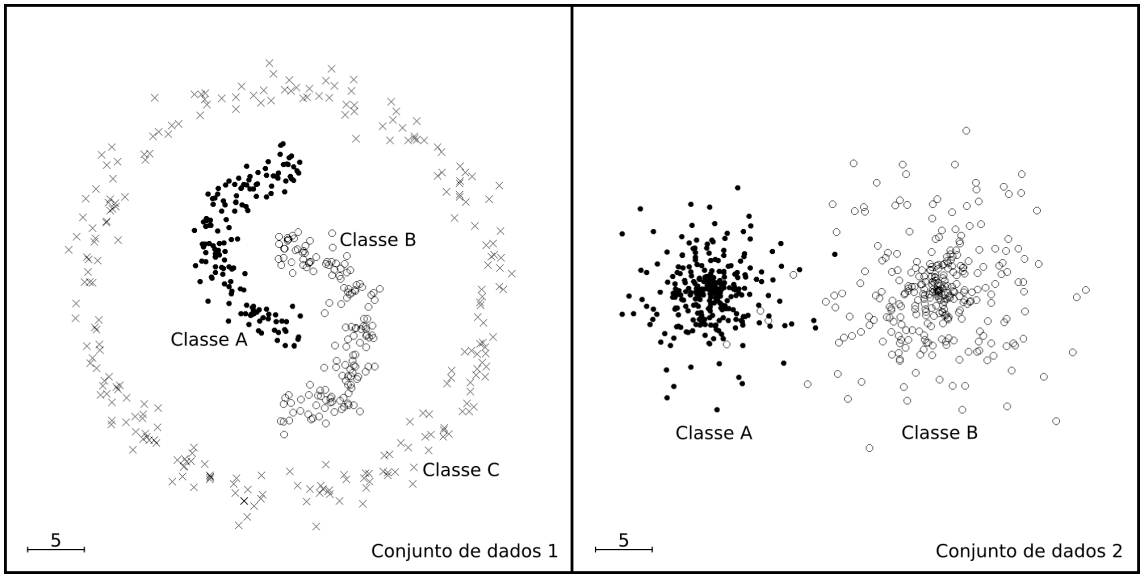


Figura 3. Dois conjuntos artificiais de dados utilizados nas simulações.

Os conjuntos de dados utilizados neste trabalho são formados por classes de objetos. Essas classes existem *a priori*. Contudo, o método de agrupamento não as "conhece", evidentemente. Seu objetivo é justamente "descobri-las". Assim, a fim de mensurar o processo de agrupamento, rastreou-se o crescimento de um *cluster* representativo de cada classe do conjunto de dados. Seja uma classe A. Então procurou-se pelo *cluster* que, ao longo de todo o processo de agrupamento, detivesse a maior quantidade de objetos da classe A. Mediu-se então a relação entre a quantidade de objetos da classe A presentes no *cluster* e o tamanho do próprio *cluster*.

3.1. Simulações com dados artificiais

Geraram-se dois conjuntos artificiais de dados, como mostra a Figura 3. Cada conjunto é identificado simplesmente como *Conjunto de dados 1* e *Conjunto de dados 2*.

O Conjunto de dados 1 contém três classes. As Classes A e B são formadas, cada uma, por 150 objetos distribuídos aleatoriamente por uma semicircunferência de raio 7,5. A Classe C, ao redor das outras duas, é formada por 250 objetos distribuídos aleatoriamente por uma circunferência de raio 18. Em todas as classes, o valor angular de cada objeto teve probabilidade uniforme pela circunferência (ou semicircunferência) e o valor radial seguiu uma distribuição de probabilidade normal (gaussiana) com valor esperado igual ao raio e desvio padrão igual a 1.

O Conjunto de dados 2 contém duas classes. Cada uma delas é formada por 300 objetos distribuídos aleatoriamente ao redor de um ponto, seu centro. Seguiu-se uma distribuição de probabilidade normal bivariada, com desvio padrão igual a 4 para a Classe A e 6 para a Classe B, em ambas as dimensões. Os centros de cada classe distam 20 unidades um do outro.

A Figura 4 mostra os resultados sobre o Conjunto de dados 1. Dentre os dois experimentos, observa-se desempenho consideravelmente melhor com *falloff* =

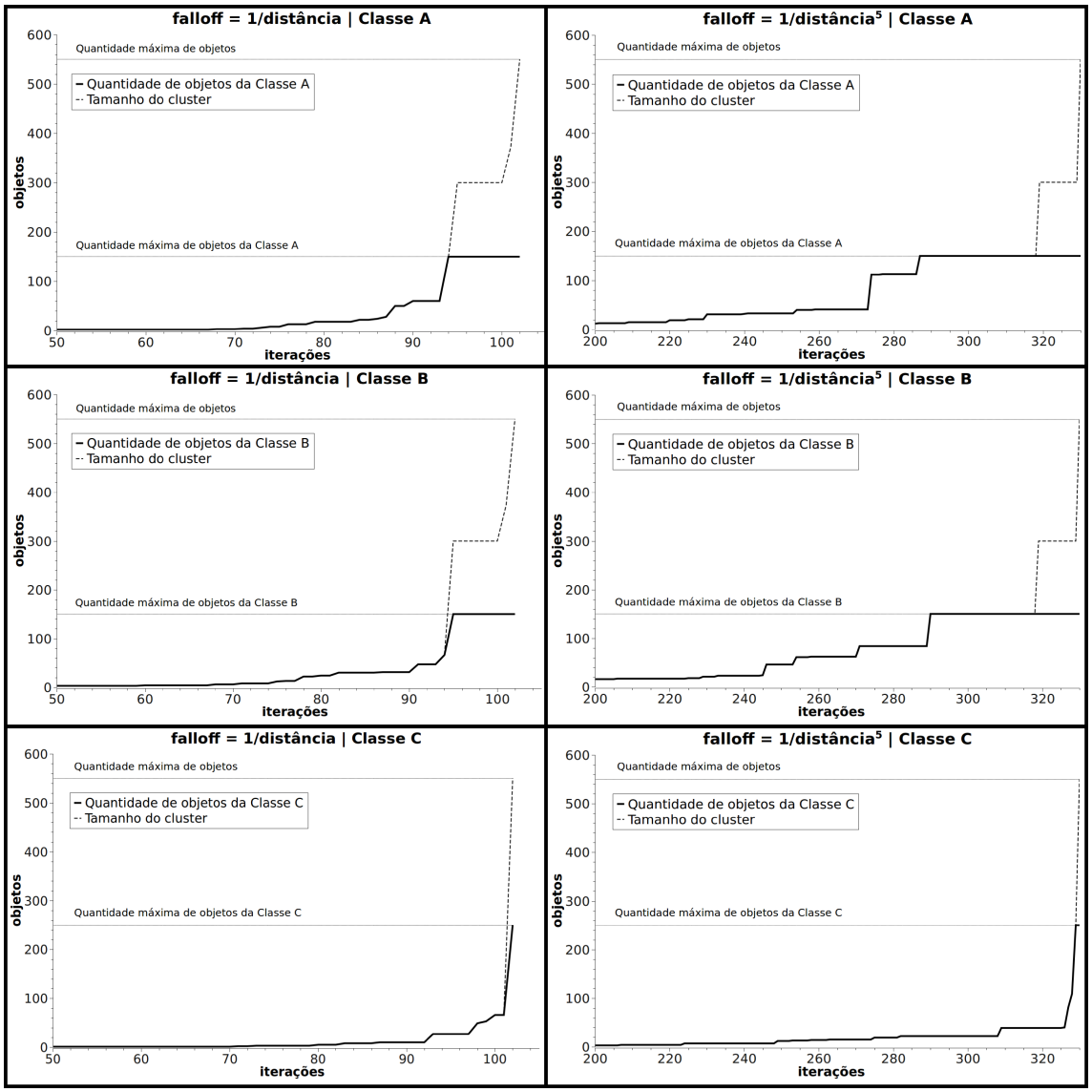


Figura 4. Resultados das simulações sobre o Conjunto de dados 1.

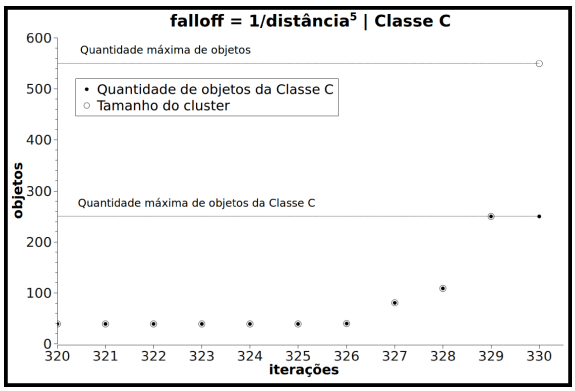


Figura 5. Evolução do cluster representativo da Classe C, nas últimas iterações da simulação sobre o Conjunto de dados 1, empregando $falloff = 1/\|\mathbf{o}_m - \mathbf{o}_n\|^5$.

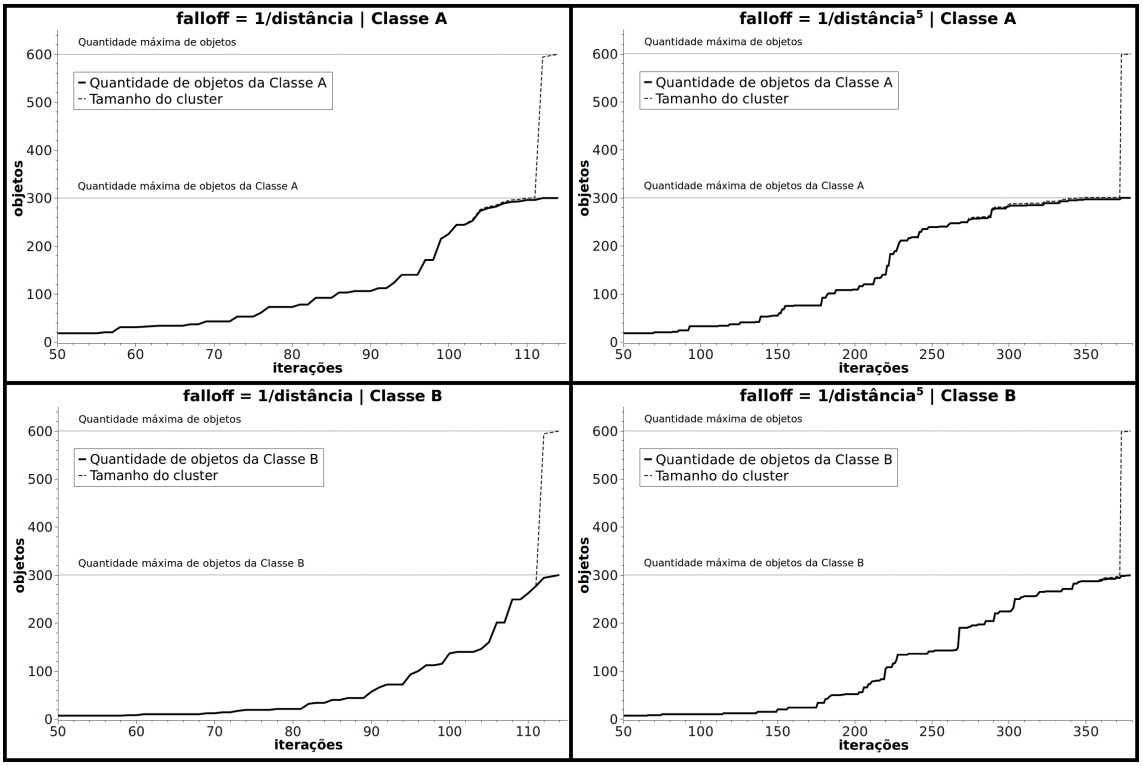


Figura 6. Resultados das simulações sobre o Conjunto de dados 2.

$1/\|\mathbf{o}_m - \mathbf{o}_n\|^5$. Nele, as Classes A e B são perfeitamente agrupadas, e permanecem assim por várias iterações, antes que comecem a se misturar. A Classe C atinge também seu estado ideal, permanecendo assim durante uma iteração somente, a iteração $t = 329$, como detalha a Figura 5.

A Figura 6 mostra os resultados sobre o Conjunto de dados 2. Observa-se desempenho ligeiramente melhor com $falloff = 1/\|\mathbf{o}_m - \mathbf{o}_n\|^5$. As duas classes mantêm-se praticamente puras por bastante tempo, mas quando chegam perto de tornarem-se completas, misturam-se bruscamente.

3.2. Simulações com dados reais

Mostram-se aqui os resultados de simulações sobre um conjunto de dados denominado *Iris Data Set*, obtido do *UCI Machine Learning Repository* [UCI]. Esse conjunto é comumente empregado para avaliação de métodos de classificação ou agrupamento de dados. Ele é composto por três classes, cada uma contendo 50 objetos. Cada classe representa uma espécie da planta *Iris*: *Iris setosa*, *Iris versicolor* e *Iris virginica*. Cada objeto é formado por quatro atributos: tamanho e largura da sépala e da pétala, em centímetros.

A Figura 7 mostra os resultados obtidos. Observa-se bom desempenho para a classe *Iris setosa*, principalmente com $falloff = 1/\|\mathbf{o}_m - \mathbf{o}_n\|^5$. Ela chega perto de ser completamente agrupada antes que comece a se misturar com as duas outras. A partir daí, a mistura é brusca. Para as outras duas classes, no entanto, houve desempenho insatisfatório. De fato, as classes *Iris versicolor* e *Iris virginica* estão "encostadas" uma na outra, enquanto a classe *Iris setosa* mantém-se distante.

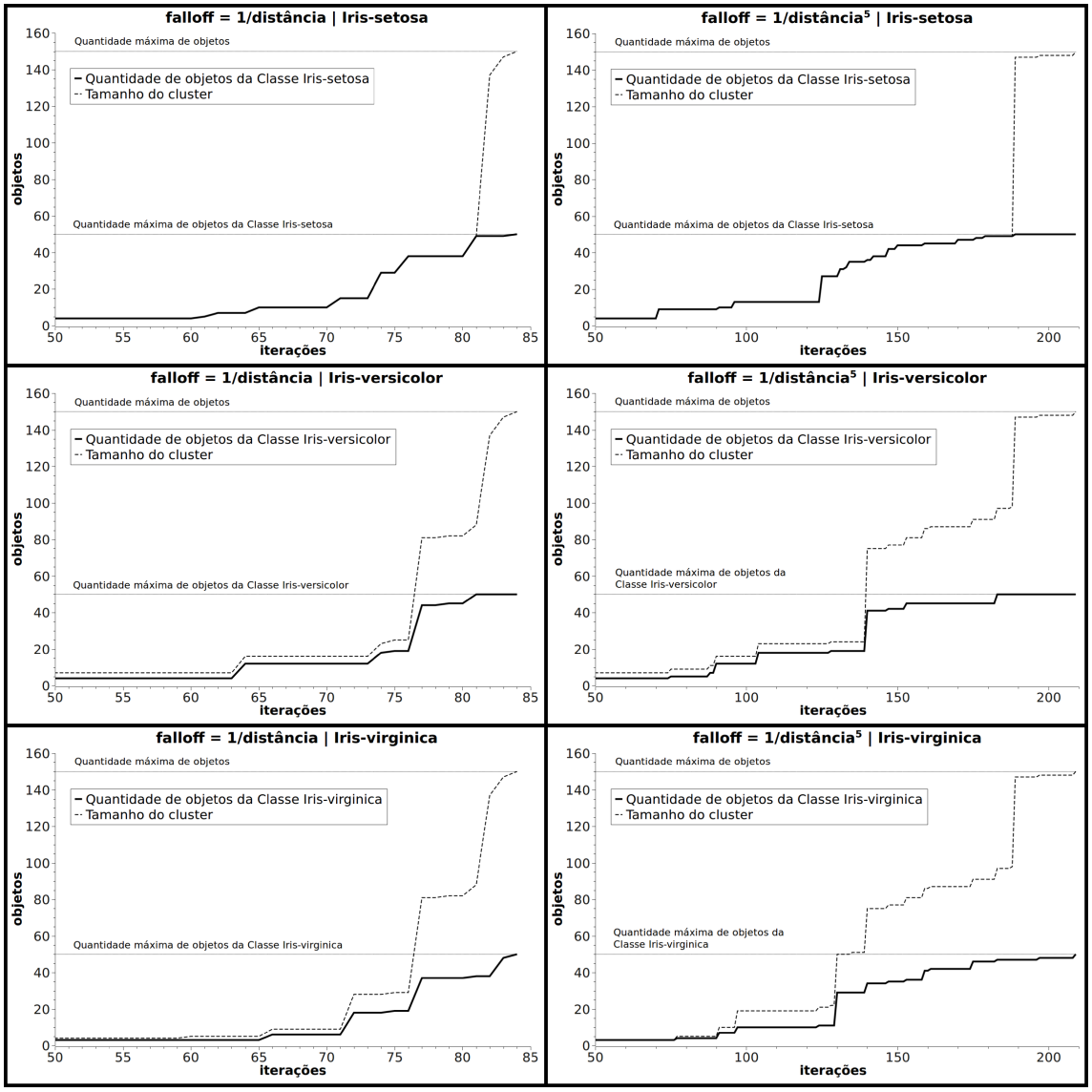


Figura 7. Resultados das simulações sobre o conjunto Iris.

4. Conclusões

O método proposto e estudado neste trabalho exibe características interessantes. Conceitualmente, é um método simples, baseado na gradativa troca de energia entre os elementos constituintes do sistema, os objetos. Contudo, essa simplicidade mostra-se capaz de agrupar objetos com distribuição relativamente complexa, como mostraram as simulações sobre o Conjunto de dados 1. Os resultados sugerem que esse método é capaz de separar e agrupar classes de formas arbitrárias, desde que a distância entre objetos adjacentes de uma mesma classe seja, em geral, menor que a distância entre objetos de classes distintas.

Alguns conjuntos de dados não exibem, entretanto, uma distribuição contendo classes tão bem-separadas. Neles ocorrem pareamento ou sobreposição de classes. Os resultados das simulações sobre o Conjunto de dados 2 e sobre o conjunto Iris exibem a dificuldade em se agrupar objetos com uma distribuição desse

tipo.

O método de agrupamento aqui apresentado é considerado como sendo um processo de aprendizado não-supervisionado, ou seja, toma um conjunto contendo somente objetos não-classificados (não-rotulados) [Mitchell 1997, Jain et al. 1999, Alpaydin 2004, Xu and Wunsch 2005, Bishop 2006, Hastie et al. 2009]. Pretende-se estender esse método de modo a contemplar conjuntos de objetos parcialmente classificados, ou seja, conjuntos contendo uma mistura de objetos classificados e não-classificados. Este último processo é denominado aprendizado semissupervisionado [Zhu 2005]. Espera-se um considerável aumento de desempenho através desse novo método semissupervisionado.

Referências

Uci machine learning repository. Disponível em <`http://archive.ics.uci.edu/ml/index.html`>. Acessado em maio de 2011.

Abraham, A., Das, S., and Roy, S. (2007). *Soft Computing for Knowledge Discovery and Data Mining*, chapter Swarm Intelligence Algorithms for Data Clustering. Springer Verlag.

Alpaydin, E. (2004). *Introduction to Machine Learning*. The MIT Press.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer.

Blatt, M., Wilseman, S., and Domany, E. (1997). Data clustering using a model granular magnet. *Neural Computation*, 9(8):1805–1842.

Castro, L. N. (2007). Fundamentals of natural computing: an overview. *Physics of Life Reviews*, 4(1):1–36.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, 2. edition.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys*, 31(3).

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill Series in Computer Science. McGraw-Hill.

Oliveira, T. B. S. (2008). Clusterização de dados utilizando técnicas de redes complexas e computação bioinspirada. Master's thesis, Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo – São Carlos.

Xu, R. and Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678.

Zhao, L., Damiance Jr., A. P. G., and Carvalho, A. C. P. L. F. (2005). *Advances in Natural Computation*, volume 3610 of *Lecture Notes in Computer Science*, chapter A Self-organized Network for Data Clustering. Springer.

Zhu, X. (2005). Semi-supervised learning literature survey. Computer Sciences, University of Wisconsin-Madison, n. 1530.

HTILDE-RT: Um algoritmo para aprender Árvores de Regressão Relacionais em grandes conjuntos de dados

Glauber M. C. Menezes [1], **Gerson Zaverucha** [1]

[1] Programa de Engenharia de Sistemas de Computação (PESC)
Universidade Federal do Rio de Janeiro (COPPE/UFRJ)
Caixa Postal 68.511 – 21.945-970 – Rio de Janeiro – RJ – Brasil

{glauber,gerson}@cos.ufrj.br

Abstract. *Currently, modern organizations store their data under the form of relational databases which grow faster than hardware capacities. However, extracting information from such databases has become crucial. In this work we propose HTILDE-RT, an algorithm to learn relational regression trees efficiently from huge databases. It is based on the ILP system TILDE and the propositional system VFDT learner. The algorithm uses Hoeffding bound to scale up the learning process. We compared HTILDE-RT with TILDE-RT in two large datasets, each with two million examples, yielding more than three times faster learning times with no statistically significant difference in Pearson correlation coefficient.*

Resumo. *Atualmente, organizações modernas armazenam seus dados sob a forma de bancos de dados relacionais que crescem numa velocidade superior à capacidade de hardware. Entretanto, extrair informações destas bases tornou-se crucial. Neste trabalho propomos o HTILDE-RT, um algoritmo capaz de aprender árvores de regressão relacionais em grandes massas de dados. O algoritmo é baseado no sistema ILP TILDE e no sistema proposicional VFDT. O algoritmo usa o limitante de Hoeffding para acelerar o aprendizado. Comparamos o HTILDE-RT com o TILDE-RT em dois conjuntos de dados grandes, com dois milhões de exemplos cada, obtendo tempos de aprendizado mais de três vezes mais rápidos sem diferenças significativas no coeficiente Pearson.*

1. Introdução

Mineração em fluxos de dados tornou-se objeto de pesquisa em variados campos da Ciência da Computação tais como Aprendizado de Máquina e Mineração de Dados. Como exemplos de onde técnicas para fluxos são úteis podemos citar: aplicações de suporte a transações de cartões de crédito, tráfego de Internet, monitoramento em telecomunicações e pesquisa científica em dados biológicos.

Os bancos dados de organizações modernas cresceram numa escala superior aos recursos de hardware e memória, entretanto, extrair informações ocultas em tais bancos tornou-se tarefa crítica para competitividade e sobrevivência das corporações.

Atualmente, as bases de dados são estruturadas sob o paradigma relacional. Porém, algoritmos de aprendizagem tradicionais não são perfeitamente aplicáveis em tal

tipo de dados, uma vez que utilizam o paradigma proposicional, não sendo capazes de descrever relações entre objetos nos dados. Uma outra importante questão é o fato de que os algoritmos tradicionais não são capazes de manipular massas de dados tão grandes, pois necessitam que todos os exemplos estejam na memória principal ao mesmo tempo, o que é impraticável neste cenário.

Em tal ambiente dados novos chegam em taxas elevadas e um algoritmo eficiente deve processá-los prontamente. Porém, há a restrição de que armazená-los como um todo na memória é irrealizável. Dado isto, as áreas de Aprendizado de Máquina e Mineração de Dados direcionaram esforços para tratar tal tipo de demanda de forma eficiente.

Domingos e Hulten [Domingos and Hulten 2000] desenvolveram uma metodologia para tratar fluxos de dados, que culminou no algoritmo VFDT (Very Fast Decision Trees). O VFDT é um algoritmo proposicional, que usa uma técnica de amostragem baseada no limitante de Hoeffding [Hoeffding 1963] para escolher qual é o melhor teste a ser introduzido no nó de uma árvore de decisão, sem a necessidade de analisar todos os exemplos ao mesmo tempo. O VFDT é o estado da arte para este tipo de problema.

O TILDE (Top-down Induction of Logical Decision Trees) [Blockeel and Raedt 1998] é a versão relacional do algoritmo de árvores de decisão, o qual é capaz de modelar a grande maioria dos esquemas relacionais nos bancos de dados modernos, sua eficiência é garantida através do emprego das técnicas de pacotes de cláusulas (Query Packs) [Blockeel et al. 2002] e Aprendizado por Interpretações (Learning from Interpretations) [Blockeel et al. 1999]. Tal algoritmo é implementado como parte do sistema de Programação em Lógica Indutiva (ILP) ACE [H. Blockeel and Fierens 2005].

Para obter a versão relacional do VFDT, Lopes e Zaverucha [Lopes and Zaverucha 2009] modificaram o TILDE, desenvolvendo o algoritmo HTILDE (Hoeffding TILDE), baseado no TILDE e no VFDT. Então, o HTILDE é a versão relacional do VFDT para problemas de classificação.

Neste trabalho apresentamos o HTILDE-RT, que é uma modificação do HTILDE para árvores de regressão, que combina a expressividade do paradigma relacional e a eficiência da técnica de amostragem apoiada no limitante de Hoeffding. Uma árvore de regressão é um algoritmo capaz de aprender funções numéricas mantendo a interpretabilidade e eficiência da versão original.

2. Conhecimentos Preliminares

Nesta seção explicaremos rapidamente o VFDT, o TILDE e o HTILDE, sendo todos base do HTILDE-RT, bem como outros trabalhos relacionados ao tema de regressão.

2.1. Limitante de Hoeffding

Seja r uma variável aleatória real cujo domínio tem tamanho R. Considere n observações independentes desta variável e seja $\overline{r}$ sua média computada. O limitante de Hoeffding [Hoeffding 1963] atesta que $P(\mu_r \geq \overline{r} - \epsilon) = 1 - \delta$, onde μ_r é a verdadeira média da variável r, ϵ é dado pela equação 1 e δ é um número muito pequeno.

$$\epsilon = \sqrt{\frac{R^2 ln(1/\delta)}{2n}} \tag{1}$$

2.2. Modelagem para Árvores de Decisão

A seguinte modelagem foi proposta por Domingos e Hulten [Domingos and Hulten 2000]. Seja G a heurística a ser maximizada, sejam X_a e X_b os atributos com os maiores valores para G. Seja $\Delta G_r = G(X_a) - G(X_b)$ a diferença nos valores das heurísticas dos atributos quando o algoritmo utiliza todo o conjunto de dados, e seja $\Delta\overline{G} = \overline{G}(X_a) - \overline{G}(X_b)$ a diferença observada quando apenas n exemplos chegaram a um nó.

Então, dado um valor para δ através do limitante de Hoeffding, sabemos que com probabilidade $1-\delta$ que $\Delta G_r \geq \Delta\overline{G} - \epsilon$, onde ϵ é dado pela equação 1. Se $\Delta\overline{G} > \epsilon$, então podemos dizer que com mesma probabilidade $\Delta G_r > 0$, logo $G(X_a) > G(X_b)$, i.e., com probabilidade $1-\delta$, X_a é o melhor atributo.

Em termos práticos, para podermos afirmar se X_a é a melhor escolha, precisamos verificar se $\Delta\overline{G} \geq \epsilon$. Então, o que precisamos fazer é processar exemplos até acontecer que $\Delta\overline{G} \geq \epsilon$, para o valor escolhido de δ e o número corrente de exemplos.

Destacamos que o limitante de Hoeffding independe da distribuição da variável considerada. Esta independência vem com o custo de ser necessário processar mais exemplos do que em métodos que usam limitantes específicos. Porém, dado o cenário das grandes massas de dados, isto não configura um problema. Além disso, para o ganho de informação temos $0 \leq G \leq lg(|C|)$, onde C é o conjunto das possíveis classes.

2.3. VFDT

Utilizando tal modelagem, Domingos e Hulten desenvolveram o algoritmo VFDT (Very Fast Decision Tree) [Domingos and Hulten 2000], um algoritmo de aprendizado proposicional e escalável em grandes bases de dados. Os autores provaram que o VFDT gera árvores assintoticamente próximas às geradas pelo algoritmo tradicional.

O VFDT tem alguns elementos para melhorar ainda mais sua performance. Iremos mencionar dois deles, sendo estes um parâmetro para acúmulo de exemplos nas folhas $emin$ e um parâmetro para desempate entre atributos τ. Mais informações sobre o VFDT podem ser encontradas em [Domingos and Hulten 2000].

O cômputo da heurística é um processo custoso. O parâmetro $emin$ reduz o tempo de indução, permitindo que o usuário determine que cada novo cômputo da heurística seja feito somente quando $emin$ novos exemplos chegaram desde a folha vazia ou o último cômputo. Isto melhora a performance pois o algoritmo não precisa computar a heurística para todos os possíveis refinamentos todas as vezes que um único exemplo chega à folha. Portanto, no contexto de conjuntos de dados gigantes, isto tem um grande impacto.

Quando dois ou mais atributos possuem valores de heurísticas muito próximos, o algoritmo pode necessitar de um número grande de exemplos até escolher o melhor atributo, o que é um desperdício, uma vez que não fará grande diferença qual atributo seja escolhido já que estão muito próximos. Esta situação é chamada de empate e para resolvê-la o VFDT usa um parâmetro fornecido pelo usuário que impõe um limiar de desempate. Caso $\Delta\overline{G} < \epsilon < \tau$, o melhor atributo é escolhido como o teste para o nó.

Terminemos esta subseção mostrando as diferenças entre o algoritmo tradicional de de árvores de decisão [Mitchell 1997] e o VFDT para comparação.

Algoritmo 1 Algoritmo de Árvore de Decisão [Mitchell 1997]

1: Raiz: $Exs =$ todos os exemplos de treino.
2: Baseado no conjunto de exemplos Exs, selecione o melhor atributo A como teste do nó.
3: Crie um ramo para cada valor v_i do atributo discreto A, correspondendo ao teste $A = v_i$.
4: Separe os exemplos entre os nós filhos do nó corrente de acordo com o valor de A, criando um novo conjunto de exemplos Exs para cada filho.
5: Repita o passo 2 para cada filho.

Algoritmo 2 VFDT [Domingos and Hulten 2000]

1: *Raiz:* $Exs = N$ *primeiros exemplos de treino, onde* N *é obtido pelo limitante de Hoeffding* ($\Delta\overline{G} > \epsilon$ *ou* $\Delta\overline{G} < \epsilon < \tau$).
2: Baseado no conjunto de exemplos Exs, selecione o melhor atributo A como teste do nó.
3: Crie um ramo para cada valor v_i do atributo discreto A, correspondendo ao teste $A = v_i$.
4: *Descarte os exemplos que estavam antes no nó. Separe os próximos exemplos entre os filhos do no corrente, de acordo com o valor de* A*, criando um conjunto de exemplos* Exs *para cada filho. O número de exemplos de cada filho será também determinado pelo limitante de Hoeffding.*
5: Repita o passo 2 para cada filho.

2.4. TILDE

TILDE (Top-down Induction of Logical Decision Trees) é a versão relacional para aprendizado de árvores de decisão, isto implica que os nós internos são representados por conjunções de literais de Lógica de Primeira Ordem em vez de atributos. Por conseqüência, os modelos gerados têm a estrutura de uma árvore binária, tal que cada nó interno possui dois ramos, sendo o esquerdo um caminho para os exemplos que sucedem no teste do nó e o direito o caminho para os exemplos que falham no teste lógico do nó. O TILDE é capaz de rodar nos modos de classificação e regressão (TILDE-RT).

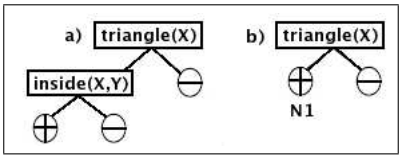


Figura 1. Exemplos de árvore gerada pelo TILDE.

Os candidatos a testes são chamados de refinamentos e são especificados pelo usuário através do viés de linguagem do problema. Quando os valores da heurística são computados o TILDE considera o caminho associado à consulta entre a raiz e a folha em avaliação.

Por exemplo, considere a árvore b mostrada na figura 1. Suponha que desejamos saber quais testes podem ser colocados no nó $N1$. Uma vez que este nó é o filho

esquerdo da raiz, sabemos que triangle(X) é verdade em $N1$ e portanto todos os refinamentos conterão este literal. Aplicando os demais refinamentos a este nó, de acordo com o viés de linguagem previamente especificado, podemos obter quatro refinamentos para $N1$: $triangle(X), inside(X, Y)$; $triangle(X), inside(Y, X)$; $triangle(X), square(Y)$ e $triangle(X), circle(Y)$. Se o melhor refinamento for o primeiro então o teste colocado no nó $N1$ será $inside(X, Y)$.

É importante notar que tais refinamentos são processados sob a condição de θ-subsumption [Blockeel et al. 1999]. Também são consideradas restrições sobre o espaço de busca escolhidas pelo usuário, que podem ser estabelecidas, principalmente, pelo uso dos tipos e modos das variáveis. Os tipos definem o domínio das variáveis restringindo a unificação. Os modos indicam se uma variável é de entrada, saída ou os dois. Uma referência mais completa sobre as configurações dos refinamentos e viés de linguagem pode ser encontrada em [H. Blockeel and Fierens 2005].

Para alcançar eficiência, o TILDE usa pacotes de cláusulas e aprendizado por interpretações para reduzir o custo computacional da prova dos exemplos. Estes mecanismos são importantes em função de o TILDE aplicar os testes aos exemplos através de provas lógicas, evitando-se provas de literais de forma repetitiva e acelerando o mecanismo de prova.

O TILDE assume que cada exemplo é uma interpretação [Blockeel et al. 1999]. Cada exemplo possui um rótulo de sua classe e um conjunto de fatos, os quais codificam propriedades e relações que podem ocorrer, de tal maneira que cada exemplo pode ser processado de forma independente dos demais. O conhecimento sobre o domínio (background knowledge) é também fornecido, através do qual fatos adicionais podem ser deduzidos, sendo representado sob a forma de um programa Prolog. O Aprendizado por Interpretações tem menor poder de expressividade que o Aprendizado por Implicação Lógica [Muggleton 1991], o qual é tradicional em ILP, porém é mais eficiente. Apesar disso, para a maioria das aplicações o aprendizado por interpretações é suficiente, sendo a total expressividade tradicional da ILP nem sempre necessária [Blockeel et al. 1999].

Um pacote de cláusulas (Query Pack) [Blockeel et al. 2002] é um método de representar cláusulas sob a forma de disjunções a fim de evitarem-se provas idênticas de literais. Cláusulas de mesma cabeça têm suas caudas reunidas sob a forma de uma disjunção, de tal modo que literais comuns a clausulas diferentes são devidamente fatorados e evidenciados. Isto é importante no momento em que o TILDE computa a heurística para cada refinamento, onde é comum que dois ou mais refinamentos tenham literais em comum.

O algoritmo 3 destaca as diferenças entre o TILDE e o algoritmo proposicional tradicional.

2.5. HTILDE

O HTILDE (Hoeffding TILDE) [Lopes and Zaverucha 2009] é um algoritmo baseado no TILDE e no VFDT, aliando expressividade e eficiência através da combinação do paradigma relacional e a metodologia proposta por Domingos e Hulten no VFDT. É um algoritmo ILP escalável que se propõe a tratar grandes bases de dados relacionais.

HTILDE processa os refinamentos da mesma forma que o TILDE e possui três parâmetros herdados do VFDT: δ, $emin$, e τ. Desta forma, o HTILDE é um algoritmo

Algoritmo 3 TILDE [Blockeel and Raedt 1998]

1: Raiz: $Exs =$ todos os exemplos de treino.
2: *Baseado no conjunto de exemplos Exs, selecione o melhor refinamento R como teste para o nó.*
3: *Crie dois ramos: um correspondente ao teste $R = True$ e outro para $R = False$.*
4: *Separe os exemplos entre os nós filhos do nó corrente de acordo com o valor lógico obtido no teste de R, criando um novo conjunto de exemplos Exs para cada filho.*
5: Repita o passo 2 para cada filho.

incremental que aprende árvores de decisão relacionais, usando o limitante de Hoeffding para tratar grandes massas de dados.

O algoritmo 4 destaca as diferenças entre o HTILDE e o VFDT. Destacamos que o HTILDE usa os mesmos parâmetros δ, $emin$, e τ que o VFDT, bem como processa os refinamentos da mesma forma que o TILDE.

Algoritmo 4 HTILDE [Lopes and Zaverucha 2009]

1: Raiz: $Exs = N$ primeiros exemplos de treino, onde N é obtido pelo limitante de Hoeffding ($\Delta\overline{G} > \epsilon \; ou \; \Delta\overline{G} < \epsilon < \tau$).
2: *Baseado no conjunto de exemplos Exs, selecione o melhor refinamento R como teste para o nó.*
3: *Crie dois ramos: um correspondente ao teste $R = True$ e outro para $R = False$.*
4: *Descarte os exemplos que estavam antes no nó. Separe os próximos exemplos entre os dois filhos do no corrente, de acordo com o valor lógico obtido no teste de R, criando um conjunto de exemplos Exs para cada filho. O número de exemplos de cada filho será também determinado pelo limitante de Hoeffding.*
5: Repita o passo 2 para cada filho.

2.6. Árvores de Regressão

Uma árvore de regressão é uma versão do algoritmo 1 que trata do aprendizado de funções numéricas. Suas principais diferenças são a função de heurística e a predição. Árvores de classificação usam heurísticas aplicáveis a domínios discretos como o ganho de informação e o índice de Gini, porém no contexto da regressão a heurística mais comumente usada é redução na variância do atributo alvo. Além disso, árvores de classificação predizem a classe majoritária da folha em que um exemplo a ser classificado alcançar e as árvores de regressão tradicionalmente predizem o valor médio observado do atributo alvo dentre os exemplos de uma folha.

A heurística padrão usada é a redução da variância [Breiman et al. 1984], dada por:

$$\phi(T, l) = S^2(L) - \sum_{T_j} \frac{|T_j|}{|L|} S^2(T_j) \tag{2}$$

onde L é o conjunto dos exemplos da folha l, T é um atributo de teste em l e T_j é o conjunto dos exemplos de l tais que o atributo T tem valor j. Se T for contínuo, tradici-

onalmente usa-se um passo de discretização para determinar os intervalos em que j deve iterar. A variância S^2 para um conjunto de exemplos A é dada por:

$$S^2(A) = \frac{1}{|A|} \sum_{i=1}^{|A|} (a_i - \overline{a})^2 \tag{3}$$

onde a_i é o valor do atributo alvo no exemplo i e $\overline{a}$ é o valor médio do predicato alvo observado nos exemplos de A.

O algoritmo FIMT-DD (Fast Incremental Model Trees with Drift Detection), proposto em [Ikonomovska et al. 2011], é uma versão proposicional para o problema de regressão baseada no VFDT e CVFDT. O algoritmo trata atributos numéricos através de árvores binarias de busca estendidas (Extended Binary Search Tree - E-BST) em vez de discretização. Sua heurística é o desvio padrão e o modelo de regressão armazenado nas folhas são perceptrons e o algoritmo é também capaz de tratar mudanças de distribuição [Hulten et al. 2001]. Mais informações sobre o sistema proposicional FIMT-DD podem ser encontradas em [Ikonomovska et al. 2011].

O algoritmo TILDE-RT [Blockeel and Raedt 1998] é a versão do TILDE para tratar problemas de regressão. O TILDE-RT usa como heurística a soma dos erros quadráticos em vez da variância, tal heurística é também listada em [Breiman et al. 1984]. Conforme o algoritmo tradicional de árvores de regressão, o modelo preditivo armazenado nas folhas é a média dos valores observados para o atributo alvo nos exemplos. A heurística do TILDE-RT será definida na seção 3.

Devido à limitação de espaço deste trabalho e ao vasto campo da regressão, recomenda-se a inspeção dos trabalhos [Breiman et al. 1984],[Vens et al. 2006] e [Ikonomovska et al. 2011] para maiores esclarecimentos sobre o tema, bem como as referências relacionadas ao tema que são citadas pelos dois últimos trabalhos.

3. HTILDE-RT

HTILDE-RT é uma modificação do HTILDE para tratar problemas de regressão e por isso processa exemplos da mesma forma. O TILDE-RT, usa como heurística a redução dos erros quadráticos [H. Blockeel and Fierens 2005], conforme equação 4, portanto nossa escolha para heurística é muito semelhante, com a alteração de ocorrer normalização para que seus valores fiquem delimitados.

Logo, desejamos o refinamento que cause a maior redução possível no erro quadrático. Além disso, o limitante de Hoeffding necessita que a variável observada seja limitada e, para tal, basta uma normalização com o fator $SS(L)$. Formalmente, desejamos o refinamento ρ que maximiza H na equação 5:

$$H(\rho, l) = SS(L) - (SS(\boldsymbol{N}(\rho)) + SS(\boldsymbol{P}(\rho))) \tag{4}$$

$$H(\rho, l) = \frac{SS(L) - (SS(\boldsymbol{N}(\rho)) + SS(\boldsymbol{P}(\rho))}{SS(L)} \tag{5}$$

onde $SS(L)$ é a soma dos erros quadráticos do predicado alvo para o conjunto de exemplos L da folha l, ρ um candidato a refinamento de l, $P(\rho)$ o subconjunto de exemplos de L que passam no teste de ρ e $N(\rho)$ o subconjunto de exemplos de L que falham em ρ. A soma dos erros quadráticos do predicado alvo num conjunto de exemplos A qualquer é dada por:

$$SS(A) = \sum_{i=1}^{|A|}(a_i - \overline{a})^2 \tag{6}$$

Note que, H na equação 5 fica limitado inferiormente por 0 e superiormente por 1, logo está devidamente delimitada, então podemos usar $R = 1$ na equação do limitante de Hoeffding 1 resultando na equação 7.

$$\epsilon = \sqrt{\frac{ln(1/\delta)}{2n}} \tag{7}$$

Sejam ρ_1 e ρ_2 os dois melhores refinamentos com heurísticas devidamente normalizadas, então a variável aleatória para o limitante de Hoeffding é $\Delta\overline{H} = \overline{H}(\rho_1) - \overline{H}(\rho_2)$, onde $\Delta\overline{H}$ é a diferença observada entre os exemplos correntes na folha, de forma análoga a $\Delta\overline{G}$. No algoritmo 5 destacamos a diferença no algoritmo do HTILDE-RT.

Algoritmo 5 HTILDE-RT

1: Raiz: $Exs = N$ primeiros exemplos de treino, onde N é obtido pelo limitante de Hoeffding ($\Delta\overline{H} > \epsilon$ *ou* $\Delta\overline{H} < \epsilon < \tau$).
2: *Baseado no conjunto de exemplos Exs, selecione o refinamento R, entre os demais, cujo valor* ***normalizado*** *da heurística seja máximo, como teste para o nó.*
3: Crie dois ramos: um correspondente ao teste $R = True$ e outro para $R = False$.
4: Descarte os exemplos que estavam antes no nó. Separe os próximos exemplos entre os dois filhos do no corrente, de acordo com o valor lógico obtido no teste de R, criando um conjunto de exemplos Exs para cada filho. O número de exemplos de cada filho será também determinado pelo limitante de Hoeffding.
5: Repita o passo 2 para cada filho.

Algumas diferenças entre os algoritmos 4 e 5 e devem ser reforçadas. O HTILDE utiliza o ganho de informação como heurística, a qual está sempre entre 0 e $lg(|C|)$, em nosso caso necessitamos do fator de normalização $SS(T(\rho))$ para que as fronteiras dos valores da heurística estejam determinadas. Além disso, o modelo preditivo é a média dos exemplos em relação ao predicado alvo em vez da classe majoritária.

4. Experimentos

Conforme mencionado em [Vens et al. 2006], há uma escassez de conjuntos de dados relacionais para regressão que sejam públicos, portanto conduzimos experimentos com dados sintéticos descritos em [Vens et al. 2006]. Usamos geradores para obter os nossos conjuntos de dados, Artificial 1 e Artificial 2, ambos com dois milhões de exemplos e tamanhos médios de 340Mb e 590Mb respectivamente.

Em ambos experimentos usamos os parâmetros do HTILDE-RT com $emin = 300$, $\tau = 0,1$ e $\delta = 10^{-6}$. Estes valores são os mesmos que os default do VFDT com exceção de τ, escolhidos por apresentaram melhores resultados em testes preliminares, porém ainda nos falta fazer o ajuste destes parâmetros. Mantivemos os parâmetros do TILDE-RT com valores default.

Avaliamos os algoritmos adicionando ruído gaussiano nos dados, conforme feito em [Vens et al. 2006]. Cada exemplo teve um valor de ruído adicionado a seu predicato alvo. Os valores de ruído foram gerados aleatoriamente com distribuição normal.

Usamos o TILDE-RT como comparativo, uma vez que ambos algoritmos rodam sob o mesmo sistema e ambos são relacionais. Marcamos os resultados referentes ao ruído com um asterisco: HTILDE-RT* e TILDE-RT*.

As medidas de interesse avaliadas são: Tempo de aprendizado (Tempo), tamanho das teorias ou número de nós na árvore (Nós), número total de literais (Literais), coeficiente de correlação de Pearson (r na equação 8) entre os valores reais e os preditos do predicado alvo, erro quadrático médio (MSE) e erro absoluto médio (MAE), equações 9 e 10, conforme [H. Blockeel and Fierens 2005].

$$r = \frac{\sum_{i=1}^{n}(p_i - \overline{p})(a_i - \overline{a})}{\sqrt{\sum_{i=1}^{n}(p_i - \overline{p})^2 \sum_{i=1}^{n}(a_i - \overline{a})^2}} \tag{8}$$

$$MSE = \sqrt{\frac{\sum_{i=1}^{n}(p_i - a_i)^2}{n}} \tag{9}$$

$$MAE = \frac{\sum_{i=1}^{n}|p_i - a_i|}{n} \tag{10}$$

Onde, p_i é o valor predito para o exemplo i e a_i seu valor real. $\overline{p}$ e $\overline{a}$ são as médias e n o número de exemplos.

4.1. Experimentos com Artificial 1

O primeiro conjunto de dados (Artificial 1) contém dois atributos preditivos: $x(X)$ (determinado) e $y(Y)$ (não determinado). Cada exemplo contém 8 literais sobre y, para os quais os valores podem ser agregados. Todos os valores numéricos são aleatórios e uniformemente distribuídos entre 0 e 10. A função objetivo para este conjunto de dados está mostrada na figura 2.

Executamos validação cruzada 5x2 [Dietterich 1998]. Os resultados indicaram que o HTILDE-RT obteve medidas de qualidade muito próximas às do TILDE-RT com tempos de aprendizado menores e árvores mais simples. Realizamos o teste t pareado e corrigido [Nadeau and Bengio 2003] [Witten and Frank 2005] com confiança de 0,001 e o HTILDE-RT obteve significância estatística em termos de ter melhor eficiência e gerar árvores menores.

Embora o HTILDE-RT tenha obtido medidas de erro ligeiramente maiores, não houve diferença no coeficiente de correlação, que é a avaliação mais importante. Em

$x(X), X < 5?$
$+yes : maxY|y(Y) < 9?$
$|+yes : 4x(X) + 6avgY|y(Y)$
$|+no : 3maxY|y(Y) + 1$
$+no : avgY|y(Y) < 4?$
$|+yes : 3x(X) + 4$
$|+no : x(X) - 2maxY|y(Y) + 3avgY|y(Y)$

Figura 2. Função Objetivo do Artificial 1

	Tempo (s)	Nós	Literais	Coef. Pearson	MSE	MAE
TILDE-RT	1046,8	290,0	580,0	0,92	28,44	3,28
HTILDE-RT	334,6	162,0	1213,0	0,92	28,55	3,29
TILDE-RT*	1054,0	280,5	561,0	0,92	29,45	3,44
HTILDE-RT*	335,1	157,8	1173,3	0,92	29,56	3,45

Tabela 1. HTILDE-RT x TILDE-RT - Artificial 1

contrapartida, o TILDE-RT foi mais de três vezes mais lento e obteve teorias quase duas vezes mais extensas, i.e., árvores com mais nós. O ruído não afetou de forma impactante nenhum dos dois algoritmos.

O número maior de literais indica que, refinamentos com conjunções maiores separam melhor os conjuntos quando menos exemplos são observados, i.e., rodando em fluxos usando amostragem. No caso do TILDE-RT, por observar todos os exemplos, bastam poucos literais para conseguir a separação.

4.2. Experimentos com Artificial 2

O segundo conjunto de dados (Artificial 2) inclui três atributos preditivos: $x(X)$, $y(C, Y)$ e $z(Z)$, dos quais y é não determinado e possui 15 literais para cada exemplo. Igualmente, os valores numéricos para x, y e z são uniformemente distribuídos entre 0 e 10. A variável C no literal de y possui domínio booleano. Seguem, a função objetivo na figura 3 e os resultados na tabela 2.

$x(X), X < 6?$
$+yes : minY|y(_, Y) < 1?$
$|+yes : 2x(X) + 3maxY|y(true, Y) + 3$
$|+no : 2x(X) + 5minY|y(_, Y)$
$+no : -2x(X) + z(Z)?$

Figura 3. Função objetivo do Artificial 2

No Artificial 2 o HTILDE-RT foi estatisticamente mais rápido, com tempos mais de três vezes menores, com confiança de 0,001. Neste conjunto de dados o HTILDE-RT parece ter conseguido uma performance melhor, por ser um conjunto de dados mais

	Tempo (s)	Nós	Literais	Coef. Pearson	MSE	MAE
TILDE-RT	2632,7	132,0	264,0	0,99	1,48	0,95
HTILDE-RT	734,8	197,0	1756,0	0,99	1,49	0,95
TILDE-RT*	2636,6	123,0	246,0	0,98	2,48	1,25
HTILDE-RT*	723,1	198,0	1692,5	0,98	2,49	1,25

Tabela 2. HTILDE-RT x TILDE-RT - Artificial 2

complexo. Não houve diferença estatística significativa para o coeficiente de correlação nem para o erro absoluto quadrático. Desta vez o ruído afetou ambos algoritmos, porém de forma semelhante e discreta. Novamente, refinamentos com mais literais foram mais utilizados pelo HTILDE-RT.

Conforme argumentado em [Ikonomovska et al. 2011], à medida que acrescenta-se ruído aos dados no fluxo o algoritmo adia suas decisões para ajustar-se melhor à perturbação, reduzindo o tamanho dos modelos gerados, conforme observado com o HTILDE-RT. Entretanto, contemplou-se efeito parecido no TILDE-RT. Isto pode ser explicado pela estratégia de dividir-para-conquistar do algoritmo tradicional, onde a decisão de dividir uma folha necessitará de tão mais exemplos quanto maior for a magnitude da perturbação nos dados, logo é razoável ter-se árvores menores.

Destacamos que melhorias são possíveis com ajustes dos parâmetros $emin$, τ e δ e que outros resultados mostraram que ajustes do $emin$ melhoram o tamanho das teorias.

5. Conclusão

Neste trabalho apresentamos um algoritmo para aprender Árvores Relacionais de Regressão, HTILDE-RT, capaz de tornar o aprendizado mais rápido em bases de dados grandes. Tal algoritmo é uma modificação do HTILDE, o qual é baseado no TILDE e VFDT, para o problema de árvores relacionais de regressão.

Os experimentos indicaram que o HTILDE-RT foi capaz de acelerar o processo de aprendizado sem perdas aos modelos gerados, chegando a obter desempenho mais de três vezes mais rápido, sem diferenças significativas no coeficiente Pearson. Além disso, no dataset Artificial 1 conseguiu teorias quase duas vezes menores, apesar de não rodar usando todos os exemplos de uma só vez. A adição de ruído afetou de forma semelhante e discreta ambos algoritmos.

Como trabalhos futuros, desejamos variar os parâmetros δ, $emin$ e τ para melhores resultados e testar o HTILDE-RT em outros conjuntos de dados relacionais gigantes no problema da regressão. Planejamos estender o uso do limitante de Hoeffding no algoritmo ReMauve (Relational Model Trees) [Vens et al. 2006] bem como desenvolver o suporte a mudanças de distribuição [Hulten et al. 2001].

6. Agradecimentos

Gostaríamos de agradecer a Hendrik Blockeel e Jan Ramon por fornecerem o código fonte do TILDE, a Celine Vens, Daan Fierens, Jan Struyf pela ajuda com o código e em particular a Celine Vens pela ajuda com os dados artificiais. Agradecemos também a Carina Lopes pela ajuda com o HTILDE. Gostaríamos de agradecer à CAPES, ao CNPq, à FAPERJ e à FACEPE pelo suporte financeiro.

Referências

Blockeel, H., Dehaspe, L., Demoen, B., Janssens, G., mon, J., and Vandecasteele, H. (2002). Improving the efficiency of inductive logic programming through the use of query packs. *Journal of Artificial Intelligence Research*, 16:135–166.

Blockeel, H. and Raedt, L. D. (1998). Top-down induction of first-order logical decision trees. *Artificial Intelligence*, 101(1-2):285–297.

Blockeel, H., Raedt, L. D., Jacobs, N., and Demoen, B. (1999). Scaling up inductive logic programming by learning from interpretations. *Data Mining and Knowledge Discovery*, 3(1):59–93.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees*. Chapman & Hall, New York, NY.

Dietterich, T. G. (1998). Approximate statistical test for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923.

Domingos, P. and Hulten, G. (2000). Mining high-speed data streams. In *Knowledge Discovery and Data Mining*, pages 71–80.

H. Blockeel, L. D. J. R. J. S. A. V. A. C. V. and Fierens, D. (2005). *The ACE Data Mining System - User's Manual*.

Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. 58:13–30.

Hulten, G., Spencer, L., and Domingos, P. (2001). Mining time-changing data streams. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 97–106, San Francisco, CA. ACM Press.

Ikonomovska, E., Gama, J., and Džeroski, S. (2011). Learning model trees from evolving data streams. *Data Mining and Knowledge Discovery*, 23:128–168. 10.1007/s10618-010-0201-y.

Lopes, C. and Zaverucha, G. (2009). Htilde: scaling up relational decision trees for very large databases. In *Proceedings of the 2009 ACM Symposium on Applied Computing (SAC), Honolulu, Hawaii, USA*, pages 1475–1479.

Mitchell, T. (1997). *Machine Learning*. McGraw-Hill, New York.

Muggleton, S. (1991). Inductive logic programming. *New Generation Computing*, 8(4):295–318.

Nadeau, C. and Bengio, Y. (2003). Inference for the generalization error. *Machine Learning*, 52(3):239–281.

Vens, C., Ramon, J., and Blockeel, H. (2006). Remauve: A relational model tree learner. In Muggleton, S., Otero, R. P., and Tamaddoni-Nezhad, A., editors, *ILP*, volume 4455 of *Lecture Notes in Computer Science*, pages 424–438. Springer.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Sys. Morgan Kaufmann, second edition.

Um Modelo Estocástico baseado em Redes Neurais

Luciana C. Dias Campos[1], **Marley Maria B. R. Vellasco**[2], **Juan G. Lazo Lazo** [2]

[1]Departamento de Ciência da Computação
Universidade Federal de Juiz de Fora (UFJF)
Rua José Kelmer - Campus Universitário, Bairro São Pedro, 36036-330
Juiz de Fora - MG – Brazil

[2]Departamento de Engenharia Elétrica
Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio)
Caixa Postal 38097 – 22451-900 – Rio de Janeiro – RJ – Brazil

luciana.campos@ufjf.edu.br, {marley,juan}@ele.puc-rio.br

Abstract. *This paper presents the proposal of a generic model of stochastic process based on neural networks, called Neural Stochastic Process (NSP). The proposal model can be applied to problems involving phenomena of stochastic behavior and / or periodic features. The NSP's neural networks capture the behavior of the time series of these phenomena without requiring a priori information about the series, and generates synthetic time series with the same probabilities as the historical series. The NSP model was applied to the treatment of monthly inflows series and the results indicated that the synthetic series generated exhibit statistical characteristics similar to historical series.*

Resumo. *Esse artigo apresenta a proposta de um modelo genérico de processo estocástico baseado em redes neurais, denominado Processo Estocástico Neural (PEN). O modelo proposto pode ser aplicado em problemas envolvendo fenômenos de comportamento estocástico e/ou com características periódicas. Através das redes neurais do PEN captura-se o comportamento da série histórica desses fenômenos, sem necessitar de informações a priori sobre a série, e gera-se séries temporais sintéticas igualmente prováveis à série histórica. O modelo PEN foi aplicado no tratamento de séries de afluências mensais e os resultados indicaram que as séries sintéticas geradas apresentam características estatísticas similares às séries históricas.*

1. Introdução

Muitos problemas reais apresentam características complexas, tais como não-linearidade e comportamento caótico, que necessitam de modelos capazes de capturar as suas reais características para obter um tratamento apropriado [Müller 2007]. Porém, os modelos existentes apresentam limitação de aplicação a certos problemas ou por serem modelos lineares (cuja aplicação obtém resultados inconsistentes ou inadequados) ou porque necessitam de uma formulação complexa ou dependem de suposições a priori, exigindo um conhecimento do problema em um nível de detalhamento muitas vezes não disponível [Sanders 1995, Luna et al. 2006].

Isso motivou a elaboração de um modelo genérico de processo estocástico baseado em redes neurais para ser aplicado em problemas envolvendo fenômenos de comportamento estocástico e/ou que apresentam características periódicas das suas propriedades

probabilísticas, como média, variância, entre outras. Esse modelo intrinsecamente não-linear, denominado Processo Estocástico Neural (PEN), o qual captura o comportamento da série histórica para gerar séries temporais sintéticas, igualmente prováveis à série histórica, na solução de diversos tipos de problemas, tais como os que envolvem fenômenos climatológicos, econômicos, entre outros.

O modelo utiliza redes neurais *Multilayer Perceptron* (MLP) [Haykin 2001], com uma única camada escondida, treinadas com o algoritmo de aprendizado supervisionado *Levenberg- Marquardt* [Hagan and Menhaj 1994], uma variação do *back-propagation* [Rumelhart et al. 1986, Haykin 2001]. As redes MLP por serem aproximadores universais [Haykin 2001], têm sido muito aplicadas no estudo de séries temporais [Abelém 1994, Vemuri and Rodgers 1994, Prudêncio 2002, Gutiérrez 2003, Fayal 2008, Batista 2009, Zhao 2009].

O modelo proposto foi aplicado no tratamento de séries hidrológicas sazonais de intervalos de tempo mensais e os resultados mostraram que o PEN consegue gerar séries sintéticas com características similares às séries históricas.

O restante desse artigo é organizado da seguinte forma: a Seção 2 apresenta a descrição do modelo PEN, mostrando a sua formulação bem como o processo de geração de séries sintéticas. A Seção 3 descreve a aplicação do modelo PEN na geração de séries sintéticas de afluências mensais, sendo que a Seção 4 apresenta os resultados obtidos neste estudo de caso. Finalmente, a Seção 5 apresenta a conclusão desse artigo.

2. Descrição do Modelo

Através da rede neural do PEN captura-se as características da série temporal sem necessitar de fazer suposições a priori sobre o comportamento da série ou realizar algum tipo de decomposição na mesma, como deixá-la estacionária ou retirar algumas características como ciclo ou tendência. Por isso, é preciso que as entradas dessa rede neural apresentem uma memória de curto prazo, que é feita através de técnicas de ”janelamento” [Gutiérrez 2003]. Essas técnicas consistem em introduzir memória nos neurônios da primeira camada escondida, fornecendo aos mesmos valores passados da série temporal analisada.

No caso mais básico, onde a série temporal a ser analisada não é periódica, o PEN é formado por um única componente estocástica (CE). Uma CE é composta por uma rede neural (a qual é alimentada por uma janela temporal contendo os termos passados da série a ser modelada para gerar uma saída), e por uma distribuição de probabilidade que fornece valores aleatórios a serem somados na saída da rede neural na geração de séries sintéticas, como ilustra a Figura 1. Essa distribuição de probabilidade é obtida através dos resíduos da rede neural calculados durante o treinamento, correspondendo à diferença entre a saída desejada e a saída dada pela rede.

Para trabalhar com séries temporais periódicas, os parâmetros do modelo PEN devem se ajustar ao intervalo de tempo da série e também ao seu período. Por essa razão, o PEN é modelado com uma CE para cada período da série. Por exemplo, no caso de período mensal, o PEN é composto por 12 CEs - uma para cada mês.

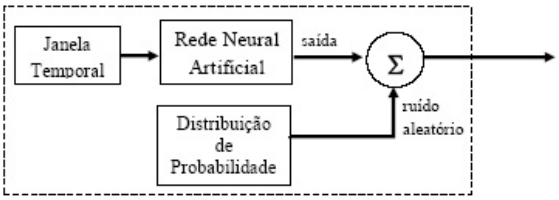


Figure 1. Componente Estocástica do Processo Estocástico Neural

2.1. Geração de séries sintéticas através do PEN

Seja $Z(t)$ uma série temporal com período sazonal s e n observações simultâneas em todos os períodos. O índice de tempo t é descrito pela equação 1:

$$t = (r-1) \cdot s + m \tag{1}$$

onde

- $r = 1 \ldots n$ é o número da observação de cada período da série;
- $m = 1 \ldots s$ corresponde a um período da série;
- $s \in N$ é o total de períodos da série;
- $n \cdot s$ é o tamanho da série observada.

Por exemplo, no caso de uma série mensal, r corresponde ao ano, m corresponde ao mês, $s = 12$ e n corresponde ao total de anos que a série foi observada. Caso a série não seja periódica, tem-se que $s = 1$.

Ao modelar essa série, o PEN é composto por s CEs, uma para cada período m da série, como ilustra o diagrama de blocos do PEN na Figura 2.

No início do processo de geração de séries sintéticas é preciso fornecer os primeiros valores de termos passados que compõem as janelas temporais das redes neurais do PEN. Esses valores iniciais são tirados diretamente da série histórica.

Quando o PEN é formado por mais de uma CE, ocorre um encadeamento entre elas, onde o valor da série fornecido pela CE de um período faz parte da janela temporal da rede neural da CE do período seguinte, como ilustra a Figura 3.

2.2. Formulação do PEN

A janela temporal da rede neural da CE do período m contém um número de termos passados da série, chamado de ordem e representada por p_m. Assim, para obter um valor para a série no instante de tempo t, $Z(t)$, o PEN acessa a componente CE do período m correspondente e a sua rede neural recebe na camada de entrada os termos: $Z(t-1), Z(t-2), \ldots, Z(t-p_m)$. Além disso, para reforçar o aprendizado do comportamento periódico da série acrescenta-se na entrada o valor da série do período anterior, correspondente a $Z(t-s)$.

A Figura 4 apresenta, em detalhes, um neurônio pertencente à camada escondida dessa rede neural de ordem p_m, cuja saída é dada pela equação 2.

$$y_i = \varphi\left(\omega_{i,0} \cdot Z(t-s) + \left(\sum_{j=1}^{p_m} \omega_{i,j} \cdot Z(t-j)\right) + \theta_i\right) \tag{2}$$

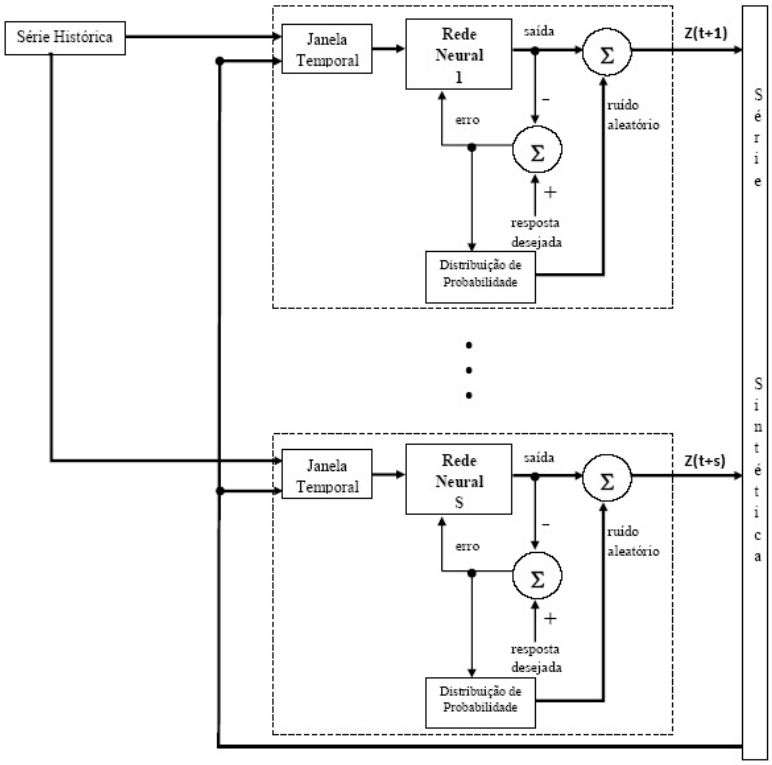


Figure 2. Geração de séries sintéticas no PEN

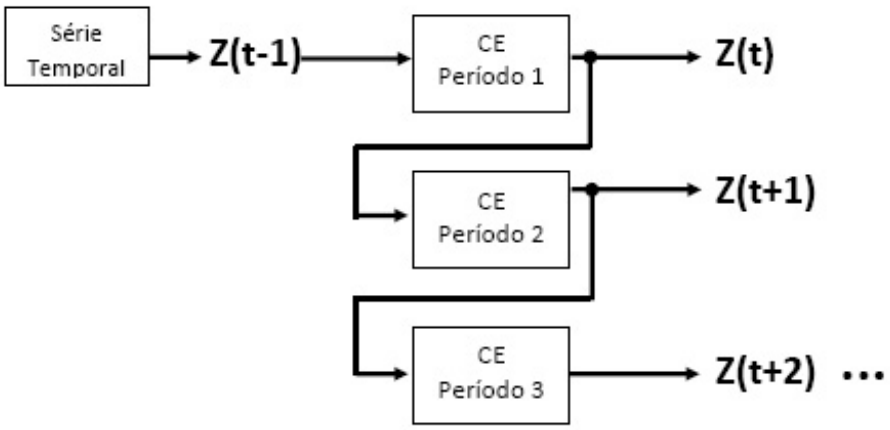


Figure 3. Encadeamento entre as CEs do Processo Estocástico Neural

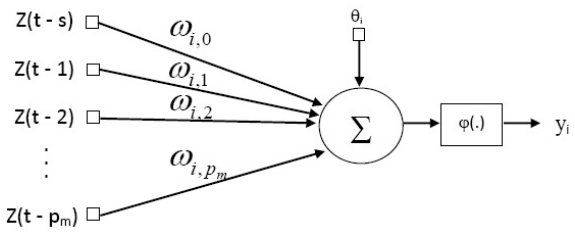


Figure 4. Neurônio da camada oculta de uma rede do PEN de ordem p_m

onde φ é a função de ativação do neurônio i, $\omega_{i,j}$ é o peso sináptico da conexão entre a entrada j e o neurônio i e θ_i é o *bias* desse neurônio.

Considerando que a rede neural de ordem p_m contém l_m neurônios na camada oculta, o neurônio da camada de saída é apresentado na Figura 5, sendo sua saída calculada pela equação 3.

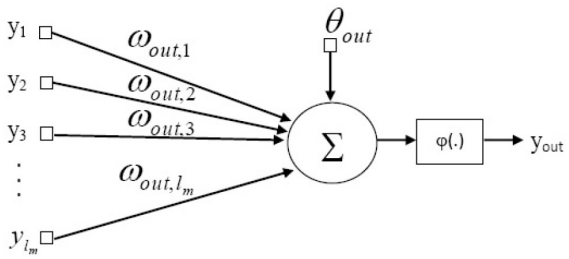


Figure 5. Neurônio de saída de uma rede do PEN com l_m neurônios na camada oculta

$$y_{out} = \varphi_{out}\left(\sum_{i=1}^{l_m} \omega_{out,i} \cdot y_i + \theta_{out}\right) \tag{3}$$

onde φ_{out} é a função de ativação do neurônio da camada de saída, representado por out, $\omega_{out,i}$ é o peso sináptico da conexão entre a entrada i (que corresponde à saída do neurônio i da camada oculta) e o neurônio out e θ_{out} é o *bias* desse neurônio.

Conforme ilustra a Figura 2, a saída de uma CE corresponde ao somatório da saída da rede neural com um valor aleatório, proveniente da distribuição de probabilidade dos resíduos da rede neural, obtidos da diferença entre a saída desejada e a saída da rede (y_{out}). Assim, a série temporal $Z(t)$, cujo índice de tempo t é descrito pela equação 1, é simulada através da equação 4.

$$Z(t) = y_{out} + \alpha(t) \tag{4}$$

onde $\alpha(t)$ é o valor aleatório proveniente da distribuição de probabilidade dos resíduos da rede neural da CE do período m correspondente.

3. Tratamento de Séries de Afluências Mensais

O Sistema Interligado Nacional (SIN) é um sistema de coordenação e controle, formado pelas empresas das regiões Sul, Sudeste, Centro-Oeste, Nordeste e parte da região Norte, o qual congrega o sistema de produção e transmissão de energia elétrica do Brasil, que é um sistema hidrotérmico de grande porte, com predominância de usinas hidrelétricas [ONS].

Atualmente, o SIN é segmentado em quatro subsistemas correspondentes aos sistemas interligados: Sul, Sudeste/Centro-Oeste, Nordeste e Norte. Por se tratar de um sistema de grande porte, nos planejamento de médio e longo prazo ocorre a agregação dos reservatórios das usinas em reservatórios equivalentes de energia, um para cada subsistema. Ocorre também a agregação das afluências às usinas em Energias Naturais Afluentes (ENA), que correspondem à estimativa da energia que pode ser gerada com todas

as vazões afluentes a cada um dos reservatórios que compõem aquele reservatório equivalente, segundo uma política de operação [Pereira and Pinto 1984].

As ENAs são séries não estacionária, devido aos períodos de cheia e seca do ano, e sazonais com períodos de 12 meses, as quais, geralmente, apresentam correlações periódicas [Luna et al. 2006]. No contexto do planejamento da operação energética do sistema hidrotérmico brasileiro a geração de cenários é muito importante pois o único cenário disponível na prática, que é o registro de afluências observadas (chamado de série histórica), é insuficiente para compor uma amostra de tamanho necessário para estimar índices de risco aceitáveis [Pereira and Pinto 1984]. Através dos cenários é possível reproduzir as características básicas da série histórica, extraindo melhor as informações dessa série temporal, permitindo a avaliação de riscos e incertezas pertinentes a um sistema hidroelétrico. Desse modo, aplicou-se o PEN para tratar as incertezas dessas séries de ENA gerando um conjunto de séries sintéticas.

A modelagem do processo estocástico neural para o tratamento de séries de ENA é composta de 4 sistemas PEN(p, l), um para cada subsistema do SIN. Como as séries de ENA são séries sazonais com períodos de 12 meses, os sistemas PEN(p, l) são compostos por 12 CEs, uma para cada mês. Tem-se que:

p = $p_1 \dots p_{12}$**:** vetor com a ordem da rede neural de cada CE;
l = $l_1 \dots l_{12}$**:** vetor com o número de neurônios na camada escondida da rede neural de cada CE.

A ordem p_m da rede neural da CE do período m pode variar de 1 até 11, que corresponde ao número de termos passados máximo permitido para manter a sazonalidade anual. Através de experimentos preliminares teve-se informações de que alguns meses normalmente precisam de ordens menores que outros. Logo, decidiu-se testar 4 tipos de ordem para cada rede: defasagem de 3, 6 , 9 e 11 meses. Decidiu-se testar, para cada ordem definida, configurações desde 1 até 20 neurônios na camada oculta, dando um total de 80 configurações a serem avaliadas para cada mês m. Além disso, cada uma dessas 80 configurações de redes neurais são treinadas 10 vezes, com diferentes inicializações dos pesos sinápticos, a fim de evitar que a escolha de uma configuração de rede seja influenciada pelo mal desempenho devido a uma inicialização ruim dos pesos sinápticos.

Para ajustar o modelo PEN(p, l) utilizou-se, para cada subsistema do SIN, um histórico consolidado de ENA que apresenta valores desde 1931 até 2005. Esse histórico de ENA foi dividido em 2 conjuntos: treinamento e validação. Durante o treinamento das redes neurais, faz-se etapas de validação do treinamento para evitar que ocorra um super ajustamento dos pesos sinápticos das redes neurais. A etapa de validação das redes neurais do PEN consiste na geração uma série de n anos composta com as saídas das redes. Por esse motivo, usa-se os n últimos anos do histórico para compor o conjunto de padrões entrada-saída de validação e, todos os anos anteriores compõem o conjunto de padrões entrada-saída de treinamento. Nesse trabalho, como decidiu-se gerar séries sintéticas na mesma quantidade de anos que é utilizado no planejamento atual do SIN, tem-se que o conjunto de validação é composto pelos últimos 5 anos do histórico.

Uma vez treinadas todas as configurações de redes neurais, escolhe-se a rede do mês m que obteve o melhor desempenho para compor a $m-$ésima CE do PEN. Como o conjunto de validação contém apenas 5 anos, decidiu-se verificar o desempenho das redes neurais treinadas em relação a um número maior de padrões de dados. Por essa razão,

unificou-se então os conjuntos de padrões de treinamento e validação. Criou-se, com esse padrão de dados unificado, uma série com as saídas das redes neurais e comparou-se essa série com as saídas desejadas do padrão de dados unificado. Utilizou-se como medida de desempenho o erro médio percentual absoluto (MAPE) [Tang et al. 1991], que é muito utilizado para validar modelos de séries temporais, calculado pela equação 5.

$$MAPE = \frac{1}{x} \cdot \sum_{k=1}^{x} \left| \frac{Z(k) - C((k-1) \cdot s + m)}{Z(k)} \right| \tag{5}$$

onde m é o mês, x corresponde ao total de elementos do mês m na série, k é o índice do elemento, $s = 12$ é o total de meses, $Z(k)$ é a k-ésima saída desejada do padrão unificado do mês m, C é a série criada com as saídas das redes neurais.

Uma vez escolhida a rede neural que irá compor a m-ésima CE do PEN, ajusta-se a série de resíduos, obtidos da diferença entre a saída da rede e os padrões de saída do conjunto de treinamento, à uma distribuição teórica de probabilidade para compor essa CE.

4. Resultados da Simulação

A Tabela 1 apresenta, para cada subsistema do SIN, as configurações de redes neurais das CEs do PEN. A tabela contém a ordem e o número de neurônios da camada escondida da rede neural de cada mês no PEN de cada subsistema.

Table 1. Configurações das Redes Neurais selecionadas

Subsistema Sudeste-Centro-Oeste												
Mês	1	2	3	4	5	6	7	8	9	10	11	12
Ordem	3	9	6	11	3	3	9	6	11	3	3	6
# Neurônios	19	20	15	7	10	10	3	16	16	19	12	9
Subsistema Sul												
Mês	1	2	3	4	5	6	7	8	9	10	11	12
Ordem	6	9	9	11	3	6	3	6	11	3	11	6
# Neurônios	3	17	1	3	19	10	14	1	2	3	3	1
Subsistema Nordeste												
Mês	1	2	3	4	5	6	7	8	9	10	11	12
Ordem	11	3	11	3	3	11	6	11	9	6	9	9
# Neurônios	9	6	6	13	2	19	1	1	3	1	5	7
Subsistema Norte												
Mês	1	2	3	4	5	6	7	8	9	10	11	12
Ordem	11	9	6	11	9	11	6	11	9	3	11	11
# Neurônios	11	14	4	15	16	3	1	8	17	19	18	15

Com cada um dos modelos PEN(p, l) da Tabela 1, gerou-se 200 séries sintéticas de 5 anos (60 meses). Com o intuito de analisar a qualidade dessas séries sintéticas geradas, utilizou-se testes estatísticos de aderência [Montgomery and Runger 1971] para verificar se as séries sintéticas são igualmente prováveis à série histórica de ENA.

Aplicou-se o teste *t* para avaliar se a média das séries sintéticas é estatisticamente igual à média histórica. A hipótese é que a média dos 200 valores de cada mês em cada ano é estatisticamente igual à média histórica do mês correspondente. Em cada teste realizado obtém-se um p-valor e caso esse p-valor esteja acima do nível de significância de 5%, aceita-se essa hipótese. A porcentagem dos 60 p-valores (um para cada mês da série) acima do nível de significância indica o desempenho do modelo PEN para gerar séries sintéticas, indicando o quanto o modelo conseguiu reproduzir o primeiro momento da série histórica.

De maneira análoga foi aplicado o teste de Levene para avaliar se a variância de cada período dos cenários é estatisticamente igual à variância histórica do mês correspondente, e o teste de Kolmogorov-Smirnov (K-S) para verificar se os cenários provêm da mesma distribuição de probabilidade do histórico, indicando que o modelo reproduziu corretamente o comportamento da série histórica.

Como o histórico de ENA foi dividido em 2 conjuntos, um para o treinamento e o outro para a validação das redes neurais do PEN, fez-se uma avaliação das séries sintéticas com esses dois conjuntos. A Tabela 2 apresenta os resultados dos testes de aderência obtidos. Observe que todos modelos PEN ajustados com as séries dos subsistemas tiveram em todos os testes aderências acima de 50%. Os melhores ajustes ocorreram com os modelos dos subsistemas Sudeste/Centro-Oeste e Sul. O modelo do Nordeste teve aderências mais altas nos testes com o histórico de Treinamento do que com o histórico de Validação. Já o modelo do Norte conseguiu melhor aderência com o teste de Levene e aderências mais baixas com o testes K-S.

Table 2. Configurações das Redes Neurais selecionadas

Subsistema	Histórico					
	Treinamento			Validação		
Teste	t	Levene	K-S	t	Levene	K-S
Sudeste/Centro-Oeste	94%	95%	94%	97%	100%	97%
Sul	75%	90%	90%	95%	99%	99%
Nordeste	97%	99%	94%	69%	64%	54%
Norte	85%	90%	64%	74%	99%	67%

A Figura 6 apresenta a envoltória dos 200 cenários (séries sintéticas) de 5 anos de ENA gerados pelo modelo PEN, a curva da média de cada cada mês em cada ano das séries sintéticas e também a curva da média mensal do conjunto histórico replicada nos 5 anos, para ser comparada com o comportamento da média dessas séries sintéticas geradas em todos esses 60 meses. Isso foi feito para os dois conjuntos históricos em cada subsistema.

Observa-se que a média dos cenários reproduziram melhor o comportamento da média histórica do conjunto de treinamento. Já o comportamento da média histórica do conjunto de validação não foi tão bem reproduzido, principalmente pelas séries sintéticas geradas pelo modelo PEN do subsistema Sul. Porém, em todos os subsistemas, a média dos conjuntos históricos ficaram dentro das envoltórias dos cenários. Assim, de uma maneira geral, pode-se concluir que para todos os subsistemas as envoltórias dos cenários acompanharam bem o desempenho das séries históricas de ENA.

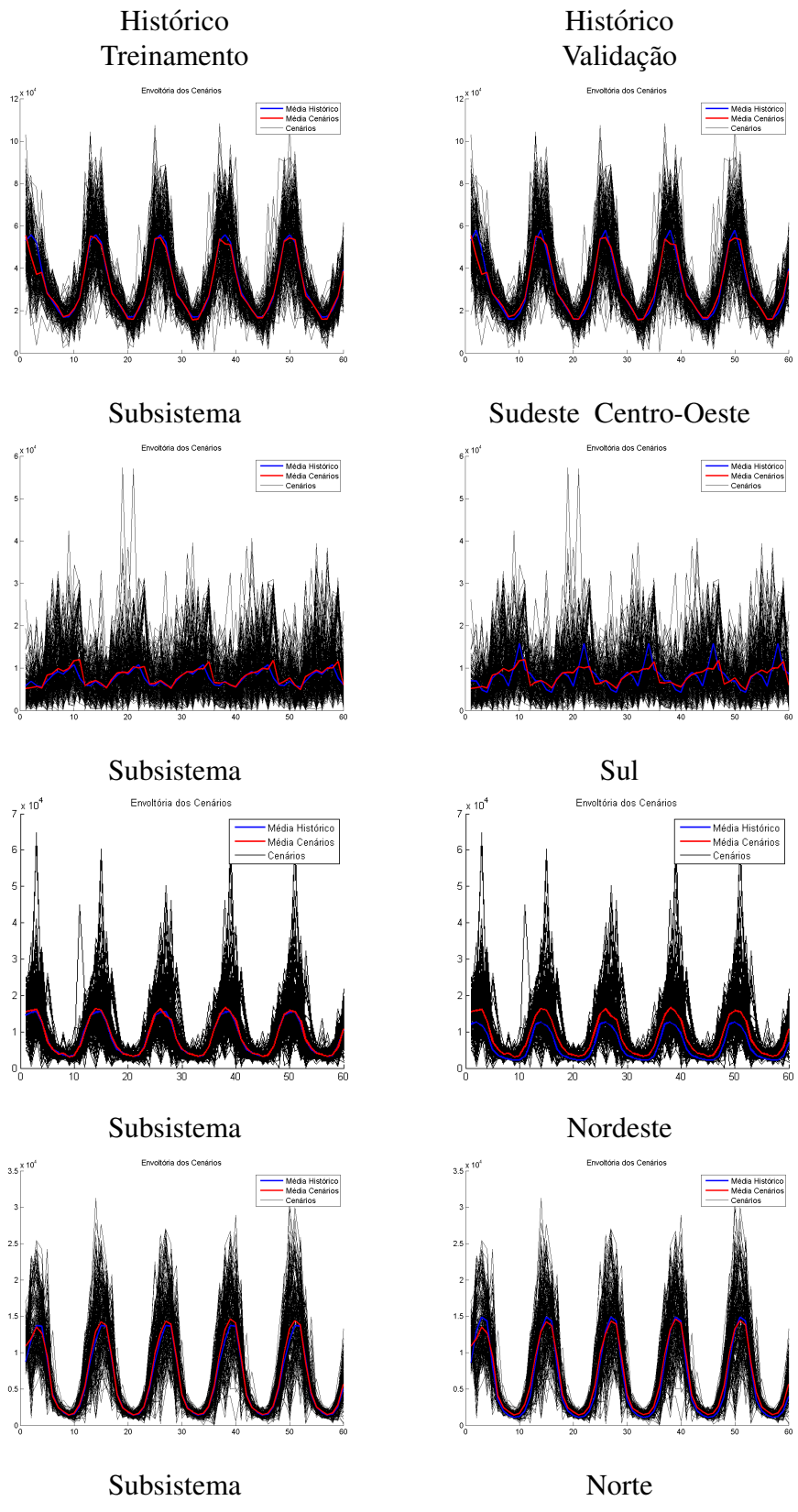


Figure 6. Envoltória dos cenários gerados no PEN

5. Conclusão

O objetivo desse trabalho foi a elaboração de um novo modelo de processo estocático genérico, intrinsecamente não-linear, passível de aplicação em uma gama de problemas de fenômenos de comportamento estocástico e/ou com características periódicas de suas propriedades.

Dado que as redes neurais são modelos não-lineares, simples de entendimento e implementação, capazes de identificar e assimilar características de séries temporais históricas, tais como sazonalidade, periodicidade e tendência, decidiu-se utilizá-las como base do modelo proposto, denominado Processo Estocástico Neural (PEN). A meta desse modelo é gerar séries temporais sintéticas igualmente prováveis à série histórica, abordando qualquer período de tempo, na quantidade necessária.

Aplicou-se o PEN no tratamento das incertezas das séries de afluências mensais, chamadas de Energia Natural Afluente (ENA). As ENAs são séries não estacionária, devido aos períodos de cheia e seca do ano, e sazonais com períodos de 12 meses, as quais, geralmente, apresentam correlações periódicas. Através do PEN gerou-se um conjunto de séries sintéticas com o intuito de reproduzir as características básicas da série histórica. Gerou-se 200 séries sintéticas de 5 anos de ENA para cada subsistema do Sistema Interligado Nacional.

Em cada conjunto de séries sintéticas fez-se uma avaliação com alguns testes de aderência para verificar se as mesmas são estatisticamente similares às séries históricas de ENA. Verificou-se que as séries sintéticas geradas por esse modelo PEN apresentaram boas aderência ao conjunto de séries históricas e que as envoltórias desses cenários englobam as médias históricas de ENA. Com isso conclui-se que o modelo PEN é capaz de capturar o comportamento da série histórica e gerar séries sintéticas com características semelhantes.

Como trabalhos futuros é interessante utilizar redes neurais recorrentes no lugar das redes neurais MLP nas componentes estocásticas do PEN. Pois as redes neurais recorrentes são capazes de representar mapas dinâmicos não-lineares arbitrários, comumente encontradas em tarefas de predição de série temporais não-lineares [Narendra and Parthasarathy 1990], apresentando melhor desempenho que técnicas lineares tradicionais e redes neurais MLP [Lendasse 2004]. Uma vez que para processar séries temporais as redes neurais feedforward MLP podem ser adaptadas através de uma entrada com sinais de atrasos (regressor) e treinadas pelo algoritmo backpropagation padrão, esses modelos assim adaptados podem ser facilmente convertidas em arquiteturas recorrentes simples. Essa conversão é feita através de realimentações de ativações dos neurônios das camadas escondidas ou de saída, dando origem, respectivamente, às redes de Elman e de Jordan [Júnior 2006], sendo esses os modelos indicados para substituir as redes MLP do PEN. Outra idéia é substituir a rede neural MLP de cada componente estocástica do PEN por uma rede do tipo NARX. Pois, segundo Bengio [Bengio et al. 1994], aprender dependências de longo prazo com técnicas de gradiente-descendente é mais eficiente numa classe de RNAs conhecida como modelos NARX (Nonlinear AutoRegressive model with eXogenous inputs) do que em modelos recorrentes baseados na rede MLP, quando aplicado à identificação de sistemas de entrada-saída não-lineares [Júnior 2006].

References

Ons - operador nacional do sistema elétrico. http://www.ons.com.br.

Abelém, A. J. G. (1994). Redes neurais artificiais na previsão de séries temporais. Dissertação de mestrado, Pontífica Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica.

Batista, A. L. F. (2009). Modelos de séries temporais e redes neurais artificiais na previsão de vazão. Mestrado em engenharia de sistemas, Universidade Federal de Lavras, Lavras, Minas Gerais.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is dificult. *IEEE Transactions on Neural Networks*, 5(2):157–166.

Fayal, M. A. A. (2008). Previsão de vazão por redes neurais artificiais e transformada wavelet. Dissertação de mestrado, Pontífica Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica.

Gutiérrez, J. L. C. (2003). Monitoramento da instrumentação da barragem de corumbá-i por redes neurais e modelos de box and jenkins. Dissertação de mestrado, Pontífica Universidade Católica do Rio de Janeiro, Departamento de Engenharia Civil.

Hagan, M. and Menhaj, M. (1994). Training feedforward networks with the marquardt algorithm. *IEEE Transactions on Neural Networks,*.

Haykin, S. (2001). *Redes Neurais: Princípios e Prática*. Bookman, Porto Alegre, RS.

Júnior, J. M. P. M. (2006). Redes neurais dinâmicas para predição e modelagem não-linear de séries temporais. Dissertação de mestrado, Universidade Federal do Ceará, Departamento de Engenharia de Teleinformática.

Lendasse, A. (2004). Time series prediction competition the cats benchmark. In *International Joint Conference on Neural Networks (IJCNN'04)*, volume S.l., pages 1615–1620.

Luna, I., Ballini, R., and Soares, S. (2006). Técnica de identificação de modelos lineares e não-lineares de séries temporais. *Revista Controle e Automação*, 17(3):245–256.

Müller, D. (2007). *Processos Estocásticos e Aplicações*, volume Colecção Económicas - 2.ª Série of *Direito Financeiro E Tributário*. Almedina.

Montgomery, D. C. and Runger, G. C. (1971). *Estatística Aplicada e Probabilidade para Engenheiros*. Editora LTC, Rio de Janeiro, RJ, 2ª edição edition.

Narendra, K. S. and Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1):4–27.

Pereira, M. V. F. and Pinto, L. M. V. G. (1984). Operation planning of large-scale hydroelectric systems. In *Proc. 8th Power System Computation Cont.*

Prudêncio, R. B. C. (2002). Projeto híbrido de redes neurais. Mestrado em ciências da computação, Universidade Federal de Pernambuco, Recife, PE.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by backpropagating errors. pages 533–536, Nature (London), 323.

Sanders, N. R. (1995). Managing the forecasting function. *Industrial Managment and Data Systems*, 95(4):12–18.

Tang, Z., Almeida, C., and Fishiwick, P. A. (1991). Time series forecasting using neural networks vs box-jenkins methodology. In *Simulation*, volume 57, pages 303–310.

Vemuri, V. R. and Rodgers, R. D. (1994). Artificial neural networks: Forecasting time series. *IEEE Computer Society Press*.

Zhao, H. (2009). A chaotic time series prediction based on neural network: Evidence from the shanghai composite index in china. In *Test and Measurement, 2009. ICTM '09. International Conference on*, volume 2, pages 382 –385.

Echo State Incremental Gaussian Mixture Network for Spatio-Temporal Pattern Processing

Rafael C. Pinto[1], **Paulo M. Engel, Milton R. Heinen**

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{rcpinto,engel,mrheinen}@inf.ufrgs.br

Abstract. *This work introduces a novel neural network algorithm for online spatio-temporal pattern processing, called Echo State Incremental Gaussian Mixture Network (ESIGMN). The proposed algorithm is a hybrid of two state-of-the-art algorithms: the Echo State Network (ESN), used for spatio-temporal pattern processing, and the Incremental Gaussian Mixture Network (IGMN), applied to aggressive learning in online tasks. The algorithm is compared against the conventional ESN in order to highlight the advantages of the IGMN approach as a supervised output layer.*

Resumo. *Este trabalho introduz um novo algoritmo de redes neurais para processamento online de padrões espaço-temporais, chamado Echo State Incremental Gaussian Mixture Network (ESIGMN). O algoritmo proposto é um híbrido de dois algoritmos estado-da-arte: a Echo State Network (ESN), usada para processamento de padrões espaço-temporais, e a Incremental Gaussian Mixture Network (IGMN), aplicada ao aprendizado agressivo em tarefas online. O algoritmo é comparado com a ESN convencional a fim de destacar as vantagens da abordagem IGMN como camada supervisionada de saída.*

1. Introduction

This work presents a novel neural network algorithm for online spatio-temporal pattern processing, called Echo State Incremental Gaussian Mixture Network (ESIGMN). The proposed algorithm is a hybrid of two state-of-the-art algorithms: the Echo State Network (ESN) [Jaeger 2001], based on the Reservoir Computing (RC) paradigm, used for spatio-temporal pattern processing, and the Incremental Gaussian Mixture Network (IGMN, previously known as Incremental Probabilistic Neural Network or IPNN on early versions) [Heinen and Engel 2010] [Heinen 2011], applied to aggressive learning of online tasks.

The ESN is a recurrent neural network which can be applied to spatio-temporal supervised learning. While being able to learn incrementally through Recursive Least Squares (RLS) [Hayes 1996] [Jaeger 2003] or conventional Least Mean Squares (LMS) [Widrow 1966], it is really powerful in batch mode, i.e. when all training data is given at once. Its incremental versions, which can be applied to online tasks, learn slowly, requiring several presentations of each pattern.

The IGMN, on the other hand, is an one-shot incremental learning algorithm, needing only a single scan through the training data in order to build a consistent model. But the IGMN is essentially a static algorithm, meaning that it basically works only for static tasks, where any pattern is independent of past ones.

The ESIGMN merges those two algorithms, giving the best of the two worlds. The ESN gives temporal processing capabilities to the IGMN, while the IGMN gives one-shot incremental learning capabilities to the ESN.

The algorithm is compared against the conventional ESN in order to highlight the advantages of the IGMN approach as a supervised output layer, instead of the conventional ESN output layer.

This work is structured as follows: In section 2, the Reservoir Computing (RC) approach is described, presenting the ESN as an instantiation of that approach. The IGMN is described in section 3. Section 4 presents the new algorithm, the ESIGMN. In section 5, the ESIGMN is compared to a conventional ESN in time-series prediction tasks. Section 6 finishes this work with concluding remarks about the new algorithm and future works.

2. Reservoir Computing

Reservoir Computing (RC) is a recently coined term for a neural pattern processing paradigm where a random, non-linear, fixed and large hidden layer with recurrent connections, called a reservoir, is used as an excitable medium where interesting dynamic features of the data stream can be extracted. It's similar to a random filter bank, producing transformations over the input data. Albeit the reservoir not being trained, its output states are sufficient to train linear regression / classification algorithms on non-linear dynamic tasks successfully, thus potentially turning any static linear algorithm into a non-linear dynamic one. But since the reservoir is random, large reservoirs are required, to increase the chances of obtaining useful transformations.

This paradigm was found independently by different researchers at different times, also in distinct research fields like computational neuroscience and machine learning: Temporal Recurrent Neural Network [Dominey 1995]; Liquid State Machines [Natschläger et al. 2002]; Echo State Networks [Jaeger 2001]; Decorrelation-Backpropagation Learning [Steil 2004].

This work will incorporate the ESN as the base for the new algorithm, since it is composed by the default neuron model used in artificial neural networks (like the Multi-Layer Perceptron), is very simple to implement, is probably the most widely used RC artificial neural network in computer science and gave excellent results in previous works, e.g., predicting chaotic dynamics (three orders of magnitude improved accuracy [Jaeger and Haas 2004]), nonlinear wireless channel equalization (two orders of magnitude improvement [Jaeger and Haas 2004]), the Japanese Vowel benchmark (zero test error rate, previous best was 1.8% [Jaeger et al. 2007]), financial forecasting (winner of the international forecasting competition NN3), and in isolated spoken digits recognition (improvement of word error rate on benchmark from 0.6% of previous best system to 0.2% [Verstraeten et al. 2006]).

2.1. The Echo State Network

The ESN consists basically of an input layer, a reservoir and an output layer. The weights between input layer and reservoir (here denoted as $\mathbf{W}_{in}$) as well as the recurrent reservoir weights ($\mathbf{W}$) are randomly chosen and fixed – no training is necessary. The weights from input layer and reservoir to the output layer ($\mathbf{W}_{out}$) are trained in batch mode by linear Least Squares Fitting, or in incremental mode by Recursive Least Squares (RLS)

or any other incremental linear learning algorithm, like conventional Least Mean Squares (LMS). This work will focus on the incremental mode, since we need to compare the ESN to an incremental algorithm (ESIGMN). The reservoir activation function ($f(.)$) is the hyperbolic tangent, and the output layer activation function ($g(.)$) is usually the identity, although any differentiable function could be used. An example of an ESN can be seen in figure 1.

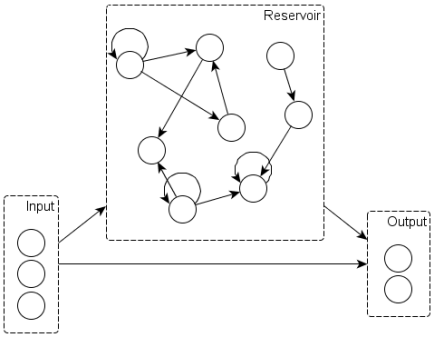


Figure 1. An example of Echo State Network with 3 input nodes, 2 output nodes and 8 reservoir nodes. All weights going to the output layer (W_{out}) must be trained, while every other weight is fixed.

In order to have a stable reservoir, however, the echo state property must be assured. It means that when a null vector is continually fed into the reservoir as its input, its state vector must decay to a null vector too, as time tends towards infinity. In practical terms, this can be ensured by rescaling the largest absolute eigenvalue $|\lambda_{max}|$ of the reservoir recurrent weight matrix $\mathbf{W}$, i.e. its spectral radius, to a value in the interval (0, 1). There is controversy about the necessity or sufficiency of this condition to ensure the echo state property [Buehner and Young 2006], but it's known to work well in practice [Jaeger 2001]. A proven and stronger sufficient condition is to rescale the largest singular value of the weight matrix to the interval (0, 1) [Jaeger 2001].

Following, a brief explanation of the ESN algorithm is shown. Given input $\mathbf{u}$ at time t, a processing step of the ESN is computed as follows:

The new state $\mathbf{x}$ of the reservoir at time t is obtained by the following equation:

$$\mathbf{x}(t) = f(\mathbf{W}_{in}\mathbf{u}(t) + \mathbf{W}\mathbf{x}(t-1)) \tag{1}$$

while the output $\mathbf{y}$ at time t of the ESN is given by (assuming identity activation function):

$$\mathbf{y}(t) = \mathbf{W}_{out}[\mathbf{u}(t); \mathbf{x}(t)] \tag{2}$$

where [.;.] is the vector concatenation operation. The prediction error $\mathbf{e}$ at time t is given by:

$$\mathbf{e}(t) = \mathbf{d}(t) - \mathbf{y}(t) \tag{3}$$

where $\mathbf{d}(t)$ is the target vector (teacher signal) at time t. The output weights $\mathbf{W}_{out}$ are modified, in the online case, by stochastic gradient descent as follows:

$$\Delta\mathbf{W}_{out} = \eta\mathbf{e}(t)[\mathbf{u}(t); \mathbf{x}(t)]^T \tag{4}$$

where η is a learning rate in the interval [0, 1].

3. Incremental Gaussian Mixture Network

The IGMN (Incremental Gaussian Mixture Network) algorithm [Heinen and Engel 2010] uses an incremental approximation of the EM algorithm [Dempster et al. 1977], the IGMM (Incremental Gaussian Mixture Model) [Engel and Heinen 2011]. It creates and continually adjusts probabilistic models consistent to all sequentially presented data, after each data point presentation, and without the need to store any past data points. Its learning process is aggressive, or "one-shot", meaning that only a single scan through the data is necessary to obtain a consistent model.

IGMN (and IGMM) adopts a gaussian mixture model of distribution components (known as a *cortical region*) that can be expanded to accommodate new information from an input data point, or reduced if spurious components are identified along the learning process. Each data point assimilated by the model contributes to the sequential update of the model parameters based on the maximization of the likelihood of the data. The parameters are updated through the accumulation of relevant information extracted from each data point.

Differently from IGMM, however, the IGMN is capable of supervised learning, simply by assigning any of its input vector elements as outputs (any element can be used to predict any other element). This architecture is depicted on figure 2.

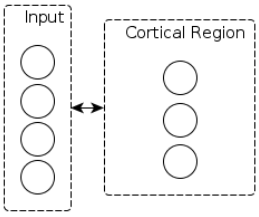


Figure 2. An example of IGMN with 4 input nodes and 3 gaussian components. Any input element can be predicted by using any other element, which means that the input vector can actually be divided into input and output elements.

3.1. Learning

The algorithm starts with no components, which are created as necessary (see subsection 3.2). Given input $\mathbf{x}$, the IGMN algorithm processing step is as follows. First, the likelihood for each component j is calculated:

$$p(\mathbf{x}|j) = \frac{1}{(2\pi)^{D/2}\sqrt{|\mathbf{C}_j|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T\mathbf{C}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)\right) \tag{5}$$

where D is the input dimensionality, $\boldsymbol{\mu}_j$ the j^{th} component mean and $\mathbf{C}_j$ its covariance matrix.

After that, posterior probabilities are calculated for each component as follows:

$$p(j|\mathbf{x}) = \frac{p(\mathbf{x}|j)p(j)}{\sum_{q=1}^{M} p(\mathbf{x}|q)p(q)} \qquad \forall j \tag{6}$$

where M is the number of components. Now, parameters of the algorithm must be updated according to the following equations:

$$v_j(t) = v_j(t-1) + 1 \tag{7}$$

$$sp_j(t) = sp_j(t-1) + p(j|\mathbf{x}) \tag{8}$$

$$\mathbf{e}_j = \mathbf{x} - \boldsymbol{\mu}_j \tag{9}$$

$$\omega_j = \frac{p(j|\mathbf{x})}{sp_j} \tag{10}$$

$$\Delta\boldsymbol{\mu}_j = \omega_j \mathbf{e}_j \tag{11}$$

$$\boldsymbol{\mu}_j(t) = \boldsymbol{\mu}_j(t-1) + \Delta\boldsymbol{\mu}_j \tag{12}$$

$$\mathbf{C}_j(t) = \mathbf{C}_j(t-1) - \Delta\boldsymbol{\mu}_j\Delta\boldsymbol{\mu}_j^T + \omega\left[\mathbf{e}\mathbf{e}^T - \mathbf{C}_j(t-1)\right] \tag{13}$$

$$p(j) = \frac{sp_j}{\sum_{q=1}^{M} sp_q} \tag{14}$$

where sp_j and v_j are the accumulator and the age of component j, respectively, and $p(j)$ is its prior probability.

3.2. Creating New Components

In order to create new components, the cortical region must reconstruct its input **x** based on the posterior probabilities obtained in equation 6. Let **x** be a concatenation of two vectors **a**, the actual input or known part, and **b**, the target / desired value ($\mathbf{x} = [\mathbf{a}; \mathbf{b}]$). Then the reconstruction of **b** given **a** is obtained by the following equation:

$$\hat{\mathbf{b}} = \sum_{j=1}^{M} p(j|\mathbf{a})\boldsymbol{\mu}_{j,b} \tag{15}$$

in the naïve approach (only diagonal covariance matrixes are used), where $\boldsymbol{\mu}_{j,b}$ is just the target part of the jth component's mean vector. Note that it's only an average of the region's means weighted by their posterior probabilities. This approach will be called here ESIGMNn (naïve). The full multivariate version (ESIGMN) is as follows:

$$\hat{\mathbf{b}} = \sum_{j=1}^{M} p(j|\mathbf{a})(\boldsymbol{\mu}_{j,b} + \mathbf{C}_{j,ba}\mathbf{C}_{j,a}^{-1}(\mathbf{a} - \boldsymbol{\mu}_{j,a})) \tag{16}$$

where $\mathbf{C}_{j,ba}$ is the submatrix of the jth component covariance matrix associating the input and output parts of the data, $\mathbf{C}_{j,a}$ is the submatrix corresponding to the input part only and $\boldsymbol{\mu}_{j,a}$ is just the input part of the jth component's mean vector. After reconstructing the input, the reconstruction error can be obtained by:

$$\epsilon = \max_{i \in D} \left[\frac{|x_i - \hat{x}_i|}{X_{max,i} - X_{min,i}} \right] \tag{17}$$

where $X_{max,i}$ and $X_{min,i}$ are the ith column's maximum and minimum values expected for the entire dataset (just approximated values are ok, since IGMN-based algorithms don't require availability of the entire dataset beforehand). If there are no components or ϵ is greater than a manually chosen threshold ϵ_{max} (e.g., 0.1), then a new component is created and initialized as follows:

$$\boldsymbol{\mu} = \mathbf{x}; \qquad sp = 1; \qquad v = 1; \qquad p(j) = \frac{1}{\sum_{i=1}^{M} sp_i}; \qquad \mathbf{C} = \sigma_{ini}^2$$

where M already includes the new component and σ_{ini} can be obtained by:

$$\sigma_{ini} = diag(\delta[X_{max} - X_{min}]) \tag{18}$$

where δ is a manually chosen scaling factor (e.g., 0.1) and diag returns a diagonal matrix having its input vector in the main diagonal.

3.3. Removing Spurious Components

A component j is removed whenever $v_j > v_{min}$ and $sp_j < sp_{min}$, where v_{min} and sp_{min} are manually chosen (e.g., 5.0 and 3.0, respectively). In that case, also, $p(q)$ must be adjusted for all $q \in M$, $q \neq j$, using equation 14.

3.4. Recalling

In IGMN, any element can be predicted by any other element. This is done by reconstructing data from the target elements (**b**) by estimating the posterior probabilities using only the input elements, as follows:

$$p(j|\mathbf{a}) = \frac{p(\mathbf{a}|j)p(j)}{\sum_{q=1}^{M} p(\mathbf{a}|q)p(q)} \qquad \forall j \tag{19}$$

It's similar to equation 6, except that it uses the actual input vector **a** instead of the full **x** vector, with the target elements **b** removed from calculations. After that, **b** can be reconstructed using equation 15 or 16.

4. The Echo State Incremental Gaussian Mixture Network

The ESIGMN is a temporal extension of the IGMN algorithm, which is augmented with an ESN-style reservoir between its input and cortical region. The reservoir is responsible

for mapping the input space into a feature space which captures temporal dynamics of the data, as with the ESN. But instead of feeding the input and the reservoir state into a linear output layer, they are fed into an unmodified IGMN in the ESIGMN algorithm. The IGMN does its usual spatial processing, but its inputs already incorporate temporal information. This architecture can be seen in figure 3.

Therefore, all IGMN equations from the previous section apply, with a difference only in the source of the IGMN input. In ESIGMN, the input data $\mathbf{u}$ is divided into input $\mathbf{u}_a$ and output/target elements $\mathbf{u}_b$, and is processed in the following way:

$$\mathbf{s}(t) = f(\mathbf{W}_{in,r}\mathbf{u}_a(t) + \mathbf{W}\mathbf{s}(t-1)) \tag{20}$$

$$\mathbf{x}(t) = [\mathbf{s}(t); \mathbf{u}(t)] \tag{21}$$

where $\mathbf{s}$ is the reservoir state (previously known as $\mathbf{x}$ in section 2), $\mathbf{u}_a$ is the known part of the input (the actual input portion of the data), excluding the target values $\mathbf{u}_b$, and $\mathbf{W}_{in,r}$ are the input weights (without the target values) to the reservoir (equation 20 is analogous to equation 1 for the ESN). The IGMN is trained by receiving $\mathbf{x}$ (the concatenation of the full input and reservoir state) as its input. By omitting $\mathbf{u}_b$ and using equation 19 for recalling, it's possible to predict the output $\mathbf{u}_b$.

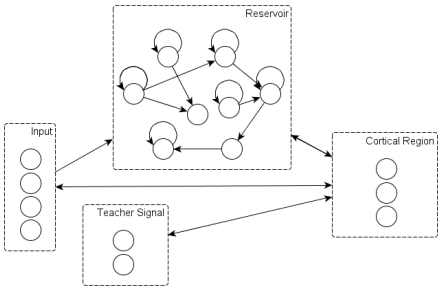


Figure 3. An example of ESIGMN with 4 inputs (u_a), 2 targets/outputs (u_b), 3 gaussian components and 8 reservoir neurons. Only the actual input portion of the data is used to update the reservoir, while the full input data (with target values / teacher signal) is fed into the IGMN together with the reservoir state s in order to train it. By omitting u_b, the IGMN can be used in recalling mode to predict u_b.

5. Experiments and Results

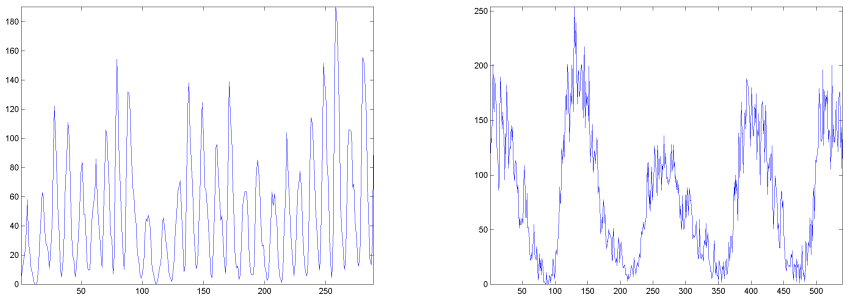
Experiments both with one-dimensional stochastic and chaotic time-series were performed. The task is to predict the scalar value $\mathbf{u}_a(t+1)$ given $\mathbf{u}_a(t)$ (the target value $\mathbf{u}_b(t)$ is $\mathbf{u}_a(t+1)$). Both the naïve and full ESIGMN were compared to the ESN (Echo State Network), Elman Network (also known as Simple Recurrent Network or SRN [Elman 1990]) and static IGMNn and IGMN (using only current input to predict the next one). The error measure used was the normalized MSE with respect to the trivial solution (always predicting the latest observation $\mathbf{u}_a(t)$ for the expected value $\mathbf{u}_a(t+1)$) and is defined as

$$NMSE_t = \frac{\frac{1}{N}\sum_{t=1}^{N} \|\mathbf{y}(t) - \mathbf{u}_a(t+1)\|^2}{\frac{1}{N}\sum_{t=1}^{N} \|\mathbf{u}_a(t) - \mathbf{u}_a(t+1)\|^2} \tag{22}$$

where N is the number of observations, $\mathbf{u}_a(t)$ is the observation at time t, $\mathbf{y}(t)$ is the predicted value at time t and $\mathbf{u}_a(t+1)$ is the desired/target value at time t. It means that solutions worse than the trivial one will have $NMSE_t$ greater than 1, while better solutions will have $NMSE_t$ smaller than 1. The runtime (in seconds) and number of epochs are also informed. The gaussian components information refers to the configuration at the end of training. The parameters of the IGMN based algorithms were the default ones suggested in section 3.1, with ϵ_{max} and δ set to 0.1. v_{min} and sp_{min} were set to 5 and 3, respectively. The ESN output layer was trained with the Conjugate gradient backpropagation with Fletcher-Reeves updates algorithm ('traincgf' in Matlab, which doesn't allow using the Levenberg-Marquardt training algorithm for recurrent networks), with early stopping and default parameters, and this same configuration was used by both layers of the Elman Network. All networks used a hidden layer with 10 neurons, and all input and output layers had size 1, since one-dimensional data was used. All reservoirs, both for ESN and ESIGMNn/ESIGMN, were scaled to a spectral radius of 0.9. All experiments were averaged over 100 runs and executed on a Intel Core 2 Quad Q8400 with 4GB RAM on Matlab 2009b without parallelization.

5.1. Yearly Mean Sunspot Numbers

This dataset consists of 289 yearly (mean) observations of a stochastic time-series, which can be seen in figure 4(a). The first 200 observations were used for training, while the remaining 89 were used for testing. For this experiment, $NMSE_t$ = 1 corresponds to $NMSE$ = 0.35 (normalized MSE w.r.t. the observations mean). Results are summarized in table 1 with mean values and standard deviations between parenthesis.



(a) The yearly mean sunspot numbers time-series.

(b) The first 550 data points of the monthly sunspot numbers time-series.

Figure 4. The two flavors of sunspot time-series.

	Elman	ESN	IGMNn	IGMN	ESIGMNn	ESIGMN
$NMSE_t$	0.97 (0.04)	0.92 (0.02)	1.38 (0.00)	0.93 (0.00)	2.23 (0.63)	0.50 (0.06)
Epochs	10.97 (5.32)	5.33 (2.26)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
Runtime	0.98 (0.77)	0.55 (0.45)	0.56 (0.34)	0.89 (0.40)	0.58 (0.18)	0.66 (0.28)
Gaussian Components	-	-	5.00 (0.00)	3.00 (0.00)	4.58 (1.23)	1.58 (0.57)

Table 1. Results of the mean yearly sunspot numbers experiment. Mean values outside parenthesis, standard deviations inside.

The ESIGMN algorithm achieved very good results in this experiment, well ahead of its competitors. The ESIGMNn, on the other hand, had the worst result, and also had high variability in the error measure, meaning it's highly sensitive to the reservoir random initialization (It's worth mentioning that a typical reservoir has hundreds of neurons, while only 10 were used in this work; larger reservoirs would potentially increase the chances of giving useful information to the IGMNn, thus reducing error mean and variance). The static IGMN could solve the problem by exploiting the fact that when the time-series is low, it tends to increase, and vice-versa. Its three gaussian components encoded low, medium and high values. A similar phenomenon happened in all experiments.

5.2. Monthly Sunspot Numbers

This dataset consists of 2987 monthly observations of a stochastic time-series, which can be seen in figure 4(b). The first 2000 observations were used for training, while the remaining 987 were used for testing. For this experiment, $NMSE_t$ = 1 corresponds to $NMSE$ = 0.1. Results are summarized in table 2.

	Elman	ESN	IGMNn	IGMN	ESIGMNn	ESIGMN
$NMSE_t$	1.02 (0.08)	0.99 (0.01)	3.67 (0.00)	0.99 (0.00)	2.59 (0.46)	0.88 (0.02)
Epochs	17.49 (16.79)	6.29 (3.83)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
Runtime	4.74 (4.27)	1.65 (1.05)	5.19 (0.94)	7.37 (1.57)	6.38 (1.42)	6.19 (2.26)
Gaussian Components	-	-	4.00 (0.00)	4.00 (0.00)	5.41 (0.82)	2.10 (0.36)

Table 2. Results of the monthly sunspot numbers experiment.

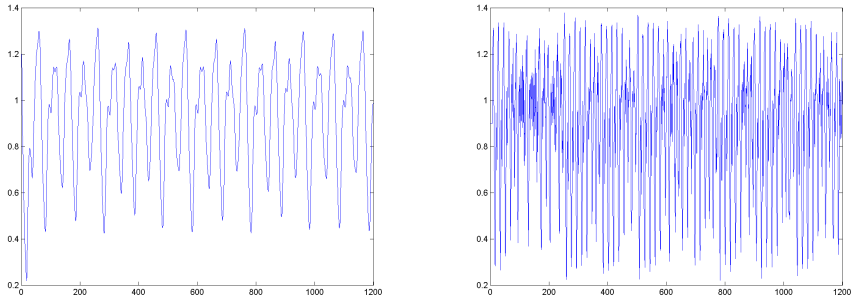
The results were very similar to the previous ones. ESIGMNn didn't do better than the trivial solution ($NMSE_t$ = 1), but the ESIGMN got the best result again.

5.3. Mackey-Glass (τ=17)

This dataset consists of 1201 observations of a chaotic time-series defined by the equation

$$\frac{dx}{dt} = 0.2\frac{x_{t-\tau}}{1 + x_{t-\tau}^n} - 0.1x_t \tag{23}$$

with $\tau = 17$, which can be seen in figure 5(a). The first 1000 observations were used for training, while the remaining 201 were used for testing. For this experiment, $NMSE_t$ = 1 corresponds to $NMSE$ = 0.0196. Results are summarized in table 3.



(a) The Mackey-Glass (τ=17) time-series. (b) The Mackey-Glass (τ=30) time-series.

Figure 5. The two flavors of Mackey-Glass time-series.

	Elman	ESN	IGMNn	IGMN	ESIGMNn	ESIGMN
$NMSE_t$	1.06 (0.19)	1.93 (0.18)	8.66 (0.00)	0.99 (0.00)	27.68 (15.63)	0.13 (0.06)
Epochs	49.81 (31.30)	7.42 (4.60)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
Runtime	7.79 (5.37)	1.01 (0.84)	2.76 (0.62)	2.45 (0.73)	3.00 (1.07)	1.78 (0.81)
Gaussian Components	-	-	4.00 (0.00)	1.00 (0.00)	4.01 (2.63)	1.04 (0.20)

Table 3. Results of the Mackey-Glass (τ = 17) experiment.

It's interesting to note that the ESIGMN could achieve the best result (by a large margin) in this experiment with only a single gaussian component, meaning that it found a linear solution over input and reservoir state space. An ESIGMN with just one component is almost equivalent to an ESN trained in batch mode, but the online ESN is inefficient with stochastic gradient descent because of the large eigenvalue spread of the reservoir states [Lukoševičius and Jaeger 2009]. The ESIGMN seems to avoid such problem.

5.4. Mackey-Glass (τ=30)

This dataset with 1500 observations is generated by the same equation 23 but with $\tau = 30$, and can be seen in figure 5(b). The first 1000 observations were used for training, while the remaining 500 were used for testing. For this experiment, $NMSE_t$ = 1 corresponds to $NMSE$ = 0.359. Results are summarized in table 4.

	Elman	ESN	IGMNn	IGMN	ESIGMNn	ESIGMN
$NMSE_t$	0.92 (0.02)	0.94 (0.01)	1.10 (0.00)	0.91 (0.00)	1.67 (0.53)	0.39 (0.08)
Epochs	29.10 (15.25)	6.39 (4.21)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
Runtime	4.61 (2.69)	0.87 (0.82)	2.83 (0.79)	4.72 (1.37)	3.38 (0.62)	3.56 (1.80)
Gaussian Components	-	-	5.00 (0.00)	6.00 (0.00)	5.41 (1.10)	4.06 (1.20)

Table 4. Results of the Mackey-Glass (τ = 30) experiment.

Again, the ESIGMN got the best results for this experiment, well ahead of its competitors. More components were needed to solve this time-series in relation to the previous one. This can be seen as a mixture of locally weighted linear experts, which enables the ESIGMN to achieve better performance than the ESN.

6. Conclusions

This work presented the ESIGMN, an one-shot, incremental and spatio-temporal learning algorithm, which has both ESN and IGMN advantages, like:

- Spatio-temporal pattern processing without the need to configure delay lines (all experiments used a reservoir with the same size)
- Online, incremental, one-shot learning
- Few non-critical parameters to be tuned manually (the same parameters were kept for all experiments)

Four experiments were executed, with 2 stochastic and 2 chaotic time-series. ESIGMN got the best results for all of them (excelling on the chaotic ones), showing that the IGMN is indeed a good replacement for the linear output layer of the ESN and also that the introduction of a reservoir increases the IGMN temporal capacity drastically (while being largely insensitive to the random nature of the reservoir). The ESN and IGMN algorithms alone yielded similar reasonable results, but together in the ESIGMN they complemented each other producing large performance gains. On the other hand, ESIGMNn got the worst results for all of them, indicating that the naïve version is not appropriate for time series prediction, at least with default parameters (experiments not shown in this work due to space restrictions show that lower ϵ_{max}, δ and sp_{min} values can give better results for the naïve version, better than the trivial solution). No parameter tuning or reservoir adaptation were done, meaning that there is room for a lot of improvement. Any improvements to reservoirs can be directly applied to ESIGMN, like Intrinsic Plasticity [Schrauwen et al. 2008], Orthogonal Reservoirs [White et al. 2004] or Critical ESN [Hajnal and Lőrincz 2006]. Also, any future improvements to the IGMM or IGMN algorithms will apply directly to ESIGMN.

The worst execution time for ESIGMN was of $\sim$5ms for each 12-dimensional data point (1 input + 10 reservoir state values + 1 target value), running on Matlab without any parallelization. This paves the way to real-time learning and prediction of sequences, which can be very useful in robotics, games, reinforcement learning, embedded systems and monitoring tools.

In future works, these other IGMN temporal extensions will be explored: Adding tapped delay lines [Kangas 1991]; Using differentiator-integrator neurons [Moser 2004]; Using recurrent connections in various ways [Pinto 2010].

References

Buehner, M. and Young, P. (2006). A tighter bound for the echo state property. *Neural Networks, IEEE Transactions on*, 17(3):820–824.

Dempster, A., Laird, N., Rubin, D., et al. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Dominey, P. F. (1995). Complex sensory-motor sequence learning based on recurrent state representation and reinforcement learning. *Biological Cybernetics*, 73(3):265–74.

Elman, J. (1990). Finding structure in time* 1. *Cognitive science*, 14(2):179–211.

Engel, P. and Heinen, M. (2011). Incremental learning of multivariate gaussian mixture models. *Advances in Artificial Intelligence–SBIA 2010*, pages 82–91.

Hajnal, M. and Lőrincz, A. (2006). Critical echo state networks. *Artificial Neural Networks–ICANN 2006*, pages 658–667.

Hayes, M. (1996). 9.4: Recursive Least Squares",". *Statistical Digital Signal Processing and Modeling*.

Heinen, M. (2011). *A Connectionist Approach for Incremental Function Approximation and On-line Tasks*. PhD thesis, Universidade Federal do Rio Grande do Sul. Instituto de Informática. Programa de Pós-Graduação em Computação.

Heinen, M. and Engel, P. (2010). An Incremental Probabilistic Neural Network for Regression and Reinforcement Learning Tasks. *Artificial Neural Networks–ICANN 2010*, pages 170–179.

Jaeger, H. (2001). The" echo state" approach to analysing and training recurrent neural networks-with an erratum note'. Technical report, Technical Report GMD Report 148, German National Research Center for Information Technology.

Jaeger, H. (2003). Adaptive nonlinear system identification with echo state networks. *Advances in Neural Information Processing Systems*, 15:593–600.

Jaeger, H. and Haas, H. (2004). Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667):78.

Jaeger, H., Lukosevicius, M., Popovici, D., and Siewert, U. (2007). Optimization and applications of echo state networks with leaky-integrator neurons. *Neural Networks*, 20(3):335–352.

Kangas, J. (1991). Phoneme recognition using time-dependent versions of self-organizing maps. In *icassp*, pages 101–104. IEEE.

Lukoševičius, M. and Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149.

Moser, L. (2004). Modelo de um neurônio diferenciador-integrador para representação temporal em arquiteturas conexionistas. Universidade Federal do Rio Grande do Sul. Instituto de Informática. Programa de Pós-Graduação em Computação.

Natschläger, T., Maass, W., and Markram, H. (2002). The "liquid computer": A novel strategy for real-time computing on time series. *Special Issue on Foundations of Information Processing of TELEMATIK*, 8(1):39–43.

Pinto, R. (2010). Um Estudo de Redes Neurais Não-Supervisionadas Temporais. Universidade Federal do Rio Grande do Sul. Instituto de Informática. Programa de Pós-Graduação em Computação.

Schrauwen, B., Wardermann, M., Verstraeten, D., Steil, J., and Stroobandt, D. (2008). Improving reservoirs using intrinsic plasticity. *Neurocomputing*, 71(7-9):1159–1171.

Steil, J. J. (2004). Backpropagation-decorrelation: online recurrent learning with o(n) complexity,. In *IJCNN*.

Verstraeten, D., Schrauwen, B., and Stroobandt, D. (2006). Reservoir-based techniques for speech recognition. In *Proceedings of the world conference on computational intelligence*, pages 1050–1053.

White, O., Lee, D., and Sompolinsky, H. (2004). Short-term memory in orthogonal neural networks. *Physical review letters*, 92(14):148102.

Widrow, B. (1966). Adaptive filters I: fundamentals (TR 6764-6).

Um Algoritmo de Mapa Auto-Organizável por Lote Baseado em Distâncias Adaptativas

Luciano D. S. Pacífico[1], **Francisco de A. T. de Carvalho**[1]

[1]Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Av. Jornalista Anibal Fernandes, s/n – Recife – PE – 50.740-560 – Brazil

{ldsp,fatc}@cin.ufpe.br

Abstract. *Clustering methods aims to organize a set of items into clusters such that items within a given cluster have a high degree of similarity, while items belonging to different clusters have a high degree of dissimilarity. The self-organizing map (SOM) is an unsupervised competitive learning neural network method which has both clustering and dimensionality reduction properties, using a neighborhood function to discover the topological structure hidden in the data set. In this paper, we introduce a batch self-organizing maps algorithm based on adaptive distances (ABSOM). Experimental results obtained in real benchmark datasets show the effectiveness of our approach.*

Resumo. *Métodos de agrupamento visam organizar um conjunto de itens em grupos de tal forma que itens de um dado grupo possuam um alto grau de similaridade, enquanto itens em grupos diferentes possuam um alto grau de dissimilaridade. O mapa auto-organizável (self-organizing map, SOM) é uma rede neural não-supervisionada de aprendizado competitivo que possui propriedades de agrupamento e de redução da dimensionalidade, usando uma função de vizinhança para descobrir a estrutura topológica escondida no conjunto de dados. Neste artigo, nós introduzimos um mapa auto-organizável por lote baseado em distâncias adaptativas (ABSOM). Resultados experimentais obtidos em bases de dados benchmark mostram a efetividade de nossa abordagem.*

1. Introdução

A análise de agrupamentos (*Clustering*) é uma tarefa bastante popular na descoberta de conhecimentos, possuindo aplicação em diversos campos como mineração de dados, reconhecimento de padrões, visão computacional, etc. Métodos de agrupamento visam a organização de um conjunto de itens em grupos de tal forma que itens de um dado grupo possuam um alto grau de similaridade, enquanto itens em grupos diferentes possuam um alto grau de dissimilaridade. As técnicas mais populares de agrupamento de dados são as hierárquicas e as de particionamento [Everitt 2001][Gordon 1999][Spaeth 1980].

Métodos hierárquicos resultam em uma hierarquia, ou seja, uma sequência aninhada de partições dos dados de entrada. Tais métodos podem ser de duas categorias: aglomerativos e divisivos. Métodos aglomerativos resultam em uma sequência de partições aninhadas, tendo início com cada item sendo ele mesmo um grupo unitário, e após aglomerações sucessivas todos os indivíduos se encontram em um mesmo grupo final. Os métodos divisivos têm início com todos os itens pertencendo a um único conjunto,

que sofre divisões sucessivas até que um critério de parada seja obtido (geralmente após a formação de uma partição com itens pertencentes a grupos unitários).

O particionamento de um conjunto de itens em um número de grupos pré-definido é um tópico importante em análise de dados, reconhecimento de padrões e processamento de imagens [Theodoris and Koutroumbas 2006][Jain et al. 1999]. Algoritmos de particionamento buscam obter uma partição dos dados de entrada em um número fixo de grupos fornecido inicialmente pelo usuário do sistema. Tais métodos sempre buscam a otimização de uma função critério de adequação (geralmente de forma local). Para a melhora da qualidade dos agrupamentos finais, o algoritmo é executado várias vezes, partindo de pontos iniciais diferentes, sendo a melhor configuração obtida usada como resposta do algoritmo. Os algoritmos mais populares desta família são os algoritmos *K-means* [MacQueen 1967] e *Fuzzy C-means* [Bezdek 1981].

O Mapa Auto-Organizável de Kohonen (SOM) [Kohonen 1995] é uma rede neural não-supervisionada de aprendizado competitivo que possui propriedades de agrupamento e de visualização. Diferente do *K-means*, a rede SOM usa um conjunto de interações de vizinhança para aproximar a interação lateral neural e descobrir a estrutura topológica escondida nos dados, e além do vetor que obteve o melhor valor de similaridade (o vencedor), seus vizinhos também são atualizados, resultando em regiões nas quais neurônios em uma mesma vizinhança sejam bastante similares. Tal método também pode ser considerado como um algoritmo que mapeia dados de alta dimensionalidade espacial, $\mathbb{R}^p$, em um espaço de dimensionalidade reduzida, geralmente 1D, 2D ou 3D, chamado mapa. Essa projeção permite a partição das entradas em grupos "similares" enquanto preserva sua topologia.

Diversas variações do algoritmo SOM básico têm sido propostas visando sua melhoria. Em Kang *et al.* [J. A. Kangas and Laaksonen 1990] duas novas abordagens baseadas nas redes SOM foram propostas: uma baseada na ponderação dinâmica dos sinais de entrada de cada neurônico, melhorando a ordenação quando diferentes sinais de entrada eram usados; o segundo consistia na definição das vizinhaças no algoritmo de aprendizagem através do uso de uma árvore de caminhos mínimos, promovendo uma melhor e mais rápida aproximação de funções de densidade. Em Badran *et al.* [F. Badran and Thiria 2005] foi aprensentado um algoritmo de mapa auto-organizável por lote similar ao algoritmo *K-means*, consistindo em uma abordagem em dois passos com uma afetação onde todas as instâncias da base de dados são afetadas ao neurônio mais próximo do mapa, e uma etapa de representação, onde todos os neurônios são atualizados.

Neste artigo o algoritmo de mapa auto-organizável por lote (BSOM) apresentado em [F. Badran and Thiria 2005] é extendido através de um algoritmo de mapa auto-organizável por lote adaptativo (ABSOM), que faz uso de distâncias euclidianas adaptativas parametrizadas por vetores de pesos nas variáveis que mudam a cada iteração do algoritmo e são diferentes para cada grupos. O cálculo desses vetores de pesos é inspirado no cálculo dos pesos para cada variável em cada grupo apresentado no algoritmo de agrupamento dinâmico baseado em distâncias adaptativas [Diday and Govaert 1977]. Os algoritmos de agrupamento dinâmicos tradicionais [Diday and Simon 1976] são algoritmos iterativos de realocação em duas etapas que envolvem a construção de novos agrupamentos a cada iteração e a identificação de representantes ou protótipos (médias,

conjuntos de regras, leis de probabilidade, funções de densidade de probabilidade, grupos de elementos, etc.) mais adequados para os novos grupos pela otimização local de uma função critério de adequação. Os métodos de agrupamento dinâmicos e adaptativos [Diday and Govaert 1977] também otimizam uma função baseado em um critério de adequação entre os grupos formados e seus protótipos, mas as distâncias para a comparação dos grupos e seus protótipos muda a cada iteração. Tais distâncias (representadas por vetores de pesos para as variáveis do problema) não são determinadas de forma absoluta e são diferentes para cada agrupamento. A vantagem dessas distâncias adaptativas está no fato de possibilitar ao algoritmo o reconhecimento de grupos de diferentes formas e tamanhos.

Como forma de avaliação da efetividade de nossa abordagem, experimentos foram realizados com a utilização de dados *benchmark* de classificação presentes no *UCI Machine Learning Repository* [Frank and Asuncion - Online].

Este artigo está organizado como se segue. A Seção 2 apresenta o mapa auto-organizável tradicional [Kohonen 1995]. O mapa auto-organizável por lote [F. Badran and Thiria 2005] é apresentado na Seção 3. Em seguida, a abordagem proposta de mapa auto-organizável por lote adaptativo é apresentado (Seção 4) e os resultados experimentais são discutidos (Seção 5). As conclusões serão apresentadas na Seção 6.

2. O Mapa Auto-Organizável Estocástico

A rede de mapas auto-organizável (SOM) estocástica [Kohonen 1995] é usada atualmente em diversos domínios e tem obtido sucesso em numerosas aplicações. É uma ferramenta muito popular para a visualização de dados de alta dimensionalidade espacial. De forma geral, pode-se considerar que a rede SOM executa a tarefa de agrupamento dos dados preservando a ordem espacial dos dados através da ordenação dos vetores protótipos (também chamados de centróides ou referências) em um espaço de saída de uma ou duas dimensões.

A rede SOM é formada por neurônios organizados em uma grade de baixa dimensionalidade, chamada mapa. Formalmente, o mapa é descrito por um grafo (C, Γ). C é um conjunto de m neurônios interconectado tendo uma topologia discreta definida por Γ. Para cada par de neurônios (c, r), $\delta(c, r)$ é definida como a função de distância entre os neurônios c e r no mapa. Essa distância impõe uma relação de vizinhança entre os neurônios. Seja $E = \{1, \ldots, n\}$ o conjunto de entradas, onde cada entrada $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})\,(i = 1, \ldots, n)$ pertence a $\mathbb{R}^p$. Cada neurônio do mapa é representado por um protótipo $\mathbf{w}_c = (w_{c1}, \ldots, w_{cp})\,(c = 1, \ldots, m)$ que também pertence a $\mathbb{R}^p$. O algoritmo SOM estocástico executa os seguintes passos:

1) Inicialização
   Fixe o número m de neurônios (grupos);
   Fixe $\eta(0)$ (taxa de aprendizado inicial);
   Fixe $h_{\delta(j,l)}(0)$ (função de vizinhança inicial, onde $\delta(j, l)$ é uma função de distâncias fixada entre neurônios j e l);
   Fixe o número de iterações N_{iter};
   Selecione aleatoriamente m protótipos distintos $\mathbf{w}_c^{(0)} \in E\,(c = 1, \ldots, m)$;
   Faça t = 1;

2) *Amostragem.*
   Pegue um vetor de entrada aleatório $\mathbf{x}_i(t)$;
3) *Seleção.*
   Encontre o melhor neurônio (vencedor) $\mathbf{w}_c(t)$ com distância euclidiana mínima em relação a $\mathbf{x}_i(t)$:

$$f(\mathbf{x}_i(t))^T = \min_{1 \leq j \leq m} \sum_{k=1}^{p} (x_{ik}(t) - w_{jk}(t))^2;$$

4) *Modificação dos pesos.*
   Para todo neurônio j faça:

$$\begin{aligned} \mathbf{w}_j(t+1) &= \mathbf{w}_j(t) + \eta(t) h_{\delta(\mathbf{w}_c, w_j)}(t)(\mathbf{x}_i(t) - \mathbf{w}_j(t)), j \in \mathbf{N}_c(t) \\ &= \mathbf{w}_j(t), j \notin \mathbf{N}_c(t) \end{aligned}$$

   onde $\mathbf{N}_c(t)$ é vizinhança do neurônio c na iteração t;
5) Atualização.
   Atualize tanto a taxa de aprendizado $\eta(t)$ quanto a função de vizinhança $h_{l,j}(t)$; faça $t = t + 1$;
6) Critério de parada.
   Se $t = N_{Iter}$, pare; caso contrário, vá para o passo 2.

3. Algoritmo de Mapa Auto-Organizável por Lote (BSOM)

Esta seção apresenta o algoritmo de mapa auto-organizável por lote introduzido por [F. Badran and Thiria 2005]. Seja $E = \{1, \ldots, n\}$ o conjunto de entradas, onde cada entrada $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})\,(i = 1, \ldots, n)$ pertence a $\mathbb{R}^p$. Cada neurônio do mapa é representado por um protótipo $\mathbf{w}_c = (w_{c1}, \ldots, w_{cp})\,(c = 1, \ldots, m)$ que também pertence a $\mathbb{R}^p$.

O algoritmo SOM de treinamento por lote [F. Badran and Thiria 2005] é um algoritmo iterativo de duas etapas (afetação e representação, discutidos em seguida) onde todo o conjunto de dados (E) é apresentado ao mapa antes de algum ajuste ser feito. O algoritmo SOM por lote minimiza a seguinte função objetivo:

$$J = \sum_{i=1}^{n} \sum_{r=1}^{m} K^T(\delta(f^T(\mathbf{x}_i), r)) d^2(\mathbf{x}_i, \mathbf{w}_r) \tag{1}$$

onde f é a função de alocação e $f(\mathbf{x}_i)$ representa o neurônio do mapa correspondente ao indivíduo $\mathbf{x}_i$, e $\delta(f(\mathbf{x}_i), r))$ é a distância entre o neurônio r do mapa e o neurônio correspondente a $\mathbf{x}_i$. Assim, K^T, que é parametrizada por T (onde T representa a temperatura), é a função *kernel* de vizinhança que define a região de influência de cada neurônio r. Essa função objetivo é minimizada apenas para um valor de T fixo.

Essa função de custo é uma extensão da função de custo do *K-means*, onde a distância euclidiana é substituída por

$$d^T(\mathbf{x}_i, \mathbf{w}_{f(\mathbf{x}_i)}) = \sum_{r=1}^{m} K^T(\delta(f^T(\mathbf{x}_i), r)) d^2(\mathbf{x}_i, \mathbf{w}_r) \tag{2}$$

onde

$$d^2(\mathbf{x}_i, \mathbf{w}_r) = \sum_{j=1}^{p} (x_{ij} - w_{rj})^2 \tag{3}$$

é a distância euclidiana. Essa distância generalizada é uma soma ponderada das distâncias euclidianas entre $\mathbf{x}_i$ e todos os vetores de referência da vizinhança do neurônio $f(\mathbf{x}_i)$. Todos os neurônios do mapa são levados em consideração.

Quanto T é fixo, a minimização de J é realizada em duas etapas iterativas: uma afetação e uma representação.

Durante a etapa de afetação, os vetores de referência (protótipos) permanecem fixos. A função de custo J é minimizada em relação à função de alocação e cada indivíduo $\mathbf{x}_i$ é assinalado ao seu neurônio mais próximo:

$$c = f^T(\mathbf{x}_i) = arg \min_{1 \leq r \leq m} d^T(\mathbf{x}_i, \mathbf{w}_r) \tag{4}$$

Durante a etapa de representação, a função de alocação é fixada. A função de custo J é minimizada em relação aos protótipos. O protótipo $\mathbf{w}_c$ é atualizado para cada neurônio de acordo com:

$$\mathbf{w}_c = \frac{\sum_{i=1}^{n} K^T(\delta(f^T(\mathbf{x}_i), c)) \mathbf{x}_i}{\sum_{i=1}^{n} K^T(\delta(f^T(\mathbf{x}_i), c))}. \tag{5}$$

O algoritmo de mapa auto-organizável por lote (BSOM) [F. Badran and Thiria 2005] pode ser resumido como:

1) Inicialização
   Fixe o número m de neurônios (grupos);
   Fixe δ; Fixe a função de *kernel* K^T;
   Fixe o número de iterações N_{iter};
   Fixe T_{min}, T_{max}; Faça $T \leftarrow T_{max}$; Faça $t \leftarrow 0$;
   Selecione aleatoriamente m protótipos distintos $\mathbf{w}_c^{(0)} \in E\,(c = 1, \ldots, m)$;
   Inicialize o mapa $L(m, \mathbf{W}^0)$, onde $\mathbf{W}^0 = (\mathbf{w}_1^{(0)}, \ldots, \mathbf{w}_m^{(0)})$;
   Afete cada objeto $\mathbf{x}_i$ ao neurônio mais próximo (grupo) do mesmo de acordo com a equação (4).
2) *Etapa 1: Representação*.
   Faça $T = T_{max}(\frac{T_{min}}{T_{max}})^{\frac{t}{N_{iter}-1}}$;
   Mantenha a função de alocação fixa;
   Calcule os protótipos $\mathbf{w}_c^{(t)}\,(c = 1, \ldots, m)$ de acordo com a equação (5);
3) *Etapa 2: Afetação*.
   Os protótipos $\mathbf{w}_c^{(t)}\,(c = 1, \ldots, m)$ são fixados. Afete cada indivíduo $\mathbf{x}_i\,(i = 1, \ldots, n$ ao neurônio mais próximo do mesmo de acordo com a equação (4);

4) *Critério de parada.*
se $T = T_{min}$, pare; caso contrário, faça $t = t + 1$ e vá para o passo 2 (Etapa 1).

4. Mapa Auto-Organizável por Lote Adaptativo (ABSOM)

Em [J. A. Kangas and Laaksonen 1990] foi apresentada uma versão adaptativa para o algoritmo SOM estocástico. Nesta seção apresentamos um algoritmo SOM por lote adaptativo que usa distâncias euclidianas paramatrizadas por vetores de pesos para as variáveis que mudam a cada iteração do algoritmo e que são diferentes para cada um dos agrupamentos. O cálculo desses vetores de pesos neste algoritmo é inspirado na abordagem usada para calcular o peso de cada uma das variáveis em cada agrupamento do algoritmo de agrupamento dinâmico baseado em distâncias adaptativas [Diday and Govaert 1977].

O algoritmo de treinamento SOM por lote adaptativo introduzido neste artigo é um algoritmo iterativo que executa em três etapas (afetação, representação e ajuste dos pesos adaptativos) no qual todo o conjunto de dados de entrada (E) é apresentado ao mapa antes de qualquer ajuste ser feito.

A função de dissimilaridade geral entre um dado de entrada $\mathbf{x}_i$ e um protótipo $\mathbf{w}_{f(\mathbf{x}_i)}$ é dada por:

$$d^T_{\mathbf{\Lambda}}(\mathbf{x}_i, \mathbf{w}_{f(\mathbf{x}_i)}) = \sum_{r=1}^{m} K^T(\delta(f^T(\mathbf{x}_i), r)) d^2_{\boldsymbol{\lambda}_r}(\mathbf{x}_i, \mathbf{w}_r) \tag{6}$$

onde

$$d^2_{\boldsymbol{\lambda}_r}(\mathbf{x}_i, \mathbf{w}_r) = \sum_{j=1}^{p} \lambda_{rj}(x_{ij} - w_{rj})^2 \tag{7}$$

é uma distância euclidiana adaptativa parametrizada por vetores de pesos para as variáveis $\boldsymbol{\lambda}_r = (\lambda_{r1}, \ldots, \lambda_{rp})$ and $\mathbf{\Lambda}$= $(\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_m)$.

Note que os vetores de pesos $\boldsymbol{\lambda}_r$ $(r = 1, \ldots, m)$ mudam a cada iteração, ou seja, eles não são determinados de forma absoluta, e são diferentes para cada neurônio do mapa.

A função de custo do algoritmo SOM por lote adaptativo é dada por:

$$J = \sum_{i=1}^{n} \sum_{r=1}^{m} K^T(\delta(f^T(\mathbf{x}_i), r)) d^2_{\boldsymbol{\lambda}_r}(\mathbf{x}_i, \mathbf{w}_r) \tag{8}$$

Quando T é fixo, a minimização de J é obtida de forma iterativa em três etapas: afetação, representação e ajuste dos pesos.

Durante a etapa de afetação, os vetores de referência (protótipos) e os vetores de pesos são fixados. A função de custo J é minimizada em respeito à função de alocação e cada indivíduo $\mathbf{x}_i$ é assinalado ao neurônio mais próximo do mesmo:

$$c = f^T(\mathbf{x}_i) = arg \min_{1 \leq r \leq m} d^T_{\mathbf{\Lambda}}(\mathbf{x}_i, \mathbf{w}_r) \tag{9}$$

Durante a etapa de representação, a função de alocação e os vetores de pesos das variáveis são fixados. A função de custo J é minimizada em respeito aos protótipos. O protótipo $\mathbf{w}_c$ é atualizado de acordo com a equação (5).

Durante a etapa de ajuste de pesos, os vetores de referência (protótipos) e a função de alocação são fixados. A função de custo J é minimizada em relação aos vetores de pesos. Os vetores de pesos $\boldsymbol{\lambda}_r = (\lambda_{r1}, \ldots, \lambda_{rp})\,(r = 1, \ldots, m)$, com $\lambda_{rj} > 0$ e $\prod_{j=1}^{p} \lambda_{rj} = 1$, têm seus pesos $\lambda_{rj}\,(j = 1, \ldots, p)$ calculados de acordo com a seguinte expressão:

$$\lambda_{rj} = \frac{\{\prod_{h=1}^{p} \sum_{i=1}^{n} K^T(\delta(f^T(\mathbf{x}_i), h))(x_{ij} - w_{hj})^2\}^{\frac{1}{p}}}{\sum_{i=1}^{n} K^T(\delta(f^T(x_i), r))(x_{ij} - w_{rj})^2} \tag{10}$$

O algoritmo de mapa auto-organizável por lote adaptativo (ABSOM) pode ser resumido como:

1) Inicialização
   Fixe o número m de neurônios (grupos);
   Fixe δ; Fixe a função de kernel K^T;
   Fixe o número de iterações N_{iter};
   Fixe T_{min}, T_{max}; Faça $T \leftarrow T_{max}$; Faça $t \leftarrow 0$;
   Faça cada componente $\lambda_{rj}(r = 1, \ldots, m; j = 1, \ldots, p)$ das matrizes de pesos adaptativas iguais a 1;
   Selecione aleatoriamente m distintos protótipos $\mathbf{w}_c^{(0)} \in E\,(c = 1, \ldots, m)$;
   Inicialize o mapa $L(m, \mathbf{W}^0)$, onde $\mathbf{W}^0 = (\mathbf{w}_1^{(0)}, \ldots, \mathbf{w}_m^{(0)})$;
   Assinale cada objeto $\mathbf{x}_i$ ao neurônio (grupo) mais próximo do mesmo de acordo com a equação (4).
2) *Etapa 1: Representação.*
   Faça $T = T_{max}(\frac{T_{min}}{T_{max}})^{\frac{t}{N_{iter}-1}}$;
   Mantenha a função de alocação fixa;
   Calcule os protótipos $\mathbf{w}_c^{(t)}\,(c = 1, \ldots, m)$ de acordo com a equação (5);
3) *Etapa 2: Ajuste dos pesos adaptativos.*
   Os vetores de referência (protótipos) e a função de alocação são mantidos fixos.
   Calcule os componentes λ_{rj} dos vetores de pesos $\boldsymbol{\lambda}_r\,(r = 1, \ldots, m; j = 1, \ldots, p)$ de acordo com a equação (10);
4) *Etapa 3: Afetação.*
   Os protótipos $\mathbf{w}_c^{(t)}\,(c = 1, \ldots, m)$ são fixados. Assinale cada indivíduo $\mathbf{x}_i\,(i = 1, \ldots, n$ ao neurônio mais próximo do mesmo de acordo com a equação (9);
5) *Critério de parada.*
   Se $T = T_{min}$ pare; caso contrário, faça $t = t + 1$ e vá para o passo 2 (Etapa 1).

5. Resultados Experimentais

Para mostrar a utilidade do método desenvolvido, experimentos foram realizados comparando o algoritmo de treinamento baseado em redes SOM por lote (BSOM) [F. Badran and Thiria 2005] com o algoritmo de treinamento baseado em redes SOM por lote adaptativo(ABSOM) desenvolvido. Os algoritmos foram testados em bases de dados quantitativas reais, obtidas através do *UCI Machine Learning Repository*

[Frank and Asuncion - Online], sendo tais bases de diferentes níveis de dificuldades para a tarefa de agrupamento dos dados.

Como forma de comparação para os agrupamentos resultantes fornecidos pelos algoritmos considerados, um índice de avaliação externo, o Índice de Rand Corrigido (Corrected Rand Index, CR) [Hubert and Arabie 1985], a taxa global de erro de classificação (Overall Error Rate of Classification, $OERC$) [L. Breiman and Olshen 1984] e *F-Measure* [van Rijsbergen 1979] serão considerados.

O CR verifica o grau de concordância (similaridade) entre duas partições dos dados, uma fornecida a priori e outra obtida pelo algoritmo de agrupamento. O CR foi usado pelo fato de o mesmo não ser sensível ao número de grupos nas partições ou a distribuição de itens nesses grupos. O CR gera resultados no intervalo [-1, 1], no qual 1 significa concordância perfeita entre as duas partições avaliadas, enquanto valores próximos de 0 ou negativos indicam que o resultado foi obtido pelo acaso [Milligan 1996].

Como indicado em [F. Badran and Thiria 2005] a performance dos métodos avaliados dependerá fortemente dos parâmetros adotados para a minimização dos mesmos. Os parâmetros mais importantes a serem escolhidos são T_{max}, T_{min}, N_{iter} e a forma de decaimento da temperatura. A tabela 1 contém os parâmetros adotados para ambos o algoritmo baseado em redes SOM por lote tradicional e o algoritmo baseado em redes SOM por lote adaptativo. Tais valores foram determinados após diferentes testes com as abordagens adotadas, tendo sido os que obtiveram os melhores desempenhos médios.

Table 1. Lista de Parêmetros

Parâmetro	Valor
N_{Iter}	500
m	16 (grade 2×8)
T_{max}	3.5
T_{min}	0.5

Neste artigo a função δ adotada foi a distância euclidiana e a função de $kernel$ de vizinhança escolhida foi

$$K^T(\delta(c,r) = exp\left(-\frac{(\delta(c,r))^2}{2T^2}\right)$$

5.1. Base de Dados Diabetes

Esta base de dados consiste em dois possíveis diagnósticos (classes) dados a uma amostra de 768 mulheres de 21 anos de idade de descendência indígena ($PimaIndians$), levando em consideração quando a paciente apresenta ou não sinais de diabetes de acordo com os critérios da Organização Mundial de Saúde ($WorldHealthOrganization$). As classes (1 para pacientes saudáveis, e 2 para pacientes interpretados como pacientes com "teste positivo para diabetes") possuem, respectivamente, 500 e 268 instâncias. Cada indivíduo é descrito por 8 variáveis reais [Frank and Asuncion - Online].

Os algoritmos de mapa auto-organizáveis por lote (tradicional e adaptativo) foram aplicados a esta base de dados. O CR, a *F-Measure* e o $OERC$ foram, respectivamente,

0.049, 0.446 e 0.320 para o algoritmo de mapa auto-organizável por lote adaptativo. Para o algoritmo de mapa auto-organizável por lote de [F. Badran and Thiria 2005], tais índices foram, respectivamente, 0.012, 0.364 e 0.338. Para esta base de dados, o algoritmo de mapa auto-organizável por lote adaptativo obteve resultados superiores ao algoritmo de mapa auto-organizável por lote tradicional no que se refere aos índices avaliados.

5.2. Base de Dados de Identificação de Vidros

Esta base de dados consiste em um estudo sobre a classificação de tipos de vidro motivada pela investigação criminal. A base consiste em sete tipos diferentes de vidro (classes), embora não exista nenhum representante da classe 4, sendo cada instância (214 no total, sendo as classes não-balanceadas) deste conjunto descrita por nove variáveis do tipo real [Frank and Asuncion - Online].

Os algoritmos de mapa auto-organizáveis por lote (tradicional e adaptativo) foram aplicados a esta base de dados. O CR, a *F-Measure* e o $OERC$ foram, respectivamente, 0.165, 0.429 e 0.383 para algoritmo de treinamento baseado em redes neurais SOM por lote adaptativo, e 0.223, 0.514 e 0.444 para o algoritmo de mapa auto-organizável por lote tradicional, respectivamente. Para esta base, o algoritmo de mapa auto-organizável por lote adaptativo foi superior ao algoritmo de mapa auto-organizável por lote apresentado em [F. Badran and Thiria 2005] no que diz respeito ao erro global de classificação ($OERC$).

5.3. Base de Dados Iris

Esta base de dados consiste em três tipos (classes) da planta iris, sendo cada classe representada por 50 instâncias, num total de 150 indivíduos. Cada indivíduo é composto por 4 variáveis do tipo real. Uma das classes é linearmente separável das outras duas, mas as últimas duas classes são não-linearmente separáveis uma da outra [Frank and Asuncion - Online].

Os algoritmos de mapa auto-organizáveis por lote (tradicional e adaptativo) foram aplicados a esta base de dados. O CR, a *F-Measure* e o $OERC$ foram, respectivamente, 0.483, 0.599 and 0.08 para o mapa auto-organizável por lote adaptativo, e 0.372, 0.570 e 0.113 para o mapa auto-organizável por lote [F. Badran and Thiria 2005]. Para esta base de dados, o algoritmo de treinamento baseado em redes SOM por lote adaptativo obteve desempenho superior ao algoritmo de mapa auto-organizável por lote tradicional no que se refere aos três índices avaliados.

5.4. Base de Dados de Segmentação de Imagens

Esta base consiste em imagens que foram obtidas de forma aleatória a partir de sete imagens de ambientes externos. As imagens foram segmentadas manualmente para a criação de sete diferentes rótulos (classes). cada classe possui 330 intâncias (num total de 2310 objetos), sendo cada instância descrita por 18 atributos do tipo real [Frank and Asuncion - Online].

Os algoritmos de mapa auto-organizáveis por lote (tradicional e adaptativo) foram aplicados a esta base de dados. O CR, a *F-Measure* e o $OERC$ foram, respectivamente, 0.391, 0.562 e 0.431 para o algoritmo de mapa auto-organizável por lote adaptativo, e foram, respectivamente, 0.105, 0.348 e 0.644 para o algoritmo de mapa auto-organizável

por lote [F. Badran and Thiria 2005]. Para esta base de dados, o algoritmo de mapa auto-organizável por lote adaptativo obteve resultados superiores ao algoritmo de mapa auto-organizável por lote tradicional no que se refere aos índices avaliados.

5.5. Base de Dados Tireóide

Esta base de dados consiste em três classes no que diz respeito ao estado da glândula tireóide. As classes (1, 2 e 3) possuem, respectivamente, 150, 35 e 30 instâncias (total de 215 instâncias). Cada instância é descrita por cinco variáveis do tipo real [Frank and Asuncion - Online].

Os algoritmos de mapa auto-organizáveis por lote (tradicional e adaptativo) foram aplicados a esta base de dados. O CR, a *F-Measure* e o $OERC$ foram, respectivamente, 0.670, 0.844 e 0.102 para o algoritmo de mapa auto-organizável por lote adaptativo, enquanto o algoritmo e mapa auto-organizável por lote [F. Badran and Thiria 2005] obteve, respectivamente, 0.112, 0.390 e 0.139. Para esta base de dados, o algoritmo de mapa auto-organizável por lote adaptativo obteve resultados superiores ao algoritmo de mapa auto-organizável por lote tradicional no que se refere aos índices avaliados.

5.6. Base de Dados de Vinhos

Esta base de dados consiste em três tipos (classes) de vinhos produzidos na mesma região da Itália, mas derivados de três diferentes cultivos. As classes (1, 2 e 3) contém, respectivamente, 59, 71 e 48 instâncias (total de 178 instâncias). Cada instância é descrita por 13 variáveis quantitativas reais, que representam quantidades dos componentes encontrados nesses vinhos [Frank and Asuncion - Online].

Os algoritmos de mapa auto-organizáveis por lote (tradicional e adaptativo) foram aplicados a esta base de dados. O CR, a *F-Measure* e o $OERC$ foram, respectivamente, 0.578, 0.709 e 0.067 para o algoritmo de mapa auto-organizável por lote adaptativo, e 0.267, 0.457 e 0.275, respectivamente, para o algoritmo de mapa auto-organizável por lote de [F. Badran and Thiria 2005]. Para esta base de dados, o algoritmo de mapa auto-organizável por lote adaptativo obteve resultados superiores ao algoritmo de mapa auto-organizável por lote tradicional no que se refere aos índices avaliados.

Podemos concluir que para a grande maioria das bases testadas o algoritmo de treinamento baseado em redes neurais SOM por lote adaptativo introduzido obteve melhores resultados no que diz respeito ao índice de Rand corrigido (CR), a *F-Measure* e ao erro global de classificação ($OERC$). No que diz respeito ao $OERC$, o método proposto obteve melhores resultados para todas as bases testadas, tendo sido inferior no que concerne ao CR e a *F-Measure* apenas no caso da base de dados de identificação de vidros (5.2).

6. Conclusões

Neste artigo, um algoritmo de mapa auto-organizável por lote adaptativo foi apresentado. Tal método combinou as características de visualização e de formação de agrupamentos das redes neurais SOM com a flexibilidade oferecida pelas distâncias adaptativas no reconhecimento de agrupamentos de tamanhos e formatos diferentes.

Experimentos realizados com bases de dados reais $benchmark$ mostraram a utilidade da abordagem proposta. A tabela 2 apresenta os resultados obtidos pelo algoritmo

de mapa auto-organizável por lote adaptativo (ABSOM) proposto e pelo algoritmo de mapa auto-organizável por lote de [F. Badran and Thiria 2005] para cada base de dados testada, sendo os melhores resultados de cada índice destacados. A precisão dos resultados obtidos pelo método introduzido foi apresentada através do índice de Rand corrigido (CR), pelo erro global de classificação ($OECR$) e pela *F-Measure* considerando as bases de dados testadas. O método de mapa auto-organizável por lote adaptativo foi superior o método tradicional de mapa auto-organizável por lote para as bases de dados selecionadas do *UCI Machine Learning Repository* levando-se em consideração os critérios de avaliação adotados e na identificação das classes a priori dos padrões.

Table 2. Resultados obtidos para cada base de dados

Base de Dados	*CR*	*F-Measure*	*OERC*
Diabetes:			
ABSOM	**0.049**	**0.446**	**0.320**
BSOM	0.012	0.364	0.338
Identificação de Vidros:			
ABSOM	0.165	0.429	**0.383**
BSOM	**0.223**	**0.514**	0.444
Iris:			
ABSOM	**0.483**	**0.599**	**0.08**
BSOM	0.372	0.570	0.113
Segmentação de Imagens:			
ABSOM	**0.391**	**0.562**	**0.431**
BSOM	0.105	0.348	0.644
Tireóide:			
ABSOM	**0.670**	**0.844**	**0.102**
BSOM	0.112	0.390	0.139
Vinhos:			
ABSOM	**0.578**	**0.709**	**0.067**
BSOM	0.267	0.457	0.275

References

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press.

Diday, E. and Govaert, G. (1977). Classification automatique avec distances adaptatives. *Informatique Comput. Sci.*, 11(4):329-349.

Diday, E. and Simon, J. C. (1976). *Clustering Analysis*. Germany: Springer-Verlag.

Everitt, B. (2001). *Cluster Analysis*. New York: Halsted.

F. Badran, M. Y. and Thiria, S. (2005). *Self-Organizing Maps and Unsupervised Classification*. in: Neural Networks: methodology and applications, G. Dreyfus, Berlin, Germany: Springer-Verlag.

Frank, A. and Asuncion, A. (2011). UCI Machine Learning Repository. *Disponível: http://archive.ics.uci.edu/ml*.

Gordon, A. D. (1999). *Classification*. Boca Raton, FL: CRC Press.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2:193-218.

J. A. Kangas, T. K. K. and Laaksonen, J. T. (1990). Variants of self-organizing maps. *IEEE Transactions on Neural Networks*, 1(1):93-99.

Jain, A., Murty, M., and Flynn, P. (1999). Data clustering: A review. *ACM Comput. Surv.*, 31(3):264-323.

Kohonen, T. (1995). *Self-Organisation Maps*. Berlin: Springer.

L. Breiman, J. Friedman, C. J. S. and Olshen, R. A. (1984). *Classification and Regression Trees*. Chapman and Hall/CRC, Boca Raton.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proc. of the Fifth Berkeley Symposium on Math., Stat. and Prob.*, 1:281–296.

Milligan, G. W. (1996). *Clustering Validation: Results and Implications for Applied Analysis*. Word Scientific.

Spaeth, H. (1980). *Cluster Analysis Algorithms*. New York: Wiley.

Theodoris, S. and Koutroumbas, K. (2006). *Pattern Recognition*. Academic Press, 3a edition.

van Rijsbergen, C. J. (1979). *Information Retrieval*. London: Butterworths.

Uso de Combinação de Reservoir para Verificação de Assinaturas Manuscritas Off-line

Saulo Henrique L. de M. Nápoles, Cleber Zanchettin

[1]Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Av. Jornalista Anibal Fernandes, s/n - Cidade Universitária – Recife – PE – Brazil

{shlmn, cz}@cin.ufpe.br

Abstract. *The handwritten signature is present in all important documents. In law, if the signature on a document is false, this document is also considered a fraud. To automate the task of verification of signatures, many classifiers have been tested, often with few practical results promising. This work proposed uses a combination of Reservoir Computings for handwritten signature verification, for their recurrent characteristics and training facility. The experiments with this promising technique are performed in a public database used in the competition for ICDAR 2009 and their results compared to the competitors.*

Resumo. *A assinatura manuscrita é marca presente em todos os documentos importantes. Perante a lei, se a assinatura em um documento é falsa, esse documento também é considerado uma fraude. Para automatizar a tarefa de verificação de assinaturas, muitos classificadores já foram testados, muitas vezes com resultados práticos pouco promissores. Neste trabalho é proposta a utilização da combinação de Reservoir Computings para verificação de assinaturas manuscritas, devido as suas características recorrentes e facilidade de treinamento. Os experimentos com esta promissora técnica são realizados em uma base de dados pública utilizada na competição do ICDAR 2009 e seus resultados comparados aos dos competidores.*

1. Introdução

Ao longo da história os documentos e as assinaturas apostas neles sempre necessitaram da garantia de sua autenticidade. Na Idade Média, a documentação régia sempre vinha com o selo real garantindo sua autenticidade. No Sistema Jurídico Visigótico, o mais intelectualizado ramo do direito germânico, existia a confirmação do documento pelas testemunhas que o tocavam, assinavam e sobrescreviam. Os documentos privados eram, em ocasiões, confirmados por documentos reais [Freitas *et al* 2007].

A assinatura é marca presente em todos os documentos importantes, desde pagamentos em cheques ou cartões de crédito, processos judiciais, até contratos e compromissos de negócios dos mais variados. Em um documento, ela prova que o assinante está de acordo com o que está escrito anteriormente, portanto, a identificação e autenticação dos documentos digitais é um dos principais aspectos a serem considerados para garantir a segurança e autenticidade das suas informações, visto que o atual aumento das informações no mundo digital vem exigindo meios mais seguros para a proteção uma vez que, a falsidade documental e o estelionato, em todos seus aspectos, constituem crime e se a assinatura em um documento é falsa, este documento também é considerado inválido [Huang e Yan 1997].

Uma das maneiras de validar uma assinatura é utilizando Sistemas de Verificação que visa examinar a autenticidade da assinatura manuscrita através de métodos que possam discriminá-la de uma falsificação [Heinen e Osório 2004]. Estes sistemas podem examinar a assinatura de maneira *on-line* ou *off-line*. Na abordagem *on-line* é necessário um hardware específico, que pode ser uma mesa digitalizadora ou uma caneta sensível e um tablet, para o indivíduo assinar, sendo assim, características dinâmicas também são observadas como: velocidade, força, pressão. Na abordagem *off-line*, a assinatura é realizada em papel e posteriormente digitalizada para serem extraídas as características estáticas, e, em seguida, fazer a verificação semelhante à verificação de imagens [Sisodia e Anand 2009].

As falsificações são classificadas em três subconjuntos: aleatórias, simples e simuladas. A falsificação aleatória é normalmente uma amostra genuína de outro autor. A falsificação simples ocorre quando o falsificador conhece o nome do autor, mas não possui um exemplo da assinatura a qual ele planeja falsificar. Por fim, a falsificação simulada ocorre quando o falsificador possui um exemplo da assinatura e faz uma imitação da assinatura genuína [Gonçalves 2008].

A verificação de assinatura consiste em, a partir de um exemplar da assinatura, verificar se ela pertence ou não ao autor. Em outras palavras, se ela foi ou não grafada pelo suposto autor. Este tipo de verificação encontra duas dificuldades: as variações interpessoal e intrapessoal. A primeira ocorrência parte de que duas pessoas não possuem a mesma escrita, já o segundo considera que ninguém escreve duas vezes exatamente igual como pode-se observar na Figura 1 com duas assinaturas genuínas sobrepostas [Freitas *et al* 2007].



Figura 1: Variação intrapessoal de assinaturas genuínas (extraído de [Freitas *et al* 2007])

O objetivo desse trabalho é desenvolver uma técnica de classificação baseada na combinação de redes *Reservoir Computing* para verificação de assinaturas manuscritas *off-line* que ofereça bons resultados práticos utilizando uma base de dados reais e pública. O restante do material se organiza da seguinte maneira: na seção 2 é realizado um levantamento bibliográfico dos sistemas de verificação de assinatura e classificadores relacionados com este trabalho. Na seção 3 é descrito como o modelo de classificação, baseado em *Reservoir*, foi construído e são apresentadas as características dos experimentos realizados. Na seção 4 são apresentados os resultados dos experimentos e uma discussão sobre os mesmos. Por fim, as conclusões e as indicações de trabalhos futuros são apresentadas na seção 5.

2. Trabalhos Relacionados

Um número considerável de trabalhos relacionados com a verificação de assinaturas *off-line* vem sendo realizados recentemente [Batista *et al* 2007] [Impedovo e Pirlo 2008] [Dimauro ET AL 2004] [Prakash e Guru 2009] [Shankar e Rajagopalan 2007] [Justino *et al* 2005]. Conforme esses estudos, o primeiro trabalho na área de verificação de assinaturas *off-line* fazendo uso de sistemas conexionistas foi proposto por Mighell *et*

al. (1989), utilizando 80 assinaturas genuínas e 66 falsificações simuladas produzidas por um mesmo indivíduo.

Atualmente os trabalhos que envolvem Sistema de Verificação de Assinaturas utilizam como características para discriminação das assinaturas a área da imagem, altura e largura, razão do tamanho, características de direção, orientação e inclinação, contornos, texturas [Dimauro *et al* 2004]. Também podem ser usados dados simbólicos como proporção contínua, intervalo discreto ou multivalorado, multivalorado com tamanho, quantitativos, categóricos, etc [Prakash e Guru 2009].

Os classificadores mais utilizados na verificação são Redes Neurais Artificiais [Bajal e Chaudhary 1997], Dynamic Time Warping (DTW) [Shankar e Rajagopalan 2007], Cadeias Escondidas de Markov (HMM) e Máquinas de Vetor de Suporte (SVM) [Justino *et al* 2005]. Algumas abordagens utilizam uma combinação dessas técnicas formando um novo método híbrido. Pode ser utilizado um Algoritmo Genético para escolha das características a serem utilizadas numa rede neural [Xuhua *et al* 1997], combinação de SVMs utilizando Algoritmo Genético [Bertolini *et al* 2010], entre outras formas e obtêm resultados satisfatórios.

As redes neurais *feed-forward* são muito utilizadas em problemas não temporais, porém para problemas temporais, como previsão do tempo e financeiro, sistemas de identificação, sistemas de visão e fala, é necessário acrescentar conexões recorrentes para obter resultados melhores. Desta forma, uma assinatura manuscrita também pode ser considerada um padrão temporal, principalmente em sistemas de verificação *on-line*. Em sistemas de verificação *off-line*, várias destas características temporais estão presentes, como variações de traços, posição de escrita e linha, e principalmente a variação da assinatura de um mesmo assinante com o passar do tempo.

Em contrapartida às vantagens dos classificadores que trabalham estas características, ao acrescentar essas conexões recorrentes, o treinamento da rede se torna mais difícil, pois o classificador é transformado num sistema dinâmico bastante complexo [Holzmann 2009]. Para minimizar essa complexidade, três estudos independentes propuseram uma solução que tinha muito em comum: *Liquid State Machine* [Maass *et al* 2002], *Echo State Machine* [Jaeger e Haas 2004] e *Backpropagation Decorrelation* (BPDC) [Steil 2004]. Cada uma dessas técnicas propostas busca evitar o problema da formação de uma rede neural recorrente enquanto continuam a ser capazes de usar sua poderosa capacidade de processamento temporal. O *Reservoir Computing* utiliza a rede como um reservatório, cujos pesos não são alterados no treinamento e são escolhidos de forma espaça e aleatória. A resposta do reservatório é observada a partir da camada de saída por uma classificação simples utilizando uma função de avaliação linear [Schrauwen *et at* 2007].

Esta abordagem combina convenientemente a memória de curto prazo do reservatório com a facilidade do treinamento e da rápida convergência de uma função simples e linear. O reservatório funciona como um complexo *kernel* que filtra e calcula as combinações aleatórias das entradas atuais e passadas, com uma memória finita. A camada de saída, então, é capaz de combinar linearmente a informação do *Reservoir* com a saída [Embrechts 2009]. O modelo do *Reservoir* pode ser vista na Figura 2. A circunferência no centro representa o *Reservoir,* na qual os pesos entre suas conexões são inseridos aleatoriamente. O retângulo representa a camada de saída. A saída do *Reservoir* é função da entrada, do bias, do próprio *Reservoir* e da camada de saída como

pode ser visto na Equação (1). A resposta da camada de saída é dada pelo somatório da atualização do *Reservoir*, da camada de entrada, da própria camada de saída e do bias, como pode ser observado na Equação (2). Dessa forma, por ter poucos parâmetros para serem otimizados, não usar gradiente descendente, a atualização dos pesos não acontecer em toda a rede, o *Reservoir Computing* apresenta uma redução no tempo e garantia de convergência da rede, além de um controle maior em outras propriedades como: raio espectral, dispersão, conexões, diferentemente das redes neurais recorrentes.

$$(1)\quad x(t+1) = f\left(W_{res}^{res}x(t) + W_{inp}^{res}u(t) + W_{out}^{res}y(t) + W_{bias}^{res}\right)$$

$$(2)\quad \hat{y}(t+1) = W_{res}^{out}x(t) + W_{inp}^{out}u(t) + W_{out}^{out}y(t) + W_{bias}^{out}$$

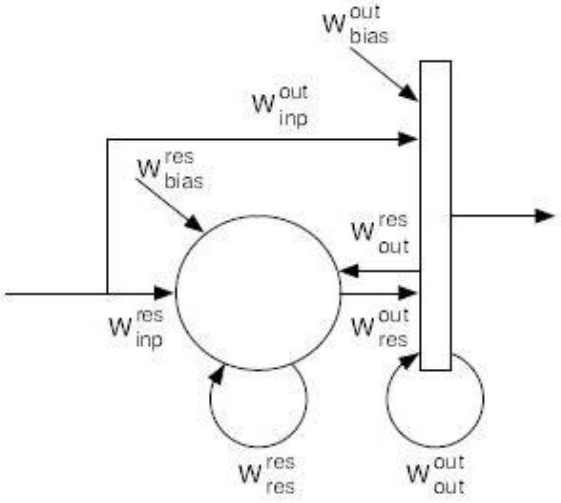


Figura 2. Topologia do *Reservoir Computing* (adaptado de [Schrauwen *et al* 2007])

3. Método Proposto

O modelo proposto neste trabalho é uma combinação de *Reservoirs Computing*. A idéia é combinar as saídas de vários classificadores com base *Reservoir* e diminuir as chances de um classificador isolado fazer uma escolha errada, em nosso caso, uma validação de uma assinatura falsa. Porém, combinar as saídas de classificadores só é útil quando estes classificadores produzem saídas distintas [Kuncheva 2004]. Apesar do método proposto utilizar diferentes instâncias do mesmo classificador, pelo fato do *Reservoir Computing* apresentar um caráter randômico nas suas ligações, é esperado que eles apresentem saídas diferentes ou que representem diferentes pontos da superfície de resposta do problema [Verstraeten *et al* 2006]. A combinação das saídas dos classificadores utilizada neste trabalho é bem simples, realizada através do voto majoritário, sendo escolhida como saída final, a resposta que mais apareceu individualmente nos classificadores *Reservoir* isolados.

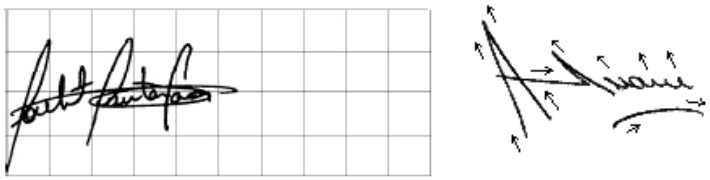
A base de dados utilizada nos experimentos foi a mesma utilizada na competição de verificação de assinaturas *off-line* do *International Conference on Document Analysis and Recognition* (ICDAR) 2009. O ICDAR é uma conferência internacional para pesquisadores e profissionais da área de reconhecimento de documentos que

acontece bienalmente. Esta conferência objetiva a identificação, o incentivo e a troca de ideias sobre o estado da arte na análise de documentos, compreensão, recuperação e avaliação de desempenho, incluindo várias formas de documentos multimídia [ICDAR 2011]. Paralela as atividades da conferência, existem várias competições, dentre as quais a competição de verificação de assinaturas *on-line e off-line*. No ano de 2009, participaram desta competição 8 equipes de pesquisadores da Espanha, França, Alemanha e Estados Unidos.

A base de dados disponibilizada na competição de 2009 é composta por 1.920 imagens sendo 60 assinaturas genuínas de 12 escritores e 1.860 falsificações do tipo simuladas, sendo 31 escritores falsos e 5 falsificações para cada assinatura genuína. As imagens foram segmentadas, visualmente inspecionadas e, em seguida, pré-processadas em tons de cinza e binarizadas (limiarizadas) com 300 e 600 dpi [Blankers 2010].

Para a extração das características das imagens foi construído um grid 4x10 na imagem, subdividindo a imagem da assinatura em 40 sub-imagens (Figura 3a). Destas sub-imagens foram extraídas as densidade dos pixels que é a relação entre a quantidade total de pixels sobre os pixels de assinatura. De acordo com os estudos de Justino (2001) esta característica incorpora um descritor estático, o qual propicia uma insensibilidade às variações intrapessoais.

Também foi extraída a inclinação axial de cada sub-imagem, pois esta característica descreve aspectos dinâmicos do traçado da assinatura. Esta tarefa realiza uma análise local sobre os segmentos mais significativos, extraindo a direção através do segmento que produziu a máxima projeção. Cada segmento é representado segundo o *chain-code* com 8-vizinhos. Pode-se observar exemplos de inclinações axiais na Figura 3b. Desta forma, o algoritmo de extração de características se utiliza de características estáticas e características que descrevem o aspecto dinâmico da assinatura.



(a) (b)

Figura 3: (a) grid 4x10 subdividindo a imagem (b) Exemplo de inclinações axiais

O método de classificação proposto utiliza 6 classificadores do tipo *Reservoir Computing*. A primeira camada possui 40 nodos correspondentes às características extraídas das assinaturas, onde cada nó de entrada recebe um par que representam as duas informações. A segunda camada é o bias, que é utilizado para aumentar o grau de liberdade, permitindo uma melhor adaptação do aprendizado. A terceira camada, o reservatório propriamente dito, tem tamanho fixo e é composto por 300 nós inseridos e conectados aleatoriamente. Foram feito alguns testes iniciais para determinar o número de nós e este número foi escolhido por apresentar um melhor resultado no menor espaço de tempo entre 150, 200, 250, 350, 400, 450 e 500. Esta camada é treinada de forma *off-line* e utiliza como função de ativação a tangente hiperbólica por ser considerada a melhor função dentre as utilizadas na literatura.

No treinamento dos *Reservoir Computing* e análise dos resultados foi utilizada a validação cruzada. Na validação cruzada, os dados empregados na etapa do treinamento não são utilizados na etapa de teste. Neste procedimento foi realizada a paginação em 5 *folds*, onde os dados são divididos em 5 subconjuntos, nos quais é realizado o treinamento com cada subconjunto e o teste feito com os dados restantes. Ao final das execuções, a taxa de erro global é calculada pela média dos 5 desempenhos obtidos com cada subconjunto.

4. Resultados e Discussão

A análise do desempenho do método proposto foi realizada de forma análoga a realizada na competição ICDAR 2009, que é baseada apenas na taxa de erro global computado a partir da base de dados testada.

Após executar o algoritmo 60 vezes, devido às características estocásticas do modelo, observou-se que a média da taxa de erro foi de 13,47% e desvio padrão de 1,63. A taxa de erro de cada execução pode ser observada na Figura 4, na qual o eixo horizontal indica o número da execução do algoritmo e o eixo vertical a taxa de erro naquele instante. Comparado estes resultados com os resultados dos algoritmos que competiram no ICDAR 2009, a combinação de *Reservoir* proposta perdeu em desempenho apenas para o algoritmo do *Centre de Morphologie Mathématique* que obteve uma taxa de erro médio de 9,15%. O desempenho dos competidores e do método proposto no trabalho pode ser observado na Tabela 1 que está ordenada pela taxa de erro médio. É importante observar que apenas a comparação entre as taxas de erro não é uma boa estatística paramétrica para comparar o desempenho de classificadores. É importante destacar também que apesar deste trabalho seguir todos os procedimentos destacados na competição ICDAR 2009, os resultados não podem ser comparados diretamente (ou pelo menos as análises precisam considerar), pois foram desenvolvidos em ambientes diferentes. Para tentar minimizar esta dificuldade foi realizado um teste de hipóteses com variância desconhecida utilizando a distribuição *t-Student*.

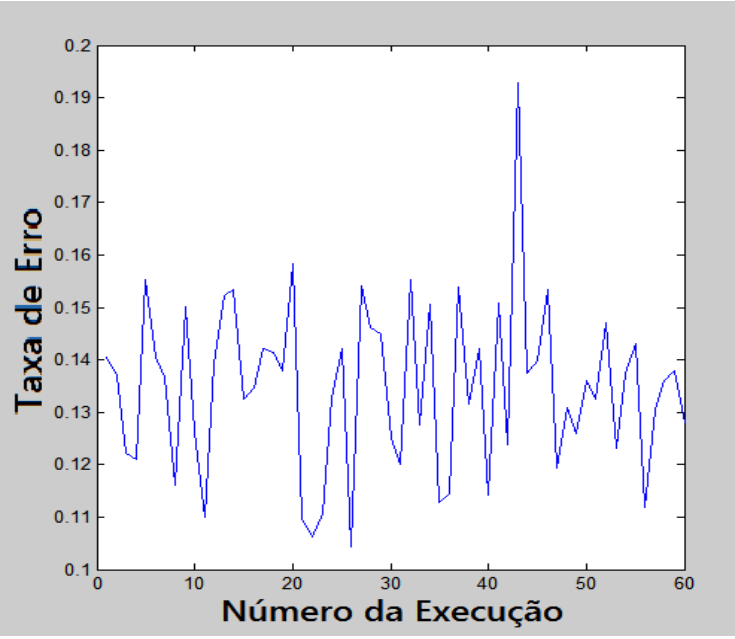


Figura 4. Taxa de erro médio no teste

Tabela 1. Taxa de Erro Médio dos competidores

Instituto de Pesquisa	País	Taxa de Erro (%)
Centre de Morphologie Math´ematique	França	9,15
Comitê de Reservoir (CIN-UFPE)	**Brasil**	**13,47**
German Research Center for Artificial Intelligence	Alemanha	15,92
Biometric Recognition Group - ATVS	Espanha	18,27
Center of Excellence for Document Analysis and Recognition (CEDAR)	EUA	23,00
Computer Vision Center	Espanha	41,12

Para realizar o teste de hipótese de cada resultado dos competidores e o algoritmo desenvolvido, iremos adotar a hipótese nula H_0 como sendo $\mu_0 = \mu$ e a hipótese alternativa H_1 como sendo $\mu_0 < \mu$ onde μ é a média do competidor, e μ_0 a média do algoritmo proposto. Foi observada a distribuição com nível de significância de 99%. O resultado pode ser conferido na Tabela 2, que confirma apenas o algoritmo francês como sendo melhor do que o algoritmo proposto.

Porém, por não se ter encontrado referência aos métodos dos competidores, fica difícil fazer mais comparações entre as técnicas confrontadas. Não se pode dizer qual técnica é melhor para ser utilizada com verificação de assinaturas, pois não foram encontradas informações de como os métodos foram construídos para esta competição ou que tenham utilizados esta mesma base de dados, isso dificulta a agregação de melhorias nas técnicas atuais porque não se sabe onde um algoritmo foi melhor do que o outro e poder juntar essas duas técnicas em apenas uma que tenha um desempenho superior as demais.

Esta dificuldade em encontrar referências formais às técnicas se deve ao fato da maioria delas serem soluções estratégicas, com aplicações comerciais. Desta forma a maioria das instituições mantém em segredo como são constituídos seus classificadores para verificação de assinaturas.

Tabela 2. Resultado dos Testes de hipóteses

Instituto	País	Erro Médio	t_{cal}	Resultado
C. de Morphologie Mathématique	**França**	**9,15**	**9,81**	**Não Rejeita H_0**
German Research Center for A.I.	Alemanha	15,92	-5,21	Rejeita H_0
ATVS	Espanha	18,27	-10,21	Rejeita H_0
CEDAR	EUA	23,00	-35,46	Rejeita H_0
Computer Vision Center	Espanha	41,12	-74,02	Rejeita H_0

5. Conclusões e Trabalhos Futuros

Neste trabalho foi desenvolvida uma técnica conexionista, baseada em *Reservoir Computing*, para verificação de assinaturas manuscritas *off-line*, que se mostrou promissora quando comparada com diferentes técnicas de classificação de assinaturas utilizadas na atualidade. O *Reservoir Computing* usa do recurso de conexões recorrentes, sem se tornar um sistema complexo, podendo ser utilizado em problemas de difícil resolução com abordagens clássicas.

Na avaliação do modelo, foi usada uma base de imagens de assinatura pública adotada na competição de verificação de assinaturas da ICDAR 2009. As imagens foram subdivididas com um *grid* 4x10 e a partir de cada um desses, foram extraídas as características densidade de pixels e inclinação de axiomas, que denotam peculiaridades estáticas e dinâmicas da assinatura. Em seguida foi submetido aos classificadores, utilizando validação cruzada com 5-*fold.* As saídas dos classificadores eram combinadas em uma única a partir do voto majoritário.

A medida de desempenho utilizada foi a taxa média de erro, que também foi usada na competição ICDAR 2009. Comparado com os demais competidores, o algoritmo proposto obteve um ótimo desempenho, perdeu apenas para um método, e o teste de hipótese realizado com os resultados dos experimentos ratifica essa conclusão.

A partir dos resultados dos experimentos realizados pode ser verificada a capacidade desta nova rede neural artificial e sua aplicabilidade no sistema de verificação de assinaturas manuscritas *off-line*, que também pode ser estendido para outros problemas, como de previsão por exemplo, devido a existência das mesmas características temporais, existente no problema abordado.

Entretanto, melhorias podem ser realizadas para que o método proposto tenha um desempenho ainda melhor, tais como: (i) mesclar com outras técnicas de classificação ou adicionar um procedimento para explorar a topologia da rede, (ii) incluir mais classificadores no comitê desenvolvido; (iii) utilizar uma regra de combinação diferente, com o tamanho do reservatório variável; (iv) pode também ser adicionado um *grid* que gere um maior número de sub-imagens e até mesmo outras características da assinatura ou acrescentar novas baseadas nos exames grafotécnicos.. Inserir, mesclar ou modificar a combinação dos classificadores pode melhorar o desempenho visto que outros pontos do espaço de soluções podem ser explorados. Modificar o *grid* e/ou aumentar o número de características exploradas facilitará na discriminação observada entre assinaturas genuínas e falsificações.

Outro trabalho futuro a ser investigado é utilizar uma base de dados com a presença de dados simbólicos, além dos dados que já são utilizados tradicionalmente para obter mais informações sobre as assinaturas.

Agradecimentos

Este trabalho recebeu apoio financeiro da Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco (FACEPE).

Referências

Freitas, C. O. A., Justino, E. J. R., Oliveira, L. E. S. (2007) “Reconhecimento de firmas por semelhança no Brasil”. In: Âmbito Jurídico, Rio Grande, v 40, Disponível em

http://www.ambito-juridico.com.br/site/index.php?n_link=revista_artigos_leitura&artigo_id=3895. Acesso em 07/04/2011.

Huang K. e Yan, Hong. (1997) "Off-line Signature Verification based on Geometric Feature Extration and Neural Network Classification". Pattern Recognition, Vol 30, No. 1, pp. 9-17.

Heinen, M. R., Osorio, F. S. (2004) "Biometria Comportamental: Pesquisa e desenvolvimento de um sistema de autenticação de usuários utilizando assinaturas manuscritas". INFOCOMP (UFLA. Impresso) Lavras, MG, Brasil, v. 3, n. 2, p. 31-37, 2004.

Sisodia, K. e Anand, M. (2009) "Off-line Handwritten Signature Verification using Artificial Neural Network Classifier". International Journal of Recent Trends in Engineering, Vol 2, no. 2, Novembro.

Gonçalves, D. B. (2008) "Agrupamento de Classificadores na Verificação de Assinaturas *off-line*". Setembro de 2008.86p. Dissertação (Mestrado em Informática) – Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Paraná, Curitiba.

Batista, L., Rivard, D., Sabourin, R., Granger, E., Maupin, P. (2007) "State Of The Art In Off-Line Signature Verification". Pattern Recognition Technologies and Applications: Recent Advances.

Impedovo, D. e Pirlo, G. (2008) "Automatic Signature Verification: The State of the Art". IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, Vol 38, pp. 609–635.

Mighell, D. A., Wilkinson, T. S., Goodman, J. W. (1989) "Backpropagation and Its Application to Handwritten Signature Verification". Advances in Neural Information Processing Systems 1, pp. 340–347.

Dimauro, G., Impedovo, S., Lucchese, M. G., Modugno, R., Pirlo, G. (2004) "Recent Advancement in Automatic Signature Verification". Proceedings of 9th International Workshop on Frontiers in Handwritting Recognition (IWFHR), pp. 179-184. Tókio, Japão

Prakash, H. N. e Guru, D. S. (2009) "Relative Orientations of Geometrics Centroids for Off-line Signature Verification". International Conference on Advanced Pattern Recognition (ICAPR-2009), ISI, Kolkata, India pp. 201-204.

Bajaj, R. e Chaudhary, S. (1997) "Signature Verification using mutiple neural classifiers". Pattern Recognition, Vol 30, pp. 1-87.

Shankar, A. P. e Rajagopalan (2007) "Off-line Signature Verification using DWT". Pattern Recognition Letters, Vol 28, PP. 1407-1414.

Justino, E. J. R. Bortolazzi, F. e Sabourin, R. (2005) "A Comparasion of SVM and HMM classifiers in the Off-line Signature Verification". Pattern Recognition Letters, Vol 26, Issue 9, pp. 1377-1385.

Xuhua, Y., Furuhashi, T., Obata, K., Uchikawa, Y. (1997) "Selection of Features for Signature Verification Using the Genetic Algorithm". Computers & Industrial Engineering Vol. 30, No. 4, pp. 1037-1045.

Bertolini, D., Oliveira, L. S., Justino, E., Sabourin, R. (2010) "Reducing Forgeries in Writer-independent Off-line Signature Verification Through Ensemble of Cassifier". Pattern Recognition, Vol 43, pp. 387-396.

Holzmann G. (2009) "Reservoir Computing: A Powerful Black-Box Framework for Nonlinear Audio Processing". Proceedings of the 12th International Conference on Digital Audio Effects (DAFx-09), Como, Italy, 2009.

W. Maass, T. Natschläger, and H. Markram. (2002) "Realtime computing without stable states: A new framework for neural computation based on perturbations". Neural Computation, Vol 14 No.11 pp.2531–2560.

H. Jaeger and H. Haas. (2004) "Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless telecommunication". Science, No. 308 pp. 78–80.

J. J. Steil. (2004) "Backpropagation-Decorrelation: Online recurrent learning with O(N) complexity." In Proceedings of International Joint Conference on Neural Network 2004 Budapeste Hungria, volume 1, pp 843–848.

Schrauwen, B. Verstraeten, D. Campenhout, J. V.(2007) "An overview of reservoir computing: theory, applications and implementations". Proceedings of the 15th European Symposium on Artificial Neural Networks (2007) pp. 471-482. Bruges, Bélgica.

Embrechts, M. J. Alexandre, L.A. Linton, J. D. (2009) "Reservoir computing for static pattern recognition". 17th European Symposium on Artificial Neural Networks - ESANN 2009, Bruges, Bélgica. pp 245-250.

Kuncheva, L. (2004) "Combining Pattern Classifier, Methods and Algorithms" Wiley, New York.

Verstraeten, D. Schrauwen, B. D'Haene, M. Stroobandt, D. (2006) "The unified Reservoir Computing concept and its digital hardware implementations". Proceedings of the 2006 EPFL LATSIS Symposium. Lausanne, Suiça

Blankers V.L., Heuvel C.E. van den., Franke K.Y., Vuurpijl L.G.(2010) "The ICDAR 2009 Signature Verification Competition". 10th International Conference on Document Analysis and Recognition. Barcelona, Espanha

Justino, E. J. R., Bortolozzi, F., and Sabourin, R. (2001) "Offline signature verification using hmm for random" In ICDAR 2001, International Conference on Document Analysis and Recognition, pp. 1031–1034.

IGMN: An Incremental Gaussian Mixture Network that Learns Instantaneously from Data Flows

Milton Roberto Heinen[1], **Paulo Martins Engel**[1] **and Rafael C. Pinto**[1]

[1]Informatics Institute – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15064, CEP 91501-970, Porto Alegre-RS, Brazil

{mrheinen,engel,rcpinto}@inf.ufrgs.br

Abstract. *This works proposes IGMN (standing for Incremental Gaussian Mixture Network), a new connectionist approach for incremental concept formation and robotic tasks. It is inspired on recent theories about the brain, specially the Memory-Prediction Framework and the Constructivist Artificial Intelligence, which endows it with some unique features that are not present in most ANN models such as MLP and GRNN. Moreover, IGMN is based on strong statistical principles (Gaussian mixture models) and asymptotically converges to the optimal regression surface as more training data arrive. Through several experiments using the proposed model it is demonstrated that IGMN is also robust to overfitting, does not require fine-tunning its configuration parameters and has a very good computational performance, thus allowing its use in real time control applications. Therefore, IGMN is a very useful machine learning tool for incremental function approximation.*

1. Introduction

Artificial neural networks (ANNs) [Haykin 2008] are mathematical or computational models inspired by the structure and functional aspects of biological neural networks. They are composed by several layers of massively interconnected processing units, called artificial neurons, which can change their connection strength (i.e., the synaptic weights values) based on external or internal information that flows through the network during learning. Modern ANNs are non-linear machine learning [Mitchell 1997] tools frequently used to model complex relationships between inputs and outputs and/or to find patterns in data. In the past several neural network models have been proposed. The most well-known model is the Multi-Layer Perceptron (MLP) [Rumelhart et al. 1986], which can be used in function approximation and classification tasks. The MLP supervised learning algorithm, called Backpropagation, uses gradient descent to minimize the mean square error between the desired and actual ANN outputs.

Although in the last decades neural networks have been successfully used in several tasks, including signal processing, pattern recognition and robotics, most ANN models have some disadvantages that difficult their use in incremental function approximation and prediction tasks. The Backpropagation learning algorithm, for instance, requires several scans over all training data, which must be complete and available at the beginning of the learning process, to converge for a good solution. Moreover, after the end of the training process the synaptic weights are "frozen", i.e., the network loses its learning capabilities. These drawbacks highly contrast with the human brain learning capabilities because: (i) we don't need to perform thousands of scans over the training data to learn

something (in general we are able to learn using few examples and/or repetitions, the so called *aggressive learning*); (ii) we are always learning new concepts as new "training data" arrive, i.e., we are always improving our performance through experience; and (iii) we don't have to wait until sufficient information arrives to make a decision, i.e., we can use partial information as it becomes available. Besides being not biologically plausible, these drawbacks difficult the use of ANNs in tasks like incremental concept formation, reinforcement learning and robotics, because in this kind of application the training data are available just instantaneously to the learning system, and in general a decision must be made using the information available at the moment.

This paper proposes a new artificial neural network model, called IGMN (standing for Incremental Gaussian Mixture Network) [Heinen 2011], which is able to tackle great part of these problems described above. IGMN is based on parametric probabilistic models (Gaussian mixture models), that have nice features from the representational point of view, describing noisy environments in a very parsimonious way, with parameters that are readily understandable. In [Heinen and Engel 2010a, Heinen and Engel 2010b] a previous version of IGMN, called IPNN, was proposed. That previous version is based on the supposition of conditional independence among distinct domains, called "naive Bayes" hypothesis. The neural network model proposed in this paper, on the other hand: (i) takes into account the correlation among distinct domains, thus fully implementing a multivariate hypothesis; (ii) uses an error driven mechanism to decide if it is necessary to add a neuron to the neural network; and IGMN (iii) has a unique neural network topology in which there is no distinction between sensory and motor stimuli.

Moreover, IGMN is inspired on recent theories about the brain, specially the Memory-Prediction Framework (MPF) [Hawkins 2005] and the constructivist artificial intelligence [Drescher 1991], which endows it with some unique features that are not present in other neural network models such as: (i) it learns incrementally using a single scan over the training data; (ii) the learning process can proceed perpetually as new training data arrive; (iii) it can handle the stability-plasticity dilemma and does not suffer from catastrophic interference; (iv) the neural network topology is defined automatically and incrementally; and (v) IGMN is not sensitive to initialization conditions. The remaining of this paper is organized as follows. Section 2 presents the ANN model proposed in this paper; Section 3 describes some experiments performed to evaluate the proposed model; and Section 4 provides some final remarks about this work.

2. Incremental Gaussian Mixture Network

Figure 1 shows the general architecture of IGMN. It is composed by an association region $\mathcal{P}$ (in the top of this figure) and many cortical regions, $\mathcal{N}^{\mathcal{A}}, \mathcal{N}^{\mathcal{B}}, \ldots, \mathcal{N}^{\mathcal{S}}$. All regions have the same number of neurons, M. Initially there is a single neuron in each region (i.e., $M = 1$), but more neurons are incrementally added when necessary using an error driven mechanism. Each cortical region $\mathcal{N}^{\mathcal{K}}$ receives signals from the kth sensory/motor modality, $\mathbf{k}$ (in IGMN there is no difference between sensory and motor modalities), and hence there is a cortical region for each sensory/motor modality.

Another important feature of IGMN is that all cortical regions $\mathcal{N}$ execute a common function, i.e., they have the same kind of neurons and use the same learning algorithm. Moreover, all cortical regions can run in parallel, which improves the performance

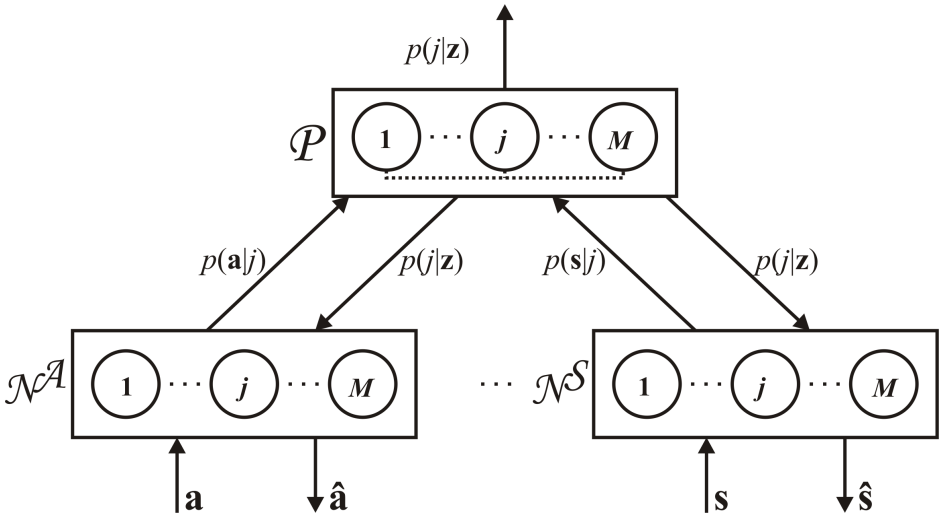


Figure 1. General architecture of the proposed neural network model

specially in parallel architectures. Each neuron j of region $\mathcal{N}^{\mathcal{K}}$ performs the following operation:

$$p(\mathbf{k}|j) = \frac{1}{(2\pi)^{D^{\mathcal{K}}/2}\sqrt{\left|\mathbf{C}_j^{\mathcal{K}}\right|}} \exp\left\{-\frac{1}{2}(\mathbf{k}-\boldsymbol{\mu}_j^{\mathcal{K}})^T \mathbf{C}_j^{\mathcal{K}-1}(\mathbf{k}-\boldsymbol{\mu}_j^{\mathcal{K}})\right\}, \tag{1}$$

i.e., a multivariate Gaussian distribution, where $D^{\mathcal{K}}$ is the dimensionality of $\mathbf{k}$ (different sensory/motor modalities $\mathbf{k}$ can have different dimensions $D^{\mathcal{K}}$). Each neuron j maintains a mean vector $\boldsymbol{\mu}_j^{\mathcal{K}}$ and a covariance matrix $\mathbf{C}_j^{\mathcal{K}}$.

In IGMN the regions are not fully connected, i.e., the neuron j of $\mathcal{N}^{\mathcal{K}}$ is connected just to the jth neuron of $\mathcal{P}$, but this connection is bidirectional. It is important to notice that there are no *synaptic weights* in these connections, i.e., all IGMN parameters are stored in the neurons themselves. A bottom-up connection between $\mathcal{N}^{\mathcal{K}}$ and $\mathcal{P}$ provides the component density function $p(\mathbf{k}|j)$ to the jth neuron in $\mathcal{P}$. Therefore, a neuron j in the association region $\mathcal{P}$ is connected with the jth neuron of all cortical regions $\mathcal{N}$ via bottom-up connections and computes the a posteriori probability using the Bayes' rule:

$$p(j|\mathbf{z}) = \frac{p(\mathbf{a}|j)\, p(\mathbf{b}|j) \ldots p(\mathbf{s}|j)\; p(j)}{\sum_{q=1}^{M} p(\mathbf{a}|q)\, p(\mathbf{b}|q) \ldots p(\mathbf{s}|q)\; p(q)}, \tag{2}$$

where it is considered that the neural network has an arbitrary number, s, of cortical regions and $\mathbf{z} = \{\mathbf{a}, \mathbf{b}, \ldots, \mathbf{s}\}$. The dotted lines in Figure 1 above indicate the lateral interaction among the association units for computing the denominator in (2).

Each neuron j of the association region $\mathcal{P}$ maintains its a priori probability, $p(j)$, an accumulator of the a posteriori probabilities, sp_j, and an association matrix to store the correlations among each sensory/motor modality. If a neural network has two cortical regions, $\mathcal{N}^{\mathcal{A}}$ and $\mathcal{N}^{\mathcal{B}}$, for instance, then the association matrix $\mathbf{C}_j^{\mathcal{AB}}$ will have two dimensions and size $D^{\mathcal{A}} \times D^{\mathcal{B}}$. Note that it is not necessary to maintain $\mathbf{C}_j^{\mathcal{BA}}$ because $\mathbf{C}_j^{\mathcal{BA}} = \mathbf{C}_j^{\mathcal{AB}^T}$. The top-down connections between $\mathcal{P}$ and $\mathcal{N}^{\mathcal{K}}$, on the other hand, returns *expectations* to $\mathcal{N}^{\mathcal{K}}$ which are used to estimate $\hat{\mathbf{k}}$ when $\mathbf{k}$ is missing. This architecture

is inspired on the memory-prediction framework (MPF) [Hawkins 2005], which states that different cortical regions are not fully connected in the neocortex. Instead, they are linked to the association areas $\mathcal{P}$ through bottom-up and top-down connections, thus providing predictions and expectations, respectively, to all cortical regions $\mathcal{N}^K$. The main advantage of this strategy is to speed up IGMN and make it more suitable to real-time and critical applications, because it is much faster to invert two covariance matrices of size M than a single covariance matrix of size $2M$. Moreover, a large number of samples is required to obtain good estimates from a large covariance matrix, and therefore using this strategy IGMN becomes more aggressive, i.e., it is able to provide good estimates using few training samples. Next subsections describe the IGMN operation in details.

2.1. IGMN operation

IGMN has two operation modes, called *learning* and *recalling*. But unlike most ANN models, in IGMN these operations don't need to occur separately, i.e., the learning and recalling modes can be intercalated. In fact, even after the presentation of a single training pattern the neural network can already be used in the recalling mode (the acquired knowledge can be immediately used), and the estimates become more precise as more training data are presented. Moreover, the learning process can proceed perpetually, i.e., the neural network parameters can always be updated as new training data arrive. As described in [Heinen 2011], BP-trained MLP neural networks cannot learn perpetually because: (i) they suffer from catastrophic interference; (ii) do not have means to handle the stability plasticity dilemma; and (iii) require that the entire database be available at the beginning of the learning process [Haykin 2008].

IGMN adopts an error-driven mechanism to decide if it is necessary to add a neuron in each region for *explaining* a new data vector $\mathbf{z}^t$. This error-driven mechanism is inspired on the Constructivist IA [Drescher 1991, Chaput 2004], where the accommodation process occurs when it is necessary to change the neural network structure (i.e. to add a neuron in each region) to account for a new *experience* which is not explained for the current schemata (i.e., the current ANN structure), and the assimilation process occurs when the new experience is well explained in terms of the existing schemata [Piaget 1954]. In mathematical terms, the ANN structure is changed if the instantaneous approximation error ε is larger that a user specified threshold ε_{max}.

The following subsections describe the IGMN operation during learning and recalling. To simplify our explanation, we will consider that the neural network has just two cortical regions, $\mathcal{N}^{\mathcal{A}}$ and $\mathcal{N}^{\mathcal{B}}$, that receive the stimuli $\mathbf{a}$ and $\mathbf{b}$, respectively. It will be also considered that we are estimating $\hat{\mathbf{b}}$ from $\mathbf{a}$ in the recalling mode. But it is important to remember that: (i) IGMN can have more than two cortical regions (one for each sensory/motor stimulus $\mathbf{k}$); and (ii) after training it can be used to estimate either $\hat{\mathbf{a}}$ or $\hat{\mathbf{b}}$ (i.e., there is no difference between *inputs* and *outputs* in IGMN).

2.2. Learning mode

The learning algorithm used by IGMN is based on the incremental Gaussian mixture model learning algorithm presented in [Engel and Heinen 2010a,Engel and Heinen 2010b], but it has many modifications which adapt it to supervised tasks such as incremental function approximation and prediction. Before learning starts the neural network is empty, i.e., all regions have $M = 0$ neurons. When the first training pattern $\mathbf{z}^1 = \{\mathbf{a}^1, \mathbf{b}^1\}$ arrives,

a neuron in each region is created centered on $\mathbf{z}^1$ and the neural network parameters are initialized as follows:

$$M = 1; \quad sp_1 = 1.0; \quad p(1) = 1.0; \quad \mathbf{C}_1^{\mathcal{AB}} = \mathbf{0};$$
$$\boldsymbol{\mu}_1^{\mathcal{A}} = \mathbf{a}^1; \quad \boldsymbol{\mu}_1^{\mathcal{B}} = \mathbf{b}^1; \quad \mathbf{C}_1^{\mathcal{A}} = {\boldsymbol{\sigma}_{ini}^{\mathcal{A}}}^2\mathbf{I}; \quad \mathbf{C}_1^{\mathcal{B}} = {\boldsymbol{\sigma}_{ini}^{\mathcal{B}}}^2\mathbf{I},$$

where the subscript '1' indicates the neuron $j = 1$ in each region, M is the number of neurons in each region (all regions have the same size M), sp is an accumulator of posterior probabilities maintained in the association region $\mathcal{P}$, $\mathbf{0}$ is a zero matrix of size $D^{\mathcal{A}} \times D^{\mathcal{B}}$, $\boldsymbol{\sigma}_{ini}^{\mathcal{A}}$ and $\boldsymbol{\sigma}_{ini}^{\mathcal{B}}$ are diagonal matrices that define the initial radius of the covariance matrices (the pdf is initially circular but it changes to reflect the actual data distribution as new training data arrive) and $\mathbf{I}$ denotes the identity matrix. $\boldsymbol{\sigma}_{ini}^{\mathcal{A}}$ and $\boldsymbol{\sigma}_{ini}^{\mathcal{B}}$ are initialized using a user defined fraction δ of the overall variance (e.g., $\delta = 1/100$) of the corresponding attributes, estimated from the range of these values according to:

$$\boldsymbol{\sigma}_{ini}^{\mathcal{K}} = \delta \left[\max(\mathbf{k}) - \min(\mathbf{k})\right] , \tag{3}$$

where $[\min(\mathbf{k}), \max(\mathbf{k})]$ defines the domain of a sensory/motor modality $\mathbf{k}$ (throughout this paper the symbol k will be used to indicate any sensory/motor modality, i.e., either a or b in this case). It is important to say that it is not necessary to know the *exact* minimum and maximum values along each dimension to compute $\boldsymbol{\sigma}_{ini}^{\mathcal{K}}$, but just the approximate domain of each feature instead.

When a new training pattern $\mathbf{z}^t$ arrives, all cortical regions are activated, i.e., $p(\mathbf{k}|j)$ is computed using Equation 1 above, and the probabilities $p(\mathbf{z}^t|j)$ are sent to the association region $\mathcal{P}$, which computes the *joint* posterior probabilities $p(j|\mathbf{z}^t)$ using the Bayes' rule in (2). After this, the posterior probabilities $p(j|\mathbf{z}^t)$ are sent back to the cortical region $\mathcal{N}^{\mathcal{B}}$, which compute its estimate as follows:

$$\hat{\mathbf{b}} = \sum_{j=1}^{M} p(j|\mathbf{z}^t)[\boldsymbol{\mu}_j^{\mathcal{B}} + \mathbf{C}_j^{\mathcal{BA}}{\mathbf{C}_j^{\mathcal{A}}}^{-1}(\mathbf{a}^t - \boldsymbol{\mu}_j^{\mathcal{A}})], \tag{4}$$

where $\mathbf{C}_j^{\mathcal{AB}}$ is the jth association matrix maintained in association region $\mathcal{P}$ and $\mathbf{C}_j^{\mathcal{BA}}$ is its transpose. Note that as the covariance matrix $\mathbf{C}_j^{\mathcal{B}}$ was already inverted when $\mathcal{N}^{\mathcal{B}}$ was activated in the bottom-up direction, we don't need to invert it again, i.e., we can temporarily store the corresponding inverse matrix to speed up the IGMN learning algorithm. Using the estimate $\hat{\mathbf{b}}$ the normalized approximation error ε is given by:

$$\varepsilon = \max_{i \in D^{\mathcal{B}}} \left[\frac{\|b_i^t - \hat{b}_i\|}{\max(b_i) - \min(b_i)} \right] \tag{5}$$

where $[\min(b_i), \max(b_i)]$ defines the domain of the sensory/motor feature b_i. Again $\min(b_i)$ and $\max(b_i)$ do not need to be the exact minimum and maximum values of $\mathbf{b}$ – they may be just approximations of the domain of each b_i feature (in fact $\min(b_i)$ and $\max(b_i)$ are just used to make IGMN independent from the range of the data features). If ε is larger than a user specified threshold, ε_{max}, than $\mathbf{z}^t$ is not considered as *represented* by any existing neuron in the cortical regions. In this case, a new unit j is created in each region and centered on $\mathbf{z}^t$ and all priors of the association region $\mathcal{P}$ are recomputed by:

$$p(j)^* = \frac{sp_j}{\sum_{q=1}^{M^*} sp_q}. \tag{6}$$

Otherwise (if $\mathbf{z}$ is well explained by any of the existing Gaussian units), the a posteriori probabilities $p(j|\mathbf{z}^t)$ are added to the current value of the $sp(j)$ on the association region:

$$sp_j^* = sp_j + p(j|\mathbf{z}^t), \quad \forall j, \tag{7}$$

and the priors $p(j)$ are recomputed using (6). Then $\omega_j = p(j|\mathbf{z}^t)/sp_j^*$ is sent back to all cortical regions, and the parameters of all neurons in $\mathcal{N}^{\mathcal{K}}$ are updated using the following recursive equations derived in [Engel and Heinen 2010a, Engel and Heinen 2010b]:

$$\boldsymbol{\mu}_j^{\mathcal{K}*} = \boldsymbol{\mu}_j^{\mathcal{K}} + \omega_j \left(\mathbf{z}^t - \boldsymbol{\mu}_j^{\mathcal{K}}\right) \tag{8}$$

$$\mathbf{C}_j^{\mathcal{K}*} = \mathbf{C}_j^{\mathcal{K}} - (\boldsymbol{\mu}_j^{\mathcal{K}*} - \boldsymbol{\mu}_j^{\mathcal{K}})(\boldsymbol{\mu}_j^{\mathcal{K}*} - \boldsymbol{\mu}_j^{\mathcal{K}})^T + \omega_j \left[(\mathbf{z} - \boldsymbol{\mu}_j^{\mathcal{K}*})(\mathbf{z} - \boldsymbol{\mu}_j^{\mathcal{K}*})^T - \mathbf{C}_j^{\mathcal{K}}\right], \tag{9}$$

where the superscript '$*$' refers to the new (updated) values. Finally the association matrix $\mathbf{C}_j^{\mathcal{AB}}$ is updated using the following recursive equation:

$$\mathbf{C}_j^{\mathcal{AB}*} = \mathbf{C}_j^{\mathcal{AB}} - (\boldsymbol{\mu}_j^{\mathcal{A}*} - \boldsymbol{\mu}_j^{\mathcal{A}})(\boldsymbol{\mu}_j^{\mathcal{B}*} - \boldsymbol{\mu}_j^{\mathcal{B}})^T + \omega_j \left[(\mathbf{a}^t - \boldsymbol{\mu}_j^{\mathcal{A}*})(\mathbf{b}^t - \boldsymbol{\mu}_j^{\mathcal{B}*})^T - \mathbf{C}_j^{\mathcal{AB}}\right], \tag{10}$$

which is derived using the same principles described in [Engel and Heinen 2010a, Engel and Heinen 2010b]. This equation is another important contribution of this paper, because it allows computing the covariances among distinct cortical regions incrementally, and thus estimating a missing stimulus $\mathbf{k}$ without having to maintain and invert a complete variance/covariance matrix. In fact, using (10) the complete variance/covariance matrix is broken down in separate submatrices that can be efficiently maintained.

2.3. Recalling mode

In the recalling mode, a stimulus (e.g., $\mathbf{a}$) is presented to a *partially* trained neural network (as the learning process proceeds perpetually, in IGMN we never consider that the training process is over), which computes an estimate for another stimulus (e.g., $\hat{\mathbf{b}}$). As said before, IGMN can be used to estimate either $\hat{\mathbf{a}}$ or $\hat{\mathbf{b}}$, but to simplify our explanation in this and the following sections we will consider that we are estimating $\hat{\mathbf{b}}$ from $\mathbf{a}$.

Initially the stimulus $\mathbf{a}$ is received in the cortical region $\mathcal{N}^{\mathcal{A}}$, where each neuron j computes $p(\mathbf{a}|j)$ using (1). These predictions are sent to the association region $\mathcal{P}$ through the bottom-up connections, which is activated using just $p(\mathbf{a}|j)$:

$$p(j|\mathbf{a}) = \frac{p(\mathbf{a}|j)\, p(j)}{\sum_{q=1}^{M} p(\mathbf{a}|q)\, p(q)}. \tag{11}$$

After this, $p(j|\mathbf{a})$ is sent to the cortical region $\mathcal{N}^{\mathcal{B}}$ via the top-down connections, and $\mathcal{N}^{\mathcal{B}}$ computes the estimated stimulus $\hat{\mathbf{b}}$ using Equation 4. More details about the IGMN learning and recalling algorithms are found in [Heinen 2011].

The main drawback of IGMN is that it is necessary to invert the covariance matrices $\mathbf{C}_j^{\mathcal{K}}$ for each training sample $(\mathbf{a}, \mathbf{b}, \ldots, \mathbf{k})$, and this requires $D^{\log_2 7}$ operations using the Strassen algorithm [Strassen 1969]. A way to reduce the computational complexity is to separate the data features in more sensory/motor areas, thus dividing a big covariance matrix in many matrices of smaller size.

3. Experiments

This section describes several experiments to evaluate the performance of IGMN in function approximation and prediction tasks. The first set of experiments, described in Subsection 3.1 presents some experiments in which IGMN is used to identify a nonlinear plant originally proposed in [Narendra and Parthasarathy 1990]. These experiments are also used to assess the sensitivity of the proposed model to its configuration parameters. The second set of experiments, described in Section 3.2 evaluates the performance of IGMN using a more complex, three-dimensional "Mexican hat" function. The computer platform used in all experiments is a Dell Optiplex 755, equipped with an Intel(R) Core(TM)2 Duo CPU 2.33GHz processor, 64 bits architecture, 1.95GB of RAM memory, GPU Intel and operating system Ubuntu Linux 10.04 LTS of 64 bits. Other experiments, some of them related to incremental robotic tasks, can be found in [Heinen 2011].

3.1. Estimating the outputs of a nonlinear plant

The first experiment consists in identifying a nonlinear plant originally proposed by [Narendra and Parthasarathy 1990] for the control of nonlinear dynamical systems using MLP neural networks. This plant is assumed to be of the form:

$$y_p(k+1) = f[y_p(k), y_p(k-1), y_p(k-2), u(k), u(k-1)], \tag{12}$$

where $y_p(k+1)$ is the next time sample of the plant output, $y_p(k)$ is the current output, $y_p(k-1)$ and $y_p(k-2)$ are delayed time samples of the output, $u(k)$ is the current input, $u(k-1)$ is the previous input, and the unknown function $f(\cdot)$ has the form:

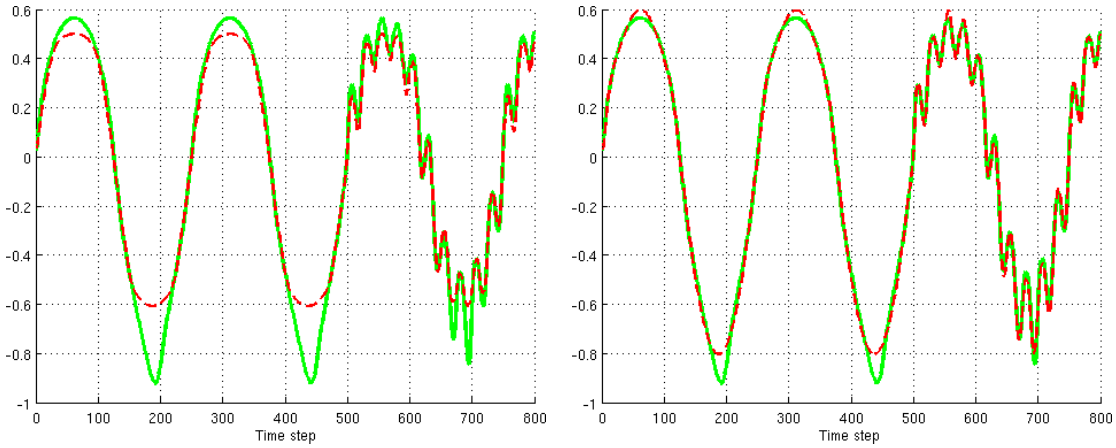
$$f[\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5] = \frac{\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_5(\mathbf{x}_3 - 1) + \mathbf{x}_4}{1 + \mathbf{x}_2^2 + \mathbf{x}_3^2}. \tag{13}$$

In [Narendra and Parthasarathy 1990] this plant was identified using a MLP neural network composed by five neurons in the input region, 20 neurons in the first hidden region, 10 neurons on the second hidden region and a single neuron in the output region. This neural network was trained using the standard backpropagation algorithm for 100,000 steps using a random input signal uniformly distributed in the interval $[-1, 1]$ and a step size of $\eta = 0.25$. During the identification procedure (i.e., the learning process) a series-parallel model was used, in which the output of the plant values (i.e., the *desired* answers) were used in the delay units $y_p(k-1)$ and $y_p(k-2)$. After identification the performance of the model was assessed using a parallel model, in which the actual neural network outputs are fed into the delay units. The identified plant (i.e., the trained neural network) was tested using the following function as input:

$$u(k) = \begin{cases} \sin(2\pi k/250) & \text{if } k \leq 500 \\ 0.8\sin(2\pi k/250) + 0.2\sin(2\pi k/25) & \text{if } k > 500 \end{cases} \tag{14}$$

In [Specht 1991] a GRNN network is used to approximate the function $f(\cdot)$ above. In that experiment the identification procedure was carried out for 1000 time steps using a random input signal uniformly distributed in the interval $[-1, 1]$, the spreading parameter was set to $\sigma = 0.315$ and 1000 pattern units were used to represent this plant (one for each random input signal used to identify the plant). Unfortunately in [Specht 1991] the

approximation error computed using the testing function (14) is not informed, and thus we reproduced that experiment 10 times using the original conditions (a standard GRNN network with $\sigma = 0.315$) and different random input signals in each replication. Figure 2(a) shows the best identification performed by GRNN. The averaged NRMS error computed over 10 replications was 0.053267, and 0.213 seconds were necessary to perform each replication.



(a) Identification using GRNN (b) Identification using IGMN

Figure 2. Identification of a nonlinear plant using 1000 training samples

To compare the performance of IGMN against the ANN models described above, we have repeated this experiment using the same conditions described above, i.e.:

- A series-parallel model for training and a parallel model for testing;
- The identification procedure was carried out for 1000 time steps using a random input signal uniformly distributed in the interval $[-1, 1]$;
- After learning the identified model was tested using inputs given by Equation 14;
- The whole experiment was repeated 10 times using different random input signals in each replication.

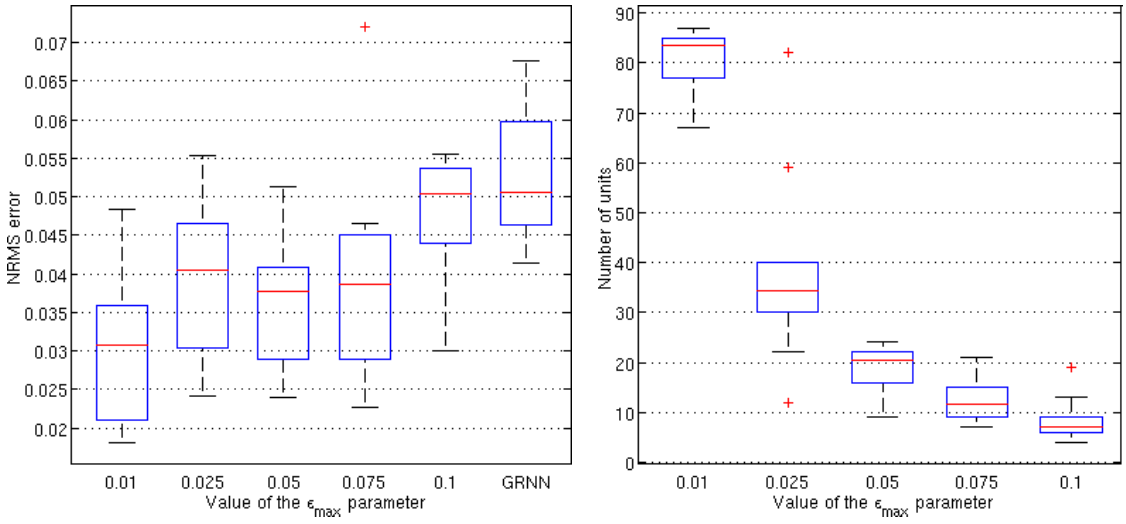
Figure 2(b) shows the identification performed by IGMN using default parameters, i.e.,$\varepsilon_{max} = 0.05$, $\delta = 0.01$ and $\Omega = 6$. Observing Figure 2(b) we can notice that the identification performed by IGMN is superior than that performed by GRNN (Figure 2(a)). The average NRMS error computed over 10 replications was 0.037, 19 neurons were added by IGMN during learning and 0.089 seconds were spent in each replication.

Table 1. Assessing the sensitivity of IGMN to the ε_{max} parameter

ε_{max}	0.1	0.075	0.050	0.025	0.010
Error	0.0474	0.0399	0.0370	0.0394	0.0304
M	8.30	12.60	19.10	38.10	80.40
Time	0.055s	0.089s	0.095s	0.125s	0.285s

To assess the sensitivity of IGMN to the ε_{max} parameter, this experiment was repeated using $\delta = 0.01$ and varying the value of the ε_{max} in the interval $[0.01, 0.1]$. Table 1 shows the results obtained in this experiment (averaged over 10 replications), and

Figures 3(a) and 3(b) show the boxplot graphs of the NRMS error and the number of units added during learning, respectively. It can be noticed in these figures that $\varepsilon_{max} = 0.05$ is a good choice, because it allows a good compromise between the model complexity (number of neurons) and the approximation level (NRMS error).



(a) NRMS error (b) Number of neurons

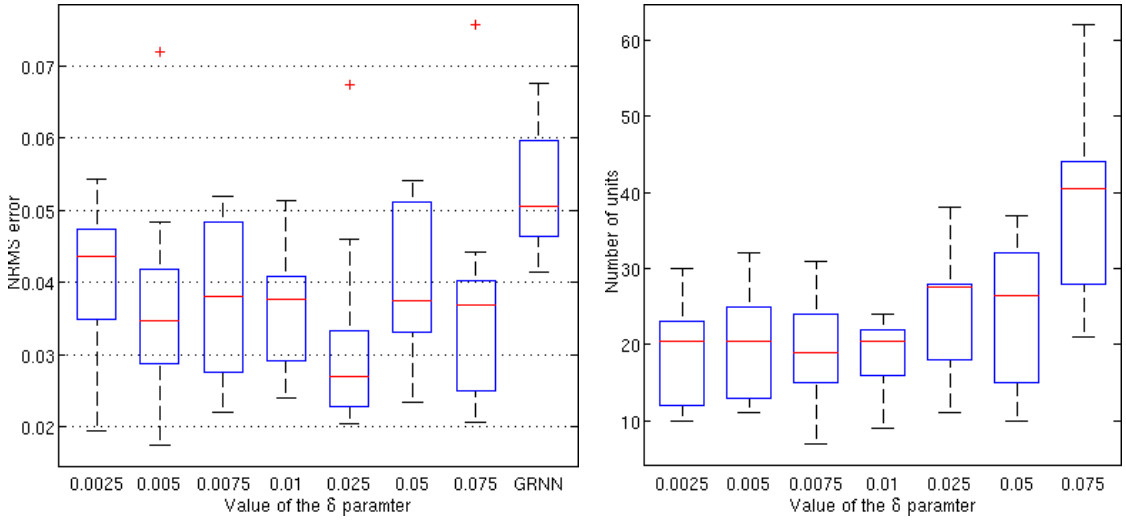
Figure 3. Assessing the sensitivity of IGMN to the ε_{max} parameter

The next experiment aims to assess the sensitivity of IGMN to δ using this plant, where ε_{max} was kept fixed at 0.05 and δ was varied in the interval $[0.0025, 0.075]$. Table 2 shows the average results obtained in this experiment, and Figures 4(a) and 4(b) show the boxplot graphs of the NRMS error and the number of units added during learning, respectively. It can be noticed that the IGMN performance is practically the same using any value of δ in the interval $[0.005, 0.05]$, although using $\delta = 0.025$ the NRMS error is slightly lower. Therefore we can notice that just extreme values of δ degrade the IGMN performance and/or require many Gaussian units.

Table 2. Assessing the sensitivity of IGMN to the δ parameter

δ	0.0025	0.005	0.0075	0.01	0.025	0.05	0.075
Error	0.042	0.037	0.037	0.037	0.032	0.039	0.037
M	18.60	20.20	19.20	19.10	24.70	24.40	38.80
Time	0.09s	0.09s	0.09s	0.09s	0.11s	0.10s	0.12s

In [Specht 1991] it is also pointed out that results similar to those presented in Figure 2(a) can be obtained using just 100 training samples. To validate this affirmation, we repeated this experiment using $N = 100$ training samples randomly chosen in the interval $[-1, 1]$. Using GRNN the NRMS error averaged over 10 replications was 0.090904, and using IGMN it was just 0.067239. Moreover, using IGMN the average number of neurons added during learning was just 3.6, whilst GRNN had used 100 pattern units in each replication.



(a) NRMS error (b) Number of neurons

Figure 4. Assessing the sensitivity of IGMN to the δ parameter

3.2. Approximating the "Mexican hat" function

In the previous experiments we have used a training data set composed by just two data features, a and b. The next experiment uses a three-dimensional data set composed by $N = 5000$ data samples given by:

$$b = \sin(a_1)/a_1 \sin(a_2)/a_2, \tag{15}$$

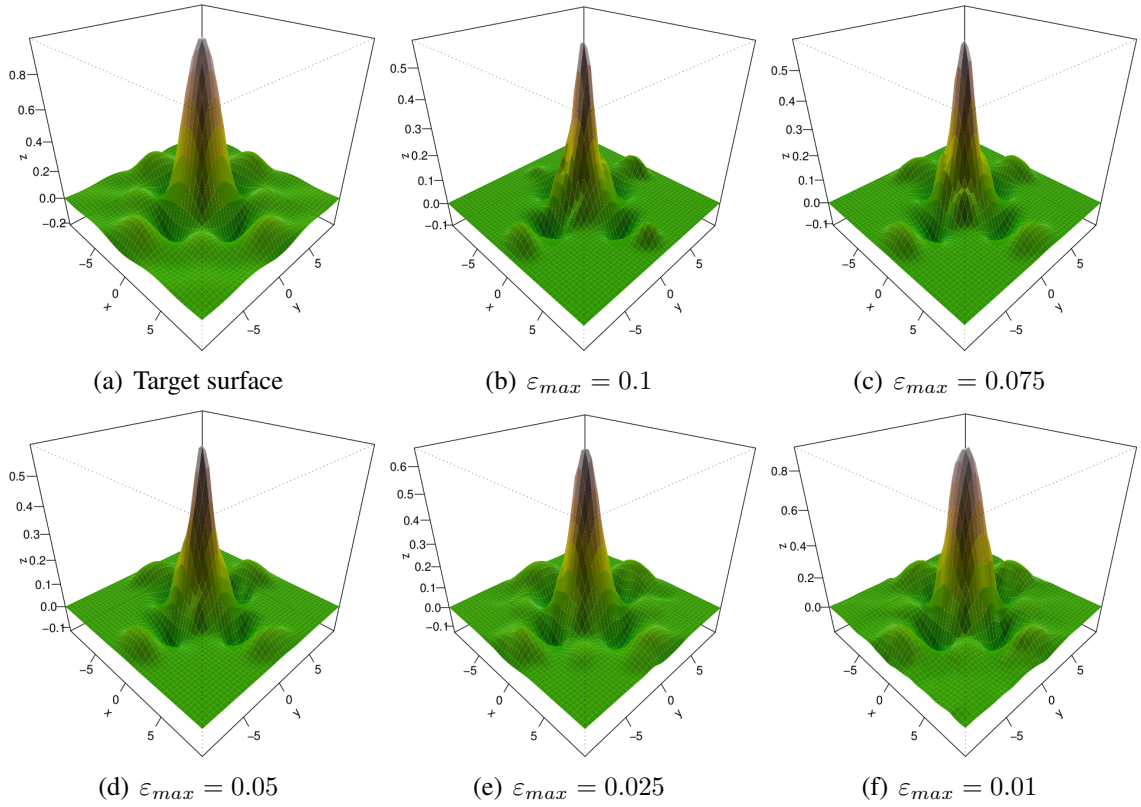
were a_1 and a_2 are randomly chosen in the interval $[-10, 10]$. Figure 5(a) shows the target surface, also known as "Mexican hat". The IGMN net used to learn this data set has two cortical regions: $\mathcal{N}^{\mathcal{A}}$, composed by two data features (a_1 and a_2), and $\mathcal{N}^{\mathcal{B}}$, composed by a single feature (b). This data set was randomly divided into two subsets, called training and testing data sets, each of them composed by 2500 data samples. Moreover, the training data was presented to IGMN in a random order (i.e., the data set was shuffled).

To assess the sensitivity of IGMN to ε_{max} over this data set we have repeated this experiment using different configurations of ε_{max}, and the results obtained in this experiment are shown in Table 3. The first row shows the configuration of the ε_{max} parameter. The following rows show, respectively, the NRMS error, number of neurons added during learning (M) and the learning time. The δ parameter was kept fixed at 0.01, but the results are practically the same using δ in the interval $0.005 \leq \delta \leq 0.05$.

Table 3. Approximating the "Mexican hat" function using IGMN

ε_{max}	0.1	0.075	0.05	0.025	0.01
NRMS	0.071302	0.067699	0.065824	0.051815	0.014288
M	19	25	37	83	184
Time	0.088s	0.104s	0.188s	0.656s	2.292s

Table 3 shows that the NRMS error is reduced as the ε_{max} parameter is decreased, whilst the number of neurons added during learning is increased. Using $\varepsilon_{max} = 0.01$,



(a) Target surface (b) $\varepsilon_{max} = 0.1$ (c) $\varepsilon_{max} = 0.075$

(d) $\varepsilon_{max} = 0.05$ (e) $\varepsilon_{max} = 0.025$ (f) $\varepsilon_{max} = 0.01$

Figure 5. Approximating the "Mexican hat" function using IGMN

for instance, the NRMS error is just 0.014288, but 184 neurons are added to each region during learning. If $\varepsilon_{max} \approx 0$ then the approximation error will be almost zero (remember that this data set is noise-free), but a huge number of neurons will be added during learning. Just for illustration purposes, Figures 5(b) to 5(f) show the surface approximated by IGMN using each ε_{max} setting. We can notice in these figures that using just 21 neurons (Figure 5(b)) the proposed model is able to learn the general structure of the target surface, and using 184 neurons (Figure 5(f)) the approximated surface is quite good.

Using a MLP neural network trained with the Levenberg-Marquardt (LM) algorithm and 20 hidden neurons, the average NRMS error computed over the testing data set was 0.015265 (we needed to repeat this experiment 10 times due to the random initialization of the synaptic weights in a MLP network), and the time required for learning was 20.633 seconds. However, using MLP we had to perform an exhaustive search to find out the best number of hidden neurons. Using GRNN with $\sigma = 1.0$, on the other hand, the NRMS error was 0.017395 and the learning time was 0.189 seconds. As said before, the number of neurons added by GRNN is equal to the number of pattern neurons, i.e., 2500 pattern neurons in this case.

4. Conclusion

This paper has presented IGMN [Heinen 2011], a new connectionist approach for incremental function approximation and prediction. IGMN is inspired on recent theories about the brain, specially the Memory-Prediction Framework (MPF) [Hawkins 2005] and the constructivist artificial intelligence [Drescher 1991], which endows it with some unique

features that are not present in other ANN models such as MLP and GRNN. Moreover, IGMN is based on strong statistical principles and asymptotically converges to the optimal regression surface as more training data arrive. The performed experiments demonstrated that IGMN is a very useful machine learning tool for incremental function approximation.

Acknowledgment

The authors acknowledge the support granted by CNPq.

References

Chaput, H. H. (2004). *The Constructivist Learning Architecture: A Model of Cognitive Development for Robust Autonomous Robots*. PhD thesis, Univ. Texas, Austin, TX.

Drescher, G. L. (1991). *Made-up Minds: A Constructivist Approach to Artificial Intelligence*. The MIT Press, Cambridge, MA.

Engel, P. M. and Heinen, M. R. (2010a). Concept formation using incremental Gaussian mixture models. In *Proc. 15th Iberoamerican Congr. Pattern Recognition (CIARP)*, LNCS, São Paulo, SP, Brazil. Springer-Verlag.

Engel, P. M. and Heinen, M. R. (2010b). Incremental learning of multivariate Gaussian mixture models. In *Proc. 20th Brazilian Symposium on AI (SBIA)*, volume 6404 of *LNCS*, pages 82–91, São Bernardo do Campo, SP, Brazil. Springer-Verlag.

Hawkins, J. (2005). *On Intelligence*. Owl Books, New York, NY.

Haykin, S. (2008). *Neural Networks and Learning Machines*. Prentice-Hall, Upper Saddle River, NJ, 3 edition.

Heinen, M. R. (2011). *A Connectionist Approach for Incremental Function Approximation and On-line Tasks*. Ph.D. thesis, Informatics Institute – Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre, RS, Brazil.

Heinen, M. R. and Engel, P. M. (2010a). An incremental probabilistic neural network for regression and reinforcement learning tasks. In *Proc. 20th Int. Conf. Artificial Neural Networks (ICANN 2010)*, LNCS, Thessaloniki, Greece. Springer-Verlag.

Heinen, M. R. and Engel, P. M. (2010b). IPNN: An incremental probabilistic neural network for function approximation and regression tasks. In *Proc. 11th Brazilian Neural Networks Symposium (SBRN)*, pages 39–44, São Bernardo do Campo, SP, Brazil.

Mitchell, T. (1997). *Machine Learning*. McGrall-Hill, New York.

Narendra, K. S. and Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Trans. Neural Networks*, 1:4–27.

Piaget, J. (1954). *The construction of Reality in the Child*. Basic Books, New York, NY.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). *Learning Internal Representations by Error Propagation*. The MIT Press, Cambridge, MA.

Specht, D. F. (1991). A general regression neural network. *IEEE Trans. Neural Networks*, 2(6):568–576.

Strassen, V. (1969). Gaussian elimination is not optimal. *Numerische Mathematik*, 13(3):354–356.

A Study on the Effects of Reputation-based Decision on the Dynamics of Public Goods Game with Punishment, Signaling and Gossiping Mechanisms

Mariana R. Mendoza[1], **Ana L. C. Bazzan**[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{mrmendoza,bazzan}@inf.ufrgs.br

Abstract. *In public goods game individuals contribute in favor of a common benefit. However this attracts free-riders, who profit the benefits generated by the group regardless their contribution decision. Although one would expect a cooperation collapse within a rational society, what is observed, in fact, is the emergence of cooperation. In the present work we not only address mechanisms such as punishment, signaling, and gossiping, but also add reputation-based decision-making to the process. We show that the dynamics of public goods game changes with the inclusion of the latter mechanism. Specifically, a decrease in the average wealth and contribution is observed because agents tend to be more selective. This holds a similarity with real-world situations: Human beings do act based on reputation about their partners.*

1. Introduction

In the last years the evolution of cooperation has been a widely studied topic in manifold disciplines, such as biology, economy and computer science, just to mention some. What intrigues researchers of this field is how cooperation is maintained in such a competitive world. Through both theoretical and experimental investigation, researchers aim to answer questions like why do people cooperate when they could just profit from the benefit generated by others. Assuming rational decision making, one would expect an uncooperative behavior from individuals. Yet, what we observe is the emergence of cooperation. In [Nowak 2006], many mechanisms involved with the evolution of cooperation are discussed. The author explains that competition takes place between all organisms, no matters their complexity, due to natural selection's principles, but that cooperation is needed in the construction of new levels of organization.

One well established paradigm for discussing altruistic behavior is the public goods game. In this game, individuals incur a cost to create a benefit for a group. Notwithstanding individuals' decision about contributing, they enjoy the benefits generated by the group. We can observe cooperation emergence in free-software development, blood donation, wikipedia, e-bay reviews about sellers, etc. This is an issue if, again, we suppose individuals act rationally. Free-riders would be attracted by the benefits, proliferate and, at some moment, cooperation would collapse. However, human societies have somehow managed to solve this and great interest and effort has been dedicated to the study of public goods game dilemma. Several mechanisms have been proposed in order to explain this phenomenon, like sanction, reputation and signaling. See [Henrich 2006, Nowak 2006] for an overview.

Punishment mechanism has been experimentally studied by [Gürek et al. 2006] with real subjects playing the public goods game. Players should select among two institutions to participate, one of which involving sanction. Authors show that individuals do prefer a sanctioning institution over a sanctioning free one, demonstrating the importance of punishment to the stabilization of cooperation. This mechanism was also addressed in [Bazzan and Dahmen 2010] to investigate evolution of cooperation among synthetic persons in a public goods game. Authors propose the creation of a more detailed scenario, where agents can punish, are exposed to bribery and can spread gossips about their acquaintances' contribution profiles.

The contribution profile is inspired by the so called green-beard model, proposed by Hamilton [Hamilton 1964] and named by Dawkins [Dawkins 1976]. In the latter, author suggests that if a gene arises that not only gives individuals a very distinguishable physical trait but also a tendency to be altruistic towards other individuals who carry the same trait, mutual altruism between these individuals could evolve. In the public goods game context, the green-beard effect is a metaphor to denote individuals who contribute when they see other green beards. [Bazzan and Dahmen 2010] modified the green-beard principle to include two kinds of beards: the blue and the red ones. These colors come from the seminal work of Nowak and May [Nowak and May 1992] on the spatial Iterated Prisoner's Dilemma, where blue is used to depict cooperation while red means defection.

In the present paper we are particularly interested in agent-based simulation of public goods game to investigate some open questions about the effect of different population sizes and punishment costs over the dynamics of the public goods game, raised in [Bazzan and Dahmen 2010]. Also, we extend the model proposed by the authors by including a reputation-based decision mechanism to bring gossip phenomenon closer to reality. In the work developed by [Bazzan and Dahmen 2010], agents believe in any rumors received about other agents' profile, regardless who the sender is. In our model, all received information is filtered according to the reputation assigned to other agents, such that only rumors sent by trustful source will be accepted.

Reputation refers to a collection of opinions that agents have about coexisting individuals in the society based in past experiences. When this observation is measured and quantified, the resulting rating may represent a valuable information to assist in decision making. In [Brandt et al. 2003], for instance, neighbors' reputation was used for deciding about whether to contribute with the common pool in spatial public goods game. Although green-beard effect somehow represents a reputation about an individual's behavior, in our work reputation has nothing to do with contribution decisions, instead, it is related to gossip spreading. The main motivation of aggregating a reputation mechanism to the model proposed in [Bazzan and Dahmen 2010] is that in real world humans do keep a register of their past experiences in their memories so that they can be later used to help them decide about what actions to take.

This paper is organized as follows. In Section 2 a brief review about the public goods game dilemma will be made. In the sequence, its modeling and dynamics according to [Bazzan and Dahmen 2010] will be explained and further details about the reputation mechanism will be given. Section 4 presents the settings and results of simulations run. Finally, in Section 5 we expose our concluding remarks.

2. Public Goods Game

In its original formulation, this game deals with public spending on libraries, community roads, etc. Individuals are offered to invest their money in a common pool, knowing that the overall invested amount is multiplied by an interest rate and equally divided by all participants, regardless their contributions. Therefore, the group as a whole does better when all individuals decide to contribute with the public pool. However, each individual faces the temptation to defect and to free-ride on the other individuals' contribution.

In theory, the Nash equilibrium in this game is simply zero contributions by all. Nevertheless, the Nash equilibrium is rarely seen in experiments; people do tend to add something into the pool. From a theoretical point of view, the reasons for this outcome are not fully understood but probably involve issues related to signalling, punishment and reputation. The two latter have been proven by [Brandt et al. 2003] as useful mechanisms to improve readiness of cooperation between agents. In [Gintis et al. 2001], the evolution of cooperation is explained as a natural consequence of the cooperation as an honest signal of the member's quality as a mate. Yet, the effects of sharing such information between agents have not been studied by the authors. Signalling mechanisms have been later explored by [Bazzan and Dahmen 2010]. Other former works have shown, for instance, that altruistic behavior prevails in the context of strong reciprocity [Gintis 2000] or between local interacting agents adopting an imitation mechanism [Bergstrom 2002].

3. Modeling

3.1. General Model

The basic behavior of our formulation of the public goods game is defined by a group of connected individuals or agents, usually disposed in a $n \times n$ grid. Each individual must decide at each time step whether or not to contribute a number of tokens to the common pool. If contributing, s/he must also decide about the value $q_i(t) \in [q_{min}, q_{max}]$ to be contributed. Overall contributions are multiplied by an interest rate $r > 0$ and are then equally divided between all participating agents, increasing their wealth. Therefore, the average contribution and the return per agent are given by Eqs. 1 and 2.

$$\bar{q}(t) = \frac{1}{n} \sum_{i=1}^{n} q_i(t) \tag{1}$$

$$R_i(t) = \bar{q}(t) \times r - q_i(t) \tag{2}$$

An accumulated wealth W is computed at each time step as shown in Eq 3. Also, each agent has an average wealth, which is calculated over time, according to Eq 4, where q_{max} is the number of tokens given to each agent at each time step.

$$W_i(t+1) = W_i(t) + R_i(t) + q_{max} \tag{3}$$

$$\bar{w_i} = \frac{W_i(t)}{\sum t} \tag{4}$$

In what follows we explain the model proposed in [Bazzan and Dahmen 2010] and already mentioned in Section 1. The dynamics of public goods game, repeated for several time steps, is summarized in three main steps:

1. select value $q_i(t) \in [q_{min}, q_{max}]$ to contribute;
2. randomly draw a free-rider from known agents list and decide whether and how much to punish; punished agents may bribe;
3. propagate gossip about other agents.

Agents are created with labels which indicate her/his contribution's profile: free-riders (FC) are in red, while high-contributors' (HC) identification is blue. The proportion of red beards in the environment is defined by parameter p_R. The idea of the green beard effect in this scenario is that individuals' profiles are visible for others agents and, most important, they influence their decision about contribution. Also, each agent keep for her/himself a list of observed blue beards (high-contributors, B_i) and red beards (free-riders, R_i). Free-riders never contribute to the common pool, i.e., they contribute $q_{min} = 0$. High-contributors decide how much to contribute depending on their knowledge about others players' profiles. They may contribute with any value $q_i(t) \in [q_{min_{HC}}, q_{max}]$, defined according to the following rules (where for instance, $|B_i|$ is the cardinality of set B_i):

- $q_i = q_{min_{HC}}$, if $|B_i| < |R_i|$;
- $q_i = q_{max}$, if $|B_i| > |R_i|$;
- $q_i = random(q_{min_{HC}}, q_{max})$, if $|B_i| = |R_i|$.

Return and wealth are then computed. After that, agents decide whether and how much they will punish. Any member of the red beards' list may be punished and the choice is random. Punishment mechanism occurs after all agents have contributed and it depends on the average wealth of a given individual and on the willingness to punish (ω). Agents will punish only if their average wealth ($\bar{w}_i$) is higher than the cost of punishing (c_{me}) multiplied by the factor ω. Punished agents may have their balance decreased by c_{you} while those who had applied the punishment may lose c_{me}. However, before punishment costs are applied, the punished agent may try to bribe her/his punisher as long as they are in close neighborhood, i.e. immediate neighbors.

Bribery is only accepted if its value is higher than the internal threshold of the agent who is practising punishment. The internal threshold is a private information computed as follows: agent i is initialized with a factor β_i whose value is drawn from a normal curve with mean 2 and deviation 1. This factor is multiplied by the agent's average wealth, determining the minimum bribe accepted by this agent. Also, agents are created with a factor λ_i, which is multiplied by its average wealth in order to specify the maximum bribe paid by this agent when s/he wishes to tempt bribe. The wealth of players i and j, involved in punishment and bribery processes, are then updated according to the following rules. If bribery did not happened or was not accepted, accumulated wealths of i and j are updated to $W_i = W_i - c_{me}$ and $W_j = W_j - c_{you}$ respectively. However, if bribery was accepted, player j is transferred from red beards' list to blue beards' list of player i and accumulated wealths are updated such that $W_i = W_i + \beta_j \bar{w}_j$ and $W_j = W_j - \lambda_i \bar{w}_i$.

In the last stage of the algorithm described in [Bazzan and Dahmen 2010], agents may spread rumors about their blue and red beards' lists with probability p_g. When this condition is verified, agent i sends a message to each one of its k neighbors containing this information. When an agent receives a message, s/he filters out all rumors involving the close neighbors. This avoids that this agent is led to believe in erroneous information about her/his own neighbors. This is a reasonable assumption since the existing relationship allows them to check directly the real beard color of its k-th neighbors. At the end of

each round, agents with negative average wealth are eliminated. The number of steps for which this dynamics will be repeated is given by t_{max}.

3.2. Reputation Mechanism

As previously mentioned, in this work we extend the model described in the previous section, aggregating a reputation mechanism between agents. Reputation refers to a collection of opinions that agents have about coexisting individuals in the society. When this observation is measured and quantified, the resulting rating may represent a valuable information to assist in decision making. In [Mui et al. 2003], a study about reputation across several subjects was discussed and authors concluded that reputation is a multiple parts notion and a context-dependent quantity. In electronic commerce environment, for instance, reputation has been applied as a modeling framework of sellers' reliability aiming to encourage transactions. In social contexts, such as the iterated prisoners' dilemma game, reputation was already used to explain cooperation between selfish individuals: agents decide whether to cooperate or defect based on the opponent's reputation, which is inferred from the ratio of cooperation over defection.

The main goal of including a reputation mechanism in the public goods game's dynamics described in Section 3.1 is to make the model even more detailed and allow agents to decide whether or not they will believe in the gossip spread by their neighbors based on the reputation associated to them. As rumors may include erroneous information about red beards who have previously bribed other agents to lie about their contribution profile, the fact that an agent will only believe in information originating from a trustful source may influence in the overall cooperation and wealth. By adding a reputation verification to the gossip mechanism, all information coming from high-reputation neighbors will be accepted, while those sent by low-reputation neighbors will be rejected. This notion of low and high reputation is variable between agents, providing more dynamism to the environment.

At the beginning of the simulation, each agent sets the reputation of her/his neighbors to 0. Simultaneously, a factor α_i is sampled for each agent from a normal distribution with mean 1 and standard deviation 1. This factor represents the agents' demanding profile in respect to their neighbors' reputation, i.e. the minimum reputation required to accept any propagated rumors. Consequently, on average individuals will demand a minimum reputation equal to 1. Also, two extreme situations will happen, characterizing agents with antagonistic profiles: some agents will be very tolerant with their neighbors ($\alpha_i \leq 0$), while others will be very rigorous about their neighbors' reputation ($\alpha_i \geq 2$). Therefore, the variation in agents' demanding profiles is associated to the α_i factor.

When rumors are received, the agent verifies if the reputation associated to the neighbor who sent them is greater than her/his internal threshold. If not, this agent will ignore all the information received. However, if the neighbor has a higher reputation, the agent will not only believe in the rumors, but also check if they do not conflict with her/his own knowledge, i.e., if agents labelled as free-riders are in fact in her/his red beards' list and if the high-contributors spread by the neighbors are in fact in her/his blue beards' list. This process is done only for gossip about agents who are already member of any of the agent's lists; unknown players will be directly added in the corresponding list according to the received information.

The reputation associated to a gossiper neighbor is a dynamic value, which is updated according to the information previously sent by her/him. This is done such that neighbors who usually propagate correct information will have higher reputation and therefore, higher chances to spread her/his gossips. If any sent gossip goes against the receiver's knowledge about other agents' contribution profile, the responsible neighbor will have her/his respective reputation decreased by a constant rate δ_{dec}. Otherwise, the reputation associated to this neighbor will be increased by δ_{inc}, as well as the probability of acceptance of her/his gossips in subsequent time steps.

3.3. Tools

The public goods game was modeled in Netlogo[1], a commonly used tool for agent-base simulations. Each agent has two types of attributes: local and global. Global attributes are shared by all agents and remain constant over time, while local attributes may vary between agents and during the simulation. Agents' attributes are summarized in Table 1, where G denotes global attributes and L stands for local attributes. The list of scenario's attributes and their possible values according to model's definitions are shown in Table 2. The environment is set up at the beginning of each simulation according to the data retrieved from those variables.

Table 1. Agents' attributes.

Attribute	Description	Type
c_{me}	cost to punish	G
c_{you}	cost if punished	G
ω	willingness to punish	G
w_0	initial wealth	G
δ_{dec}	decrease rate of reputation	G
δ_{inc}	increase rate of reputation	G
q_{min}	minimum contribution for free-riders	G
$q_{min_{HC}}$	minimum contribution for HC	G
q_{max}	maximum contribution	G
T_i	tag of agent *i*	L
N_i	neighbors' list of *i*	L
B_i	blue beards' list of *i*	L
R_i	red beards' list of *i*	L
q_i	contribution of *i*	L
W_i	accumulated wealth of *i*	L
R_i	return received by *i*	L
$\bar{w}_i$	average wealth of *i* over time	L
β_i	percentage of $\bar{w}_i$ offered as bribe	L
λ_i	factor to compute minimum accepted bribe	L
α_i	minimum reputation required by *i*	L
γ_i	reputation associated by *i* to its neighbors	L

4. Simulations and Results

4.1. Settings

Different tests were made with the model created in Netlogo. The experiments' goal is to investigate the impact of the population size, the punishment costs and the in-

[1]http://ccl.northwestern.edu/NetLogo/

clusion of a reputation mechanism over the basic dynamic exposed in Section 3.1. The main model's parameters were configured exactly as the variables' values used in [Bazzan and Dahmen 2010], so that results comparison is possible: $r = 1.2$, $c_{me} = 5$, $c_{you} = 20$, $\omega = 3$, $q_{min} = 0$, $q_{min_{HC}} = 10$, $q_{max} = 20$ and $p_R = \{25, 50, 75\}$. The scenario is originally composed by $n = 225$ agents disposed in a 15×15 grid. For experiments involving the test of different population sizes or punishment costs, we varied parameters n and the rate c_{me}/c_{you}, respectively, while the tests of the reputation mechanism were performed maintaining the original variables' values. Results' analysis is made based on the average contribution (Eq. 1) and the average accumulated wealth (Eq. 4) collected at the last time step over 30 runs of the same simulation setting. We remark that contribution is an instantaneous score.

Table 2. Model's attributes and their possible values.

Attribute	Description	Possible values
n	population size	$\{21, 49, 81, 121, 225\}$
p_R	percentage of free-riders	$\{25, 50, 75\}$
r	interest rate	$[1.0 : 0.1 : 1.5]$
p_g	probability of gossip	$[10 : 10 : 100]$
t_{max}	number of game steps	$[50 : 50 : 300]$
n_s	number of simulations to perform	$[1 : 1 : 30]$
$p?$	activate/deactivate punishment	$\{true, false\}$
$g?$	activate/deactivate gossip	$\{true, false\}$
$r?$	activate/deactivate reputation	$\{true, false\}$

4.2. Experiments

The first experiments derive from [Bazzan and Dahmen 2010] and consist of a variation of the free-riders' percentage, combining it with the activation of punishment and bribery mechanisms and with different gossip probabilities. The purpose of these experiments was to test the model and verify if the observed behavior matches with the one described at [Bazzan and Dahmen 2010][2]. Results are shown in Table 3. As expected, the greater the percentage of free-riders in the environment, the lower the average wealth and contribution of players, since the decision of whether and how much to contribute is directly related to the number of red beards an agent knows. Besides, the activation of punishment, and hence bribery, and gossip mechanisms seems to increase the average contribution due to more elimination of free-riders.

In the sequence, we repeated the experiments of [Bazzan and Dahmen 2010] for different population sizes, aiming to examine if the system dynamics is affected by this parameter. We tested $n = \{49, 81, 121\}$ for a scenario with and without punishment mechanism and compared it to the results published by the authors for a scenario with 225 agents. The simulations' results are shown in Figure 1. It is possible to observe that the mean for both average contribution and average accumulated wealth haven't suffered significant variation, meaning that this parameter seems not to interfere in the basic dynamics of the public goods game. The most visible changes have happened when punishment mechanism is activated (right column), specially for a scenario with 50% of free-riders.

[2]The original implementation was not made in Netlogo, but using SeSAm.

Table 3. Average wealth and average contribution in the last time step over 30 simulation runs obtained with our model for scenarios proposed in [Bazzan and Dahmen 2010].

			$< \bar{w}_i >$		$< q_i >$	
p_R	$p?$	p_g	avg.	std.	avg.	std.
25%	false	0	22.97	0.16	13.86	0.86
	true	0	21.97	0.84	18.86	0.46
	true	20	24.04	0.04	19.98	0.03
	true	50	24.02	0.03	20.00	0.00
	true	70	24.02	0.03	20.00	0.00
50%	false	0	21.62	0.16	7.13	0.85
	true	0	17.30	0.94	13.50	1.67
	true	20	23.42	0.26	19.93	0.09
	true	50	23.62	0.13	19.99	0.00
	true	70	23.66	0.10	20.00	0.00
75%	false	0	20.77	0.10	2.89	0.50
	true	0	16.34	0.49	6.59	0.76
	true	20	13.70	0.56	8.08	2.25
	true	50	14.20	0.54	7.82	2.91
	true	70	14.13	0.45	8.76	2.75

Later, we ran simulations using distinct values of punishment costs, varying the rate between c_{me}/c_{you}. We tested ratios 1:1, 2:1, 3:1 e 5:1. Ratio 4:1 is the original one, used in [Bazzan and Dahmen 2010], thus it was not included in this set of simulations. The results, depicted in Figure 2, show that the greater the difference between c_{me} and c_{you}, the higher the contribution level and, therefore, the average wealth of agents. Thus, this parameter has a direct effect over the game's dynamics. This behavior is probably due to the fact that a higher punishment cost causes more free-riders to be eliminated, which affects the decision of other agents about how much to contribute. At this point it is worth to remind that the decision about whether and how much to punish is directly related to the number of free-riders known by each agent. The effect over the mean average wealth was more remarkable for ratio 5:1, in which $c_{me} = 5$ and $c_{you} = 25$. In this case, a significant raise in individuals' wealth can be observed for all percentages of free-riders. Other proportions have resulted in similar marks for scenarios with 50% and 70% of free-riders, and a more expressive variation is observed when the rate of free-riders is 25%.

The last step performed in this study consists of the inclusion of the reputation mechanism, described in Section 3, by which agents will consider the rumors received from their neighbors depending on the reputation assigned to them. Reputations are dynamically modified based on the veracity of information received. When an agent receives rumors from her/his neighbor that goes against her/his own knowledge, this neighbor will have the reputation decreased by factor δ_{dec}. Otherwise, if the information matches or if it refers to a new knowledge, the neighbor's reputation will be increased by factor δ_{inc}. The motivation is that individuals will become more selective about the rumors spread by their neighbors, ignoring the erroneous information usually sent by those with low reputation.

First, we tested the case where $\delta_{dec} = \delta_{inc} = 0.2$ and observed how the inclusion of a reputation mechanism affects the basic dynamics of the public goods game. Figure 3 shows a comparison between the average wealth and contribution for a scenario with

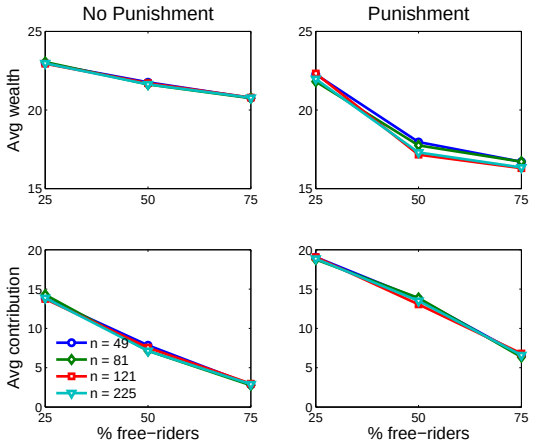


Figure 1. Average contribution and accumulated wealth of simulations with different population size for scenarios with and without punishment mechanism. The graphs' legends are shown in the bottom left subfigure.

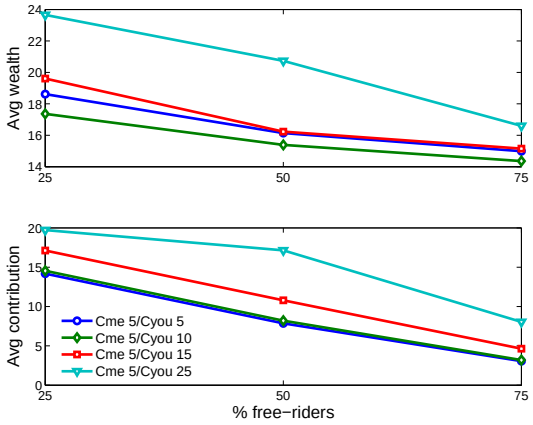


Figure 2. Average contribution and wealth for different ratios c_{me}/c_{you}.

and without this mechanism. The graphs' legends are shown in the bottom left subfigure. To help in results' interpretation, we also plot the average number of agents eliminated. As one can notice, for most of used settings, the activation of this mechanism caused a reduction in the average wealth and contribution. The lower average contribution is closely related to the reduction in the amount of eliminated individuals (Figure 3-c), who generally have a free-rider profile.

A larger concentration of free-riders in the scenario causes contribution levels to decrease. This fact alone justifies the small variation in wealth and contribution values in the scenario with 25% of free-riders, in which the elimination of non-contributing agents has little impact on the dynamics. However, an unexpected small increase in the average wealth may be observed when the percentage of free-riders is equal to 75%. We suspect that the reason of this behavior is related to the higher wealth concentration between free-riders. Since the only cost they have is the punishment cost, when not punished their accumulated wealth increases substantially. When more free-riders agents remain in the scenario, which is the case as players elimination rate decreased, total accumulated wealth is higher and, consequently, also the average wealth.

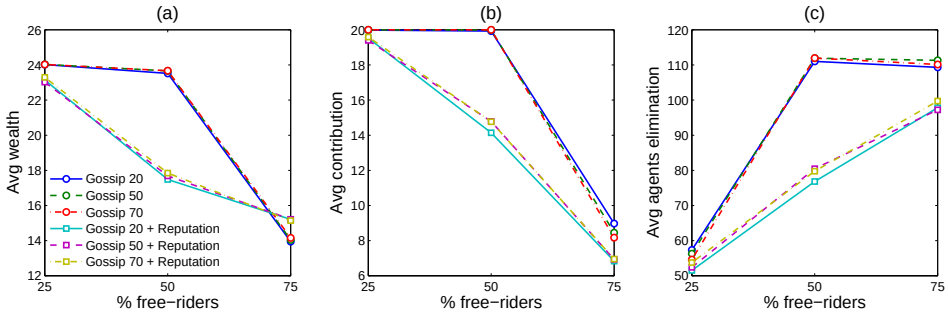


Figure 3. (a) average wealth, (b) average contribution and (c) average number of eliminated players between scenarios with and without reputation mechanism. The graphs' legends are shown in the left subfigure.

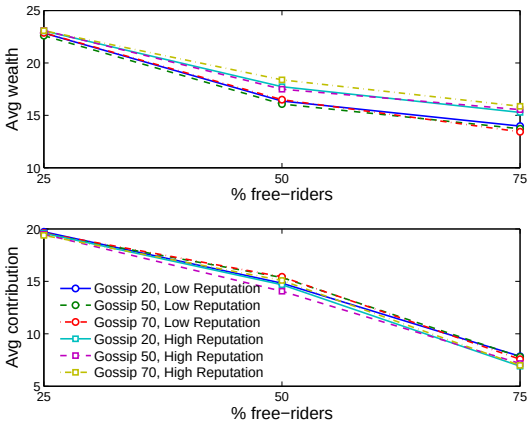
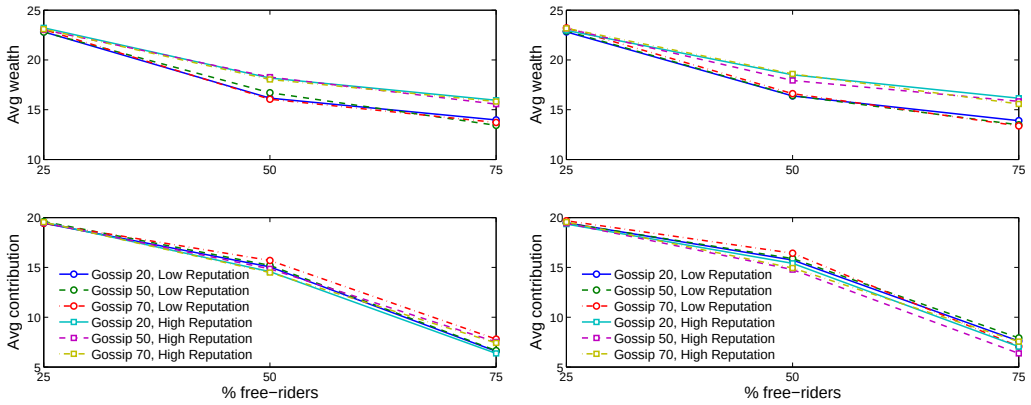


Figure 4. Average contribution and wealth for group of players with lowest and highest demand about their neighbors' reputation.

To better understand the consequences of including a reputation mechanism in a scenario like the public goods game, we run some simulations following more closely agents who have antagonistic profiles. We selected the group of agents who require low reputation from their neighbors, i.e. agents who have $-1.0 \leq \alpha_i \leq 0$, whom we call tolerant agents, and the group of most demanding players, i.e. those who have $2.0 \leq \alpha_i \leq 3.0$, to whom we refer to as rigorous agents. For these, we plot the average contribution and wealth in Figure 4. In this figure one can observe that the difference is modest but yet, the group of rigorous individuals tend to contribute less as they rarely accept information about briberies propagated by their neighbors. Therefore, their red beards' list is more extensive than their blue beards' list. Thus, according to the relationships in Eqs. 2 and 3, their average accumulated wealth is higher on average.

Two other variants were tested: i) the case where $\delta_{dec} = 0.2$ and $\delta_{inc} = 0.3$; and ii) $\delta_{dec} = 0.3$ and $\delta_{inc} = 0.2$. These values simulate the cases in which agents attach distinct relevance levels to mistakes and hits contained in received rumors. Results are depicted respectively in Figures 5-a and 5-b. No significant variation in the behavior of curves for average accumulated wealth and average contribution plotted in both figures is observed. Therefore, one concludes that the fact that an individual attaches more importance to the mistakes ($\delta_{dec} > \delta_{inc}$) or hits ($\delta_{inc} > \delta_{dec}$) made by her/his neighbors seems not influence the results qualitatively. In quantitative terms, the changes are minor and occur mainly in

scenarios with 50% of free-riders.



(a) (b)

Figure 5. Average contribution and wealth for agents with lowest and highest demand about their neighbors' reputation when a) $\delta_{dec} < \delta_{inc}$ and b) $\delta_{dec} > \delta_{inc}$.

5. Conclusion

In this work we have investigated the effects of parameters such as population size and punishment costs on the dynamics of public goods game using an agent-based simulation. Also, we proposed a more detailed scenario for gossip spreading based in the concept of reputation. Our main concluding remarks may be summarized as follows. The basic dynamics, in which the more red beards are seen the lesser the contribution and wealth of agents, is insensible to variations in the size of population. We tested values $n = \{49, 81, 121, 225\}$ and no significant change in average wealth and contribution was observed. Regarding the punishment costs, we modified the ratio 4:1 applied in [Bazzan and Dahmen 2010], where $c_{me} = 5$ and $c_{you} = 20$, using ratios 1:1, 2:1, 3:1 e 5:1. We observed that the greater the difference between c_{me} and c_{you}, the higher the contribution level and, therefore, the average wealth of agents. This effect is related to the fact that a higher punishment cost causes more free-riders to be eliminated, which affects the decision of other agents about how much to contribute.

The reputation mechanism caused a decrease in the average wealth and contribution. The lower average contribution is closely related to the reduction in the amount of eliminated agents, who generally have a free-rider profile. Also, we observed that the rigorous agents rarely accept information about briberies and, therefore, they tend to contribute less because their red beards' list is more extensive than their blue beards' list. The relevance of such simulations is the studying more realistic behaviors. In fact, our results hold a similarity with real-world situations: Human beings do select in which partners to believe, according to a reputation degree assigned to them. This reputation reflects somehow the trustful level of agents based on past experiences. However it causes a loss of utility (here contribution level). For instance, someone may believe in a partner even after this partner has shared wrong information a couple of times. However, at some point, a bad reputation will be attached to the sender and no more gossip spread by her/him will be accepted in the future.

Once a bad reputation is earned, agents will no longer have the opportunity to

spread their gossips to those neighbors whose threshold is higher than their own reputation. This is surely not representative of real life. Therefore, it would be interesting to refine the model so that it becomes even more close to reality. The first direction in which it could be modified is to allow agents to decide, with a given probability, if they wish to give a second chance for neighbors with low reputation. This may be referred as the ”forgiveness probability”. Also, we judge interesting to test a different and nonlinear strategy of reputation update, which is more characteristic of the real world.

6. Acknowledgments

Authors would like to thank the anonymous reviewers for their valuable suggestions of paper improvements. Both authors are supported by CNPq.

References

Bazzan, A. L. C. and Dahmen, S. R. (2010). Bribe and punishment: Effects of signaling, gossiping, and bribery in public goods games. *Advances in Complex Systems*, 13(6):755–771.

Bergstrom, T. (2002). Evolution of social behavior: Individual and group selection. *Journal of Economic Perspectives*, 16(2):67–78.

Brandt, H., Hauert, C., and Sigmund, K. (2003). Punishment and reputation in spatial public goods games. *Proc Biol Sci*, 270(1519):1099–1104.

Dawkins, R. (1976). *The Selfish Gene*. Oxford University Press, New York. 224 pp.

Gintis, H. (2000). Strong reciprocity and human sociality. *Journal of Theoretical Biology*, 206(2):169–179.

Gintis, H., Smith, E. A. L. D. E. N., and Bowles, S. (2001). Costly signaling and cooperation. *Journal of Theoretical Biology*, 213(1):103–119.

Gürek, O., Irlenbusch, B., and Rockenbach, B. (2006). The competitive advantage of sanctioning institutions. *Science*, 312(5770):108–111.

Hamilton, W. D. (1964). The genetic evolution of social behavior ii. *J Theoretical Biology*, 7(1):17–52.

Henrich, J. (2006). Cooperation, punishment, and the evolution of human institutions. *Science*, 312(5770):60–61.

Mui, L., Halberstadt, A., and Mohtashemi, M. (2003). Evaluating reputation in multiagents systems. In *Proc. of the AAMAS*, number 2631 in LNAI, pages 123–137.

Nowak, M. A. (2006). Five rules for the evolution of cooperation. *Science*, 314(5805):1560–1563.

Nowak, M. A. and May, R. M. (1992). Evolutionary games and spatial chaos. *Nature*, 359:826–829.

An Architecture Based on M5P Algorithm for Multiagent Systems

Alex Machado[1], David Carvalho[1], Esteban Clua[1], Cristiano G. Duarte[2], Marcos V. Montanari[2], Willian M. P. Reis[2]

[1]Instituto de Computação - Universidade Federal Fluminense - Niterói, RJ - Brazil

[2]Departamento de Computação - Instituto Federal do Sudeste de Minas Geral - Rio Pomba, MG - Brazil

alexcataguases@hotmail.com, dbcdavid.clk@gmail.com, esteban@ic.uff.br, crgdcomp@gmail.com, marcosviniciusmontanari@gmail.com, willianmpreis@yahoo.com.br

Abstract—Character-based interactive storytelling, life simulation and game difficulty dynamic balancing are examples of topics that need to deal with autonomous agent evolution. Although the commercial appeal of such kind of feature, the research of new behaviors emergence in virtual societies is restricted to intelligent agents that do not learn. This work proposes a novel architecture of a multiagent system based on the M5P algorithm for generation of emergent behaviors in complex virtual worlds. Based on our obtained results, we describe the advantages of the reinforced learning based on numerical classifiers when compared with traditional interaction adaptations.

1. Introduction

In a previous work [Machado et al 2010], we defined an architecture for implementing strategy games with the possibility of occurrence of emergent behaviors. Through experiments, we demonstrated the advantages of using the M5P algorithm over others qualitative attributes classifiers. In this paper we extend the experiments for a multiagent based environment.

In a similar way, [Bakkes et al 2009] define a game strategy as a configuration of parameters that determine strategic behavior (such as how many aircraft units should be constructed, maximum air group size or maximum number of storage buildings). In this paper, domain knowledge is required in order to adapt the characters to game circumstances automatically by the game AI, and is exploited immediately to evoke effective behavior in a controlled manner in online matches. Experiments in a strategy game shown that proposed case-based adaptive game AI provides a strong basis for effectively adapting game AI. Our approach differs from this, since defines a simplest architecture in order to achieve the same goals and makes an evaluation about the interaction between the multiagents systems characters. Our method also maintains the rapid and reliable adaptation of game AI characteristics.

We maintain the emergent property achieved in [Machado et al 2010], enhancing knowledge properties presented by Crocomo et al. [Crocomo & Simões 2008] and [Hong & Cho 2005], where it is shown that machine learning processes in games that rules NPCs (non player character) behaviors (in this case, evolutionary algorithms) are

able to emerge more complex strategies from simple behaviors.

2. Related Work

According to [Manslow 2002], the usage of adaptive learning in games is a trend and must be one of the most important improvements to the field. This topic is important not only for the improvement of AI behaviors, but specially for generating more robust and smart gameplay, according to the player characteristics. Within this field, our work is more focused on reinforcement and supervised learning approaches.

Our approach uses the supervised learning related to the examples input requirements and induction rules. We also use reinforcement learning, since the presence of rewards is a key factor for adapting the system for new cases.

In [Andrade et al 2005] and [Spronck et al 2004] it is shown efficient implementation of reinforcement learning as a mechanism of player adaption. The first work introduces the concept of a challenge function, which estimates the current player´s level and changes on the action selection mechanism for fighting games. The second work, presents rules that can be applied for each opponent type of the game.

This paper proposes the creation of a novel architecture using adaptive intelligent agents in a game, with dynamic parameters configuration based on the M5P algorithm.

AI-based storytelling is the mechanisms for automatic story generation, which can be based on autonomous artificial actors' or on explicit plot representations [Cavazza et al 2002]. In the first the story diversity emerges from dynamic interaction between characters.

Although we present in this paper a novel adaptive game balancing, we also discuss the possibility of the usage of the technique for a narrative mediation approach in order to achieve more freedom and coherence for the plot generation guidelines, as presented in [Riedl 2003].

Social AI in games study is important because in a reasonably deep virtual world, there are always "boring" roles, that typical players are not going to feel pleasure to control, such as quest-givers, merchants, henchmen and thugs [Witze el al 2002]. A detailed study about life simulation and interaction can be found in [Pallay 2009]. The virtual world chosen for this study was Second Life [L.Lab 2011].

In [Goertzel et al 2008] the authors proposed a general approach for teaching behaviors to AI-controlled artificial animals embodied in Second Life [L.Lab 2011] using imitative-reinforcement-corrective learning. It is shown the possibility of teaching an agent (animal) to generate new behaviors with gestures.

We proposed use machine learn techniques for a social environment instead for just one autonomous agent.

3. Advantages of M5P Learning

M5P [Quinlan 1992] is a reconstruction of Quinlan's M5 algorithm for inducing trees of regression models. M5P combines a conventional decision tree with the possibility of linear regression functions at the nodes.

In previous work [Machado et al 2010], we experimentally demonstrate the advantages of using that algorithm (M5P) over decision tree learner (C4.5) and probabilistic learn method (Nayve Bayes). This behavior is due to its numerical classification, which guarantees the least loss of information (so accurately) for the next generations (stages). While its competitors just map the set of attribute values to a categorical target value.

When we build an architecture so that the M5P algorithm stays responsible to on-line generate the evaluation function, we create an agent capable of adapting itself to dynamic and inaccessible ambient. In this case, this possible architecture takes advantage in comparison with traditional methods like the genetic algorithm, considering that although the latter does not require knowledge derived from the problem, it requires a pre-defined evaluation method of the result [Goldberg 1989].

According to [Etemad-Shahidi & Mahjoobi 2009] and [Bhattacharya & Solomatine 2005], neural networks (NNs) are not as transparent as semi-empirical regression-based models. For this reason, the main advantage of model trees is that, compared to NNs, they represent understandable rules. In addition, NNs approach needs to find network parameters such as number of hidden layers and neurons by trial and error, which is time consuming.

In [Bhattacharya & Solomatine 2005] the predictive accuracy of the M5P model was observed to be very high and with similar results than the NN model built with the same data.

4. Development Framework

4.1. Approach

Since the environment where the character is inserted is complex (inaccessible, stochastic, sequential and dynamic), this work discusses the intelligent agent architecture with individual learning. This not only happens in situations where each agent pursues its own learning objectives, but also when it's learning can be affected by other agents.

Altogether these agents form a coordinated multi-agent system [Malone & Crowston 2004], where the coordination between the agents are defined by the dependency relations of shared resources. A multi-agent system is composed by two or more agents, in society form, with potential for self-sufficiency and emergency behavior.

4.2. System Architecture Abstraction

M5P-Reinforcement Learn Architecture (Figure 1) uses the M5P algorithm, which is capable of generating rules dynamically, with reinforcement learning, to update the database instances. Therefore this architecture has the characteristics of evolutionary classifier systems [Booker, Goldberg and Holland 1989] that are designed to absorb new information and devise continuously competing sets of hypotheses (expressed as rules) without disturbing already significantly capabilities acquired. Its steps are:

1. Test of random cases at the environment: as happens in an on-line learning system, the agent starts with no previous knowledge, so at the risk of making mistakes. It explores the environment several times, with its attributes and/or actions

defined in a varied form. The interaction mechanisms with the environment and the other agents are sensors and actuators.

2. Incorporation to the instances base: in this step each case previously lived is added to the instances base, which functions as a "memory" of the agent that stores its experiences with their results. These results serve as reward of the architecture, characteristic of the reinforcement learning.

3. Induction of learning rule: using the M5P algorithm, the trees and rules are induced from the base of instances.

4. Generation of candidate instances, classification and selection of the best: this is a module diversifier so that the generated solution does not get stuck in local optima. Uses the elitist criterion to define the best solution of a new set randomly generated and ranked (classified) by the learning rule.

5. Test of this case in the environment: this new generated instance is released in the environment for testing and evaluation. At this moment, the agent, with the same interaction mechanisms, explores the environment creating a new set of experiences for a later restart of the evolution system process.

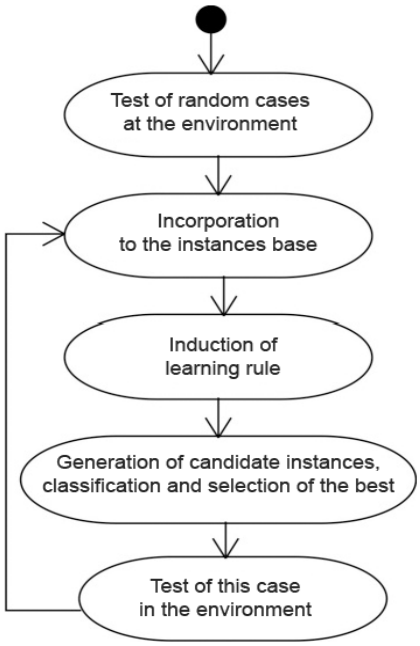


Figure 1 - M5P-Reinforcement Learn Architecture

Although this work evaluates a multi-agent system, this architecture specifies the learning process of only one agent. In the next session, this architecture is redesigned, applied and evaluated in a multi-agent system.

5. Case Study

5.1. Virtual environment plot

The developed application, called “New World” (Figure 2), created for testing purposes of M5P-Reinforcement Architecture, fits in the context of life simulation. Its 3D environment reconstructs aspects of a real city, with a resting place and options of work

and leisure. Each NPC is considered foreigner, initially. As such, it must experience/explore the city to be able to define the routine that best suits it later.



Figure 2 – "New World", the life simulation experiment.

The agents have two types of intelligent behavior: simple reactive and with learning aspects. The first refers to its instincts, such as moving, mapping a route between origin and destination, avoiding obstacles, stopping to chat when it meets with another agent (with no exchange of information). The second relates to decide, based on experience, which is the best routine for its day.

The options the city offers to them, here called "waypoints", may be one of the types: work (with payment), leisure (that, although has financial costs, gives satisfaction) or inn/hotel (to spend the night with no expenses and no satisfaction). A routine consists of the waypoints that the agent decided to visit during the working time of the day.

The day has 24 hours. Around 23:00 the agents must go rest until 7:00. So they have an average of 16 working hours to assemble an agenda. Thus, given the need for money (which can be obtained at the waypoints that offer work) to achieve satisfaction (at the waypoints that provide entertainment), the NPC needs to explore the environment to learn how to optimize its routine during the day.

Among the complications of this virtual environment, in which the programmer or a player's character can influence, we can cite:

- Assuming that the agent is employed, if another company discloses the increased value of the base salary, the agent can end by pleading for this job.
- Assuming that a new leisure option appears in the city, the agent may want to experience.
- In a particular route can be placed a "thief", in this case its diary routine may be compromised, so the agent will need to define a new one.

5.2. Developed application

The application New World was developed on the M5P-Reinforcement Architecture, the suggested technology in [Machado et al 2010] and the exchange of information made through threads for the multi-agent system. The experiments in this environment were realized through alterations in the waypoints parameters.

As [Machado et al 2010], to develop an intelligent agent architecture with learning applied to the generation of new strategies for a game, it was used a framework

integrating Unity 3D (game engine), Microsoft Visual Studio (server development IDE), C#, Weka and IKVM.

Despite all the advantages of using Mono, the C# compiler version included in the Unity3D package, it had incompatibilities with the Weka DLL interpreted by IKVM in the experiments we ran. More specifically, the problem was related to the garbage collector. To circumvent this problem, we opted for integrating Unity3D with a server module developed using Visual Studio and making the connections through sockets.

This client-server system can be seen as implementing a kind of reinforcement learning strategy because it generates an output that has never been presented to it, which is rewarded with a score (which represents the knowledge generated by the classifiers).

A multithreading system was developed with a communication protocol between the client and the server to guarantee the generation of multiple agents in the environment.

5.3. Definition of the Instances Base

For the proposed application, the instance base consists of the waypoints that the agent can visit. It is defined dynamically and its attributes are numerical as specified in the M5P algorithm. They are named as follows:

```
@relation RotinaAgente        (1)
@attribute wp1 numeric
@attribute wp2 numeric
…
@attribute wpN numeric
@attribute satisfaction numeric
```

In this first version, the implemented waypoints are (Table 1):

Table 1. Waypoints and its characteristics

Attr.	Waypoint	Salary	Cost	Satisfaction	Time
wp1	Office	8	0	0	5
wp2	Police station	5	0	0	4
wp3	Hospital	3	0	0	3
wp4	Industry	2	0	0	2
wp5	Pizza Store	0	4	5	4
wp6	Boite	0	3	3	3
wp7	Circus	0	2	2	2
wp8	Square	0	0	1	3

The value of the salary represents the remuneration per day that the waypoint provides. In this first version, the money does not accumulate from one day to another. The cost is the value that the waypoint requires for the agent to use it. The satisfaction is the classifier attribute of the system. It is the sum of the accumulated values of satisfaction on a day that defines whether a routine is good or not. The time represents the hours that each waypoint takes to be used by the agent. Like said before, the sum of the times cannot be higher than16. Analyzing two simple instances we have [Table 2]:

Table 2. Examples of instances

wp1	wp2	wp3	wp4	wp5	wp6	wp7	wp8	satisfaction
2	0	0	0	0	1	1	0	5
0	1	0	0	1	0	0	2	7

Where in the first case the agent worked two shifts at the Office, then went to the boite, and finally to the circus. He raised a total of 16 and spent 5. In the second case it worked a shift at the Police station, went to the pizzeria and spent the rest of the time in the square, having a total compensation of 5 and a total expense of 4. The better of these two instances was the second one, due to its satisfaction value.

If the agent does not have enough money to pay for the leisure waypoint that is in its instance, it will be standing in the front of the waypoint until its end time. The agent will not receive satisfaction.

If the agent decides to go to a waypoint that gives remuneration and the same is closed, destroyed or out of work, the agent will be standing in front of the waypoint until its end of time. The agent will not receive remuneration.

5.4. System Architecture

To avoid the elaboration of more complex schemes, we decided to join, in an activity diagram, the server and the client modules to illustrate the applied architecture (as in Figure 3).

The client initializes after the server, instantiates a predetermined number of agents and warns the latter. The server, in turn, creates a thread for each agent. Each process created uses the learning module with the M5P algorithm. As described in [Machado et al 2010], in this moment the Weka's DLL starts to be interpreted by IKVM.

The client, in turn, starts the simulation. Each agent thread has the client-server communication protocol. As can be seen in the Agent Thread Module (Figure 3), at this moment the M5P-Reinforcement Learn is implemented.

Considering the agent in an individual manner we have it beginning by generating an instance (set of waypoints that forms a routine) randomly. After, it leaves the hostel and visits the waypoints until runs out of valid period of the day. We consider as trial period, this 10 initial days constant that the agents will use to collect information from the environment. Once the visits are over, the routine information and the satisfaction gained each day are sent for the server that induces the tree and linear equations. Then generates a set of 40 valid instances (which sum of hours of their times are less or equal to 16), rank them through the learning rule, elects the better and sends to the client.

This ranking is done by the comparison of the attributes classified by the satisfaction field of the instances base. As can be seen, best instance definition is not based on simple summation of the satisfaction of each waypoint, at the moment. This fact is justified since not necessarily an instance with only entertainment is good, since the agent must work to have money to spend on them. Hence the necessity of learning rule.

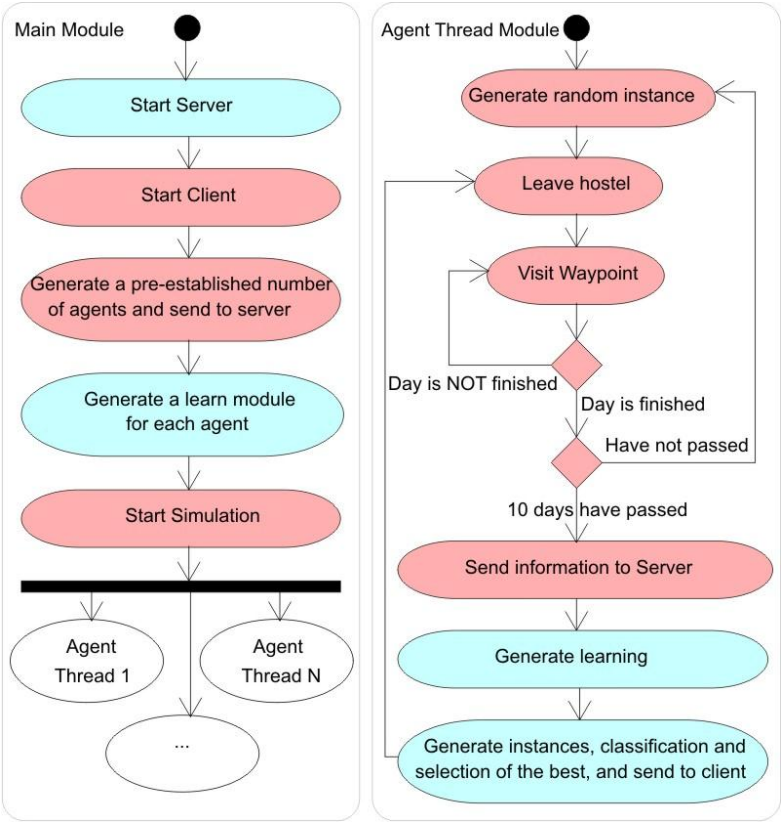


Figure 3 - Example of an application of the M5P-Reinforcement Learn Architecture to a simulation of artificial life. The states represented by blue are executed by the Server and the states in pink are executed by the client (virtual environment).

The simulation, on the client side, restarts from there. If the number of agents and the characteristics of the waypoints do not change, every new day the learned instance tends to be similar to the previous. Otherwise changes may occur. The random component of the stage of diversification of the algorithm ensures that, even if the agent has a belief that a routine is good, perchance it might know another.

Another important feature of the system is the dynamic definition of the attributes of the instances base, which ensures that the agent can receive new information about waypoints that were not programmed by the developer.

5.5. Experiments and preliminary results

We realized two batteries of experiments, one to evaluate the evolution of the characters and another to verify their adaptation to the environment.

The first set of experiments was realized with 4 individuals during 20 days. In the first 9, they gathered data and in the following they defined their routine based on experience to optimize their satisfaction (the higher the better satisfaction). This experiment was repeated 10 times and the average of satisfaction per individual can be seen in the Figure 4.

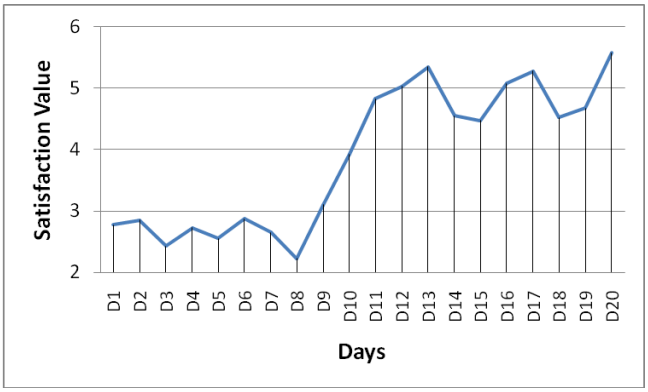


Figure 4: Results of the satisfaction increase based on the learning period

As from the 10th day, satisfaction grew and stabilized close to the value 5, it was shown that the agents are capable to perceive the environment and optimize their results.

To test the adaptation, we compromised the environment right after the agent's learning. Since the 10th day, each agent started to use its learning and since the 15th day we raised the salary of the office, reduced the salary of the industry and took the entire satisfaction of the boite. In this experiment, the instances base for learning was tested in two different ways: with no limit of records and with limit (containing only the 10 and 5 last days). It is observed that 10 tests were performed for each of the three forms described. Results can be seen in the Figure 5:

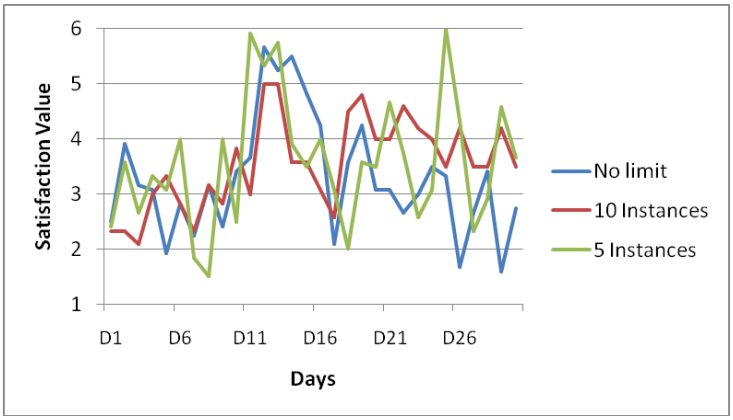


Figure 5: Agent adaption with three experimental bases and different information values.

In this experiment, although since day 15 the three configurations had a drastic and expected income fall, what really matters is the evolution from that day on. When we store the good and bad cases and then completely change the values of the classifiers attributes, our instances base passes through a latency of learning that can compromise next results. This is what happened in the configuration without limit, which until the 30th day failed to achieve the performance demonstrated at its peak period (days 10 to 14). However, if we have a small instance base we will not have a rule ready to handle all cases, and although a drastic change can be quickly bypassed, its result does not have constancy, it is what happens with the base with the last 5 instances. The best case of adaptation has been generated in the intermediate case, where with 10 instances the individual can create a consistent learning rule and at the same time short enough to fit the new cases without much learning latency.

From these experiments we can conclude that when we use the M5P-Reinforcement Learn Architecture and we configure the learning with a regular tax of experience storage we can ensure satisfactory evolution and adaptation of the agents.

Greedy methods such as stabilize the behavior keeping the best rule were not evaluated.

6. Emergent Behaviors Generated

As in [Machado et al 2010], through the resource management system for generating strategies present in the application it was able to adapt to the enemy and environment using emergent behaviors, in the experiments done in the New World environment, the agents have adapted, in their own way, to the situation. As can be seen in Table 3 in a same test and environment, and with the same configuration of waypoints, different agents were able to generate completely different efficient rules.

Table 3. Rules Generated by M5P Algorithm of Three Individuals in the Day 20

Pizza_Store <= 0.5 : LM1 (15/26.974%) Pizza_Store > 0.5 : LM2 (14/51.011%) LM num: 1 satisfaction = 0.6785 * office + 1.759 * Pizza_Store + 0.9362 * circus + 1.3564 * park + 0.3925 LM num: 2 satisfaction = 1.2996 * office + 1.8197 * Pizza_Store + 0.8736 * boite + 2.565
LM num: 1 satisfaction = 1.4356 * police + 0.8362 * Pizza_Store - 1.187 * boite + 1.3149 * circus + 2.0125
LM num: 1 satisfaction = -0.6679 * industry + 2.9391 * Pizza_Store + 2.5821

Reviewing the rules of each individual (Table 3) we realize that the first, if doesn't have much chance to go to the pizzeria, prefers to work in the office and have fun at the pizzeria, at the circus and/or at the square, but if it has the chance to go to the pizzeria, prefers more work in the office and also have fun at the pizzeria and/or at the boite. The second decided much more for the work at the police station, and entertainment at the pizzeria and circus, this avoids the most the club. The third is most simple: works anywhere but avoids the industry and tends to have fun ate the pizzeria.

7. Architecture Applications

Considering a multi-agents environment with learning utilizing a M5P Reinforcement Architecture we can highlight the applications:

- Using criteria such as the restriction on learning mediated by a challenge function, each intelligent agent can learn just enough to ensure good gameplay regarding the balancing of the difficulty in a game of strategy.
- The intelligent collective behavior generated by the interaction of the agents with learning can provide an environment even more diversified for character-based interactive historytelling. For such it would only be necessary the incorporation of a module to ensure consistency of the storyline.

- In applications in the style of life simulation, the use of learning ensures that the agents are ready to adapt to new situations for which they were not previously programed. So responding in a human-like manner to the global interactions.

8. Conclusions

In this work, the M5P Reinforcement Architecture was developed, applied to intelligent agents with learning immersed in complex environments in the field of electronic games and simulation. Its implementation and experiments were conducted in a multi-agent environment called "New World", whose technology was first defined in [Machado et al 2010].

This architecture was suggested for any electronic game genre with multiple agents, especially for character-based interactive storytelling, life simulation and game difficulty dynamic balancing.

The M5P Reinforcement Architecture proposes the on-line training of the individual in the environment, feeding the numerical classifier with the system reward and inducing a tree with linear equations easy to interpret, solving many of the challenges cited by [Manslow 2002].

The experiments demonstrated that the numerical M5P classifier, present in this architecture, offers better performance for a partial instance base (not complete). Which guarantees a satisfactory evolution and a better adaptation of the agents (questioning presented in [Rollings and Ernest 2006]).

In our experiments all the agents had the same loading parameters. A future project would be to evaluate the behavior of this multi-agent system with agents with different profiles. For example, an individual with less willingness to work, other with higher yield (works better or moves faster), another with a slower rate of adaptation (which can take more time to identify a good schedule). Evaluation of the system performance in an overgrowth in the generation of such individuals is also considered future work.

References

Andrade, G., Ramalho, G., Santana, H. and Corruble, V., (2005) "Automatic computer game balancing: A reinforcement learning approach" in *Proceedings of the International Conference on Autonomous Agents*, pp.1229-1230.

Bakkes, S., Spronck P., and van den Herik, J. (2009) "Rapid and Reliable Adaptation of Video Game AI." in *IEEE Transactions on Computational Intelligence and AI in Games*. Vol.1, No, 2.

Bhattacharya, B. and Solomatine, D. P. (2005) "Neural networks and M5 model trees in modelling water level-discharge relationship" in *Neurocomputing*, Vol. 63, pp. 381-396.

Booker, L.B., Goldberg, D.E. and Holland, J.H. (1989) "Classifier Systems and Genetic Algorithms", Artificial Intelligence, vol. 40, pp. 235-282.

Cavazza, M., Charles, F. and Mead, S. J., (2002) "Emergent situations in interactive storytelling" in *SAC '02 Proceedings of the ACM symposium on Applied computing,* pp. 1080-1085.

Crocomo, M. K. and Simões, E. V. (2008) "Um Algoritmo Evolutivo para Aprendizado On-line em Jogos Eletrônicos" In: *SBGames.*

Etemad-Shahidi, A. and Mahjoobi, J., (2009) "Comparison between M5′ model tree and neural networks for prediction of significant wave height in Lake Superior " in *Ocean Engineering* Volume 36, Issues 15-16, pp 1175-1181.

Goertzel, B., Pennachin, C., Geisweiller, N., Looks, M., Senna, A., Silva, W., Heljakka, A. and Lopes, C., (2008) "An integrative methodology for teaching embodied non-linguistic agents, applied to virtual animals in second life". In: *Proceedings of the First Artificial General Intelligence Conference.*

Goldberg, D. E., (1989) "Genetic Algorithms in Search, Optimization and Machine Learning ", in *Addison-Wesley Longman Publishing Co.*.

Hong, J. and Cho S., (2005) "Evolving Reactive NPCs for the Real-Time Simulation Game" In: *IEEE 2005 Symposium on Computational Intelligence and Games.*

L.Lab, Secondlife.http://secondlife.com/, (2011).

Machado, A. F. V., Clua, E. W. and Zadrozny B. (2010) "A Method for Generating Emergent Behaviors using Machine Learning to Strategy Games". In: *SBGames.*

Malone, T. W. and Crowston, K., (2004) "The interdisciplinary study of coordination" in *ACM Computing Surveys (CSUR)*, Volume 26 Issue 1, pp. 87-119.

Manslow, J., (2002) "Learning and Adaptation" in *AI GAME PROGRAMMING WISDOM,* pp. 557-566.

Pallay, C., Rehm M. and Kurdyukova E., (2009) "Getting acquainted in Second Life: human agent interactions in virtual environments." In: *Proceeding ACE '09 Proceedings of the International Conference on Advances in Computer Enterntainment Technology* .

Quinlan, J. R., (1992) "Learning with Continuous Classes" in *AI' 92*, pp. 343-348.

Riedl, M., Saretto, C. J., Michael Young, R. (2003) "Managing Interaction Between Users and Agents in a Multi-agent Storytelling Environment." In: *AAMAS '03 Proceedings of the second international joint conference on Autonomous agents and multiagent systems.*

Rollings, A., Ernest, A., (2006) "Fundamentals of Game Design", *New Challenges for Character-Based AI for Games. Chapter 20: Artificial Life and Puzzle Games. Prentice Hall,* pp. 573-590.

Spronck, P., Kuyper, S. I. and Postma, E., (2004) "Difficulty scaling of game AI" in *Proceedings of the 5th International Conference on Intelligent Games and Simulation*, pp. 33-37.

Witze, A., Zvesper, J. A. and Kennerly, E., (2008) "Explicit Knowledge Programming for Computer Games" in *AIIDE.*

Disseminação Bio-Inspirada de Eventos em Redes Dinâmicas e Descentralizadas

Adam S. Banzi, Aurora T. R. Pozo, Elias P. Duarte Jr.

Departamento de Informática – Universidade Federal do Paraná (UFPR)
Caixa Postal 19018, Curitiba PR CEP 81531-980 Brasil

{adam,aurora,elias}@inf.ufpr.br

Resumo. *Este trabalho apresenta uma estratégia baseada em inteligência coletiva para a disseminação de eventos em redes dinâmicas e descentralizadas. Um evento corresponde à mudança do estado de um nodo ou enlace da rede. Através de formigas, que são abstrações para agentes móveis, as informações são disseminadas na rede. Um nodo que detecta o evento na sua vizinhança dispara a disseminação, que é controlada por feromônios. O nível de feromônios é utilizado para controlar o tamanho da população de formigas e determinar as rotas que percorrem. Um estudo empírico foi realizado, comparando a estratégia proposta com a disseminação por inundação e* gossip*. Resultados permitem observar que o algoritmo apresenta uma solução de compromisso entre o tempo necessário para a disseminação e a sobrecarga em termos do número de mensagens utilizadas.*

Abstract. *This work presents a strategy based on* swarm intelligence *for spreading events in dynamic and decentralized networks. An event is defined as a state transition of a node or link. By means of ants, which correspond to mobile agents, information is spread throughout the network. A node which detects an event in its neighborhood starts disseminating the new information. Pheromones are used to both control the ant population and help define the paths that the agents take. An empirical study was performed, in which the proposed strategy was compared with* flooding *and* gossip *algorithms. Results obtained show that the proposed strategy presents a good trade-off between the time required to disseminate information and the overhead in terms of the number of messages.*

1. Introdução

A disseminação de eventos é um dos desafios das redes dinâmicas, tais como redes móveis *ad hoc* e *peer-to-peer*. Em modelos tradicionais de rede, com pouca modificação da composição, o controle e administração dos recursos são, comparativamente, simplificados [Dong et al. 2006]. Por outro lado, nas redes dinâmicas a própria frequência com que nodos entram e saem é alta, impedindo que as decisões de disseminação sejam tomadas de forma antecipada a partir de um conhecimento prévio da rede [Ahmed and Far 2007]. Outra característica relevante é a descentralização, ou seja, a ausência de um ponto único que possa ser responsabilizado pelo controle. Devido ao alto dinamismo, a manutenção do funcionamento da rede deve ser realizada conjuntamente pelos nodos ativos.

Para auxiliar na tarefa de gerenciamento de recursos envolvidos, inclusive na disseminação de informações, é importante que os integrantes da rede tornem-se cientes das mudanças ocorridas na topologia de maneira rápida e eficiente. Em modelos

centralizados e com pouco dinamismo, pode ser usada uma estratégia determinística para disseminação de eventos. Para as redes dinâmicas, a descoberta e disseminação de novidades aos participantes necessita de estratégias também dinâmicas e descentralizadas, que se adaptam a alterações imprevistas.

Uma abordagem promissora para tratar esse problema origina-se da observação da natureza. Certos grupos de seres vivos, tais como as colônias de formigas e enxames de abelhas, apresentam a chamada inteligência coletiva [Bonabeau et al. 1999]. Inteligência coletiva é a propriedade apresentada por sistemas constituídos por agentes dotados de comportamento autônomo, capazes de executar ações simples. Tais agentes, coletivamente, podem apresentar comportamento inteligente [White and Pagurek 1998]. Em outras palavras, as interações entre os diversos elementos é a chave para a realização da tarefa desejada.

A estratégia apresentada neste trabalho é baseada no algoritmo *EvAnt*, que utiliza inteligência coletiva para disseminar eventos em redes dinâmicas e descentralizadas. O algoritmo de inteligência coletiva é requisitado somente em situações nas quais há novidades a serem informadas aos elementos da rede. Estudos empíricos foram realizados a fim de avaliar a eficiência e eficácia da abordagem. A comparação da estratégia proposta foi realizada com a inundação (*flooding*) [Mullender 1993] e a disseminação epidêmica (*gossip*) [Eugster et al. 2004]. Dentre os aspectos avaliados encontram-se a velocidade que o algoritmo consegue disseminar novidades e a carga imposta à rede durante tal processo.

O restante deste trabalho está organizado conforme descrito a seguir. A seção 2 apresenta a estratégia proposta, sendo seguida da descrição das simulações e avaliação de resultados na seção 3. Na seção 4 são descritos os trabalhos relacionados. Por fim, as conclusões seguem na seção 5.

2. *EvAnt*

A solução proposta possui como base o algoritmo *EvAnt*, elaborado para este trabalho. O algoritmo faz uso de inteligência coletiva para disseminar, através de agentes móveis, novidades em redes dinâmicas e descentralizadas. Tais redes possuem comportamento imprevisível e podem sofrer alterações (a saída e entrada de nodos e links) frequentes.

2.1. Disseminação de novidades baseada em eventos

O algoritmo é disparado com a ocorrência de um evento. Denomina-se *evento*, ou *novidade*, alguma alteração no estado de um componente da rede. Uma alteração corresponde à criação ou eliminação de um enlace bem como o surgimento ou a saída de algum nodo da rede. Após a ocorrência de um evento, os nodos que o *detectaram* são aqueles que iniciam a *disseminação* das informações. Os nodos que inicialmente tornam-se cientes dos eventos e disparam a disseminação diferem de acordo com o tipo de evento:

surgimento de um nodo: a disseminação da novidade inicia-se pelo próprio nodo que acabou de surgir, bem como pelos nodos que são seus vizinhos;

saída de um nodo: os vizinhos do nodo que saiu da rede iniciam a disseminação;

criação de um enlace: os nodos que estão ligados pelo novo enlace iniciam a propagação do evento;

eliminação de um enlace: os nodos que estavam ligados pelo enlace eliminado iniciam a propagação da novidade.

2.2. Modelo do sistema

O ambiente em que o algoritmo é executado é uma rede dinâmica e descentralizada, como são as redes *peer-to-peer* não estruturadas e redes *ad hoc*. A entrada e saída de *hosts* pode ser intensa, de modo que as mudanças no estado da rede são frequentes e imprevisíveis. Outro ponto a ser observado é a total ausência de componentes que centralizam a tarefa de coletar e disseminar as novidades.

O modelo de sistema a ser considerado no algoritmo *EvAnt* é definido como um grafo não direcionado $G(t) = (V_t, E_t)$ [Nassu et al. 2007]. Tal grafo é composto, em um instante de tempo t, de um conjunto de vértices V_t e um conjunto de arestas E_t. Por ser definido em função do tempo, pode ocorrer que, para dois instantes de tempo físico distintos t_1 e t_2, $G(t_1) \neq G(t_2)$. As seguintes características definem o modelo de sistema considerado:

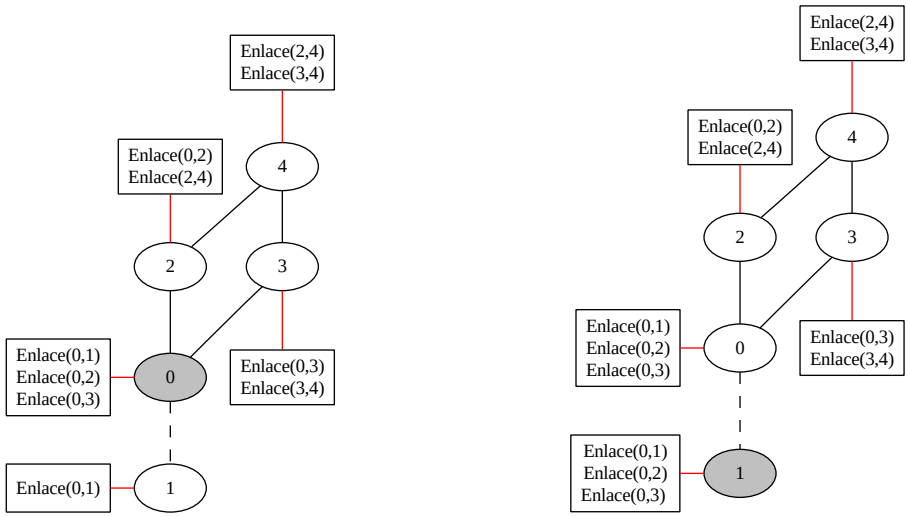
1. Os nodos sempre possuem, para um dado instante de tempo t, o conhecimento correto sobre o estado dos vizinhos, isto é, dos vértices aos quais estão diretamente conectados através de um único enlace.
2. O envio de mensagens é livre de erros. O tempo para o envio de informações é não desprezível e não há limite de tempo definido para que uma mensagem chegue ao destino.
3. Não há relógio global compartilhado pelos nodos. Cada nodo possui apenas um relógio local e os diversos relógios locais dos nodos não são sincronizados entre si. Logo, a diferença entre dois instantes de tempo físico t e $(t+1)$ não é a mesma para os vários participantes da rede.
4. O grafo que representa o sistema é não direcionado. Os enlaces são bidirecionais, ou seja, vale a igualdade $e_{ijt} = e_{jit}$.

2.3. Disseminação das formigas

As informações são disseminadas pela rede através de formigas, que são abstrações para agentes móveis. Um exemplo de como as mensagens são transmitidas entre os nodos é mostrado na Figura 1. Na primeira figura, 1(a), a formiga localiza-se no nodo 0, mostrado em cor cinza. Neste caso, a formiga carrega informações relativas aos enlaces que o nodo 0 possui, conforme mostrado no quadro à esquerda do nodo. O destino escolhido pela formiga é o nodo 1, caminho mostrado pelo enlace pontilhado. Na segunda figura, 1(b), o nodo 1 já foi atualizado com as informações trazidas pela formiga.

A disseminação das novidades, com o objetivo de permitir que as formigas consigam transmitir as informações para toda a rede, utiliza a abordagem baseada na estigmergia. Cada nodo contém um depósito de feromônios associado a cada um dos enlaces que possui. Quando uma formiga viaja de um nodo i em direção a algum vizinho j, a concentração $C_{ij}(t)$ de feromônios daquele enlace é incrementada em uma unidade, conforme mostra a expressão $C_{ij}(t) = C_{ij}(t-1) + 1$ [Nassu et al. 2007].

Os feromônios também são utilizados para auxiliar na escolha do próximo destino das formigas. Essa escolha é feita de forma probabilística: enlaces com menor concentração de feromônios possuem maior probabilidade de serem escolhidos. A expressão $P_{ij}(t) = \frac{\left(C_{ij}(t)+1\right)^{-\alpha}}{\sum_{k=1}^{|e_{it}|}\left(C_{iE_{it}(k)}(t)+1\right)^{-\alpha}}$ define a probabilidade de uma formiga migrar do nodo i para o j no instante de tempo t. O tamanho da vizinhança do nodo i é representado



(a) Formiga (localizada no nodo cinza) migra do nodo 0 para o nodo 1, carregando as informações do nodo 0.

(b) Nodo 1 é atualizado com as informações trazidas pela formiga.

Figura 1. Disseminação de informações pela formiga.

por $|e_{it}|$ e $E_{it}(k)$ denota o k-ésimo vizinho do nodo i no tempo t. A constante α, tal que $\alpha > 0$, representa a força dos feromônios. Maior o valor de α, maior a diferença entre as probabilidades obtidas. Os feromônios, assim, servem para "afastar" as formigas umas das outras, fazendo com que elas sejam melhor distribuídas pela rede.

Com o passar do tempo, os feromônios enfraquecem. O enfraquecimento dos feromônios é dado pela expressão $C_{ij}(t) = C_{ij}(t_0) \cdot (1 - \rho)^{(t-t_0)}$. A nova concentração $C_{ij}(t)$ de feromônios do enlace e_{ij} no tempo t é dada em função da concentração do enlace em um instante de tempo anterior t_0 e uma constante ρ, tal que $0 \leq \rho < 1$. Essa constante controla a velocidade da evaporação. Maior o ρ, mais rapidamente os feromônios evaporam.

2.4. Controle da população de formigas

A quantidade de formigas na rede possui grande impacto no desempenho do algoritmo. O excesso de agentes faz com que as mesmas informações sejam repetidamente disseminadas pela rede. Por outro lado, com um número insuficiente de formigas as novidades não conseguem alcançar todos os nodos. Para controlar a população de formigas de maneira dinâmica, também são utilizados feromônios.

A concentração total $C_i(t)$ de feromônios de um nodo i é dada pela a soma dos valores de feromônio de cada enlace desse nó, conforme mostrado na expressão $C_i(t) = \sum_{k=1}^{|e_{it}|} C_{iE_{it}(k)}(t)$. Esse valor é então comparado aos limites mínimo $\big(L_{min}(t)\big)$ e máximo $\big(L_{max}(t)\big)$, definidos pelas expressões $L_{min}(t) = \gamma_{min} \cdot |e_{it}|^{\delta}$ e $L_{max}(t) = \gamma_{max} \cdot |e_{it}|^{\delta}$, respectivamente. A constante δ, tal que $\delta \geq 0$, define o papel do tamanho da vizinhança sobre a definição dos limites. À medida que o valor de δ aumenta, maior a influência que o tamanho da vizinhança terá sobre os valores de $L_{min}(t)$ e $L_{max}(t)$. As constantes γ_{min} e γ_{max} devem ser escolhidas de modo que $\gamma_{min} > 0$ e $\gamma_{max} > \gamma_{min}$.

Caso $C_i(t)$ seja menor ou igual a $L_{min}(t)$, uma formiga é criada pelo nodo. Essa formiga carregará inicialmente as mesmas informações que aquele nodo possui. Se, por outro lado, $C_i(t)$ for superior ou igual a $L_{max}(t)$, todas as formigas localizadas no nodo são destruídas.

2.5. Especificação do algoritmo

A estratégia proposta é composta de duas partes principais: a primeira, apresentada no algoritmo 1 (*EvAnt*: ciclo do nodo), descreve o comportamento de um nodo i. O algoritmo 2 (*EvAnt*: ciclo da formiga), por sua vez, mostra as ações executadas por uma formiga f.

Inicialmente, no algoritmo 1, um novo evento faz com que os nodos que detectaram o acontecimento criem formigas para disseminar a novidade pela rede. Os nodos que detectaram o evento criam formigas em toda rodada na qual a concentração local de feromônios $C_i(t)$ for menor ou igual ao limite mínimo $L_{min}(t)$. Nodos que receberem formigas com novidades também criam um novo agente a cada rodada, enquanto a condição $C_i(t) \leq L_{min}(t)$ for respeitada. Caso, em alguma rodada, o nodo i possua a concentração local de feromônios $C_i(t)$ maior ou igual ao limite superior $L_{max}(t)$, todas as formigas que no momento estão no nodo i são destruídas.

O ciclo das formigas, mostrado no algoritmo 2, é composto por 5 ações. Inicialmente, a formiga, vinda do nodo i, chega ao nodo j e incrementa a concentração de feromônios relativa ao enlace (i, j). Em seguida, a formiga atualiza o nodo com as novidades que carrega consigo. Após a formiga ficar inativa no nodo por um determinado período de tempo, que é parâmetro do algoritmo, a formiga então se atualiza com informações que ela não possui. Por fim, a formiga seleciona o próximo destino, de acordo com a expressão mostrada na linha 6 do algoritmo.

Algoritmo 1: EvAnt: ciclo do nodo i

```
1  while true do
2      evapora feromônios → C_ij(t) = C_ij(t_0) · (1 − ρ)^(t−t_0)
3      if i detectou evento or (i recebeu formiga que carrega novas informações and C_i(t) ≤ L_min(t)) then
4          cria nova formiga
5          escalona ciclo para a nova formiga (Algoritmo 2)
6      end
7      else
8          if C_i(t) ≥ L_max(t) then
9              destrói todas as formigas localizadas no nodo i
10         end
11     end
12 end
```

Algoritmo 2: EvAnt: ciclo da formiga f

```
1  while true do
2      formiga chega ao nodo j, vinda do nodo i → C_ij(t) = C_ij(t − 1) + 1
3      formiga atualiza informações do nodo
4      formiga fica inativa no nodo por um determinado tempo
5      formiga atualiza-se com informações do nodo
6      formiga seleciona o próximo destino → P_ij(t) = (C_ij(t)+1)^(−α) / Σ_{k=1}^{|e_it|} (C_{iE_it(k)}(t)+1)^(−α)
7  end
```

3. Resultados das simulações

Ao longo desta seção serão apresentados resultados de estudos empíricos realizados para validar e avaliar o algoritmo proposto. Será mostrada uma comparação de resultados obtidos através da simulação entre o algoritmo *EvAnt*, a inundação (*flooding*) [Mullender 1993] e a disseminação epidêmica (*gossip*) [Eugster et al. 2004]. Os três algoritmos são utilizados na disseminação de informações sobre um evento detectado. Ambos os algoritmos foram escolhidos para a comparação por serem métodos de referência em disseminação de informações. Os três métodos serão avaliados através de simulações com o objetivo de observar a eficiência e eficácia dos algoritmos na atualização das informações da rede baseada na detecção de eventos.

3.1. Descrição das simulações

As simulações foram conduzidas conforme descrito a seguir. Inicialmente, todos os nodos da rede possuem, em suas descrições locais, informações corretas sobre o estado de todos os nodos da rede. Então, a partir desse estado, eventos são disparados ao longo do tempo e informações relativas ao processo da detecção desses eventos são coletadas. As métricas escolhidas possuem como objetivo avaliar a velocidade de detecção dos acontecimentos, bem como o peso que cada método impõe à rede para todos os nodos ficarem cientes dos eventos.

Para avaliar a velocidade da detecção de eventos, foram utilizadas duas métricas: latência (*latency*) e a taxa de ciência (*awareness rate*). A latência é definida como o número de rodadas necessário para que todos os eventos sejam detectados. A taxa de ciência tem como função mostrar a quantidade de nodos cientes dos eventos ao longo do tempo. Esta taxa pode ser obtida através da expressão $\frac{\sum_{i=1}^{|\varepsilon|} |\mu_i|}{|\varepsilon|}$, onde μ_i denota o número de nodos cientes do i-ésimo evento e $|\varepsilon|$ representa o tamanho do conjunto de eventos. Por exemplo: para uma rede com 100 nodos na qual foram disparados 3 eventos ($|\varepsilon| = 3$), a taxa de ciência é dada por $\frac{(|\mu_1|+|\mu_2|+|\mu_3|)}{3}$. Caso não haja eventos ($|\varepsilon| = 0$), a métrica não se aplica.

As 2 métricas utilizadas para medir a carga imposta pelo método de detecção de eventos são o número de mensagens trocadas (*# messages exchanged*) e a quantidade de mensagens criadas (*# agents created*). O número de mensagens trocadas representa a quantidade de formigas que se deslocaram de um nodo a outro, mais uma vez considerando um determinado intervalo de tempo. As mensagens criadas denotam o número de novas formigas criadas ao longo do processo de detecção de eventos.

Redes de topologia *powerlaw* [Bu and Towsley 2002] e aleatória foram utilizadas. Uma rede de topologia *powerlaw* apresenta um determinado conjunto de nodos fortemente conectado. Os nodos restantes são ligados a algum membro desse conjunto. Essa distribuição de nodos e enlaces é semelhante à encontrada na Internet. Uma rede de topologia aleatória possui os enlaces escolhidos de maneira randômica, sendo conexa e cada nodo com ao menos um enlace. As redes aleatórias possuem uma distribuição mais uniforme dos graus de seus nodos, se comparadas às redes *powerlaw*. Foram utilizadas redes de 1000, 2000 e 2500 nodos. Para cada tamanho, 20 redes foram geradas, 10 com topologia *powerlaw* e outras 10 com topologia aleatória. As redes *powerlaw* foram criadas de acordo com o seguinte critério [Bu and Towsley 2002]: $m_0 = 20\%$ do tamanho da rede e

sementes que variam de 0 a 9. As redes de topologia aleatória possuem o mesmo número de enlaces que as redes *powerlaw*.

As simulações tiveram tempo limite de 50 unidades de tempo. As rodadas, definidas como o intervalo no qual as métricas são coletadas, possuem duração de 0.2 unidades de tempo. Para cada uma das redes, um conjunto aleatório de eventos correspondente a 1% do tamanho da rede foi gerado. Esse conjunto de eventos foi estabelecido de modo que, após a sua ocorrência, a rede continue conexa. Os eventos possíveis são a criação e remoção de nodos e enlaces. Cada evento é disparado em intervalos de 5 rodadas (1 unidade de tempo).

A simulação do algoritmo *EvAnt* termina quando todos os eventos forem detectados (*awareness rate* igual ao tamanho da rede) ou o tempo limite da simulação for atingido. O algoritmo *EvAnt* foi comparado com a inundação (*flooding*) e a disseminação epidêmica (*gossip*).

No *flooding* [Mullender 1993], a disseminação de informações inicia-se a partir do ponto de origem dos eventos. Quando um nodo receber uma mensagem que traga novidades às informações contidas naquele nodo, essas informações locais são atualizadas bem como a mensagem de modo que ambos possuam as mesmas informações. Essa nova mensagem é então disseminada por inundação a partir do nodo. Os termos mensagem, agente e formiga referem-se à mesma entidade, responsável por disseminar os eventos pela rede. O algoritmo termina após uma rodada na qual nenhum nodo disseminou novas mensagens.

No *gossip* [Eugster et al. 2004], a disseminação de informações também é iniciada no ponto onde os eventos são originados. Os nodos que recebem mensagens contendo novas informações são atualizados com as novidades e atualizam a mensagem de modo que ambos contenham as mesmas informações. O nodo então dissemina essa mensagem para $f \geq 1$ vizinhos escolhidos aleatoriamente. Tal mensagem é disseminada pelo nodo durante $t \geq 1$ rodadas. Os parâmetros t e f são os mesmos para todos os nodos da rede.

O *gossip* utilizado neste artigo possui como parâmetros $f = \log_{10}(N)$ e $t = 1$. Esse valor foi escolhido de acordo com o critério explicado em [Bollobas 2001]. Os parâmetros utilizados para o algoritmo *EvAnt* são mostrados à seguir:

Força dos feromônios (α)**:** 8
Taxa de evaporação dos feromônios (ρ)**:** 0.4
Peso da vizinhança (δ)**:** 0.5
Limite inferior para a concentração local de feromônios (γ_{min})**:** $\frac{\log(N)/4}{|\text{vizinhança média}|^{\delta}}$
Limite superior para a concentração local de feromônios (γ_{max})**:** $100 \times \gamma_{min}$
Tempo de inatividade das formigas: 0.2 segundos (1 rodada).

Para obter uma comparação com o *gossip*, foi estipulado que o *EvAnt* deve disseminar mensagens por, em média, $\log_{10}(N)$ rodadas. Assim, o limite mínimo $L_{min}(t)$ da concentração de feromônios foi fixado como $\log_{10}(N)$, então, o valor da constante γ_{min} foi obtido. Com o valor de γ_{min} fixado, os demais valores foram obtidos através de análise experimental, ou seja, diversos valores foram simulados até que a melhor combinação de constantes fosse obtida.

Pode-se obter um paralelo sobre a maneira como o *gossip* e o *EvAnt* disseminam as mensagens. No *gossip*, cada nodo dissemina $\log_{10}(N)$ mensagens, todas na mesma

rodada. Por outro lado, no algoritmo *EvAnt* cada nodo dissemina, em média, $\log_{10}(N)$ mensagens, uma por rodada.

3.2. Resultados obtidos

Os resultados das simulações são mostrados a seguir. Na Figura 2 temos os gráficos referentes às médias obtidas das 10 redes *powerlaw* de 2500 nodos. A Figura 2(a) apresenta a latência obtida por cada estratégia. Nas Figuras 2(b) e 2(c) podemos observar a o número de mensagens trocadas e a quantidade de formigas criadas ao longo do tempo, respectivamente. De acordo com a Figura 2(a), o *gossip* não foi capaz de deixar todos os nodos cientes das novidades. Os outros dois algoritmos conseguiram comunicar todos os eventos, com o *flooding* sendo, em média, 18.2% (28 rodadas) mais rápido que o *EvAnt*. Para tal, o *flooding* trocou aproximadamente 19.7% (ou 42000) mensagens a mais do que o *EvAnt*, conforme pode ser visto em 2(b). A diferença entre a quantidade de mensagens criadas ao longo das simulações favorece o *EvAnt*, pois ele não necessita criar cópias das mensagens que dissemina aos vizinhos. Cerca de 99.4% agentes a mais foram criados pelo *flooding* (Figura 2(c)). O *gossip*, mesmo não conseguindo atingir todos os nodos da rede, criou cerca de 96000 mensagens a mais que o algoritmo *EvAnt*, uma diferença de 98.65%. A Figura 2(d) mostra a *awareness rate*, ou seja, reflete com mais detalhes os resultados mostrados em 2(a).

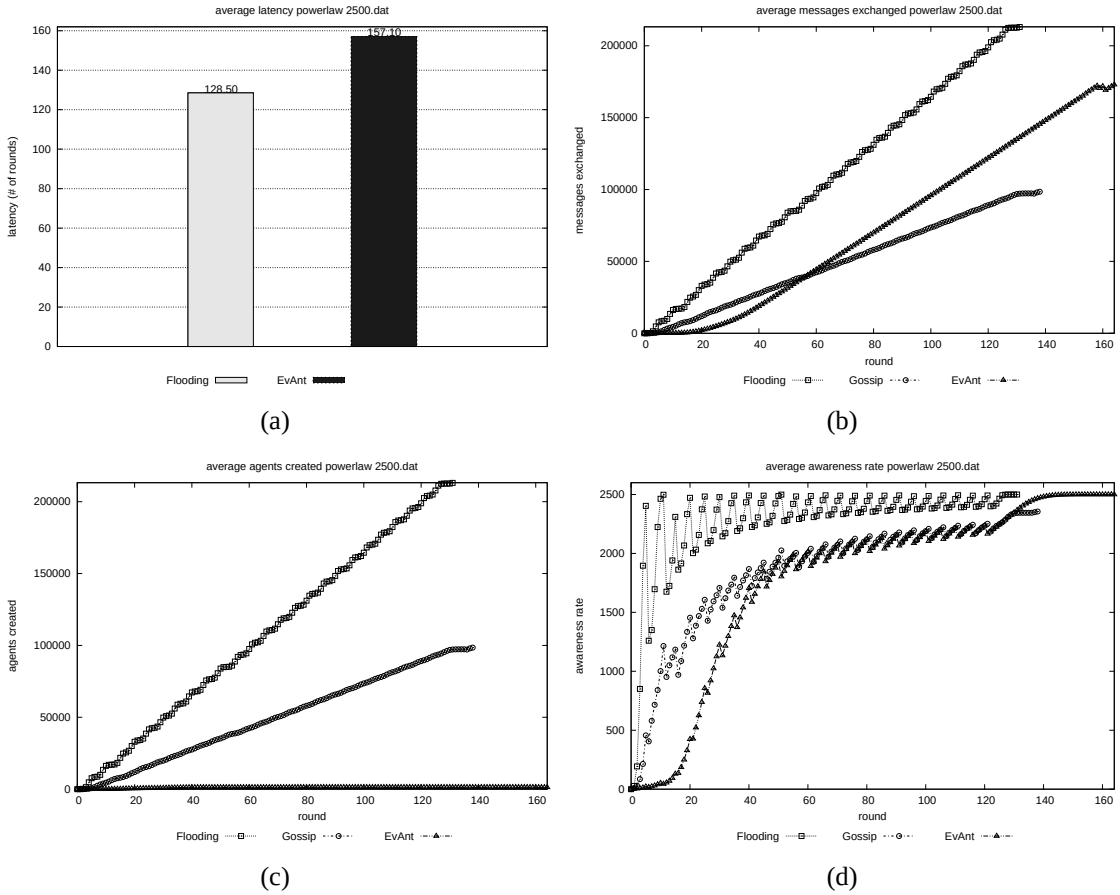


Figura 2. Gráficos referentes às redes *powerlaw* de 2500 nodos.

Os gráficos referentes às redes de topologia aleatória de 2500 nodos, bem como

os resultados referentes às redes de 1000 e 2000 nodos são omitidos aqui por restrições de espaço. Os resultados obtidos em tais redes foram semelhantes aos apresentados nas redes *powerlaw* de 2500 nodos.

Um resumo dos resultados referentes às redes de 2500 nodos é mostrado na Tabela 1. Para cada métrica, o valor mostrado na parte superior refere-se à média dos dados obtidos nas simulações. Os valores mostrados na parte inferior dizem respeito aos desvios padrão obtidos.

Para as duas topologias, *powerlaw* e aleatória, pode ser observado que tanto no *flooding* quanto no *EvAnt* toda a rede consegue ficar ciente de todas as novidades. O *flooding*, como um algoritmo de disseminação confiável, sempre consegue tornar a rede ciente de todos os acontecimentos. Uma vez que a *awareness rate* do algoritmo *EvAnt* é a mesma obtida pelo *flooding*, conclui-se que o algoritmo proposto consegue também deixar toda a rede ciente dos eventos. O *gossip* em nenhum caso conseguiu disseminar os eventos para todos os nodos. A latência obtida pelo *flooding*, como era de se esperar, é menor que a obtida pelo *EvAnt*, em todos os casos.

Ao se analisar o número de mensagens trocadas, percebe-se uma diferença considerável entre os métodos. O algoritmo *EvAnt* sempre consegue disseminar todas as informações pela rede com um menor número de mensagens que o *flooding*, essa diferença chegando a ser de mais de 29%, conforme pode ser observado nas redes aleatórias de 2500 nodos. Em alguns casos, como para as redes *powerlaw* de 2500 nodos, o *gossip* troca menos mensagens que as outras duas estratégias. Porém este algoritmo não atingiu o objetivo de disseminar as informações para todos os nodos, como explicado acima. Ainda, em outras situações, como nas redes de topologia aleatória de 2500 nodos, o *gossip*, mesmo não sendo capaz de tornar toda a rede ciente das novidades, trocou mais mensagens do que o algoritmo *EvAnt*.

Esse é um forte indício da eficiência do algoritmo *EvAnt*, uma vez que ele é capaz de alcançar a mesma taxa de ciência do *flooding* com uma menor quantidade de mensagens disseminadas ao longo do processo.

Por fim, quanto ao número de mensagens criadas ao longo das simulações, há novamente um ganho significativo por parte do *EvAnt*. Isso ocorre pois, a cada vez que o *flooding* ou o *gossip* recebem uma mensagem que carrega novidades, eles necessitam criar *cópias* dessa mensagem para então espalhá-las pela rede. No algoritmo *EvAnt*, é criada apenas uma formiga por nodo, a cada rodada na qual a condição $C_i(t) \leq L_{min}(t)$ for atendida.

	powerlaw			topologia aleatória		
	flooding	*gossip*	*EvAnt*	*flooding*	*gossip*	*EvAnt*
latência	128.5	–	157.1	131.1	–	161.2
	0.527	–	3.178	0.737	–	1.98
mensagens trocadas	212591.8	97304.1	170525.2	211328.4	185029	148998.3
	2351.7	1942	4232.6	2377.9	1516.7	2820.335
mensagens criadas	212591.8	97304.1	1311.2	211328.4	185029	1098
	2351.7	1942	25.7	2377.9	1516.7	15.8
awareness rate	2500	2344.9	2500	2500	2497	2500
	0	10.5	0	0	0.44	0

Tabela 1. Resultados para as redes de 2500 nodos.

A fim de verificar estatisticamente se os resultados podem ser considerados re-

levantes, foi utilizado o *teste de Wilcoxon* [Wilcoxon 1945]. Esse teste é utilizado para verificar se um determinado conjunto de dados apresenta resultados significativamente diferentes em relação a outro. A comparação foi realizada apenas entre os algoritmos *EvAnt* e *flooding*, uma vez que, conforme já mostrado, o *gossip* não consegue tornar toda a rede ciente dos eventos. O teste de Wilcoxon confirmou que, para todas as métricas estudadas, com exceção da *awareness rate*, igual para ambos os métodos, os conjuntos de dados obtidos pelos algoritmos *EvAnt* e *flooding* são mesmo distintos.

Através da análise conjunta dos resultados do teste de Wilcoxon com a Figura 1, pode-se observar um claro caso de escolha conflitante (*trade-off*) entre velocidade e carga imposta à rede. De um lado, o *flooding* oferece rápida detecção de eventos porém com um alto número de mensagens trocadas e criadas ao longo do processo. Do outro lado há o algoritmo *EvAnt*, que, embora demore algumas rodadas a mais para tornar toda a rede ciente das novidades, realiza essa tarefa com uma menor quantidade de mensagens trocadas e com uma diferença muito expressiva no número de mensagens criadas.

4. Trabalhos relacionados

As duas estratégias utilizadas para comparação com a solução proposta são consideradas referência para a disseminação de eventos: a inundação (*flooding*) [Mullender 1993], que garante que as informações chegarão a todos os nodos da rede, e a disseminação epidêmica (*gossip*) [Eugster et al. 2004], que é uma alternativa menos custosa que o *flooding*, mas é probabilística, não garantindo que todos os nodos serão alcançados.

Alguns métodos têm utilizado inteligência coletiva para realizar determinadas tarefas em redes *ad hoc* e *peer-to-peer*. Aqui são citados aqueles que possuem maior relação com a estratégia proposta neste trabalho. Nenhum desses métodos é diretamente comparável pelas diferenças de asserções e/ou objetivos. Em [Brocco et al. 2009], com o objetivo de localizar recursos em *grid*, a disseminação de informações ocorre através de agentes móveis (formigas) especializadas em várias funções, tais como a manutenção de enlaces em situações de baixo tráfego. Um ponto importante a ser destacado neste trabalho é que ele também cria formigas quando o estado dos enlaces dos nodos é alterado. O algoritmo altera a topologia da rede, isto é, cria e remove enlaces, de modo a manter uma rede fracamente conexa.

O trabalho [Ahmed and Far 2007] também trata sobre disseminação de agentes em redes dinâmicas, porém sem utilizar a abordagem da Otimização por Colônia de Formigas. Segundo os autores, as estratégias baseadas em formigas até então apresentadas continuavam a disseminar agentes mesmo em situações nas quais não haviam novidades a serem disseminadas. Os agentes são hierarquizados em dois tipos. Um tipo de agente percorre a rede em busca de informações e as informa ao outro tipo. Há uma diferença conceitual neste trabalho, uma vez que existe a hierarquização dos nodos em dois tipos.

A disseminação de informações através de formigas é utilizada em [Aissani et al. 2007] para auxiliar no roteamento. Tabelas cache em cada nodo colaboram para as mensagens enviadas encontrarem o destino. Este trabalho também possui formigas especializadas em realizar determinadas tarefas. A circulação de pacotes na rede é permanente, com o objetivo de manter as tabelas cache dos nodos atualizadas.

O roteamento também é tratado em [Gopalan et al. 2006] por meio de agentes móveis, especificamente utilizando Otimização por Colônia de Formigas. Porém, este

trabalho possui um componente centralizador, uma vez que um determinado nodo é que concentra as informações sobre as rotas da rede.

Em [Wei et al. 2009] é proposto um protocolo para a descoberta da topologia em redes *ad hoc* através do uso de agentes móveis. Uma taxa de mudança dos vizinhos de um nodo foi estabelecida. Quanto maior for essa taxa, maior chance que os agentes terão de migrarem para aquele nodo. Por outro lado, nodos cuja vizinhança pouco muda ou muito visitados destroem os agentes que ali chegam.

O trabalho [Nassu et al. 2007] trata sobre a descoberta e disseminação da topologia de redes dinâmicas e descentralizadas através de agentes móveis, utilizando para isso inteligência coletiva, mais especificamente a Otimização por Colônia de Formigas. Informações contidas tanto pelos nodos quanto pelos agentes, após passarem de uma determinada idade, são descartadas. O algoritmo também dissemina formigas continuamente pela rede, em contraste com a proposta de disseminação de informações baseada em eventos. Além disso, os relógios locais de cada nodo são sincronizados entre si, de modo que a diferença entre dois instantes de tempo físico é a mesma para todos os nodos. Os conceitos de estigmergia e método de disseminação de informações apresentados pelo algoritmo *EvAnt* foram adaptados deste trabalho. Uma comparação direta dos dois métodos não foi possível pois cada estratégia baseia-se em um modelo de sistema diferente.

5. Conclusão

Este artigo apresentou o algoritmo *EvAnt*, uma estratégia para a disseminação de informações em redes dinâmicas e descentralizadas. Através de formigas que se deslocam pela rede, as informações são repassadas aos nodos. Cada nodo possui um depósito de feromônios relativo a cada enlace. Quanto maior a concentração de feromônios desse depósito, maior a frequência com que o enlace foi utilizado. A escolha dos destinos das formigas é tomada com base apenas em informações locais ao nodo no qual ela se localiza. Para que os agentes sejam distribuídos de maneira adequada na rede, os enlaces menos utilizados possuem maior probabilidade de serem escolhidos. Os depósitos de feromônios também são utilizados para controlar a população de formigas, de modo que não haja excesso ou falta de agentes.

Com o objetivo de não gastar recursos desnecessários da rede, a movimentação de agentes ocorre apenas quando há novidades a serem disseminadas. A estratégia mostrou-se eficiente e eficaz na tarefa à qual se propõe, uma vez que os experimentos mostraram que o algoritmo *EvAnt* é capaz de deixar toda a rede ciente dos eventos com uma menor quantidade de mensagens do que os métodos com o qual foi comparado.

Trabalhos futuros incluem considerar alterações em determinados pontos do modelo de sistema apresentado, como por exemplo considerar uma rede com enlaces direcionados. Uma implementação do algoritmo *EvAnt* para redes *peer-to-peer* deverá ser desenvolvida.

6. Agradecimentos

Este trabalho foi possível graças ao apoio do CNPq, projetos 304013/2009-9 e 303761/2009-1. O trabalho foi parcialmente financiado pela FINEP através dos projetos CT-INFRA/UFPR.

Referências

Ahmed, A. and Far, B. (2007). Performance of mobile agent based network topology discovery. In *Electrical and Computer Engineering, 2007. CCECE 2007. Canadian Conference on*, pages 66–69.

Aissani, M., Fenouche, M., Sadour, H., and Mellouk, A. (2007). Ant-dsr: Cache maintenance based routing protocol for mobile ad-hoc networks. In *Telecommunications, 2007. AICT 2007. The Third Advanced International Conference on*, pages 35–35.

Bollobas, B. (2001). *Random graphs*. Cambridge Univ. Press.

Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm intelligence: from natural to artificial systems*. Oxford University Press, USA.

Brocco, A., Malatras, A., and Hirsbrunner, B. (2009). Proactive information caching for efficient resource discovery in a self-structured grid. In *BADS '09: Proceedings of the 2009 workshop on Bio-inspired algorithms for distributed systems*, pages 11–18, New York, NY, USA. ACM.

Bu, T. and Towsley, D. (2002). On distinguishing between Internet power law topology generators. In *IEEE INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, volume 2.

Dong, C., rong Cheng, X., and quan Zhang, M. (2006). Research of mobile agent based network topology discovery. In *Innovative Computing, Information and Control, 2006. ICICIC '06. First International Conference on*, volume 1, pages 733–736.

Eugster, P., Guerraoui, R., Kermarrec, A., and Massoulie, L. (2004). Epidemic information dissemination in distributed systems. *Computer*, 37(5):60–67.

Gopalan, N., Mala, C., Shriram, R., and Agarwal, S. (2006). Multicast tree computation for group communication in mobile networks using optimization techniques. In *Ad Hoc and Ubiquitous Computing, 2006. ISAUHC '06. International Symposium on*, pages 88–93.

Mullender, S. (1993). *Distributed Systems*. Addison Wesley Publishing Company.

Nassu, B., Nanya, T., and Duarte, E. (2007). Topology discovery in dynamic and decentralized networks with mobile agents and swarm intelligence. *Proceedings of 7th ISDA, IEEE Computer Society, Washington*, pages 685–690.

Wei, J., Guo, W., Su, J., and Tang, W. (2009). Mobile agent based topology discovery in mobile ad hoc networks. In *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, pages 1–4.

White, T. and Pagurek, B. (1998). Towards multi-swarm problem solving in networks. In *Proceedings of Third International Conference on Multi-Agent Systems (ICMAS'98)*, pages 333–340.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83.

Novas Medidas de Relevância para Seleção *Lazy* de Atributos *

Douglas B. Pereira[1], **Alexandre Plastino**[1], **Rafael B. Pereira**[1]
Bianca Zadrozny[2], **Luiz Henrique de C. Merschmann**[3], **Alex A. Freitas**[4]

[1]Instituto de Computação – Universidade Federal Fluminense (UFF)
Niterói – RJ – Brasil

[2]IBM Research Brasil
Rio de Janeiro – RJ – Brasil

[3]Instituto de Ciências Exatas e Biológicas – Universidade Federal de Ouro Preto (UFOP)
Ouro Preto – MG – Brasil

[4]Computing Laboratory – University of Kent
Canterbury – Kent – United Kingdom

douglasuff@vm.uff.br, {plastino,rbarros}@ic.uff.br, biancaz@br.ibm.com
luizhenrique@iceb.ufop.br, a.a.freitas@kent.ac.uk

Abstract. *Attribute selection is a data preprocessing step used to identify attributes relevant to the classification task. Recently, a lazy technique which postpones the choice of attributes to the moment an instance is submitted to classification was proposed. In the original lazy technique proposal, a measure based on the entropy concept was presented to evaluate the quality of the attributes. In this work, we propose four new measures, based on: the chi-square statistic test, the Cramer coefficient, the Gini index and the gain ratio concept. Experimental results show the relevance of this proposal since, for a large number of datasets, the best performance of the lazy selection strategy was achieved when the new measures were used.*

Resumo. *Seleção de atributos é uma etapa de pré-processamento que identifica atributos relevantes para a tarefa de classificação. Recentemente, foi proposta uma técnica lazy que adia a escolha dos atributos ao momento em que uma instância é submetida à classificação. Na proposta da técnica lazy original, uma medida baseada no conceito de entropia foi apresentada para avaliar a qualidade dos atributos. Neste trabalho, propõem-se outras quatro medidas, baseadas: no teste estatístico chi-quadrado, no coeficiente de Cramer, no índice Gini e no gain ratio. Experimentos evidenciam a importância dessa proposta uma vez que, para um número significativo de bases de dados, o melhor desempenho da seleção lazy foi obtido com a utilização das novas medidas.*

1. Introdução

Diversos estudos evidenciam que o desempenho de estratégias de classificação está diretamente relacionado, entre outros fatores, à qualidade dos dados da base de treinamento [Guyon et al. 2006, Liu and Motoda 2008]. Atributos redundantes e irrelevantes

*Trabalho resultante de pesquisa parcialmente financiada pelo CNPq e pela FAPERJ.

podem, não somente prejudicar a acurácia de um classificador, mas tornar o processo de construção do modelo ou a execução do algoritmo de classificação mais lento. Técnicas de seleção de atributos visam eliminar da base de treinamento atributos que não contribuem ou mesmo prejudicam o desempenho de um classificador. Desse modo, para bases que contêm atributos redundantes ou irrelevantes, técnicas de seleção de atributos são fundamentais para melhorar a qualidade dos dados que serão utilizados pelo classificador.

Tradicionalmente, técnicas de seleção de atributos são executadas na fase de pré-processamento dos dados e suas decisões são definitivas para a fase de construção do modelo ou classificação propriamente dita. Porém, em [Pereira et al. 2008, Pereira et al. 2011], foi proposta uma técnica de seleção de atributos cuja característica principal é adiar a seleção dos atributos relevantes – de forma *lazy* – ao momento em que uma instância é submetida ao processo de classificação, ao invés de se fazer a seleção previamente – de forma *eager*. Essa proposta tem como hipótese que o conhecimento dos valores dos atributos da instância a ser classificada pode contribuir para a identificação dos melhores atributos para aquela instância em particular. Dessa forma, para diferentes instâncias a serem classificadas, subconjuntos distintos de atributos, e customizados para cada instância, poderão ser selecionados. Nessa proposta, foi utilizado, como medida de qualidade dos valores dos atributos, o conceito de entropia da distribuição de classes.

Neste trabalho, serão propostas e avaliadas quatro novas medidas de qualidade de atributos para serem utilizadas com a abordagem *lazy* de seleção de atributos. Essas novas medidas são baseadas nos seguintes conceitos: teste estatístico chi-quadrado [Liu and Setiono 1995, Han and Kamber 2006], no coeficiente de Cramer [Spiegel 1993], no índice Gini [Breiman et al. 1984, Han and Kamber 2006] e no *gain ratio* [Quinlan 1986, Han and Kamber 2006]. Cabe informar que a proposta baseada no teste estatístico chi-quadrado evoluiu de uma versão preliminar apresentada em [Menezes et al. 2009].

O principal objetivo deste trabalho é mostrar a importância da utilização de outras medidas de relevância de atributos, junto à abordagem de seleção *lazy*, diferentes daquela baseada no conceito de entropia, definida em [Pereira et al. 2008, Pereira et al. 2011]. Os experimentos permitem verificar que, considerando-se diferentes bases de dados, com características variadas, cada uma das medidas avaliadas – a original e as quatro novas – pode levar, isoladamente, a técnica de seleção *lazy* a atingir seu melhor desempenho.

O restante deste trabalho está organizado da seguinte forma. Na Seção 2, realiza-se uma revisão sobre seleção de atributos e medidas de relevância de atributos comumente utilizadas. A seleção *lazy* de atributos e a adaptação do conceito de entropia para o contexto *lazy* são revisadas na Seção 3. As quatro novas medidas para seleção *lazy* são propostas na Seção 4. Na Seção 5, são analisados os experimentos computacionais e, finalmente, na Seção 6, são apresentadas as conclusões e direções para trabalhos futuros.

2. Seleção de atributos

Estratégias de seleção de atributos são categorizadas como: embutidas, *wrapper* ou do tipo filtro [Guyon et al. 2006, Liu and Motoda 2008]. Técnicas embutidas aparecem incorporadas ao algoritmo de classificação e realizam a seleção de atributos no seu processo de treinamento. Um exemplo típico são os algoritmos de indução de árvores de decisão, pois selecionam os atributos que farão parte da árvore. Técnicas *wrapper* e filtro pro-

curam pelo subconjunto de atributos mais adequado que será utilizado pelo algoritmo de classificação. No caso da seleção *wrapper*, o próprio algoritmo de classificação é utilizado para avaliar a qualidade dos atributos. Técnicas filtro são independentes do algoritmo de classificação e avaliam a qualidade dos atributos por meio de medidas específicas.

Uma maneira de se medir a qualidade de um atributo para classificação é avaliar o seu grau de associação com a classe. Para tanto, existem diversas medidas, que serão descritas a seguir. Considere que essas medidas sejam empregadas em uma base de dados $D(A_1, A_2, ..., A_n, C), n \geq 1$, com $n+1$ atributos, onde C é o atributo classe, com domínio $\{c_1, c_2, ..., c_m\}, m \geq 2$. Assume-se que os valores dos atributos e da classe são discretos.

2.1. Entropia

O conceito de entropia tem origem na área de teoria da informação e pode ser especificado da seguinte forma. A entropia da distribuição de classes em D, representada por $Ent(D)$ é dada por [Han and Kamber 2006]:

$$Ent(D) = -\sum_{i=1}^{m}[p_i \times log_2(p_i)], \tag{1}$$

onde p_i, $1 \leq i \leq m$, é a probabilidade de ocorrência de cada valor c_i da classe C, dada pela razão entre o número de instâncias em que c_i ocorre e o número total de instâncias.

Seja A um atributo de D, com domínio $\{a_1, a_2, ..., a_k\}, k \geq 1$, e seja D_{a_i} a partição de D composta por todas as instâncias cujo valor de A é igual a a_i. A entropia da distribuição de classes em D, condicionada aos valores do atributo A, representada por $Ent(D, A)$, é definida por:

$$Ent(D, A) = \sum_{i=1}^{k}\left[\left(\frac{|D_{a_i}|}{|D|}\right) \times Ent(D_{a_i})\right]. \tag{2}$$

Quanto mais informativo um atributo A for em relação à classe C, menor será a entropia condicional $Ent(D, A)$.

2.2. Chi-quadrado

A medida Chi-quadrado (ou χ^2) avalia a qualidade de um atributo de acordo com a sua correlação com a classe por meio de um teste estatístico χ^2. Para cada valor a_i do atributo A ($1 \leq i \leq k$) e para cada valor c_j da classe C ($1 \leq j \leq m$), existe uma frequência esperada quando ($A = a_i$) e ($C = c_j$), que pode ser calculada pela fórmula [Han and Kamber 2006]:

$$e_{ij} = \frac{count(A = a_i) \times count(C = c_j)}{n}, \tag{3}$$

onde n é o número total de instâncias, $count(A = a_i)$ é o número de instâncias em que ocorre o valor a_i do atributo A, e $count(C = c_j)$ é o número de instâncias que pertencem à classe c_j. A partir da frequência esperada de todas as combinações de valores i e j, pode-se calcular a medida χ^2 pela fórmula:

$$\chi^2 = \sum_{i=1}^{k}\sum_{j=1}^{m}\left[\frac{(o_{ij} - e_{ij})}{e_{ij}}\right], \tag{4}$$

onde o_{ij} é a frequência observada da conjunção $(A = a_i) \wedge (C = c_j)$, ou seja, a razão entre o número de instâncias em que ocorre o valor a_i do atributo A simultaneamente com o valor c_j da classe C, e o número total de instâncias da base. O teste estatístico pode ser aplicado para determinar se o atributo A e a classe C são independentes, usando o número de graus de liberdade dado por $(k-1) \times (m-1)$, que é proporcional ao número de ocorrências distintas de valores de A multiplicado pela cardinalidade da classe C. Se essa hipótese puder ser rejeitada, de acordo com um nível de significância estatística pré-determinado e uma distribuição χ^2, significa que o atributo A é fortemente relacionado à classe C [Han and Kamber 2006].

2.3. Coeficiente de *Cramer*

O valor calculado pela medida Chi-quadrado está diretamente relacionado ao número de instâncias de uma base e à cardinalidade dos atributos. Já o coeficiente de *Cramer* não é afetado pelo tamanho da amostra, sendo muito útil quando se suspeita que um aumento significativo do Chi-quadrado é resultante do grande tamanho da amostra, em vez de uma relação entre os atributos. Esse coeficiente é interpretado como uma medida da correlação entre dois atributos e varia de 0 a 1. Quanto mais próximo de 0, menor a correlação e, quanto mais próximo de 1 maior a correlação. O valor do coeficiente de *Cramer* (V) de um atributo A é calculado pela fórmula [Spiegel 1993]:

$$V = \sqrt{\frac{\chi^2}{n(q-1)}}, \tag{5}$$

onde χ^2 é o valor de Chi-quadrado de A, n o é total de instâncias da base e q é o mínimo entre o número de valores distintos do atributo e a cardinalidade da classe.

2.4. Índice *Gini*

O índice *Gini* é uma medida estatística de desigualdade (ou impureza), comumente utilizado para calcular a desigualdade de distribuição de renda. É utilizado na seleção de atributos no algoritmo CART [Breiman et al. 1984] de árvores de decisão, com o objetivo de encontrar o atributo que contém o melhor particionamento de valores. Esse índice é máximo (pior) quando as instâncias de uma determinada partição estão igualmente distribuídas entre todas as classes, e o índice é mínimo (melhor) quando todas as instâncias pertencem a uma única classe. O índice *Gini* de cada valor do atributo A é dado por:

$$Gini(D, A, a_i) = 1 - \sum_{j=1}^{m} (p_{j|i})^2, \tag{6}$$

onde m é o número de valores distintos da classe e $p_{j|i}$ é a probabilidade da instância pertencer à classe c_j $(1 \leq j \leq m)$ quando ocorre o valor a_i em A, dada pela razão entre o número de instâncias da classe c_j em que a_i ocorre e o número de ocorrências de a_i.

O índice *Gini* do atributo A é dado por [Breiman et al. 1984]:

$$Gini(D, A) = \sum_{i=1}^{k} p_i \times Gini(D, A, a_i), \tag{7}$$

onde k é o número de valores distintos do atributo A, p_i $(1 \leq i \leq k)$ a razão entre o número de instâncias da base em que ocorre o valor a_i para o atributo e o número total de instâncias e $Gini(D, A, a_i)$ é dado pela Equação 6.

2.5. *Gain ratio*

A medida *Gain ratio* [Quinlan 1986] é baseada no conceito de entropia, mais especificamente, no ganho de informação, ou seja, na diferença entre a entropia da classe e a entropia da classe condicionada a determinado atributo. Na tentativa de não priorizar atributos que têm muitos valores diferentes, a medida faz um tipo de normalização, que é a divisão do ganho de informação pelo $SplitInfo(D, A)$, definido por [Han and Kamber 2006]:

$$SplitInfo(D, A) = -\sum_{k=1}^{i} w_i \times log_2(w_i), \tag{8}$$

onde w_i pode ser descrito como o peso da partição que contém os valores a_i do atributo A, dado pela razão entre número de ocorrências de a_i e o total de instâncias da base.

O *Gain ratio* pode ser obtido através da fórmula [Han and Kamber 2006]:

$$GainRatio(D, A) = \frac{Ent(D) - Ent(D, A)}{SplitInfo(D, A)}, \tag{9}$$

onde $Ent(D)$ é a entropia da distribuição de classes em D (Equação 1) e $Ent(D, A)$ é a entropia de classe em D condicionada ao atributo A (Equação 2).

3. Seleção *lazy* de atributos

Estratégias convencionais de seleção de atributos – que neste trabalho serão referenciadas como estratégias de seleção *eager* – são executadas na etapa de pré-processamento e os atributos não selecionados são descartados da base de dados, não participando do processo de classificação. Proposta em [Pereira et al. 2008], a estratégia de seleção de atributos *lazy* baseia-se na hipótese de que o conhecimento dos valores dos atributos da instância a ser classificada pode contribuir para o processo de escolha dos atributos da base mais adequados para a classificação dessa instância em particular. A seleção de atributos, para cada instância, é, portanto, realizada apenas no momento da sua classificação. Para diferentes instâncias, subconjuntos distintos de atributos podem ser selecionados.

Essa proposta de seleção *lazy* de atributos pode ser aplicada em conjunto com qualquer classificador do tipo *lazy*, tais como o k-NN e o Naive Bayes, ou com qualquer versão *lazy* de classificadores tradicionalmente *eager*, como por exemplo as árvores de decisão *lazy* [Friedman et al. 1996].

O conceito definido na Equação 2 é usado pela estratégia *eager* de seleção de atributos conhecida como *Information Gain Attribute Ranking* [Yang and Pedersen 1997] para medir a capacidade de um atributo discriminar os valores da classe. Para a seleção *lazy* de atributos, proposta em [Pereira et al. 2008, Pereira et al. 2011], define-se, então, a entropia da distribuição de classes em D, restrita ao valor de a_i do atributo A, representada por $Ent(D, A, a_i)$, da seguinte forma:

$$Ent(D, A, a_i) = Ent(D_i), \tag{10}$$

onde a entropia $Ent(D_i)$ pode ser obtida através da Equação 1.

A Equação 10 é usada pela estratégia de seleção *lazy* para medir a capacidade de um valor especifico a_i, de um atributo particular A, de discriminar as classes. Quanto

mais próxima a entropia $Ent(D, A, a_i)$ é de zero, maior é a chance de o valor de a_i do atributo A determinar bem alguma classe.

Os parâmetros de entrada da estratégia *lazy* são: uma base de dados D com n atributos, uma instância $I = (v_1, v_2, ..., v_n)$ e um número $r, 1 \leq r \leq n$, que representa o número de atributos a serem selecionados. Para selecionar os r melhores atributos para classificar a instância I, os n atributos são avaliados baseados na medida *lazy* proposta em [Pereira et al. 2008, Pereira et al. 2011] (Equação 11) que estabelece que, para cada atributo A_j $(1 \leq j \leq n)$, se a capacidade discriminatória do valor específico v_j de A_j $(Ent(D, A_j, v_j))$ é melhor que (menor que) a capacidade de discriminação geral do atributo A_j $(Ent(D, A_j))$, então a primeira será considerada como medida *lazy* de A_j. Dessa forma, a medida proposta para avaliar a qualidade de cada atributo A_j é definida por:

$$Ent_{Lazy}(D, A_j, v_j) = Min(Ent(D, A_j, v_j), Ent(D, A_j)), \tag{11}$$

onde Min() retorna o menor entre seus parâmetros.

Após calcular o valor $Ent_{Lazy}(D, A_j, v_j)$ para cada atributo A_j, a estratégia *lazy* selecionará os r atributos que apresentem os r menores valores dessa medida.

O desempenho dessa proposta foi avaliado quando utilizada em conjunto com o algoritmo de classificação *k-NN* [Dasarathy 1991]. A partir de uma base de dados D, uma instância I e um valor de k, sendo $k \geq 1$, basicamente, o algoritmo *k-NN* atribui a I a classe majoritária entre os seus k elementos mais próximos. A distância entre I e as instâncias de D é calculada a partir de uma função definida sobre os valores dos atributos das respectivas instâncias. Dessa forma, a utilização da seleção *lazy* de atributos, em conjunto com o algoritmo *k-NN*, por exemplo, implica que o cálculo da distância entre instâncias poderá ser realizado utilizando-se diferentes subconjuntos de atributos, para diferentes instâncias de entrada.

A seleção de atributos *lazy* baseada no conceito de entropia apresentou resultados interessantes, demonstrando ser uma técnica bastante promissora. Para permitir uma avaliação mais abrangente, tentando dar evidências de que a técnica *lazy* pode ter um bom desempenho independentemente do tipo de medida utilizada, neste trabalho, propõem-se quatro novas medidas de relevância de atributos, que serão apresentadas a seguir.

4. Novas medidas para seleção lazy

As quatro novas medidas para seleção *lazy* de atributos, propostas neste trabalho, estão baseadas nos seguintes conceitos: teste estatístico chi-quadrado [Liu and Setiono 1995, Han and Kamber 2006], o coeficiente de Cramer [Spiegel 1993], o índice Gini [Breiman et al. 1984, Han and Kamber 2006] e a medida gain ratio [Quinlan 1986, Han and Kamber 2006].

4.1. Chi-quadrado *lazy*

No caso da seleção *lazy* de atributos, a medida chi-quadrado deve levar em consideração a capacidade de cada valor da instância a ser classificada determinar o atributo classe. Portanto faz-se necessária uma adaptação das Equações 3 e 4 para que o valor χ^2 possa ser utilizado também na abordagem *lazy* de seleção de atributos. Neste trabalho, propõe-se a Equação 12 que avalia o quanto o valor a_{ji} do atributo A_j está correlacionado com o

atributo classe C:

$$\chi^2(D, A_j, v_j) = \sum_{k=1}^{m} \frac{(o_k - e_k)}{e_k}, \tag{12}$$

onde m é o número de valores distintos do atributo classe C, o_k é a frequência observada do par $\{A_j = v_j, C = c_k\}$ e e_k é a frequência esperada do mesmo par. Quanto maior for o valor de χ^2, maior a chance de a_{ji} ser um bom determinante de uma classe.

A frequência esperada do par $\{A_j = v_j, C = c_k\}$ é definida na Equação

$$e_k = \frac{count(A_j = v_j) \times count(C = c_k)}{N}, \tag{13}$$

onde $count(A_j = v_j)$ é o número de instâncias em que ocorre o valor v_j do atributo A_j, $count(C = c_k)$ é o número de instâncias que pertencem à classe c_k e N é o número total de instâncias da base.

A seleção dos r melhores atributos para classificar a instância I é realizada a partir da avaliação dos n atributos pela medida χ^2_{Lazy} definida pela Equação 14, onde χ^2 é o valor de χ^2 calculado para o atributo A_j (Equação 4) e Max() é a função que retorna o maior valor entre seus parâmetros:

$$\chi^2_{Lazy}(D, A_j, v_j) = Max\left(\frac{\chi^2}{n_j}, \chi^2(D, A_j, v_j)\right), \tag{14}$$

onde n_j que é o número de valores distintos de A_j. Quanto maior o valor de χ^2_{Lazy}, maior a correlação entre o valor v_j da instância a ser classificada e o atributo classe C.

Como o valor de χ^2 é obtido a partir do acumulo dos n_j valores do domínio do atributo A_j e o valor de $\chi^2(D, A_j, v_j)$ é obtido por uma única parcela desta soma (referente ao valor v_j), existe a necessidade de torná-los comparáveis. Para tanto, foi introduzido o denominador n_j. Após calcular o valor χ^2_{Lazy} para os n atributos, a estratégia *lazy* selecionará os r atributos que apresentam os r maiores valores de χ^2_{Lazy}.

4.2. Coeficiente de *Cramer lazy*

A proposta de adaptação do coeficiente de Cramer para a abordagem *lazy* é representada pela Equação 15 que avalia o quanto o valor v_j do atributo A_j está correlacionado com o atributo classe C:

$$V(D, A_j, v_j) = \sqrt{\frac{\chi^2(D, A_j, v_j)}{N(q-1)}}, \tag{15}$$

onde $\chi^2(D, A_j, v_j)$ é o valor calculado de Chi-quadrado para o valor v_j do atributo A_j dado pela Equação 12, N o é total de instâncias e q é o mínimo entre o número de valores distintos do atributo A_j e a cardinalidade da classe C. Quanto maior for o valor de $V(D, A_j, v_j)$, maior a chance de v_j ser um bom determinante de uma classe.

A seleção dos r melhores atributos para classificar a instância I é realizada a partir da avaliação dos n atributos de acordo com a medida V_{Lazy} definida pela Equação 16, onde χ^2 é o valor de χ^2 calculado para o atributo A_j (Equação 4).

$$V_{Lazy}(D, A_j, v_j) = Max\left(\sqrt{\frac{\chi^2/n_j}{N(q-1)}}, V(D, A_j, v_j)\right) \tag{16}$$

Quanto maior o valor de V_{Lazy}, maior correlação entre o valor v_j e o atributo classe C.

Houve a necessidade de introduzir o denominador n_j na parcela referente ao valor de V obtido pela técnica *eager* pelo mesmo motivo apresentado no caso da medida Chi-quadrado *lazy*. Após calcular o valor V_{Lazy} para os n atributos, a estratégia *lazy* selecionará os r atributos que apresentam os r maiores valores de V_{Lazy}.

4.3. Índice *Gini lazy*

A adaptação *lazy* proposta neste trabalho para o índice Gini é definida por:

$$Gini(D, A_j, v_j) = 1 - \sum_{k=1}^{m} (p_k)^2, \tag{17}$$

onde m é o número de valores distintos do atributo classe, p_k é a razão entre o número de instâncias da base que pertencem à classe c_k em que ocorre o valor v_j, e o número de ocorrências do valor v_j do atributo A_j.

A seleção dos r melhores atributos para classificar a instância I é realizada a partir da avaliação dos n atributos de acordo com a medida $Gini_{Lazy}$ definida pela Equação 18, onde $Gini(D, A_j)$ é o índice *Gini* calculado para o atributo A_j (Equação 7).

$$Gini_{Lazy}(D, A_j, v_j) = Min(Gini(D, A_j), Gini(D, A_j, v_j)) \tag{18}$$

Quanto menor o valor de $Gini_{Lazy}$ menor a desigualdade na relação dos valores do atributo A_j com os da classe C.

Após calcular o valor $Gini_{Lazy}$ para os n atributos, a estratégia *lazy* selecionará os r atributos que apresentam os r menores valores de $Gini_{Lazy}$.

4.4. *Gain ratio lazy*

No caso da estratégia *lazy*, a adaptação proposta para o cálculo do $SplitInfo(D, A_j, v_j)$ é definida por:

$$SplitInfo(D, A_j, v_j) = -log_2(w_j), \tag{19}$$

onde w_i é definido pela razão entre número de ocorrências de v_j no atributo A_j e o total de instâncias da base.

A partir das definições anteriores, pode-se calcular o valor de $GainRaito(D, A_j, v_j)$ a partir da Equação:

$$GainRatio(D, A_j, v_j) = \frac{Ent(D) - Ent(D, A, v_j)}{Split(D, A_j, v_j)}. \tag{20}$$

A seleção dos r melhores atributos para classificar a instância I é realizada a partir da avaliação dos n atributos pela medida $GainRatio_{Lazy}$ definida pela Equação 21.

$$GainRatio_{Lazy}(D, A_j, v_j) = Max(GainRatio(D, A_j), GainRatio(D, A_j, v_j)) \tag{21}$$

Um maior valor de $GainRatio_{Lazy}$ indica um aumento no ganho de informação da base obtido pela escolha do atributo A_j considerando o valor v_j.

Após calcular o valor $Gain_{Lazy}$ para os n atributos, a estratégia *lazy* selecionará os r atributos que apresentam os r maiores valores de $Gain_{Lazy}$.

5. Resultados Experimentais

Para realizar os experimentos computacionais, a estratégia de seleção *lazy* foi implementada utilizando cada uma das medidas propostas e integrada à ferramenta Weka (Waikato Environment for Knowledge Analysis) [Waikato 2009]. O efeito da seleção de atributos foi avaliado utilizando-se o classificador *k-NN*. A seleção ocorre quando o classificador recebe uma nova instância a ser classificada. Uma vez que, para cada nova instância, um subconjunto distinto de atributos pode ser considerado pelo classificador, os atributos não selecionados pela estratégia *lazy* para uma dada instância não são removidos do conjunto de dados, mas somente desconsiderados pelo classificador.

Os experimentos foram realizados com 40 bases de dados, que possuem uma ampla variedade de tamanho, complexidade e áreas de aplicação. Essas bases foram retiradas do repositório da UCI [Asuncion and Newman 2007]. O valor ausente de um atributo A em uma instância t foi substituído pela moda dos valores de A das instâncias da classe de t, no caso de A categórico, ou pela média dos valores de A das instâncias da classe de t, no caso de A contínuo. As medidas utilizadas para avaliar a qualidade dos atributos necessitam de valores de atributos discretos. Por esse motivo foi adotado um método supervisionado de discretização baseado em ganho de informação, proposto em [Fayyad and Irani 1993], para discretizar os atributos contínuos dessas bases. Sem atributos contínuos, não houve a necessidade do processo de normalização de atributos.

O algoritmo de classificação utilizado foi o *k-NN*, especificamente, a implementação da ferramenta Weka conhecida como *IBk*. Cabe ressaltar que o código do *k-NN* teve que ser adaptado para considerar, no cálculo da distância entre elementos, apenas os atributos selecionados de forma *lazy*.

Os resultados obtidos estão sumarizados na Tabela 1. A seleção *lazy* de atributos foi utilizada com o classificador *k-NN*, com k igual a 1, para as 40 bases de dados, em conjunto com cada uma das cinco medidas: ENT (entropia), CHI (chi-quadrado), CRV (coeficiente de Cramer), GIN (índice Gini) e GRT (*gain ratio*). Na primeira coluna, estão apresentados o nome da base de dados e, entre parênteses, respectivamente, o número de atributos, o número de classes e o número de instâncias da base. A última coluna representa a execução do classificador sem a execução da seleção de atributos, ou seja, com todos os atributos da base de dados. Para cada base de dados, o melhor comportamento está indicado por valores em negrito. Caso represente um melhor desempenho obtido isoladamente, o valor estará sublinhado.

Cada valor da tabela representa a melhor acurácia obtida pelo classificador, quando utilizada a seleção *lazy* e a medida em questão, ao se variar o percentual de atributos selecionados de 10% a 90% do total de atributos, com um incremento regular de 10%. Cada valor de acurácia é obtido a partir da média das acurácias resultantes do processo de validação cruzada com dez partições [Han and Kamber 2006], na qual cada partição foi obtida de maneira aleatória. A acurácia correspondente a cada execução do classificador é uma média dos valores obtidos em cada partição. As mesmas partições foram utilizadas nas execuções com as diferentes medidas.

Inicialmente, observa-se a importância da seleção de atributos: para apenas uma base, a *pendigits*, o desempenho do classificador sem seleção foi o melhor obtido sem que nenhuma outra estratégia tivesse levado ao mesmo valor. Para as outras 39 bases, a

Tabela 1. Comparativo entre medidas lazy usando 1-NN

Base	ENT	CHI	CRV	GIN	GRT	Sem sel.
anneal (38,5,898)	**99,55**	99,44	99,22	99,44	**99,55**	99,22
audiology (69,24,226)	76,99	78,32	77,88	77,88	**78,76**	76,11
autos (25,6,205)	87,32	**89,27**	86,83	87,80	88,78	85,85
breast-cancer (9,2,286)	74,13	**75,52**	**75,52**	73,08	73,43	69,93
breast-w (9,2,699)	**97,14**	**97,14**	**97,14**	97,00	95,99	**97,14**
chess-Kr-vs-Kp (36,2,3196)	**96,81**	96,50	96,50	96,75	96,21	96,56
credit-a (15,2,690)	85,51	**86,38**	**86,38**	85,22	86,23	82,32
diabetes (8,2,768)	**77,99**	74,61	74,61	68,49	73,05	76,43
flags (29,8,194)	60,82	60,82	61,34	61,34	**65,98**	59,79
glass (9,6,214)	**77,10**	**77,10**	**77,10**	**77,10**	**77,10**	**77,10**
heart-cleveland (13,2,303)	**82,84**	**82,84**	**82,84**	82,51	81,52	80,53
heart-hungarian (13,2,294)	81,63	82,65	82,65	81,63	**83,67**	80,27
hepatitis (19,2,155)	86,45	85,81	85,81	86,45	**87,10**	83,87
horse-colic (27,2,368)	83,70	**84,78**	**84,78**	84,24	**84,78**	78,53
hypo-thyroid (29,4,3772)	**96,98**	93,61	93,61	93,19	93,45	91,52
ionosphere (34,2,351)	**94,59**	93,45	93,45	94,30	93,16	92,59
labor (16,2,57)	**100,00**	98,25	98,25	**100,00**	**100,00**	96,49
letter-recog (16,26,20000)	91,93	**92,09**	90,06	91,95	91,86	91,87
lymph (18,4,148)	**85,14**	83,78	**85,14**	84,46	84,46	82,43
mol-bio-promot (57,2,106)	**89,62**	87,74	87,74	88,68	86,79	80,19
mol-bio-splice (60,3,3190)	90,72	88,21	88,21	87,49	**90,88**	73,32
mushroom (22,2,8124)	**100,00**	**100,00**	**100,00**	**100,00**	**100,00**	**100,00**
optdigits (64,10,5620)	94,84	94,73	94,73	**94,86**	94,52	94,25
pendigits (16,10,10992)	96,79	96,63	96,55	96,71	96,74	**97,05**
postoperative (8,3,90)	**71,11**	70,00	70,00	**71,11**	**71,11**	63,33
primary-tumor (17,21,339)	42,48	**43,07**	40,71	41,30	41,89	38,35
solar-flare1 (12,6,323)	71,52	70,90	70,28	**72,14**	70,59	65,94
solar-flare2 (12,6,1066)	**76,27**	75,89	74,95	**76,27**	75,05	73,45
sonar (60,2,208)	**86,54**	82,21	82,21	82,21	83,17	**86,54**
soybean-large (35,19,683)	**93,41**	91,80	92,09	93,27	91,51	92,24
spambase (57,2,4601)	93,68	93,61	93,61	**93,72**	**93,72**	92,98
statlog-heart (13,2,270)	**85,19**	84,07	84,07	**85,19**	84,07	84,07
statlog-segment (19,7,2310)	94,68	94,68	94,33	94,68	**94,76**	94,68
statlog-vehicle (18,4,846)	71,39	**71,87**	70,80	71,28	70,33	70,92
thyroid-sick (29,2,3772)	97,48	**97,83**	**97,83**	97,59	97,56	97,45
vote (16,2,465)	**96,09**	95,63	95,63	**96,09**	95,17	92,18
vowel (13,11,990)	89,80	89,80	88,18	**89,90**	89,80	89,80
waveform-5000 (40,3,5000)	75,52	75,54	75,54	**76,34**	75,86	73,82
wine (13,3,178)	**98,88**	**98,88**	97,75	**98,88**	**98,88**	98,31
zoo (17,7,101)	97,03	**98,02**	96,04	96,04	97,03	96,04
Num.Vitórias	19	14	9	13	14	5
Num.Vitórias.Isoladas	6	5	0	4	6	1

seleção de atributos com alguma das medidas obteve resultado melhor do que sem seleção em 35 bases, ou resultado igual ao do classificador sem seleção em quatro bases.

Na base da tabela, a linha com o rótulo *Num.Vitórias* representa o número de vezes em que a medida em questão levou o classificador a obter o melhor desempenho. A linha *Num.Vitórias.Isoladas* representa o número de vezes em que a medida levou o classificador ao melhor desempenho isoladamente, isto é, sem que nenhuma outra medida

obtivesse o mesmo desempenho.

O objetivo principal deste trabalho era evidenciar que a seleção de atributos *lazy* pode se beneficiar de outras medidas de relevância de atributo além da medida ENT, proposta originalmente. Observa-se então que a medida ENT, apesar de ter levado o classificador a obter o melhor desempenho em 19 bases, levou ao melhor desempenho isoladamente em apenas seis bases de dados. Para outras 15 bases, o melhor desempenho foi obtido isoladamente com a utilização de uma das três medidas: CHI (cinco bases), GIN (quatro bases) ou GRT (seis bases). Verifica-se ainda que em outras cinco bases (*breast-cancer*, *credit-a*, *horse-colic*, *spambase* e *thyroid-sick*) duas ou mais das novas medidas obtiveram resultado melhor do que a medida ENT. Observa-se que apenas a medida CRV não foi capaz de obter um melhor desempenho de forma isolada e contribuir para a confirmação da importância da existência de medidas alternativas à ENT.

Para algumas bases de dados, a acurácia obtida por uma das medidas propostas foi significativamente maior do que a obtida com a medida ENT. Por exemplo, as diferenças absolutas entre as acurácias obtidas pelas medidas GRT e ENT, para as bases *audiology*, *flags* e *heart-hungarian* foram, respectivamente: 1,77; 5,16 e 2,04.

Uma vez que se tenha uma base de treinamento sobre a qual será realizada a tarefa de classificação, a disponibilidade de um conjunto amplo de medidas permite que se faça uma avaliação, através de técnicas como validação cruzada ou *leave-one-out*, da medida mais adequada à base e, dessa forma, se obtenham melhores acurácias preditivas.

Vale informar que experimentos também foram realizados utilizando-se o classificador *k-NN* com os valores de k igual a 3 e 5. Porém, como o desempenho do *k-NN* com k igual a 1 foi, de maneira geral, superior ao das outras duas parametrizações e essas não permitiram conclusões adicionais, por economia de espaço, decidiu-se apresentar apenas a análise dos melhores resultados obtidos.

6. Conclusão

Neste trabalho, propôs-se um novo conjunto de medidas para seleção *lazy* de atributos baseadas no teste chi-quadrado, no coeficiente de *Cramer*, no índice *gini* e na medida *gain ratio*. O objetivo principal era disponibilizar diferentes medidas de relevância que pudessem ser adotadas pela abordagem de seleção *lazy*, além da originalmente definida em [Pereira et al. 2008, Pereira et al. 2011], baseada no conceito de entropia.

As avaliações das novas medidas foram realizadas a partir de 40 bases de domínio público adotadas frequentemente em experimentos de mineração de dados, obtidas do repositório da UCI [Asuncion and Newman 2007].

Os resultados experimentais mostraram que, para um número significativo de bases, as medidas propostas levaram o classificador a obter as melhores acurácias. Com isso, o paradigma de seleção *lazy* de atributos conta agora com um número maior de medidas de relevância de atributos, permitindo que, para diferentes bases de dados, medidas distintas e mais apropriadas sejam adotadas.

Pretende-se, mais adiante, adaptar, para o contexto de seleção *lazy*, medidas que avaliam subconjuntos de atributos, tais como o *Correlation-based Feature* [Hall 2000] e o *Consistency-based Feature* [Liu and Setiono 1996], que medem a capacidade de um determinado conjunto de atributos discriminar os valores do atributo classe.

Referências

Asuncion, A. and Newman, J. (2007). Uci machine learning repository. http://www.ics.uci.edu/ mlearn/mlrepository.html.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and regression trees*. Wadsworth & Brooks/Cole Advanced Books & Software.

Dasarathy, B. V. (1991). *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press.

Fayyad, U. M. and Irani, K. B. (1993). Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, pages 1022–1029.

Friedman, J., Kohavi, R., and Tun, Y. (1996). Lazy decision trees. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI'96)*, pages 717–724.

Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L., editors (2006). *Feature Extraction, Foundations and Applications*. Springer.

Hall, M. A. (2000). Correlation-based feature selection for discrete and numeric class machine learning. In *Procs. of 17th Intl. Conf. on Machine Learning*, pages 359–366.

Han, J. and Kamber, M. (2006). *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2nd edition.

Liu, H. and Motoda, H. (2008). *Computational Methods of Feature Selection*. Chapman & Hall/CRC.

Liu, H. and Setiono, R. (1995). Chi2: Feature selection and discretization of numeric attributes. In *7th Intl. Conference on Tools with Artificial Intelligence*, pages 388–391.

Liu, H. and Setiono, R. (1996). A probabilistic approach to feature selection: A filter solution. In *Procs. of the 13th Intl. Conference on Machine Learning*, pages 319–327.

Menezes, R., Plastino, A., Zadrozny, B., Pereira, R., Merschmann, L. H., and Freitas, A. (2009). Avaliação de uma nova medida para seleção *lazy* de atributos baseada no teste chi-quadrado. In *Anais do V Workshop em Algoritmos e Aplicações de Mineração de Dados (WAAMD 2009/SBBD 2009)*, pages 58–65.

Pereira, R., Plastino, A., Zadrozny, B., Merschmann, L., and Freitas, A. (2008). Seleção lazy de atributos – uma nova perspectiva. In *Anais do IV Workshop em Algoritmos e Aplicações de Mineração de Dados (WAAMD 2008/SBBD 2008)*, pages 1–9.

Pereira, R., Plastino, A., Zadrozny, B., Merschmann, L., and Freitas, A. (2011). Lazy attribute selection – choosing attributes at classification time. *Intelligent Data Analysis*, to appear.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1:81–106.

Spiegel, M. R. (1993). *Estatística*. Makron Books.

Waikato (2009). Weka (waikato environment for knowledge analysis) machine learning project http://www.cs.waikato.ac.nz/ml/weka/.

Yang, Y. and Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In *Procs. of the 14th Intl. Conf. on Machine Learning*, pages 412–420.

Caracterização de Classes via Otimização em Redes Complexas

Lilian Berton[1], Liang Zhao[2]

[1,2]Instituto de Ciências Matemáticas e de Computação (ICMC) – Universidade de São Paulo (USP)
Caixa Postal 668 – 13560-970 – São Carlos – SP – Brasil

{lberton,zhao}@icmc.usp.br

Abstract. *Complex networks have emerged as an important way of data representation and abstraction able of capturing the topological structure presented in databases. This work proposes a method for building a network from a vector based dataset. It is based on the optimization of an energy function that considers purity and extension measures of the network. The constructed network was used to characterize mixing level among data classes in classification problem. Class characterization is an important issue, but it is not well studied. Therefore, we consider this work a contribution in this direction.*

Resumo. *As redes complexas surgiram como uma importante maneira de representação e abstração de dados capaz de capturar as relações topológicas presentes em bases de dados. Este trabalho propõe um método para construção de rede a partir de uma base de dados vetorial, o qual é baseado na otimização de uma função de energia que considera medidas de pureza e extensão da rede. A rede construída foi utilizada para caracterizar mistura entre classes de dados em problema de classificação de dados. A caracterização de classes é uma questão importante na classificação de dados, porém ainda é pouco estudada. Desta forma, consideramos este trabalho uma contribuição nesta direção.*

1. Introdução

Nos últimos anos as redes complexas têm se destacado fazendo com que o interesse nesta área aumentasse. As redes complexas são grafos de grande escala que possuem padrões de conexão não trivial. Redes como a Internet [Faloutsos et al. 1999], a *World Wide Web* [Albert et al. 1999], redes de interações sociais entre indivíduos [Scott 2000], redes neurais biológicas [Sponrs 2002], entre outras, têm sido estudadas sob essa abordagem. As redes complexas têm se mostrado uma maneira interessante para representação e abstração de dados, mudando a maneira com que vários sistemas interconectados são modelados [Bornholdt e Schuster 2003].

Tradicionalmente as redes complexas eram descritas de acordo com o modelo de Erdös e Rényi (1959), conhecido como Redes Randômicas. No entanto, Watts e Strogatz (1998) descobriram que a média de caminhos mais curtos em uma rede pode ser reduzida por alteração aleatória de poucas ligações em uma rede regular. A rede resultante foi denominada Rede de Pequeno Mundo. Posteriormente, Barabási e Albert (1999) descobriram que muitas redes reais têm distribuição de grau dos nós que obedece a lei de potência: $P(k) \sim k^{-\gamma}$, onde k é o número de conexões de um nó escolhido

aleatoriamente e γ é o expoente de escala, o que significa que existe um pequeno conjunto de nós que possui um grande número de ligações (*hubs*) e um grande número de nós com poucas ligações. Estas redes são denominadas Redes Livre de Escala.

Métodos de agrupamento de dados baseados em redes complexas, também conhecidos como detecção de comunidades, têm sido extensivamente explorados na literatura. Tais comunidades podem ser definidas como grupos de vértices da rede densamente conectados, enquanto que conexões entre vértices de grupos diferentes são esparsas [Newman e Girvan 2004]. Muitas técnicas foram desenvolvidas baseadas em diversas ideias para detecção de comunidades. Alguns exemplos podem ser encontrados em: [Zhou 2003a; Zhou 2003b; Newman 2004; Newman e Girvan 2004; Reichardt e Bornholdt 2004; Boccaletti et al. 2007; Quiles et al. 2008]. De maneira geral, a estrutura em comunidades revela similaridade por meio de conexões entre os vértices pertencentes a um mesmo grupo. Essas similaridades, por sua vez, podem revelar grupos nos dados em problemas de agrupamento e, de maneira análoga, podem evidenciar classes em problemas de classificação. Desse modo, as redes complexas também estão sendo usadas com sucesso na classificação de dados.

A classificação de dados lida com a detecção automática de padrões em conjuntos de dados. Por padrões entendem-se relações ou estruturas inerentes a alguns conjuntos de dados [Mitchell 1997]. Com a detecção de padrões significantes nos dados disponíveis (conjunto de treinamento), espera-se que um classificador possa realizar predições de classes em novos dados de entrada. Existem muitos problemas importantes que podem ser resolvidos utilizando-se dessa abordagem, abrangendo diversas áreas como bioinformática [Baldi e Brunak 1998], mineração de dados [Cook e Holder 2000], reconhecimento de escrita [LeCun et al. 1989], entre outras.

O desempenho de um classificador depende de diferentes fatores. Um fator muito importante se refere às características dos dados a serem classificados. Na prática nenhum classificador é o melhor em todos os problemas dados, fenômeno que pode ser explicado pelo teorema *no free lunch* [Worlpert e Macready 1997]. Como nem sempre se tem conhecimento prévio sobre a distribuição de um conjunto de dados ou sobre as características das classes, neste trabalho propõe-se um método que auxilie na caracterização de classes de dados. Para isso, uma rede é construída a partir de uma base de dados vetorial, sendo a mesma o resultado do processo de otimização de uma função de energia que considera medidas de pureza e extensão da rede. Para validar a proposta, executou-se o algoritmo em alguns cenários de dados artificiais e reais, buscando analisar a mistura nas classes de dados.

A seguir, na Seção 2 são apresentados alguns trabalhos relacionados. Na Seção 3 é descrito o método proposto, apresentando a função de energia utilizada, as medidas de extensão e pureza e o algoritmo com detalhes da implementação. Na Seção 4 são mostrados os resultados obtidos a partir da aplicação do método em redes artificiais e reais, bem como análise dos mesmos. Por fim, na Seção 5 são apresentadas as principais conclusões do trabalho e perspectivas de trabalhos futuros.

2. Trabalhos relacionados

Esta Seção descreve alguns dos métodos utilizados para construção de redes a partir de dados, com enfoque para as redes *K*-associados e sobre um processo de otimização que inspirou o desenvolvimento deste trabalho.

Construção de redes a partir de dados no formato atributo-valor

A construção de redes a partir de dados é uma fase importante no aprendizado baseado em grafos, as abordagens mais conhecidas são baseadas em relações de vizinhança, como grafos *KNN* [Zhu 2005] e ϵ [Belkin e Niyougi 2003]. Nos grafos *KNN* um parâmetro *K* é usado para controlar o número de vértices vizinhos que serão utilizados nas conexões. Nos grafos ϵ, as conexões são baseadas num raio ϵ, de modo que um vértice v_i será conectado com um vértice v_j, se a distância $d(v_i, v_j) < \epsilon$. Outras abordagens incluem relações de vizinhança usando uma função gaussiana $f(x_i, x_j) = \exp\left(-\|x_i - x_j\|^2 / 2\sigma^2\right)$ na qual x_i representa um dado de entrada e σ controla o tamanho da vizinhança [Von Luxburg 2007].

Outra abordagem proposta por Lopes et al. (2009) pode ser considerada como um *KNN* adaptativo. A seguir esta técnica é descrita com mais detalhes.

Seja $k_{nn}(v_i)$ o conjunto dos *K*-vizinhos mais próximos do vértice v_i por uma medida de similaridade dada, observe que vértices em $k_{nn}(v_i)$ podem ter diferentes rótulos de classes. A vizinhança N_{K_i} para um vértice v_i é definida como seus *K*-vizinhos conectados como segue: $N_{K_i} = \{v_j \mid e_{ij} \in E \text{ e } v_j \in \text{classe}(v_i) \text{ e } v_j \in k_{nn}(v_i)\}$. O grau g_i é definido como o número de arestas para seus vizinhos N_{K_i} e isto é no máximo $2K_i$ (*K* arestas de v_i para seus vizinhos e *K* de seus vizinhos para v_i).

Seja g_i o grau do vértice v_i, *N* o número de dados no conjunto de treinamento (número de vértices), *K* um parâmetro para controlar o número de vizinhos usados na construção da rede. A fração $g_i/2K$ corresponde à conexão entre o vértice v_i e os vértices em seu próprio componente, note que cada classe pode possuir um ou mais componentes da rede e cada componente é um conjunto de vértices conectados. Essa proporção varia entre 0 e 1, inclusive. Em seguida, o total de conexões entre N_c vértices num componente *C* é dado pela Equação (1).

$$|E_c| = \frac{1}{2}\sum_{i=1}^{N_c} g_i = \frac{N_c}{2}\sum_{i=1}^{N_c}\frac{g_i}{N_c} = \frac{N_c}{2}\langle G_c\rangle \quad (1)$$

Onde $\langle G_c\rangle$ corresponde ao grau médio do componente *C*. O número máximo de arestas entre N_c vértices é KN_c desde que $K < N_c$. Com isso, a probabilidade de arestas entre vértices no mesmo componente *C* (componentes inter conectados) é dada pela Equação (2).

$$P_i = \frac{\frac{N_c\langle G_c\rangle}{2}}{KN_c} = \frac{\langle G_c\rangle}{2K} \quad (2)$$

Na Equação (2) $P_i = 1$ quando há somente vértices com o mesmo rótulo na *K*-vizinhança de todo componente de v_i. Desse modo, $\langle G_c\rangle/2K$ pode ser visto como uma medida de *pureza* na região do componente *C*.

No processo descrito até agora, cada *K* gera uma rede e certamente algumas redes terão componentes melhores do que outras, de acordo com a noção de pureza. Raramente uma rede obtida de um único *K* terá os melhores componentes entre todos os componentes em todas as *K* redes possíveis. A partir disso, o objetivo é obter uma rede com a melhor organização de todos os dados nos componentes independentemente de se ter um único *K*. Fazendo isto, a ideia é variar *K* mantendo os melhores componentes encontrados. Esse processo irá resultar em uma rede chamada rede ótima, com

componentes formados por valores distintos de *K*. A rede ótima é a estrutura final obtida por meio desse processo. Essa rede pode ser vista como o resultado do processo de aprendizagem supervisionado e será usada no classificador *K*-associado não-paramétrico proposto para classificar novos dados [Lopes et al. 2009].

Otimização em redes complexas

Cancho e Solé (2003) utilizam um algoritmo evolutivo envolvendo a minimização da quantidade de arestas e a média do menor caminho e encontram quatro principais tipos de redes: exponenciais, livres de escala, estrelas e altamente densas. A seguir o método é descrito com mais detalhes.

Em um tempo $t = 0$ tem-se um grafo aleatório com distribuição de grau de Poisson, no qual dois vértices são conectados com uma probabilidade p. A função de energia do algoritmo de otimização é definida como:

$$E(\lambda) = \lambda d + (1 - \lambda)\rho \tag{3}$$

na qual $0 \leq \lambda, d, \rho \leq 1$. λ é um parâmetro que controla a combinação linear de d e ρ. O número normalizado de arestas, ρ é definido em termos de a_{ij} como:

$$\rho = 1/\binom{n}{2} \sum_{i<j} a_{ij} \tag{4}$$

A distância vértice-vértice normalizada, d, é definida como $d = D/D^{linear}$, tal que D é a média da distância mínima entre os vértices e $D^{linear} = (n + 1)/3$ é o valor máximo de D que pode ser encontrado em uma rede conexa, considerando-se um grafo linear.

$$D = 1/\binom{n}{2} \sum_{i<j} D_{ij} \tag{5}$$

A minimização de $E(\lambda)$ envolve a minimização simultânea da distância e do número de arestas. Essas duas restrições incluem dois aspectos relevantes em uma rede, o custo das ligações físicas entre unidades e a velocidade de comunicação entre elas.

O algoritmo de minimização procede da seguinte maneira: No tempo $t = 0$, a rede é inicializada com uma densidade $\rho(0)$ seguindo a distribuição de grau de Poisson. Num tempo $t > 0$, o grafo é modificado alterando-se aleatoriamente a conexão entre alguns pares de vértices. Com uma probabilidade *v*, cada a_{ij} pode mudar de 0 para 1 ou de 1 para 0. A nova matriz de adjacência é aceita se $E(\lambda, t + 1) < E(\lambda, t)$. Caso contrário, diferentes alterações são realizadas e testadas novamente. O algoritmo é interrompido quando as modificações em $A(t)$ não são aceitas após *T* execuções. A minimização do algoritmo funciona como a técnica de *simulated annealing*.

O trabalho de Cancho e Solé (2003) mostra que a otimização tem um papel fundamental na formação e evolução das redes complexas. A Figura 1 sugere que quatro fases estão presentes no processo.

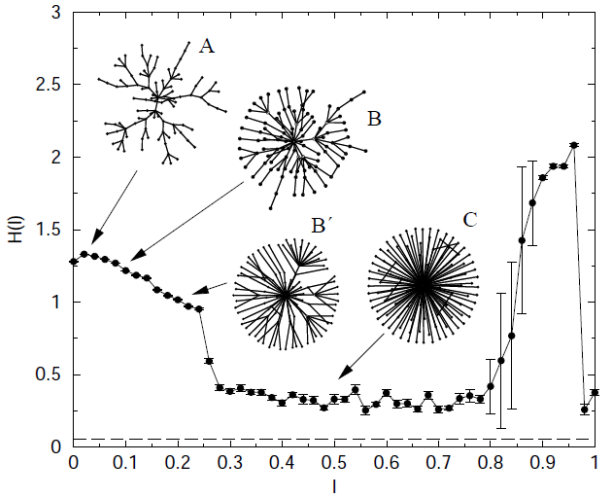


Figura 1: Redes ótimas para determinados valores de λ. Média sobre 50 redes otimizadas com n = 100, T = $\binom{n}{2}$***, v =*** $2/\binom{n}{2}$ ***e*** $\rho(0) = 0.2$***. A: uma rede exponencial com*** $\lambda = 0.01$***. B: uma rede livre de escala com*** $\lambda = 0.08$***. Hubs com múltiplas conexões e a dominância de vértices com apenas uma conexão podem ser vistos. C: uma rede estrela com*** $\lambda = 0.5$***. B': uma rede intermediária entre B e C com muitos hubs podem ser identificadas, (Cancho e Solé 2003).***

3. O Método Proposto

A motivação para este trabalho partiu das redes *K*-associados [Lopes et al. 2009]. Essas redes são construídas com base em uma medida de pureza e como esta medida não considera o tamanho dos componentes de maneira explícita, ela favorece a formação de muitos componentes pequenos quando se tem um nível alto de mistura nas classes. Uma boa classificação depende do equilíbrio entre esses dois fatores, por essa razão, buscou-se desenvolver uma técnica considerando não apenas o fator de *pureza*, mas também *extensões* de classes formadas.

No método proposto, uma rede é construída a partir de um conjunto de dados proposicionais representados no formato atributo-valor. A rede é construída com base nas relações de similaridade entre os vértices, de modo que cada vértice irá alterar suas conexões com um *k*-vizinho mais próximo que seja de sua mesma classe, somente se essa alteração maximize a *função de energia* aqui sugerida. A utilização dessa função de energia foi inspirada pelo trabalho de Cancho e Solé (2003), no qual se buscava a minimização da quantidade de arestas e da média do menor caminho.

Buscamos definir uma função de energia que considere a relação entre a pureza e a extensão da rede, visto que os dois fatores são opostos, pois o aumento da pureza tende a diminuir a extensão e vice-versa. A função de energia é representada pela Equação (6):

$$E = \lambda d + (1 - \lambda)p \tag{6}$$

na qual, d representa a extensão e p representa a pureza, tal que p e $d \in [0,1]$. A partir das medidas de pureza e extensão, investigamos o comportamento delas na rede alterando gradativamente o peso de cada uma na função de energia, para tal um parâmetro $\lambda \in [0,1]$ foi introduzido.

Para a proposta considerada, a medida de extensão é computada do seguinte modo: o caminho mínimo é calculado para *N* vértices de cada componente formado na rede. O valor máximo encontrado para cada componente *C* representa o diâmetro do mesmo. Podemos obter assim a média de diâmetro dos componentes:

$$\langle d \rangle = \left(\sum_{i=1}^{C} d_i\right)/C \tag{7}$$

Para que $\langle d \rangle$ permaneça no intervalo entre 0 e 1, é normalizado novamente pelo maior diâmetro possível de ser encontrado quando consideramos a maior classe como um grafo linear. Obtemos assim *d,* que representa a extensão da rede.

$$d = \langle d \rangle / d_{max} \tag{8}$$

A pureza de um vértice *i* se refere à relação entre o total de ligações que um vértice estabeleceu e o número máximo que ele poderia ter estabelecido. Seja g_i o grau de saída do vértice v_i, k um parâmetro para controlar o número de vizinhos usados na construção da rede. A Equação (9) corresponde a relação entre o número de conexões entre o vértice v_i e os vértices em seu próprio componente.

$$p_i = g_i/k \tag{9}$$

A medida *p* se refere a pureza de toda a rede, sendo a média de pureza de todos os vértices.

$$p = \left(\sum_{i=1}^{N} p_i\right)/N \tag{10}$$

Algoritmo 1

Entrada: Conjunto de vértices: $V = v_1, \ldots, v_n$
Conjunto de classes: $L = classe(v_1), \ldots, classe(v_n)$
Parâmetro $\lambda \in [0,1]$

Saída: Rede gerada: R

1) **Para** cada vértice v_i de V
Para cada vértice v_j de V
S = Calcula_similaridade (v_i, v_j);
S = Ordena_crescente (S);
R = V;

2) p = Calcula_pureza (R);
d = Calcula_extensão (R);
E = Calcula_Energia (p, d, λ);

3) **Para** *cont* = 1 até N
R_Δ = Modifica_rede (R, S, V, L);
p_Δ = Calcula_pureza (R_Δ);
d_Δ = Calcula_extensão (R_Δ);
E_Δ = Calcula_Energia $(p_\Delta, d_\Delta, \lambda)$;
Se $E_\Delta > E$
$E = E_\Delta$;
$R = R_\Delta$;

3.1) Modifica_rede (R, S, V, L)
v_i = *random*();
k = *random*();
Se $(k < n)$
v_k = Busca_vertice (S);
Se $\left(classe(v_i) == classe(v_k)\right)$
Se (!Existe_aresta(v_i, v_k))
$R_\Delta \leftarrow R_\Delta \cup$ Insere_aresta (V);
Senão
$R_\Delta \leftarrow R_\Delta -$ Remove_aresta (V);

4) **Retorna** R.

O Algoritmo 1 descreve com detalhes a implementação do método. A entrada do algoritmo é um conjunto de dados representados na forma atributo-valor, com a respectiva classe associada e um parâmetro λ.

No passo 1 o algoritmo calcula a partir dos dados de entrada a matriz de similaridade, nesse caso foi calculada a distância euclidiana entre todos os pares de vértices. Posteriormente cada linha dessa matriz é ordenada em ordem crescente. A rede R é inicializada, de modo que cada objeto de dado é representado em um vértice.

No passo 2 são calculadas as medidas de extensão da rede, Equação (8), de pureza, Equação (10), e a partir destas é realizado o cálculo da energia, Equação (6), para determinado valor de λ.

O passo 3 é repetido N vezes. Outra opção que poderia ser utilizada como critério de parada, seria executar o algoritmo até que N vezes o valor de E não sofra alteração.

O passo 3.1 *Modifica-rede* é executado retornando uma nova rede com modificações nas ligações para um vértice i. A partir da nova rede, as medidas de pureza e extensão são recalculadas e a função de energia é reavaliada para estas novas medidas. Caso o valor da nova função de energia seja maior que o anterior, a rede antiga é substituída pela nova rede que sofreu modificações.

No passo 3.1 um vértice i é escolhido aleatoriamente para sofrer alteração nas ligações. Também é escolhido aleatoriamente um vértice k, que esteja dentro do conjunto dos n-vizinhos mais próximos do vértice i, para se conectar/desconectar com este. Nos testes realizados, o vértice k é sorteado num intervalo entre 1 e 5.

Destaca-se que o vértice k é selecionado aleatoriamente, porém o 1-vizinho tem uma probabilidade maior de ser escolhido, que o 2-3...n-vizinho respectivamente. Do mesmo modo, o 2-vizinho tem uma probabilidade menor que o 1 de ser selecionado, porém sua probabilidade é maior que o 3-4...n-vizinho respectivamente. E assim por diante. O objetivo disto é fazer com que os vértices mais similares de um vértice i tenham preferência nas conexões.

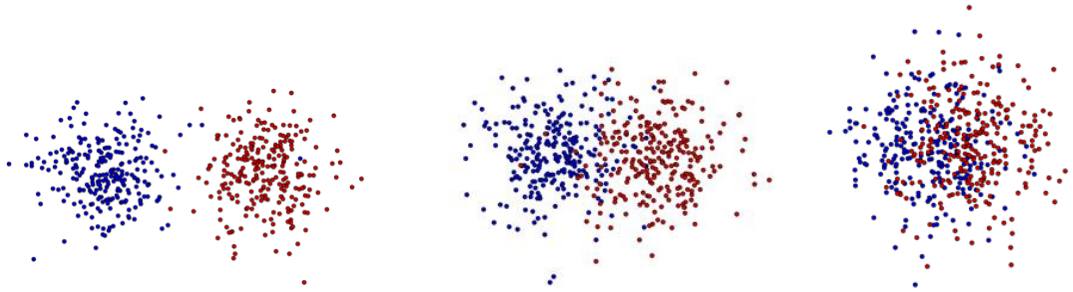
Caso o vértice k escolhido não esteja ligado com o vértice i, uma ligação entre eles será estabelecida se ambos pertencerem a mesma classe. Além disso, o vértice i irá estabelecer ligações com todos os k-vizinhos menores que k. Porém, se o vértice k já estiver conectado com o vértice i, esta conexão é removida, bem como todas as conexões existentes entre o vértice i e os vértices maiores que k.

Essa nova rede irá conter alterações nas ligações para um dado vértice i e será retornada para o Passo 3. Por fim, o passo 4 retorna a rede final, a qual apresenta a maximização da função de energia, para um dado valor de λ.

4. Simulações em Redes Artificiais e Reais

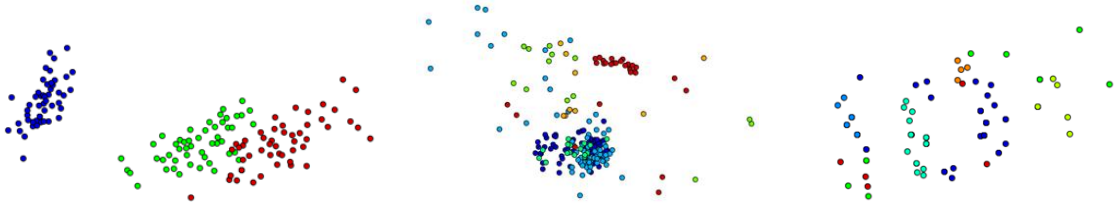
O algoritmo foi executado para um conjunto de dados artificiais e em três bases de dados reais a fim de analisar seu comportamento nestas bases. Testou-se para um conjunto de dados com distribuição gaussiana (Figura 2) de modo que, para cada conjunto de dados aumentava-se gradativamente a mistura dos elementos e nos

conjuntos de dados Iris, Glass e Zoo, obtidos no repositório UCI[1]. Para a geração das figuras foi utilizado o software PEx[2], com o método de *Sammon´s Mapping* para redução de dimensionalidade.



Gaussianas 1 *Gaussianas 2* *Gaussianas 3*

Figura 2: Base de dados *Gaussianas 1-2-3*. Cada figura representa dois conjuntos de dados com 250 elementos e distribuição gaussiana, a cor vermelha representa uma classe e a cor azul outra.



Iris *Glass* Zoo

Figura 3: Base de dados Iris, Glass e Zoo. Cada cor nos conjuntos de dados representa uma classe.

Na Figura 4 são mostradas as redes finais formadas para as bases *Gaussianas 1-2-3*, após a execução do *Algoritmo 1* para os valores de λ igual a 0 e 1. Nota-se que quando $\lambda = 0$ apenas a pureza está sendo levada em conta e quando $\lambda = 1$ apenas a extensão é considerada.

As redes *A* e *B* foram formadas a partir da base de dados *Gaussianas 1*. Nota-se que ambas as redes formaram poucos componentes, porém na rede *A* há mais conexões entre os vértices e na rede *B* o diâmetro dos componentes é maior que na rede *A*. As redes *C* e *D* foram formadas a partir da base de dados *Gaussianas 2*. Como o nível de mistura foi aumentado um número maior de componentes é formado, porém a rede C apresenta mais componentes que a rede *D*. As rede *E* e *F* foram formadas a partir da base de dados *Gaussianas 3* cujo nível de mistura está maior que das bases *Gaussianas 1* e *2*. Com isso, o número de componentes formados é bem alto quando $\lambda = 0$ (rede *E*), pois como cada vértice apresenta muitos vizinhos com classe diferente da sua, estabelece ligações com pouco deles para permanecer como uma pureza alta. Quando $\lambda = 1$, o número de componentes formados na rede *F* diminui com relação a rede *E*.

Analisando os resultados obtidos, nota-se que conforme a mistura entre dados de diferentes classes aumenta o número de componentes formados na rede também aumenta, tanto para $\lambda = 0$, como para $\lambda = 1$. Porém para $\lambda = 1$ a tendência é de formar menos componentes que para $\lambda = 0$.

[1] http://archive.ics.uci.edu/ml/
[2] http://infoserver.lcad.icmc.usp.br/infovis2/PExDownload

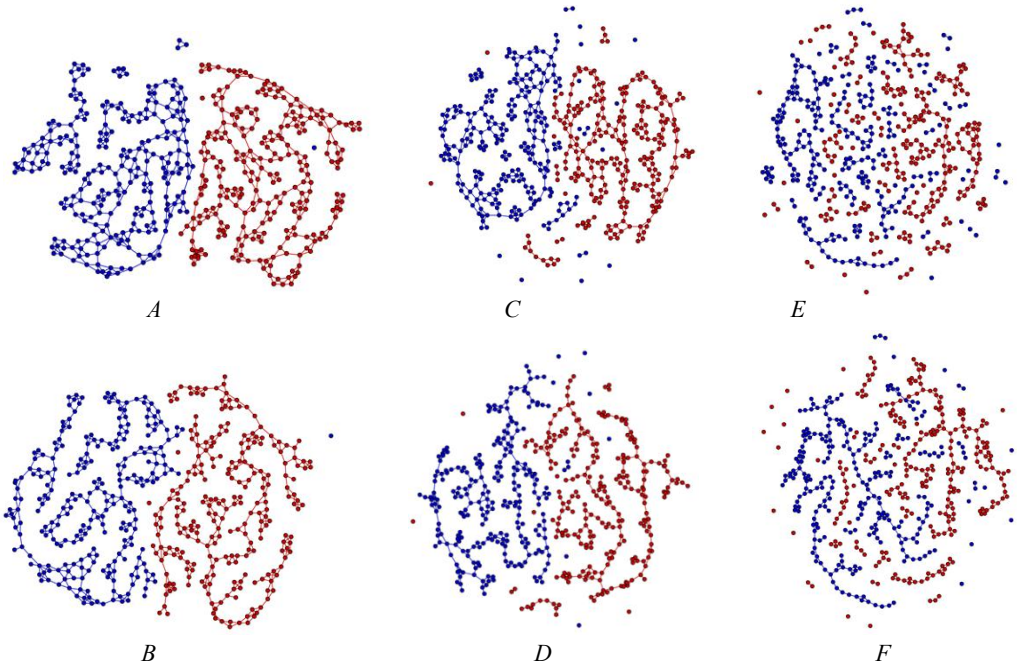


Figura 4: *A* e *B*: redes finais formadas para a base de dados *Gaussianas 1*. *C* e *D*: redes finais formadas para a base de dados *Gaussianas 2*. *E* e *F*: redes finais formadas para a base de dados *Gaussianas 3*. As redes *A*, *C* e *E* foram construídas com λ igual a 0 e as redes *B*, *D* e *F* foram construídas com λ igual a 1. Foram considerados *n* = 5 e *N* = 10000 no *Algoritmo 1*.

Na Figura 5 são mostradas as redes finais formadas para as bases de dados Iris, Glass e Zoo, após a execução do *Algoritmo 1* para λ igual a 1. Nota-se que na base Iris os componentes ficaram bem conexos, já que esta base apresenta menos mistura entre as classes, ao contrário da rede Glass e Zoo.

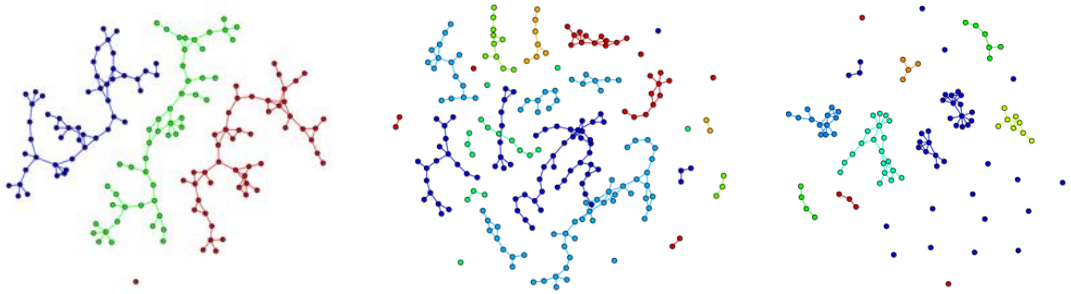


Figura 5: Redes finais formadas para as bases de dados Iris, Glass e Zoo. As redes foram construídas com λ igual a 1. Foram considerados *n* = 5 e *N* = 10000 no *Algoritmo 1*.

A Figura 6 mostra os valores obtidos para a pureza, extensão e energia das redes finais geradas a partir da base de dados *Gaussianas 1-2-3*, quando a função de energia é maximizada em alguns valores de λ.

Nota-se que o valor da pureza para a base *Gaussianas 3* é menor que da base *Gaussianas 2* e *Gaussianas 1* respectivamente. Isso porque a base *Gaussianas 3* apresenta alto nível de mistura entre os elementos de diferentes classes, com isso, componentes "menos puros" são formados. Ou seja, cada vértice estabelece ligações

com um número menor de vizinhos que estava sendo considerado no momento. Além disso, a pureza em cada base se mantém praticamente constante, diminuindo levemente conforme λ se aproxima de 1. Isso acontece, pois os vértices conseguem completar suas ligações tanto para um *k*-vizinho menor como para um *k*-vizinho maior, permanecendo com um valor alto de pureza.

Observa-se que o valor da extensão para a base *Gaussianas 3* é menor que da base *Gaussianas 2* e *Gaussianas 1* respectivamente, devido ao nível de mistura apresentado na base *Gaussianas 3*. Com isso, os vértices não conseguem formar um único componente para cada classe, diminuindo o valor para a extensão. Além disso, o valor de extensão aumenta conforme λ se aproxima de 1, já que esta medida passa a ter mais destaque no cômputo da função de energia.

A função de energia decai conforme λ se aproxima de 1 porque passa a dar mais peso para a extensão e essa medida obtém um valor menor que o da pureza para todas as redes.

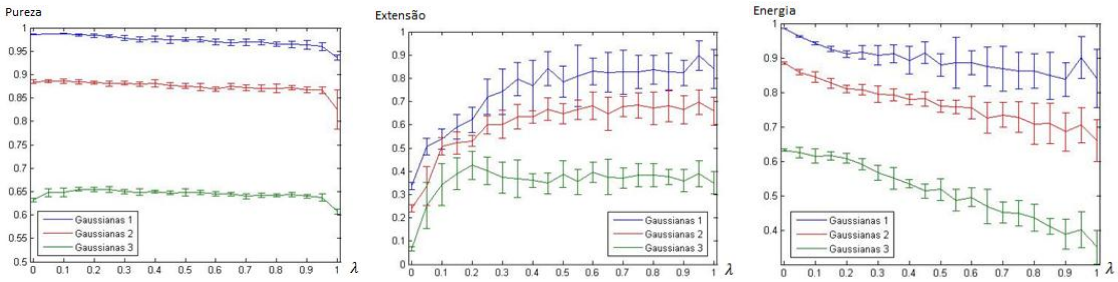


Figura 6: Representação da pureza, extensão e energia para as redes finais geradas a partir das bases de dados *Gaussianas 1-2-3*, com média sobre 30 execuções do algoritmo.

A Figura 7 mostra a pureza, extensão e energia das redes finais geradas quando a função de energia é maximizada em alguns valores de λ para as bases de dados Iris, Glass e Zoo. Nota-se que o valor da pureza para a base Iris assume valores mais próximos de 1, já que esta base apresenta menos mistura entre as classes, diferentemente das bases Glass e Zoo.

O valor da extensão para a base Iris assume valores maiores que as demais bases, pois consegue formar componentes conexos, além disso, o tamanho das três classes é igual. Já as bases Glass e Zoo apresentam além da mistura entre os componentes, classes com tamanhos variados, de modo que quando o cálculo da extensão é realizado, ele é ponderado pelo tamanho da maior classe, e com isso, o valor da extensão alcança valores mais baixos.

A energia apresenta um comportamento semelhante ao da base *Gaussianas*, ou seja, decai conforme λ se aproxima de 1 porque passa a dar mais peso para a extensão e essa medida obtém um valor menor que o da pureza para as redes testadas.

Analisando os resultados para pureza, extensão e energia, nota-se que as três medidas obtém valores mais baixos conforme o nível de mistura aumenta, de modo que estas medidas podem ser utilizadas para caracterizar a mistura nas classes dos dados. Além disso, observa-se que conforme a extensão das redes aumenta a pureza diminui levemente, indicando que seria possível utilizar uma rede formada para um valor maior de λ já que esta teria um número menor de componentes formados e poderia ser mais interessante para o processo de classificação.

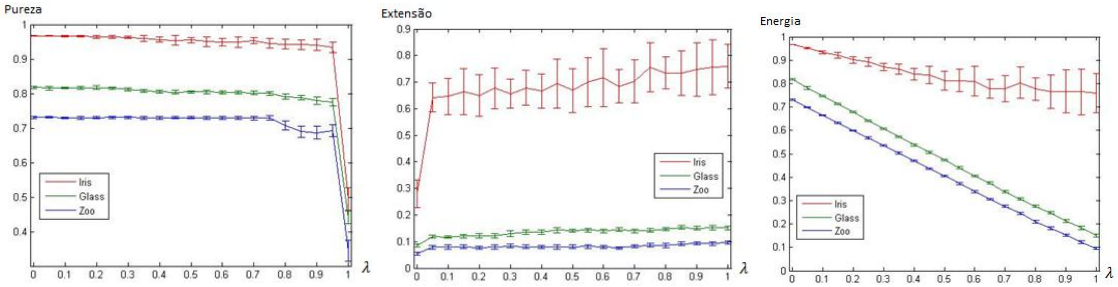


Figura 7: Representação da pureza, extensão e energia para as redes finais geradas a partir das bases de dados Iris, Glass e Zoo, com média sobre 30 execuções do algoritmo.

4. Conclusões

Este artigo apresenta um método para construção de redes baseado nas relações de similaridade entre os vértices da mesma classe e em uma função de energia que pondera medidas de pureza e extensão da rede. A construção de redes a partir de dados é uma fase importante no aprendizado baseado em grafos, de modo que este trabalho pode ser visto como uma contribuição para esta etapa.

O método foi aplicado em alguns conjuntos de dados artificiais e reais para caracterizar mistura entre as classes de dados. Os resultados obtidos mostraram que conforme a mistura dos dados aumenta, a pureza, a extensão e a energia diminuem, indicando que estas medidas podem ser utilizadas para caracterizar mistura de classes. Observa-se também que conforme a extensão aumenta a pureza não diminui de maneira considerável, indicando que redes formadas com um valor maior de λ poderiam ser utilizadas para a classificação quando se tem muita mistura nos dados, pois desse modo, se consideraria redes com um número menor de componentes formados.

A técnica de caracterização de classes embora apresente resultados preliminares, é um assunto importante e novo. Pode-se considerar que o presente trabalho é uma das primeiras tentativas nesta direção, além de ser uma contribuição para a construção de grafos a partir de dados proposicionais.

Como trabalho futuro pode-se aplicar o método em outros dados reais a fim de estudar seu comportamento nestes dados e caracterizá-los. O método pode ser aprimorado, de modo que diferentes medidas podem ser propostas e exploradas, visando extrair outras características de dados com diferentes classes. Podem ser testadas ainda outras medidas de similaridade para composição da rede, além da distância euclidiana.

Referências

Albert, R., Jeong, H. e Barabási, A. L. (1999) "Internet: Diameter of the world wide web". Nature 401, p.130-131.

Baldi, P. e Brunak, S. (1998) Bioinformatics: the machine learning approach, MIT Press.

Barabási, A. L. e Albert, R. (1999) "Emergence of scaling in random networks", Science 286, p. 509-512.

Belkin, M. Niyogi, P. (2003) "Laplacian eigenmaps for dimensionality reduction and data representation", Neural Computation 15, p. 1373-1396.

Boccaletti, S., Ivanchenko, M., Latora, V., Pluchino, A. e Rapisarda, A. (2007) "Detecting complex network modularity by dynamical clustering", Physical Review E 75, p. 1-4.

Bornholdt, S. e Schuster, H. G. (2003) Handbook of Graphs and Networks: From the Genome to the Internet, Wiley-VCH.

Cancho, F. e Solé, R. V. (2003) "Optimization in complex networks. Statistical Mechanics of Complex Networks", Lecture Notes in Physics 625, p. 114-125.

Cook, D. J. e Holder, L. B. (2000) "Graph-based data mining", IEEE Intelligent Systems 15, p. 32-41.

Erdös, P. e Rényi, A. (1959) "On random graphs", Publicationes Mathematicae 6, p. 290-297.

Faloutsos, M., Faloutsos, P. e Faloutsos, C. (1999) "On power-law relationship of the internet topology", Computer Communication Review 29, p. 251-262.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. e Jackel, L. D. (1989) "Backpropagation applied to handwritten zip code recognition", Neural Computation 1, p. 541-551.

Lopes, A. A., Bertini, Jr. J. R., Motta, R. e Zhao, L. (2009) "Classification Based on the Optimal K-Associated Network", Proceedings of The First International Conference on Complex Sciences: Theory and Applications, p. 1-11.

Mitchell, T. M. (1997) Machine learning. McGraw-Hill Series in Computer Science, McGraw-Hill.

Newman, M. E. J. (2004) "Fast algorithm for detecting community structure in networks", Physical Review E 69 066133.

Newman, M. E. J. e Girvan, M. (2004) "Finding and evaluating community structure in networks", Physical Review E 69 026113.

Quiles, M. G., Zhao, L., Alonso, R. L. e Romero, R. A. F. (2008) "Particle competition for complex network community detection", Chaos 18, p. 1-10.

Reichardt, J. e Bornholdt, S. (2004) "Detecting fuzzy community structures in complex networks with a Potts model" Physical Review Letters 93, p. 1-4.

Scott, J. (2000) Social network analysis: a handbook, Sage.

Sponrs, O. (2002) "Networks analysis, complexity, and brain function", Complexity 8, p. 56-60.

Von Luxburg, U. (2007) "A tutorial on spectral clustering", Statistical Computation 17, p. 395-416.

Watts, D. J. e Strogatz, S. H. (1998) "Collective dynamics of 'small-world' networks", Nature 393, p. 440-442.

Wolpert, D. H. e Macready, W. G. (1997) No Free Lunch Theorems for Optimization, IEEE Transactions on Evolutionary Computation 1, p. 67.

Zhou, H. (2003a) "Network landscape from a Brownian particle´s perspective". Physical Review E 67 041908.

Zhou, H. (2003b) "Distance, dissimilarity index, and network community structure". Physical Review E 67 061901.

Zhu, X. (2005) "Semi-supervised learning literature review". Technical Reporter 1530, Computer-Science, University of Wisconsin-Madison.

Uso de Critérios Multiobjetivo Baseados em Enxames na Escolha dos Melhores Métodos para Seleção de Atributos em Microarranjos Gênicos

Rodolfo Garcia[1], Júlio Cesar Nievola[1]

[1]Programa de Pós-Graduação em Informática (PPGIa) – Pontifícia Universidade Católica do Paraná (PUCPR)
CEP 80.215-901 – Curitiba – PR – Brasil

{rodolfobbgarcia,nievola}@ppgia.pucpr.br

Abstract. *Microarray technique is responsible for extracting information about expression of large numbers of genes. Knowing that only few genes participate in the activation or inhibition of each characteristic, this paper aims to evaluate methods to select relevant features in genes datasets obtained by microarray technique. The criteria used to measure the quality of each selection method characterize a multiobjective problem and the proposed swarm technique called MOPSO evaluates the methods and defines satisfactorily the best solutions, facilitating the genes expression datasets analysis.*

Resumo. *A técnica de microarranjo é responsável pela extração de informação sobre a expressão de grande quantidade de genes. Sabendo-se que poucos são os genes que fazem parte da ativação ou inibição de uma característica, este trabalho objetiva avaliar métodos para selecionar os atributos mais relevantes em bases de dados obtidas pela técnica de microarranjo. Os critérios utilizados para medir a qualidade de cada método de seleção caracterizam um problema multiobjetivo e uma proposta técnica baseada em enxames, o MOPSO, avalia os métodos e define satisfatoriamente as melhores soluções, facilitando a análise de bases de dados contendo expressões gênicas.*

1. Introdução

O aumento da quantidade de informações resultantes de pesquisas relacionadas ao Projeto Genoma Humano faz com que análises manuais feitas por cientistas demorem até anos para serem completadas. Para agilizar esse trabalho, necessita-se da alta velocidade computacional, além de sua capacidade de analisar simultaneamente vários genes [Dy 2008].

A análise das expressões gênicas é de extrema importância para diagnosticar doenças e identificar o estado de determinados genes em ciclos específicos [NIH 2001]. A técnica de microarranjo é responsável pela extração de informação sobre a expressão de grande quantidade de genes em relação a uma determinada característica. Essa técnica tem sido utilizada para buscar os possíveis genes envolvidos em doenças como o câncer, Alzheimer, Parkinson e diabetes [D'haesleer 2005].

Apesar de obter informações simultaneamente de milhares de genes, sabe-se que poucos são aqueles que participam da ativação ou inibição de uma característica, o que

torna necessário o uso de métodos que possam selecionar os genes mais relevantes e obter resultados mais compreensíveis [Handl e Knowles 2007]. Este trabalho objetiva avaliar métodos de seleção de atributos, o C-FOCUS e o *Relief*-F, em bases de dados de expressões gênicas obtidas pela técnica do microarranjo. Para medir a qualidade de cada seleção são usados critérios como a quantidade de genes selecionados e o índice Jaccard, resultante do agrupamento das instâncias de uma base pelo algoritmo K-*means*.

Otimizar vários critérios caracteriza um problema multiobjetivo. Neste trabalho, a avaliação dos métodos de seleção aqui utilizados e a definição dos métodos ótimos para cada base de dados são feitas através de uma técnica multiobjetivo baseada em enxames, o *Multi-Objective Particle Swarm Optimization*. Um método de seleção é tido como ótimo se gerar uma base de dados composta pelos atributos mais relevantes, o que torna sua análise mais simples, já que seus atributos representam da melhor forma possível cada classe existente.

A seção 2 expõe a importância da seleção de atributos na análise de bases de dados gênicos e os algoritmos usados neste trabalho. Na seção 3 é mostrada a etapa do agrupamento, sua influência sobre a seleção de atributos, o método K-*means* e o índice Jaccard. A seção 4 apresenta a técnica *Multi-Objective Particle Swarm Optimization* baseado na frente de Pareto e seu funcionamento. Na seção 5 é apresentada detalhadamente a proposta deste trabalho. Em seguida, na seção 6, são descritos os experimentos, juntamente com os resultados e, na seção 7, as últimas considerações.

2. Seleção de Atributos

Muitas aplicações do mundo real, como análise de expressões gênicas obtidas pelos microarranjos, apresentam grande quantidade de atributos [Dash et al. 2002]. O problema é que, apesar do microarranjo obter informações de milhares de genes simultaneamente, sabe-se que poucos são aqueles que fazem parte da ativação ou inibição de uma característica.

Para os genes, ou atributos, que não melhoram os resultados das análises são atribuídos o nome de redundantes ou irrelevantes e devem ser removidos [Kohavi e John 1997]. A redução da quantidade de genes pode garantir resultados mais compreensíveis, obtidos em menos tempo e com altas taxas de acerto, já que são usados apenas atributos relevantes [Dy 2008].

Os algoritmos utilizados neste trabalho fazem parte do método de filtro e podem ser utilizados por qualquer outro algoritmo pertencente às demais etapas da mineração de dados, como o agrupamento. Segundo [Yu e Liu 2004], esses métodos são úteis em bases de grande dimensionalidade, como no problema de seleção de genes.

O primeiro algoritmo, o C-FOCUS, é um extensão do FOCUS para ser usado em atributos nominais e discretos [Azofra, Sanchez e Peña 2003]. Seu objetivo é eliminar inconsistências e redundâncias retornando o menor subconjunto ótimo, chamado *Min-Features*, formado apenas por atributos relevantes [Kohavi e John 1997].

Outro método usado é o *Relief*-F, uma extensão do *Relief* que envolve mais de duas classes, o qual escolhe uma quantidade de atributos especificada pelo usuário. Quanto maior o poder de um atributo em distinguir instâncias de classes diferentes, maior é a probabilidade dele ser selecionado [Kononenko e Sikonja 2008].

3. Agrupamento

A etapa de agrupamento objetiva organizar um conjunto de instâncias de forma que aquelas com comportamentos similares pertençam ao mesmo grupo [Xu e Wunsch 2005]. Um bom agrupamento é sinônimo de que os atributos das bases de dados são relevantes e suas instâncias definem bem as classes a que pertencem. Logo, essa etapa pode servir como avaliador dos métodos de seleção de atributos. Nas bases de dados de expressão gênica, instâncias com comportamentos similares podem representar a mesma característica.

O algoritmo usado neste trabalho é o K-*means,* que é baseado em centróide, bastante popular e simples. A partição resultante é formada pela quantidade de grupos especificada pelo usuário e cada objeto pertence a um único grupo [Berkhin 2002].

Neste trabalho, a qualidade do agrupamento é calculada pelo índice Jaccard, que compara o agrupamento obtido (A_o) com um agrupamento de referência (A_r). Segundo [Faceli, Carvalho e Souto 2005], duas instâncias são ditas:

- SS, se pertencem ao mesmo grupo em A_o e ao mesmo grupo em A_r;

- SD, se pertencem ao mesmo grupo em A_o e a grupos diferentes em A_r;

- DS, se pertencem a grupos diferentes em A_o e ao mesmo grupo em A_r;

Em que a_1, a_2 e a_3 são, respectivamente, a quantidade de pares SS, SD e DS. O índice Jaccard, calculado pela Equação 1, pertence ao intervalo [-1, 1] e o valor máximo representa agrupamentos idênticos [Faceli, Carvalho e Souto 2005].

$$jaccard = \frac{a_1}{(a_1 + a_2 + a_3)} \tag{1}$$

4. Técnicas Multiobjetivo

O uso de vários critérios facilita na avaliação dos métodos de seleção, pois nem sempre o melhor método em um único critério é o melhor método possível. Um problema de otimização que trata simultaneamente de vários critérios, ou funções objetivo, é chamado de multiobjetivo. Em [Suresh et al. 2009] encontra-se a formulação matemática para um problema de otimização multiobjetivo composto por m objetivos, cada um com n variáveis de decisão, que são as soluções para o problema (Equação 2).

$$Otimizar\ \vec{Y} = \vec{f}(\vec{Y}) = \left(f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n)\right) \tag{2}$$

No problema multiobjetivo, uma variável de decisão pode não ser a melhor solução para todos os objetivos, sendo possível eleger mais de uma variável como melhor solução. Essa é a finalidade da técnica de enxames *Multi-Objective Particle Swarm Optimization* (MOPSO) baseada na frente de Pareto.

O MOPSO é um algoritmo evolucionário que se baseia na simulação de vôo que uma população de pássaros faz ao buscar alimentos [Kennedy e Eberhart 1995]. Além de ser simples, outro fator que o tornou popular foi sua eficiência em varias aplicações, produzindo bons resultados com baixo custo computacional [Chuang et al. 2008].

Cada membro da população, ou enxame, é chamada de partícula e representa uma possível solução que se movimenta pelo espaço de busca. Sua posição $\vec{x}_i(t)$,

(Equação 3), em um determinado tempo t é definida pela sua posição no tempo $t-1$ acrescida pelo valor do operador $\vec{v}_i(t)$, que simula a velocidade.

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (3)$$

As partículas que estiverem situadas em regiões promissoras são tidas como as melhores soluções e são chamadas de líderes, podendo ser mais de uma porque se trata de um problema multiobjetivo. Uma partícula é eleita líder se, primeiramente for definida para um problema de maximização ou minimização, e satisfizer a chamada regra de dominância de Pareto [Coello, Lamont e Vldhuizen 2007].

Dadas duas soluções $\vec{x}$ e $\vec{y}$, pertencentes ao espaço de soluções π, dizemos que $\vec{x}$ domina $\vec{y}$, ou $\vec{x} \prec \vec{y}$, em um problema de minimização com n critérios, se:

- Existe pelo menos um critério $i = 1,\ldots,n$ em que x_i é estritamente melhor que y_i, ou seja $x_i \prec y_i$;

- x_i não é pior que y_i em nenhum critério i, ou seja $x_i \leq y_i$;

Se não existir solução $\vec{x}' \in \pi$, onde $f(\vec{x}') \prec f(\vec{x})$, $\vec{x}$ é Pareto Ótimo e eleito um dos líder do enxame pelo MOPSO [Suresh et al. 2009]. Por sua vez, determinando o conjunto de todos os vetores do Pareto Ótimo, que geralmente consiste em um problema NP-Completo, é possível gerar a frente de Pareto [Carvalho e Pozo 2009].

O conjunto de líderes do enxame são armazenados em um repositório no qual seu conteúdo é retornado como resultado ao término da execução do algoritmo [Sierra e Coello 2006]. Dentro deste conjunto, a melhor solução deve ser selecionada pelo *decision maker*, que pode ser o próprio usuário [Coello, Lamont e Vldhuizen 2007].

Para cada partícula, um líder é escolhido para servir-lhe de guia no espaço de busca. A escolha desse guia pode ser feita pelo método sigma ($\vec{\sigma}$), apresentado em [Mostaghim e Teich 2003] que é muito eficaz em problemas multiobjetivo baseados em técnicas de enxames. O guia só é eleito se fizer parte da mesma vizinhança da partícula. Por isso, é extremamente importante estabelecer como as partículas estão interligadas, ou seja, estabelecer uma topologia [Sierra e Coello 2006].

O vetor $\vec{\sigma}$, calculado pela Equação 4, realiza a combinação dos objetivos e é composto por $\binom{m}{2}$ elementos, em que m é a dimensão do espaço objetivo. O líder cujo vetor sigma é mais próximo que o vetor sigma da partícula, e se eles pertencerem a mesma vizinhança, será eleito líder para essa partícula [Mostaghim e Teich 2003].

$$\vec{\sigma} = \begin{pmatrix} {f_1}^2 - {f_2}^2 \\ {f_1}^2 - {f_3}^2 \\ \cdots \\ {f_{m-1}}^2 - {f_m}^2 \end{pmatrix} \Big/ \left({f_1}^2 + {f_2}^2 + {f_3}^2 + \cdots + {f_{m-1}}^2 + {f_m}^2\right) \quad (4)$$

Para um problema de dois objetivos f_1, f_2, o vetor sigma terá apenas um elemento, calculado pela Equação 5.

$$\sigma = \frac{f_1(x)^2 - f_2(x)^2}{f_1(x)^2 + f_2(x)^2} \quad (5)$$

O cálculo da velocidade de uma partícula, na Equação 6, depende de sua melhor posição, $\vec{x}_{pbest_i}$, e da melhor posição do seu líder, $\vec{x}_{leader}$ [Carvalho e Pozo 2009]. Seu resultado é influenciado pelo valor da inércia W, que mede o impacto da velocidade anterior sobre a velocidade atual, das constantes positivas de aprendizagem cognitiva C_1

e de aprendizagem social C_2, que são as influências que uma partícula tem, respectivamente, sobre sua própria velocidade e sobre a velocidade da vizinhança, e pelos valores aleatórios $r_1, r_2 \in [0,1]$ [Sierra e Coello 2006].

$$\vec{v}_i(t) = W\vec{v}_i(t-1) + C_1 r_1 \left(\vec{x}_{pbest_i} - \vec{x}_i(t-1)\right) + C_2 r_2 (\vec{x}_{leader} - \vec{x}_i(t-1)) \quad (6)$$

5. Proposta

Este trabalho tem como meta avaliar, por meio do MOPSO, métodos de seleção de atributos em bases de dados obtidas pela técnica de microarranjo. Como resultado, o MOPSO seleciona os métodos de seleção ótimos e identifica a quantidade dos atributos mais relevantes para uma característica, afim de facilitar na análise da base de dados obtidas pelo microarranjo, que são formadas por milhares de atributos.

As bases de dados usadas neste trabalho, apresentadas na Tabela 1, são formadas por instâncias com expressões gênicas de pessoas saudáveis e de portadoras de doenças relacionadas ao câncer. Algumas dessas bases foram usadas em [Borges 2006] e estão disponíveis no formato "arff", juntamente com suas documentações descrevendo as instâncias e revelando a qual classe cada uma pertence, na página da Kent Ridge[1].

Tabela 1: Descrição das bases

Nome	Número de atributos	Número de instâncias	Número de grupos*
DLBCL-Stanford	4026	47	2
DLBCL-Tumor	7129	77	2
DLBCL-NIH	7399	80	2
Leukemia-ALL/AML	7129	58	2
Leukemia-MLL	12582	57	3

* Conhecimento *a priori*

Essas bases foram submetidas ao processo de seleção de atributos por dois métodos, C-FOCUS e o *Relief*-F, este último reduzindo cada base a 10%, 25%, 50% e 75% da quantidade de seus atributos originais. Ao final desse processo existirão, para cada base original, cinco novas bases reduzidas.

Cada base resultada pela seleção de atributos, assim como as bases originais, gera uma solução composta por dois critérios: o número de atributos e o valor do índice Jaccard. Esse índice foi calculado pela comparação entre os grupos construídos pelo método K-*means* e o agrupamento real, já que há conhecimento, *a priori*, da composição das bases de expressão gênica. O conhecimento prévio também foi usado para determinar a quantidade de grupos que deviam ser gerados pelo K-*means*.

O uso de vários critérios caracteriza um problema multiobjetivo. Este trabalho faz uso do funcionamento evolucionário do método MOPSO baseado na frente de Pareto para movimentar as soluções, ou partículas, seguindo seus guias, escolhidos pelo método sigma usando dois critérios, para regiões promissoras. Os líderes eleitos pela regra de dominância de Pareto ao final da execução, quando não houver alteração no

[1] http://datam.i2r.a-star.edu.sg/datasets/krbd/, acessado em 07 de Abril de 2011.

repositório, serão considerados soluções ótimas, revelando os melhores métodos de seleção de atributos para esses critérios.

Os objetivos utilizados neste trabalho visam a minimização da quantidade de atributos e a maximização do índice Jaccard. As regiões promissoras, por sua vez, são compostas pelas partículas que otimizam esses objetivos.

No cálculo da velocidade de cada partícula, $r_1, r_2 \in [0,1]$ foram escolhidos aleatoriamente a cada iteração. O valor da inércia foi fixada em 1 para facilitar a exploração global [Sierra e Coello 2006]. As constantes de aprendizagem também foram fixadas e o valor 2 foi escolhido a partir de experiência adquirida por [Eberhart e Shi 2001].

Por se tratar de poucas soluções, todas as partículas fazem parte da mesma vizinhança, formando um grafo completamente conectado. Segundo [Sierra e Coello 2006], essa topologia converge para o resultado final mais rapidamente.

Todos os experimentos aqui foram realizados em uma máquina Intel ® Core™ I7 com 1.73GHz e 6GB de memória RAM. O MOPSO baseado na frente de Pareto foi desenvolvido na linguagem Java, assim como os métodos de seleção de atributos, o K-*means* e o índice Jaccard, que utilizaram a biblioteca do *software* Weka[2].

6. Experimentos e Resultados

Os experimentos aqui realizados visam cumprir a meta já citada na proposta. Além das bases originais, são avaliadas as bases reduzidas pelos métodos de seleção C-FOCUS e *Relief*-F, totalizando um conjunto de seis bases para cada característica.

Cada partícula gerada pelas bases de dados caracteriza a posição no espaço de buscas e o conjunto dessas partículas forma o enxame para uma determinada característica. Os enxames estão apresentados nas tabelas a seguir contendo a identificação do método de seleção, o número de atributos selecionados por esse método e o valor do índice Jaccard.

As partículas são submetidas a otimização pelo MOPSO baseado na frente de Pareto. Os gráficos abaixo, cujos eixos representam os objetivos, mostram o posicionamento final das partículas, representadas pelos círculos brancos, e dos líderes eleitos pela regra de dominância de Pareto, representados pelos círculos pretos.

A identificação dos métodos de seleção usados para cada solução obtida está numerada, tanto nas tabelas quanto nas figuras, da seguinte forma:

1- Solução obtida pelo método C-FOCUS;

2- Solução obtida pelo método *Relief*-F com 10% do tamanho original;

3- Solução obtida pelo método *Relief*-F com 25% do tamanho original;

4- Solução obtida pelo método *Relief*-F com 50% do tamanho original;

5- Solução obtida pelo método *Relief*-F com 75% do tamanho original;

6- Solução obtida pela base original;

[2] http://www.cs.waikato.ac.nz/ml/weka/, acessado em 07 de Abril de 2011.

Na Tabela 2 estão presentes as soluções geradas pelas bases DLBCL-Stanford. Como previsto, o C-FOCUS selecionou o menor conjunto de atributos. Observam-se baixos valores para o índice Jaccard, em que a solução 2 se mostrou melhor.

Tabela 2: Soluções das bases DLBCL-Stanford

Número do método	Número de atributos	Índice Jaccard
1	3	0.33
2	402	0.56
3	1006	0.39
4	2013	0.33
5	3020	0.33
6	4027	0.34

Ao final da execução do MOPSO, o repositório dos líderes eleitos pela regra de dominância de Pareto retornou as soluções 1 e 2, justamente as soluções que obtiveram, respectivamente, a menor quantidade de atributos e o maior índice Jaccard (Figura 1).

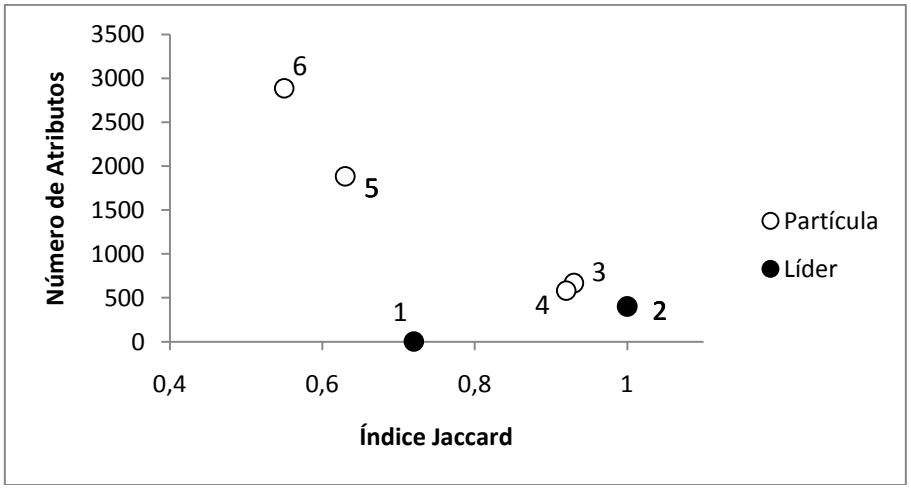


Figura 1. Posicionamento final das soluções das bases DLBCL-Stanford

As soluções geradas pelas bases DLBCL-Tumor são apresentadas na Tabela 3, onde pode ser visto que a solução 1 obteve a menor quantidade de atributos e a solução 5 conseguiu o maior valor Jaccard.

Tabela 3: Soluções das bases DLBCL- Tumor

Número do método	Número de atributos	Índice Jaccard
1	4	0.37
2	713	0.39
3	1782	0.4
4	3565	0.4
5	5347	0.46
6	7130	0.44

Na Figura 2, o repositório de líderes retornado pelo MOPSO para as bases DLBCL-Tumor é formado somente pela solução 1, que otimizou os dois critérios simultaneamente e dominou todas as outras soluções.

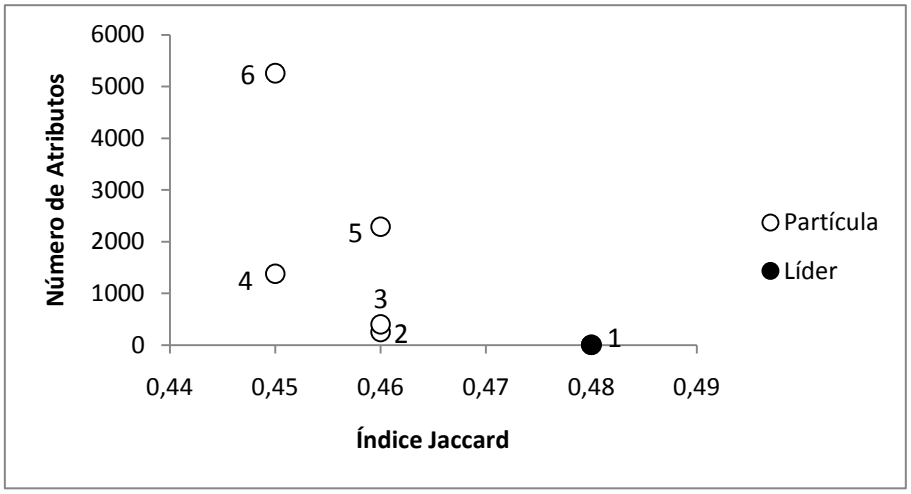


Figura 2. Posicionamento final das soluções das bases DLBCL-Tumor

Na Tabela 4, entre as soluções geradas pelas bases DLBCL-NIH, pode ser visto que a solução 1 otimizava todos os critérios, sendo o único líder do enxame. Depois de processados pelo MOPSO, a solução 2 também foi deslocada para a região promissora, tornando-se líder (Figura 3). Diferente dos demais experimentos até aqui vistos, a base de dados original dominou as solução 4 e 5, não sendo a pior solução.

Tabela 4: Soluções das bases DLBCL- NIH

Número do método	Número de atributos	Índice Jaccard
1	5	0.37
2	740	0.34
3	1850	0.353
4	3700	0.359
5	5550	0.36
6	7400	0.36

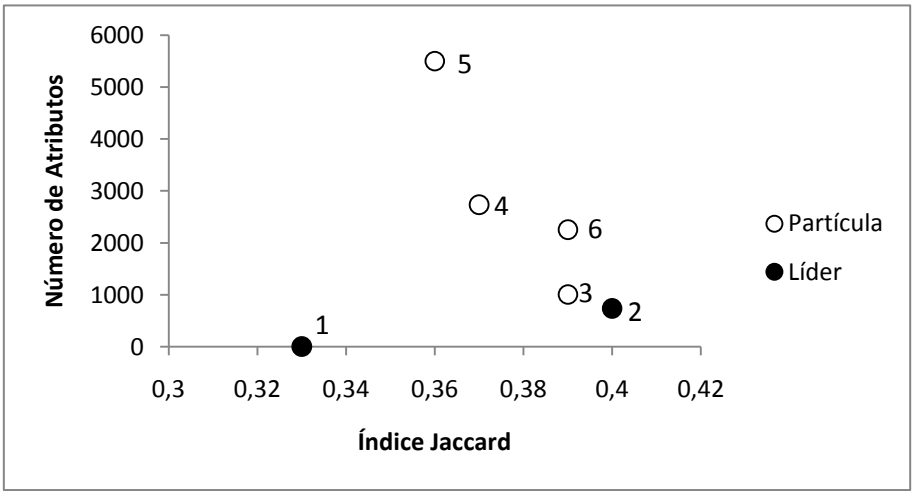


Figura 3. Posicionamento final das soluções das bases DLBCL-NIH

Os resultados das bases relacionadas a Leukemia-ALL/AML estão na Tabela 5. Apesar da solução 1 ter apenas um atributo em sua base, o valor obtido pelo índice Jaccard foi o pior entre as bases dessa característica. O melhor valor desse índice, ainda distante do valor máximo, foi obtida pela solução 5. Esse fato mostra que o agrupamento realizado foi ruim.

Tabela 5: Soluções das bases Leukemia-ALL/AML

Número do método	Número de atributos	Índice Jaccard
1	1	0.32
2	713	0.44
3	1782	0.44
4	3565	0.45
5	5347	0.47
6	7130	0.45

Na Figura 4 estão os posicionamentos das partículas após a execução do MOPSO. A solução 2, apesar de ter obtido o mesmo valor Jaccard que a partícula 3, domina-a, além das partículas 4,5 e 6, por ser composta por menos atributos. A outra solução tida como ótima é a de número 1, que se deslocou para a região de menor quantidade de atributos.

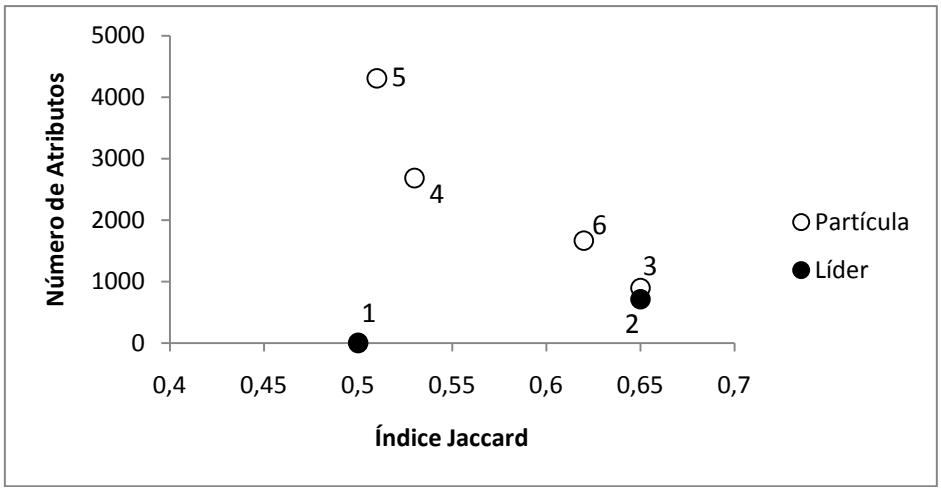


Figura 4. Posicionamento final das soluções das bases Leukemia-ALL/AML

No experimento realizado pelas bases Leukemia-MLL, a partícula 1 da Tabela 6 obteve os menores valores para o número de atributos e para o índice Jaccard. As soluções que obtiveram o melhor valor Jaccard foram as de números 5 e 6, esse último representando a base de dados original. Entretanto, como pode ser visto na Figura 5, as soluções que o MOPSO migrou para a região promissora foram as de números 1,2 e 3.

Tabela 6: Soluções das bases Leukemia-MLL

Número do método	Número de atributos	Índice Jaccard
1	3	0.21
2	1258	0.34
3	3145	0.36

4	6291	0.37
5	9437	0.53
6	12583	0.53

Nesse caso, mostra-se mais importante a necessidade de um *decision maker* para fazer a escolha pela melhor solução.

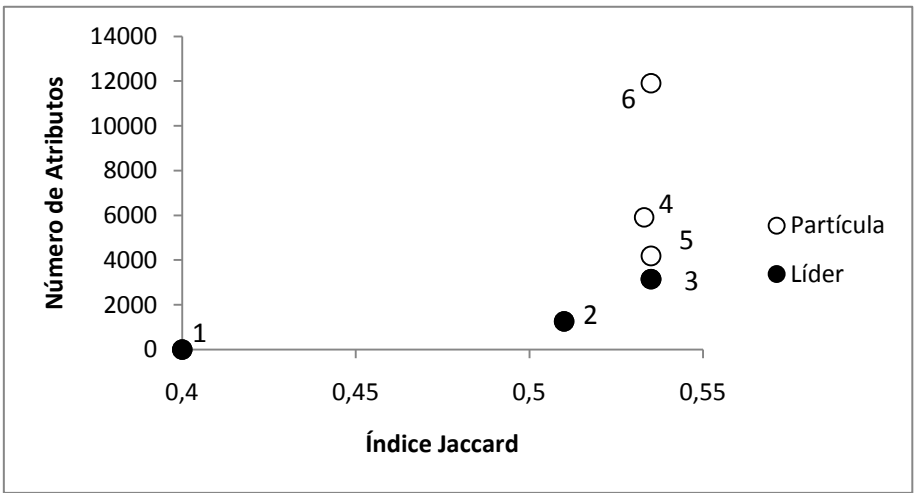


Figura 5. Posicionamento final das soluções das bases Leukemia-MLL

7. Conclusões

Este trabalho avaliou, por meio do MOPSO, métodos de seleção de atributos em bases de expressões gênicas obtidas pelo microarranjo. A primeira observação é que a redução de dimensionalidade de uma base melhora os resultados, já que nenhuma solução obtida pela base original foi classificada como ótima pelos experimentos realizados. Além disso, pelo fato de selecionar o menor subconjunto de atributos relevantes, o C-FOCUS sempre foi incluído como solução ótima, mostrando que poucos são os genes que realmente fazem parte do processo de ativação ou inibição de uma característica.

O método MOPSO baseado na frente de Pareto teve um desempenho satisfatório por ter selecionado coerentemente as soluções. Seu funcionamento evolucionário movimentou as soluções geradas pelas menores bases para as regiões mais promissoras, destacando-as das demais soluções do enxame.

Pelo fato de terem sido avaliadas soluções com poucos critérios, é possível analisá-las individualmente e comprovar a qualidade do balanceamento multiobjetivo, já que nos experimentos das Tabelas 3, 5 e 6, as soluções com o pior valor Jaccard foram selecionadas como líderes ao final do experimento por terem quantidade de atributos muito inferior das demais soluções e assim, migraram para a região promissora.

A regra de dominância de Pareto resultou vários líderes. Esse fato mostra a importância de um *decision maker* para decidir qual das soluções é a melhor. Um usuário que tenha conhecimento profundo das bases utilizadas poderá também validar o conjunto de atributos que foram selecionados, se eles realmente são os mais relevantes para cada característica abordada. Na falta do *decision maker*, o uso da regra de dominância de Pareto pode se tornar indevida pelo fato de gerar vários líderes. Nesse

caso, é mais interessante utilizar outro método multiobjetivo, que retorna apenas uma solução ótima.

A utilização do índice Jaccard mostrou que a etapa do agrupamento pode servir como avaliador dos métodos de seleção de atributos. Porém, o maior valor obtido, na solução 2 da Tabela 2, cujo valor é 0.56, mostra que o K-*means* não é a melhor escolha para agrupar dados de expressões gênicas. Além disso, caso não haja um conhecimento *a priori* das bases de dados, o K-*means* pode ser substituído pelo ISODATA, que tem o mesmo funcionamento e realiza um auto-balanceamento da quantidade de grupos [Xu e Wunsch 2005]. O mesmo ocorre com o índice Jaccard, que pode ser substituído por outro índice que não necessite de informações além das contidas na base de dados.

O objetivo principal deste trabalho foi mostrar que um método de otimização multiobjetivo pode avaliar métodos de seleção de atributos, escolher os melhores para uma característica específica, afim de tornar a análise de bases de dados de expressões gênicas obtidas pela técnica de microarranjo mais fácil e rápida.

Como proposta de trabalhos futuros para obtenção de resultados melhores e mais precisos, será estudado o método lexicográfico de otimização multiobjetivo, que estabelece uma sequência de preferência de critérios para escolher a melhor e única solução. Um método de agrupamento baseado em densidade, o DBSCAN, poderá ser usado para comparar os resultados obtidos pelo K-*means* neste trabalho.

Agradecimentos

Os autores deste trabalho gostariam de agradecer ao CNPq pelo suporte financeiro (projeto Seleção de Atributos Usando Critérios Multiobjetivo Baseados em Enxames para a Seleção de Genes em Microarranjos, referência 555264/2009-2).

Referências Bibliográficas

Azofra, A.A., Sanchez, J.M.B e Peña J.L.C. (2003) "C-FOCUS: A continuous Extension of FOCUS", Advances in Soft Computing - Engineering, Design and Manufacturing, p. 225-232.

Berkhin, P. (2002) "Survey of clustering data mining techniques", Technical report, Accrue Software, EUA.

Borges, H.B. (2006) "Redução de Dimensionalidade em Bases de Dados de Expressão Gênica", Dissertação de Mestrado, PPGIa-PUCPR.

Carvalho, A.B. e Pozo, A.T.R. (2009) "Otimização por Nuvem de Partículas Multiobjetivo na Aprendizagem Indutiva de Regras: Extensões e Aplicações", Dissertação de mestrado, Universidade Federal do Paraná.

Chuang, L.Y., Chang, H.W., Tu, C.J. e Yang, C.H. (2008), Computational Biology and Chemistry, Elsevier, vol. 32, p. 29-38.

Coello, C.A., Lamont, G.B. e Veldhuizen, D.A.V. (2007), Evolutionary Algorithms for Solving Multi-Objective Problems, segunda edição, Springer-Verlag.

Dash, M., Choi, K., Scheuermann, P. e Liu, H. (2002) "Feature Selection for Clustering – A Filter Solution", Em: 2nd International Conference on Data Mining, p. 115-122.

Dy, J. (2008) "Unsupervised Feature Selection", Computational Methods of Feature Selection, Editado por Huan Liu e Hiroshi Motoda, Chapman & Hall/CRC, p. 19-39.

D'Haeseleer, P. (2005) "How Does Gene Expression Clustering Work?", Nature Biotechnology, vol. 23, no. 12, p. 1499-1501.

Eberhart, R.C. e Shi, Y. (2001) "Particle Swarm Optimization: Developments, Applications and resources", Em: IEEE International Conference Evolutionary Computation, vol. 1, p. 81-86.

Faceli, K., Caravalho, A. e Souto, M. (2005) "Validação de Algoritmos de Agrupamento", Relatórios Técnicos do ICMC.

Handl, J. e Knowles, J. (2007) "An Evolutionary Approach to Multiobjective Clustering", Em: IEEE/ACM Transactions on Evolutionary Computation, vol. 11, no. 1, p. 56-76.

Kennedy, J. e Eberhart, R.C. (1995) "Particle Swarm Optimization", Em: IEEE International Conference on Neural Networks, p. 1942-1948, IEEE Press.

Kohavi, R. e John, G.H. (1997) "Wrappers for Feature Subset Selection", Artificial Intelligence, p. 273-324.

Kononenko, I. e Sikonja, M.R. (2008) "Non-Myopic Feature Quality Evaluation with (R)ReliefF", Computational Methods of Feature Selection, Editado por Huan Liu e Hiroshi Motoda, Chapman & Hall/CRC, p. 169-191.

Mostaghim, S. e Teich, J. (2003) "Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO)", Em: IEEE Swarm Intelligence Symposium, p. 26-33.

NIH- National Institutes of Health (2001) "Genetic Basic", http://www.nigms.nih.gov, acessado em 07 de Abril de 2011.

Sierra, M.R. e Coello, C.A.C. (2006) "Multi-objective Particle Swarm Optimizers: A Survey of The State-of-the-art", Em: International Journal of computational Intelligence Research, vol. 2.

Suresh, K., Kundu, D., Ghosh, S. e Das S. (2009) "Data Clustering Using Multi-objective Differential Evolution Algorithm", Fundamenta Informaticae, vol. 21, IOS Press, p. 1001-1024.

Xu, R. e Wunsch, D. (2005) "Survey of Clustering Algorithms", Em: IEEE Transactions on Neural Networks, vol. 16, no. 3.

Yu, L. e Liu, H. (2004) "Redundancy Based Feature Selection for Microarray Data", Em: 10th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.

A study on the selection of local training sets for hierarchical classification tasks

Jean Metz[1], **Alex A. Freitas**[2], **Maria Carolina Monard**[1], **Everton Alvares Cherman**[1]

[1]Institute of Mathematics and Computer Science, University of São Paulo (USP)
Av. Trabalhador São Carlense 400, Centro, CEP: 13566-590, São Carlos – SP – Brasil

[2]School of Computing, University of Kent
CT2 7NF, Canterbury – Kent – U.K.

{metzz,mcmonard,echerman}@icmc.usp.br, A.A.Freitas@Kent.ac.uk

Abstract. *In hierarchical classification tasks using the local approach, an important decision concerns the selection of training examples to build the local classifiers. To this end, several policies, which take into account the class taxonomy information, have been proposed. However, a study of a comprehensive comparison concerning the performance of these policies is still lacking. This paper presents a comprehensive empirical evaluation of eight different policies using 13 datasets. The results have shown that three of these policies outperformed the other five policies with statistically significant differences.*

1. Introduction

Hierarchical classification is the task of categorizing examples into a set of classes consistent with a predefined taxonomy organized in a hierarchical fashion. In other words, the classes in the taxonomy relate to each other by means of generalization and specialization relationships. Nowadays, a growing number of applications are specially suited for the task of classifying objects into hierarchical structures [Silla Jr and Freitas 2011].

Hierarchical classification methods can be categorized according to several different criteria, such as data structure and the use of hierarchical information during the training and prediction phase. Regarding the data structure, the class hierarchy is usually represented by a tree or a direct acyclic graph (DAG) [Vens et al. 2008], where the main difference is related to the number of parent classes per node. When a tree structure is used, each node has only one parent node, with the obvious exception of the root node that has no parent. On the other hand, whenever DAG is used, each non-root node is allowed to have more than one parent. A direct consequence of this fact is the possibility of reaching a certain node going through various paths in the class hierarchy. This increases the complexity of the methods that use the DAG data structure. This paper focuses on tree-based class hierarchy.

Considering the use of hierarchical information, the methods can be divided into two types [Kiritchenko et al. 2006]: *global* (or big-bang) and *local* (or top-down level-based). A learning approach is called global if it builds only one classifier to discriminate all categories in a hierarchy, somehow respecting the relationships between classes in different levels of the taxonomy, and it is called local if it builds separate classifiers associated to some of the nodes in the hierarchy, called local classifiers. During the classification stage, a local approach usually proceeds in a top-down fashion, also known as

Pachinko Machine strategy, first predicting the most relevant category(ies) of the top level and then recursively predicting its child classes. This top-down prediction strategy stops when a stop criterion is met or a leaf node is reached [Sun et al. 2003].

Furthermore, according to [Freitas and de Carvalho 2007], there are two distinct broad types of prediction strategy for hierarchical classification methods: *Mandatory leaf node prediction* (MLN), where the most specific class assigned to an instance must be a leaf class, and *Non-mandatory leaf node prediction* (NMLN), where the most specific class assigned to an instance may be either an internal or a leaf class in the class taxonomy.

A popular strategy for hierarchical classification using local methods builds a local binary classifier for each node except the root. We call this method *Hierarchical Binary Relevance* (HBR) in accordance with the *Binary Relevance* method for multilabel classification introduced in [Tsoumakas et al. 2010]. It is worth noting that a hierarchical classification task is in essence a multilabel task, since each example is possibly associated with more than one class at a time, one for each level of the taxonomy. However, in the hierarchical context, the multiple labels assigned to each instance obey a specific rule respecting the hierarchical organization of classes. In this paper, we focus on this local HBR approach, which has the advantage of having a standard flat classification algorithm which can be used to produce the local classifiers, avoiding the need for designing a complex global hierarchical classification algorithm.

In the HBR approach, the main question is how to select the local training sets used by the learning algorithm running at each node of the class hierarchy. Using a straightforward strategy, for each class in the taxonomy, the examples of the entire training set are labeled as positive if the instance is annotated with the respective class and negative otherwise, resulting in a new binary training set of examples which is used to build the local classifier. This naive approach solves the problem of selecting local training sets, but it does not consider the hierarchical information in the class taxonomy and, consequently, may result in suboptimal training sets.

Nonetheless, there are several other methods that can be applied to refine this selection process, considering local information and paying attention to the topology of the classification scheme. Although these methods have been proposed in the literature, to the best of our knowledge, there is no work presenting a comprehensive empirical comparison of training set selection methods across several different application domains. In this work, we experimentally analyzed the results obtained by 8 different training set selection policies in 13 hierarchical datasets. Six of these policies are based on mathematics set operations and 2 are based on distance between the examples. Results show that the *set operation based* policies are more efficient and effective than the *distance based* policies. Moreover, three of the *set operation based* policies present better results.

The remainder of this work is organized as follows: Section 2 presents fundamental concepts of the hierarchical classification tasks, as well as the selection of local training set policies and the evaluation measures. Moreover, Section 2 also discusses related work. The experimental scenarios and analyses are presented in Section 3. Finally, in Section 4 the conclusions and the future work directions are given.

2. Hierarchical classification

As stated before, a tree data structure or a DAG can be used to represent the class hierarchy. In this work, we use methods based on the tree data structure and assume the existence of the root class, which is the ancestor of all the other categories in the hierarchy. Thus, the class taxonomy is defined as a partially order finite set $\tau = (L, \prec)$ that enumerates all class concepts (labels) in the application domain. $L = \{y_1, y_2, ..., y_q\}$ represents the finite set of class labels and the operator $\prec$ represents the *is-a* relationship among the labels in L. Thus, whenever an instance belongs to a class y_j it also belongs to all ancestor nodes of that class (*true path rule*). In this way, the assigned labels would be clearly consistent with the class taxonomy structure. Following [Fagni and Sebastiani 2007], other operators to describe hierarchical relationships are defined in Table 1.

Table 1. Hierarchical relationships.

Operator	Description
$\uparrow y_j$	The parent of class y_j
$\downarrow y_j$	The set of child classes of the class y_j
$\Uparrow y_j$	The set of ancestor classes of the class y_j
$\Downarrow y_j$	The set of descendant classes of the class y_j
$\leftrightarrow y_j$	The set of sibling classes of the class y_j

The HBR is a popular local approach for hierarchical classification. In order to build each local classifier in the HBR method, it is necessary to select the local training set related to each class y_j, namely Tr_{y_j}. Therefore, HBR builds $|L| - 1$ local binary classifiers considering the respective training set, *i.e.*, one classifier for each class y_j in the taxonomy except the root node — Equation (1).

$$HBR = \{C_{y_j}(\mathbf{x}, y_j) \rightarrow \hat{y}_j \in \{0, 1\} |\ y_j \in (L, \prec)\ \wedge\ y_j \neq root\} \tag{1}$$

Each of these local binary classifiers is able to predict whether a given example is positive ($\hat{y}_j = 1$) or negative ($\hat{y}_j = 0$) for the respective class. Using a top-down prediction strategy, whenever an example is predicted as positive, it is recursively presented to the local child classifiers to perform the classification. This process continues until a stop criterion is met or a leaf node is reached. The final hierarchical classification for a given example is the aggregation of the labels positively predicted by all the independent binary classifiers that respect the *true path rule*. It is worth noting that this method is naturally multilabel in the sense that each example may be classified in different branches of the class taxonomy. This aspect may be avoided by only assigning to the given example the most likely class at each level, forcing only the children of the assigned class to classify the example at the next level. In this work, we apply this strategy to constrain each example to be assigned just one class at each level of the class hierarchy. Next we present the local training set selection policies studied in this paper.

2.1. Local training set selection policies

Supervised learning is a two-stage process where known examples (labeled examples) of a domain, named global training set Tr, are used by the learning algorithm to build (train) a classifier, which is used during the second stage to make predictions of examples whose classifications are unknown (unlabeled examples).

During the HBR training stage, for each class y_j in the taxonomy a local classifier C_{y_j} is built by a learning algorithm using the positive set of examples $Tr^+_{y_j}$ (those that

represent the node y_j) and the negative set of examples $Tr^-_{y_j}$ (those that represent instances outside the node y_j). Several policies have been proposed to select these sets of examples from Tr in order to build each local classifier C_{y_j}, described in Table 2.

Six of theses policies are defined based on mathematical operations over sets of examples considering the class taxonomy and the most specific class assigned to each example in Tr. Furthermore, the operator $*$, which is applied to a set of class labels, represents all examples *annotated* with one of the specified labels as its most specific class. The other two policies (*B-Global* and *B-Local*) are based on the distance between examples.

Table 2. Policies for local training sets selection.

Policy	$Tr^+_{y_j}$	$Tr^-_{y_j}$	Reference
All (*L-Inclusive*)	$*(y_j \cup \Downarrow y_j)$	$Tr - Tr^+_{y_j}$	[Mladenić and Grobelnik 1998]
Hierarchical training set (*S-Inclusive* §)	$*(y_j \cup \Downarrow y_j)$	$*(\leftrightarrow y_j \cup \Downarrow (\leftrightarrow y_j))$	[Ceci and Malerba 2003]
Proper training set (*S-Exclusive* §)	$*(y_j)$	$*(\leftrightarrow y_j)$	[Ceci and Malerba 2003]
Inclusive	$*(y_j \cup \Downarrow y_j)$	$Tr - (Tr^+_{y_j} \cup *(\Uparrow y_j))$	[Eisner et al. 2005]
Exclusive	$*(y_j)$	$Tr - Tr^+_{y_j}$	[Eisner et al. 2005]
L-Exclusive	$*(y_j)$	$Tr - *(y_j \cup \Downarrow y_j)$	[Eisner et al. 2005]
B-Global	$*(y_j \cup \Downarrow y_j)$	$\underset{E^- \notin Tr^+_{y_j}}{\overset{\eta}{\operatorname{argmin}}} \sigma(\zeta_{y_j}, E^-)$	[Fagni and Sebastiani 2007]
B-Local	$*(y_j \cup \Downarrow y_j)$	$\underset{E^- \notin Tr^+_{y_j}}{\overset{\eta}{\operatorname{argmin}}} \sum_{E^+ \in \chi^{\eta}_{E^-}} \sigma(E^+, E^-)$ $\chi^k_{E^-} = \underset{E^+ \in Tr^+(y_j)}{\overset{k}{\operatorname{argmin}}} \sigma(E^-, E^+)$	[Fagni and Sebastiani 2007]

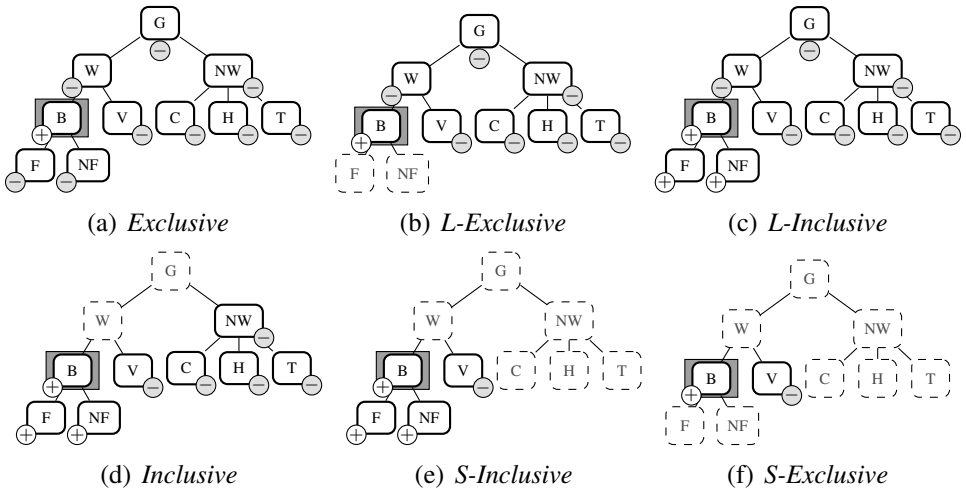
§ The *S* in these policies means siblings.

The rationale behind the *All* policy [Mladenić and Grobelnik 1998] is the use of hierarchical information to select the local training set Tr_{y_j}. Therefore, examples explicitly annotated with y_j as well as any of the descendant classes of y_j are included in $Tr^+_{y_j}$, since by definition $\forall y_k \in (\Downarrow y_j), y_k \prec y_j$. All examples not included in the positive set are used as negative training examples. In other words, $Tr^-_{y_j} = Tr - Tr^+_{y_j}$. This policy was later used in [Eisner et al. 2005] and was called *Less-Inclusive*. From now on, the *L-Inclusive* term is adopted to refer to this policy.

The concept of *exclusive* and *inclusive* classifiers, presented in [Eisner et al. 2005], can be used to clarify the definitions of some policies. According to the authors, a hierarchical classifier is considered *exclusive* if for a given instance which is to be assigned to a class y_j, only the local classifier C_{y_j} predicts positive, while every other local classifier in the hierarchy predicts negative, including the predecessor classifiers of C_{y_j}. In other words, the C_{y_j} classifier excludes (negates) every instance except the ones explicitly labeled with class y_j, *i.e.*, *excludes* all examples in the set $\neg(*y_j)$. On the other hand, the classifier is *inclusive* whenever an example belongs to $*y_j$ and the classifier C_{y_j}, as well as all ancestor classifiers predict positive for this example. Thus, it considers (includes) the prediction of the ancestor classifiers and not only C_{y_j}.

All the 6 *set operator* based policies work in a similar way, first selecting the examples in $Tr^+_{y_j}$ based on the most specific class ($*y_j$), and applying afterwards a mathematical set operation over Tr and $Tr^+_{y_j}$ to define the $Tr^-_{y_j}$ set — Table 2. The main difference among the 6 policies lies on the operation used and whether it is applied over all examples not in $Tr^+_{y_j}$ or a sub-set of it. Figure 1 illustrates the use of these policies to select the local training set of class B, namely Tr_B, where $L =$

$\{G, W, B, F, NF, V, Z, NW, C, H, T\}$. The gray square indicates the target class for which the training set is selected. The other class nodes participate with different roles for each policy, determining whether an example is positive ($\oplus$), negative ($\ominus$) or ignored (nodes with dotted lines).



(a) *Exclusive* (b) *L-Exclusive* (c) *L-Inclusive*

(d) *Inclusive* (e) *S-Inclusive* (f) *S-Exclusive*

Figure 1. Illustration of the 6 set operation based policies.

According to the *S-Inclusive* policy, $Tr^-_{y_j}$ is chosen among the training examples that are not positive for C_{y_j} and may be assumed to be mostly correlated to y_j, *i.e.*, it consists of all training examples labeled with classes that are siblings of the positive class or descendants of those siblings. The basic idea is that $Tr^-_{y_j}$ includes the "quasi-positive" examples of y_j [Schapire et al. 1998]. In other words, the examples not in $Tr^+_{y_j}$ that are closest to the boundary between the positive and the negative region of y_j, and are thus the most informative negative examples that can be used to build the local classifier. This is beneficial also from the standpoint of efficiency, since fewer training examples are involved. A variant of the *S-Inclusive* policy is called *Siblings-Exclusive* (*S-Exclusive*), which restricts the selection of examples to the ones explicitly labeled with class y_j and their siblings, ignoring descendant classes.

The *distance* based policies (*B-Global* and *B-Local*) [Fagni and Sebastiani 2007] also make use of the "quasi-positive" notion, more precisely, the "query-zoning" selection strategy introduced in [Singhal et al. 1997]. In the vector space model, a query zone can be envisioned as a volume of the vector space around a query object. In the context of classification, a query zone for a new example can be simulated by considering a set of training examples that have some similarity to the new example. Moreover, this concept can be interpreted as a boundary region on the input space between positive and non-positive examples. Therefore, it can be used to select the non-positive examples that lie on the query-zone of the $Tr^+_{y_j}$ examples. The basic idea is that more informative examples that lie on the borderline between positive and negative classes might increase the classifier's prediction power with a thinner adjustment on the hyperplanes between classes.

In both *distance* based policies, the $Tr^+_{y_j}$ examples are defined as the set of examples with the most specific class label in the set $\{y_j \cup (\Downarrow y_j)\}$, *i.e.*, the examples explicitly labeled with one of the classes in the sub-tree rooted at y_j. The $Tr^-_{y_j}$ examples are selected

from the non-positive example set, depending on the policy. To this end, the *B-Global* policy finds the centroid of $Tr^{+}_{y_j}$ and calculates the distances between the centroid and each non-positive example. After that, it selects the η closest non-positive examples and includes them in $Tr^{-}_{y_j}$. The *B-Local* policy uses a similar approach, but tries to identify the "quasi-positive" examples within a more complex boundary region. In order to find this boundary, *B-Local* uses not only one centroid but a small subset of k positive examples to identify the closest non-positive examples while considering the overall shape of the positive example space. Putting it in a different perspective, the *B-Local* policy calculates for each non-positive example the sum of the distances to the k closest positive examples, and selects the η non-positive examples that minimizes the calculated distance.

Although some papers report the impact of these policies on the prediction performance of the HBR method, summarized in Table 3, to the best of our knowledge there is no study showing conclusive experimental results comparing all these policies on different domains and/or several datasets. Note that each of the projects mentioned in Table 3 carried out experiments with at most 4 training set selection policies and 2 datasets, whilst in this paper we perform experiments with 8 policies and 13 datasets.

Table 3. Policies considered in related work (▲ better, and ▼ worse).

Reference	Policy							
	Exclusive	*L-Exclusive*	*L-Inclusive*	*Inclusive*	*S-Inclusive*	*S-Exclusive*	*B-Global*	*B-Local*
[Ceci and Malerba 2003]					▲	▼		
[Eisner et al. 2005]	▼	▼	▲	▲				
[Fagni and Sebastiani 2007]			▲		▲		▼	▼

2.2. Hierarchical classifier evaluation

Conventional flat classification measures are not suitable for hierarchical classification tasks, since they cannot handle different types of misclassification errors present in the hierarchical scenario. Intuitively, a sibling or a parent node misclassification is less harmful then a more distant node misclassification. For that reason, specialized measures consistent with conventional non-hierarchical measures, as well as with the class taxonomy, allowing for less severe misclassification errors, such as generalization and specialization errors, are needed. Furthermore, flat classification measures cannot handle a partially correct prediction, which is essential to evaluate hierarchical classifiers performance. In this work, the hierarchical version of the standard precision and recall measures proposed in [Kiritchenko et al. 2006] and defined by Equations (2) and (3), respectively, are used, where TP refers to *true positive*, FP to *false positive* and FN to *false negative*. It is important to mention that these measures are actually a micro-average of the traditional precision (Pr_j) and recall (Re_j) obtained for each label in the hierarchy. The use of averaged measures contributes to a more precise overall evaluation of the experimental results, since local evaluation would produce as many precision and recall values as the number of classes in the taxonomy, making the analysis impractical when there are too many classes in the taxonomy.

Another measure commonly used combines the precision and recall values into one single value called *f-Measure* — Equation (4). In our experimental evaluation, we used $\beta = 1$, giving precision and recall equal weights.

$$hPr^{\mu} = \frac{\sum_{j=1}^{|L|} |TP_{y_j}|}{\sum_{j=1}^{|L|}(|TP_{y_j}| + |FP_{y_j}|)} \quad (2) \qquad hRe^{\mu} = \frac{\sum_{j=1}^{|L|} |TP_{y_j}|}{\sum_{j=1}^{|L|}(|TP_{y_j}| + |FN_{y_j}|)} \quad (3)$$

$$F_{\beta} = \frac{(\beta^2 + 1) \times hPr^{\mu} \times hRe^{\mu}}{\beta^2 \times hPr^{\mu} + hRe^{\mu}};\ where \beta \in [0, \infty) \quad (4)$$

3. Experimental Analysis

The main purpose of this experimental analysis is to assess the impact of the 8 different policies to select the local training sets on the performance of the HBR method.

3.1. Datasets and Protocol

The experiments were carried out with 13 datasets from three different domains. Table 4 describes the datasets, where #E, #F and $|L|$ are the number of examples, number of features and number of class labels, respectively. Column $|L|$ *per level* shows the number of labels in each level of the class taxonomy, except for the root. The last column (*Label Cardin.*), shows the average number of class labels per example.

Table 4. Datasets description.

Dataset	Domain	MLN	#E	#F	$\|L\|$	$\|L\|$ *per level*	*Label Cardin.*
HGlass	Glass Identification	yes	214	9	9	2, 5, 2	2.682
Fabien	Music Genre Classification	yes	4188	40	20	2, 9, 9	2.645
Marsyas	Music Genre Classification	yes	4188	30	20	2, 9, 9	2.645
ThomasRH	Music Genre Classification	yes	4188	60	20	2, 9, 9	2.645
ThomasSSD	Music Genre Classification	yes	4188	168	20	2, 9, 9	2.645
G-Pfam †	Protein Function	no	7053	75	192	12, 52, 79, 49	2.841
G-Prints †	Protein Function	no	5404	283	179	8, 46, 76, 49	3.009
G-Prosite †	Protein Function	no	6246	129	187	9, 50, 79, 49	2.951
G-Interpro †	Protein Function	no	7444	450	198	12, 54, 82, 50	2.823
E-Pfam ‡	Protein Function	no	13987	708	333	6, 41, 96, 190	3.667
E-Prints ‡	Protein Function	no	14025	382	351	6, 45, 92, 208	3.699
E-Prosite ‡	Protein Function	no	14041	585	324	6, 42, 89, 187	3.690
E-Interpro ‡	Protein Function	no	14027	1216	330	6, 41, 96, 187	3.660

†: Classes are functions of GPCR proteins; ‡: Classes are functions of enzymes

The HBR method with top-down prediction strategy implemented in HARPIA [1], a framework for hierarchical classification, was used. This framework makes extensive use of the Weka[2] java library for machine learning. The experiments were carried out using two different flat classification algorithms to build the local binary classifiers: J48, a decision tree learning algorithm and Naïve Bayes (NB), both implemented in the Weka library [Witten et al. 2011].

The 8 selection policies were used to construct the training sets to build the local classifiers. As the *distance* based policies are parametric, for a fair comparison with the *set operator* based policies, we set up the number of negative examples (η) to be equal to the number of positive examples. Moreover, 5 different values for k (1, 3, 5, 7 and 9) were used with the *B-Local* policy.

All the reported results were obtained using 5×2-fold cross validation with paired folds, *i.e.*, the same training and testing partitions were used to assess the performance of

[1]http://labic.icmc.usp.br/software-and-application-tools

[2]http://www.cs.waikato.ac.nz/ml/weka

all different configurations of the HBR method. In order to analyze whether there is a difference among the policies impact on the predictive accuracy, we run the Friedman's test[3] with the null hypothesis that the performance of the algorithms using any of the policies are comparable considering all the results. When the null hypothesis is rejected by the Friedman's test, at 95% of confidence level, we proceed with the Nemenyi's post-hoc test to detect which differences among the methods are significant. According to this test, the performance of two methods is significantly different if the corresponding average ranks differ by at least the critical difference (CD). The results and discussion are presented next.

3.2. Evaluation

The evaluation considers (a) whether the degree of *inclusion* affects the overall performance of the algorithm when using a *set operation* based policy; (b) analyzing the effect of the k parameter for the *B-Local* policy; and (c) the overall comparison of all policies.

Figure 2 shows the results obtained with J48 as a base classifier using *set operation* based policies, where policies are indicated on the X-axis and the average of the $F_1^{\mu 4}$ values are shown on the Y-axis. For the sake of visualization, only 6 datasets are shown in this figure, since the results for some datasets were similar. More specifically, there is no significant difference using *E-Pfam*, *E-Prints*, *E-Prosite* and *E-Interpro* datasets. The same happened for other pairs of datasets, such as *G-Pfam* and *G-Prosite*; *G-Interpro* and *G-Prints*; *Marsyas* and *ThomasSSD*; and finally *Fabien* and *ThomasRH*. Due to space restriction, as the results using NB as the base classifier follow the same pattern, the corresponding figure is omitted[5].

Figure 2 clearly shows the tendency of improvement on the overall performance of the HBR method using more *inclusive* policies, as for all cases the *exclusive* policies were outperformed by the *inclusive* ones. In fact, when using *exclusive policies* with datasets where all examples are annotated with only leaf classes, the local training sets selected to build the the internal classifiers will be empty, resulting in very poor classifiers that reject (predict negative) any new example. These "empty" classifiers have a strong influence on the overall degradation of the HBR global performance. Even when the datasets have examples explicitly annotated with internal classes, there may exist some classes with no examples, also contributing to the degradation of the final model.

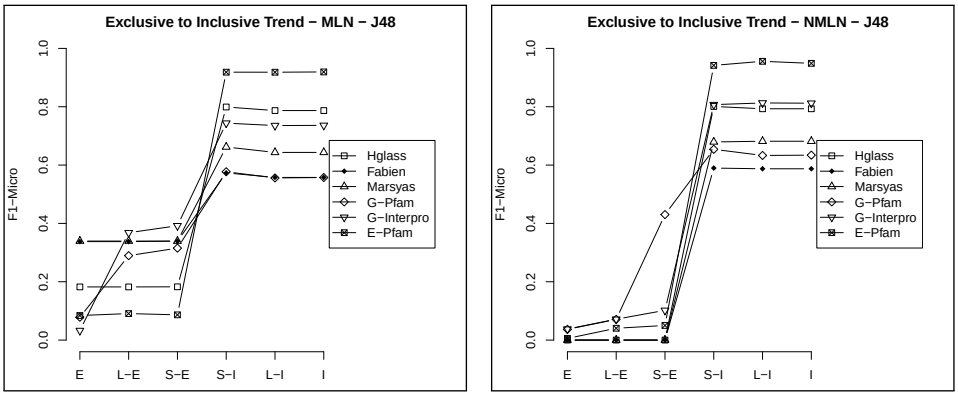
When the prediction type is set to MLN (Figure 2(a)), *i.e.*, mandatory leaf node prediction, even when examples are classified as negative by all the classifiers at a certain level, they are inspected by all the immediate lower level classifiers in the branch where the confidence on the negative prediction is lower ("less negative"), *i.e.*, the classification continues through the branch being mostly "positive". This means the final classification is incorrect with a high probability, or even random. On the other hand, if the prediction type is NMLN, *i.e.*, non-mandatory leaf node prediction, whenever an example is predicted as negative by all classifiers at a certain level, it is not inspected by deeper classifiers.

As the first level local classifiers might be "empty" for all *exclusive* policies, they would predict any presented examples as negative, resulting in cumulative errors and

[3]See [Demsar 2006] for a thorough discussion regarding statistical tests in machine learning research.

[4]F-Measure using $\beta = 1$ and the micro version of the precision (hPr^{μ}) and recall (hRe^{μ}).

[5]All tabulated results can be found in http://www.icmc.usp.br/ metzz/archives/harpia/

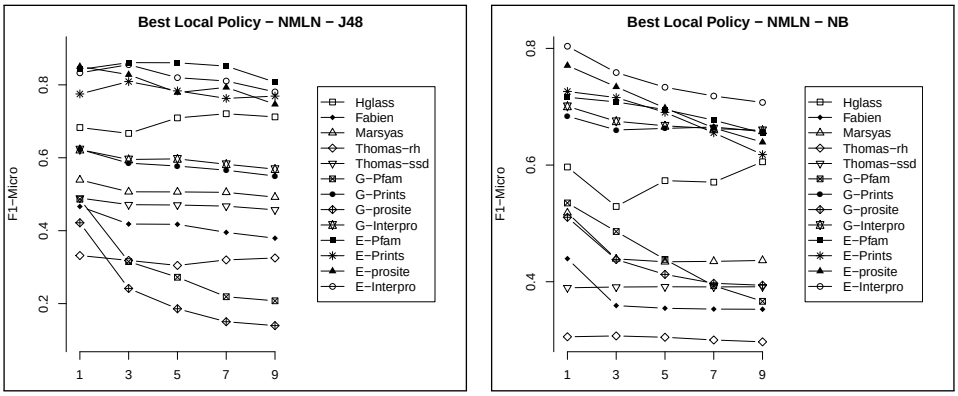


(a) (b)

Figure 2. Set operator based policies. *Exclusive* (E), *L-Exclusive* (L-E), *S-Exclusive* (S-E), *S-Inclusive* (S-I), *L-Inclusive* (L-I) and *Inclusive* (I).

degrading the HBR global performance. This explains Figure 2(b), where the use of the policies *Exclusive*, *L-Exclusive* and *S-Exclusive* with the datasets *Fabien* and *Marsyas* resulted in 0.00 F_1^μ. Similar results were obtained with *G-Pfam*, *G-Interpro* and *E-Pfam* datasets, showing F_1^μ values close to zero. In contrast, the more *inclusive* policies, which prevent the creation of classifiers with empty training sets, show better results.

As the *exclusive* policies performed clearly worse than the *inclusive* ones, from now on we focus our analysis on the *inclusive* and *distance* based policies. Figure 3 shows the effect of the variation on the k parameter for the *B-Local* policy using the NMLN prediction strategy, where the different k values are indicated on the X-axis and the average of the F_1^μ values are shown on the Y-axis. It can be observed that, in most cases, increasing the value of k results in smaller F_1^μ values. The only exception is for the *HGlass* dataset. In some cases, the effect of increasing k shows slight degradation or even no degradation at all, as for *ThomasRH* and *ThomasSSD* datasets in Figure 3(a). In other cases, there is a considerable deterioration of F_1^μ, such as with *G-Pfam* and *G-Prosite* datasets. Similar behavior was observed for the MLN prediction strategy. As the best F_1^μ values of the *B-Local* policy were obtained for $k = 1$, we proceed our analysis of the *B-Local* policy only considering $k = 1$.



(a) (b)

Figure 3. The influence of the k parameter value on the *B-Local* policy.

The results for the *inclusive* and *distance* based policies are presented in Table 5, which is divided into 4 quarters, each one corresponding to a combination of one base

classifier (J48 or NB) and one prediction type (MLN or NMLN). For each quarter, the average of the F_1^μ measure obtained by the HBR method using the different policies to generate the local training sets is shown. The numbers in brackets correspond to the standard deviation. Moreover, the last line (labeled as # wins) shows the number of datasets for which the respective policy achieved the greatest F_1^μ value. For the sake of visualization, the best values for each dataset in each quarter are shown in bold.

Table 5. Average F_1^μ values using NB and J48 with MLN and NMLN prediction types.

	J48									
Dataset	*B-Local* $k-1$	*B-Global*	*S-Inclusive*	*L-Inclusive*	*Inclusive*	*B-Local* $k-1$	*B-Global*	*S-Inclusive*	*L-Inclusive*	*Inclusive*
	MLN					NMLN				
HGlass	0.68(0.09)	0.68(0.07)	**0.80(0.02)**	0.79(0.02)	0.79(0.02)	0.68(0.09)	0.68(0.07)	**0.80(0.03)**	0.79(0.02)	0.79(0.02)
Fabien	0.47(0.03)	0.38(0.03)	**0.57(0.01)**	0.56(0.01)	0.56(0.01)	0.47(0.03)	0.37(0.03)	**0.59(0.01)**	**0.59(0.01)**	**0.59(0.01)**
Marsyas	0.53(0.02)	0.47(0.02)	**0.66(0.01)**	0.64(0.01)	0.64(0.01)	0.54(0.02)	0.47(0.02)	**0.68(0.01)**	**0.68(0.01)**	**0.68(0.01)**
ThomasRH	0.34(0.03)	0.30(0.03)	**0.53(0.01)**	0.52(0.01)	0.52(0.01)	0.33(0.03)	0.29(0.03)	**0.55(0.01)**	**0.55(0.01)**	**0.55(0.01)**
ThomasSSD	0.49(0.02)	0.42(0.03)	**0.66(0.01)**	0.65(0.01)	0.65(0.01)	0.49(0.02)	0.42(0.03)	0.68(0.01)	**0.69(0.01)**	**0.69(0.01)**
G-Pfam	0.44(0.04)	0.33(0.05)	**0.58(0.01)**	0.56(0.01)	0.56(0.01)	0.49(0.04)	0.38(0.05)	**0.65(0.00)**	0.63(0.01)	0.63(0.01)
G-Prints	0.59(0.05)	0.56(0.05)	**0.73(0.01)**	0.72(0.01)	0.72(0.01)	0.62(0.05)	0.58(0.04)	0.80(0.00)	**0.81(0.01)**	**0.81(0.00)**
G-Prosite	0.39(0.03)	0.22(0.05)	**0.56(0.01)**	0.54(0.01)	0.54(0.01)	0.42(0.02)	0.25(0.05)	**0.64(0.01)**	0.61(0.01)	0.61(0.00)
G-Interpro	0.60(0.07)	0.58(0.04)	**0.74(0.01)**	**0.74(0.00)**	**0.74(0.00)**	0.62(0.07)	0.60(0.04)	**0.81(0.00)**	**0.81(0.00)**	**0.81(0.00)**
E-Pfam	0.80(0.06)	0.57(0.10)	**0.92(0.00)**	**0.92(0.00)**	**0.92(0.00)**	0.84(0.06)	0.63(0.09)	0.94(0.00)	**0.96(0.00)**	0.95(0.00)
E-Prints	0.74(0.04)	0.51(0.03)	**0.90(0.00)**	**0.90(0.00)**	**0.90(0.00)**	0.78(0.04)	0.56(0.03)	0.92(0.00)	**0.94(0.00)**	0.93(0.00)
E-Prosite	0.82(0.06)	0.58(0.14)	**0.93(0.00)**	0.92(0.00)	**0.93(0.00)**	0.85(0.06)	0.63(0.14)	0.94(0.00)	**0.96(0.00)**	0.95(0.00)
E-Interpro	0.79(0.06)	0.72(0.03)	**0.93(0.00)**	0.92(0.00)	**0.93(0.00)**	0.83(0.05)	0.77(0.03)	0.95(0.00)	**0.96(0.00)**	**0.96(0.00)**
# wins	0	0	13	3	5	0	0	7	10	7
	NB									
HGlass	0.62(0.08)	0.63(0.08)	0.68(0.04)	**0.75(0.03)**	**0.75(0.03)**	0.60(0.08)	0.61(0.07)	0.68(0.04)	**0.75(0.03)**	**0.75(0.03)**
Fabien	0.44(0.04)	0.37(0.01)	0.51(0.01)	**0.52(0.01)**	**0.52(0.01)**	0.44(0.04)	0.37(0.01)	0.51(0.00)	**0.52(0.01)**	**0.52(0.01)**
Marsyas	0.53(0.03)	0.48(0.01)	**0.65(0.00)**	**0.65(0.00)**	**0.65(0.00)**	0.52(0.03)	0.47(0.01)	**0.66(0.00)**	**0.66(0.00)**	**0.66(0.00)**
ThomasRH	0.35(0.01)	0.34(0.00)	**0.51(0.00)**	**0.51(0.00)**	**0.51(0.00)**	0.31(0.00)	0.29(0.01)	**0.51(0.00)**	**0.51(0.00)**	**0.51(0.00)**
ThomasSSD	0.45(0.01)	0.42(0.01)	**0.60(0.01)**	0.57(0.01)	0.57(0.01)	0.39(0.01)	0.40(0.01)	**0.60(0.01)**	0.57(0.01)	0.57(0.01)
G-Pfam	0.53(0.01)	0.46(0.01)	**0.57(0.00)**	**0.57(0.00)**	**0.57(0.00)**	0.54(0.01)	0.48(0.01)	**0.64(0.00)**	**0.64(0.00)**	**0.64(0.00)**
G-Prints	**0.66(0.01)**	0.63(0.01)	**0.66(0.00)**	**0.66(0.00)**	**0.66(0.00)**	0.68(0.01)	0.63(0.01)	**0.72(0.01)**	**0.72(0.01)**	**0.72(0.01)**
G-Prosite	0.50(0.01)	0.41(0.01)	**0.55(0.01)**	**0.55(0.01)**	**0.55(0.01)**	0.51(0.01)	0.42(0.01)	0.61(0.00)	0.61(0.00)	**0.62(0.00)**
G-Interpro	**0.68(0.01)**	0.67(0.01)	0.66(0.01)	0.66(0.01)	0.66(0.01)	0.70(0.01)	0.68(0.01)	0.72(0.00)	**0.73(0.00)**	**0.73(0.00)**
E-Pfam	0.68(0.03)	0.55(0.03)	0.80(0.00)	0.80(0.00)	**0.81(0.00)**	0.72(0.02)	0.59(0.03)	**0.82(0.00)**	0.79(0.01)	0.79(0.00)
E-Prints	0.70(0.02)	0.58(0.03)	0.83(0.00)	**0.84(0.00)**	**0.84(0.00)**	0.73(0.02)	0.61(0.03)	**0.85(0.00)**	0.84(0.00)	0.84(0.00)
E-Prosite	0.75(0.04)	0.65(0.02)	**0.85(0.00)**	**0.85(0.00)**	**0.85(0.00)**	0.77(0.04)	0.67(0.02)	**0.87(0.00)**	0.86(0.00)	0.86(0.00)
E-Interpro	0.77(0.02)	0.77(0.01)	**0.80(0.00)**	**0.80(0.00)**	**0.80(0.00)**	0.80(0.02)	0.81(0.02)	**0.83(0.01)**	0.81(0.00)	0.81(0.00)
# wins	2	0	8	10	11	0	0	9	7	8

An interesting aspect to be evaluated is whether the MLN prediction is better than the NMLN prediction. Considering the performance for each dataset, NMLN has outperformed MLN in more than 80% of the experiments, while for the other portion, both prediction types performed equally. Summing up, NMLN would be a preferable prediction type.

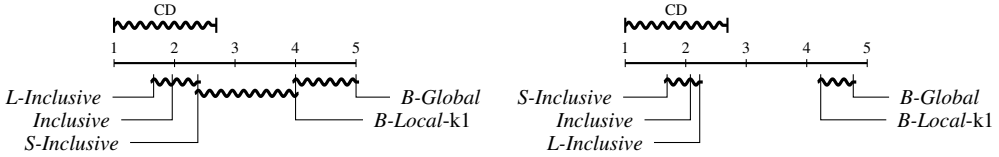
Regarding the policies and considering the number of times that the highest F_1^μ value was obtained (Table 5), it is easily observed that the *S-Inclusive* policy outperforms the others, and that the *set operation* based policies outperform the *distance* based policies. These numbers, for the 52 experiments, are summarized in Table 6.

Table 6. Comparison of Policies.

B-Local $k-1$	*B-Global*	*S-Inclusive*	*L-Inclusive*	*Inclusive*
2	0	37	30	31

In order to analyze whether the difference among the policies impact on the HBR performance, we run the Friedman's hypothesis test with the null hypothesis that the different policies do not affect the final performance of the HBR method. As the hypothesis

was rejected, we run the post-hoc test Nemenyi, that shows significant differences between methods whenever their average ranks differ by at least the critical value (CD). The results of the Nemenyi's test can be represented in a simple diagram (Figure 4), where the main line is the X-axis on which the average ranks of the methods are plotted (lower average rank is better). The secondary lines beneath the main axis connect groups that are not significantly different. Moreover, the critical difference (CD) is shown just above the main line. These diagrams ensure the observations aforementioned, that the *set operation* based policies are better than the *distance* based policies. Notwithstanding, they also show that there is no statistically significant difference among *S-Inclusive*, *L-Inclusive* and *Inclusive* policies. Thus, a good criterion to choose one among these policies could be the number of examples included in the negative local training set, since the smaller the set the faster the training stage is. Taking this into account, *S-Inclusive* policy should be chosen, since it always selects less examples to use in the local training sets than the other two policies (see Figure 1).



(a) J48 base classifier. (b) NB base classifier.

Figure 4. Visualization of the Nemenyi post-hoc test. Groups of methods that are not significantly different (at $p < 0.05$) are connected.

4. Conclusions

This paper presented a discussion on 8 different policies to select local training sets for the *Hierarchical Binary Relevance* method, a popular hierarchical classification method. The policies were evaluated using two different base classifiers in 13 datasets from three different domains. Results show that the *inclusive set operation* based policies are better than the *exclusive* and the *distance* based ones. Moreover, hypothesis tests show that using NB as the base classifier, all *inclusive set operation* policies are significantly better (at $p < 0.05$) than the others. Using J48 as the base classifier, although the *S-Inclusive* policy shows better results than the *Exclusive* and the *distance* based policies, the difference is not statistically significantly better. Regarding *L-Inclusive* and *Inclusive* policies, both were significantly better than distance based policies.

As future work, we intend to extend the experimental evaluation of the three best policies, considering not only datasets from other domains but also a more detailed analysis, as well as a greater number of evaluation measures.

Acknowldgements: This research was supported by the Brazilian Research Councils FAPESP and CAPES. We would like to thank Mr. Carlos Silla Jr for providing the datasets with protein descriptors used in this paper.

References

Ceci, M. and Malerba, D. (2003). Hierarchical classification of HTML documents with WebClass II. In *Proceedings of the 25th European conference on IR research*, ECIR'03, pages 57–72, Berlin, Heidelberg. Springer-Verlag.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30.

Eisner, R., Poulin, B., Szafron, D., Lu, P., and Greiner, R. (2005). Improving protein function prediction using the hierarchical structure of the gene ontology. In *In Proc. IEEE CIBCB*, pages 1–10.

Fagni, T. and Sebastiani, F. (2007). On the selection of negative examples for hierarchical text categorization. In *Proceedings of The 3rd Language Technology Conference*, pages 24–28.

Freitas, A. and de Carvalho, A. C. (2007). A tutorial on hierarchical classification with applications in bioinformatics. In Taniar, D., editor, *Research and Trends in Data Mining Technologies and Applications*, chapter 7, pages 175–208. IGI Global.

Kiritchenko, S., Matwin, S., Nock, R., and Famili, A. (2006). Learning and evaluation in the presence of class hierarchies: Application to text categorization. In Lamontagne, L. and Marchand, M., editors, *Advances in Artificial Intelligence*, volume 4013 of *Lecture Notes in Computer Science*, pages 395–406. Springer Berlin / Heidelberg.

Mladenić, D. and Grobelnik, M. (1998). Feature selection for classification based on text hierarchy. In *Text and the Web, Conference on Automated Learning and Discovery CONALD-98*, pages 1–6.

Schapire, R. E., Singer, Y., and Singhal, A. (1998). Boosting and Rocchio applied to text filtering. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 215–223, New York, NY, USA. ACM.

Silla Jr, C. and Freitas, A. (2011). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 1:1–42.

Singhal, A., Mitra, M., and Buckley, C. (1997). Learning routing queries in a query zone. In *Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '97, pages 25–32, New York, NY, USA. ACM.

Sun, A., Lim, E., and Ng, W. (2003). Performance measurement framework for hierarchical text classification. *Journal of the American Society for Information Science and Technology*, 54:1014–1028.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2010). Mining multi-label data. In Maimon, O. and Rokach, L., editors, *Data Mining and Knowledge Discovery Handbook*, pages 667–685. Springer US.

Vens, C., Struyf, J., Schietgat, L., Džeroski, S., and Blockeel, H. (2008). Decision trees for hierarchical multi-label classification. *Machine Learning*, 73(2):185–214.

Witten, I. H., Frank, E., Holmes, G., and Hall, M. (2011). *Data Mining: Practical machine learning tools and techniques*, volume 1. Morgan Kaufmann Publishers Inc., San Francisco, California, USA, 3rd edition.

O Papel da Inferência Seletiva em Aprendizado de Máquina

Ígor Assis Braga, Maria Carolina Monard

Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo – São Carlos, SP – Brasil

{igorab,mcmonard}@icmc.usp.br

Abstract. *One of the challenges in machine learning is the automatization of empirical inference processes. In this paper we deal with selective inference, which is an inference type that remains unexplored in machine learning. Despite the possibility of using more general inference algorithms for solving selective inference, we argue that an algorithm for solving it directly can achieve success even when more general inference algorithms fail. We report on initial experiments conducted on text data sets that support this idea.*

Resumo. *Um dos desafios em aprendizado de máquina é a automatização de processos de inferência empírica. Neste artigo é apresentado um tipo de inferência ainda não explorado em aprendizado de máquina, denominado inferência seletiva. Apesar de algoritmos de inferência mais gerais poderem ser empregados na realização de inferência seletiva, é defendida a ideia de que um algoritmo para realizá-la diretamente pode ter sucesso mesmo quando algoritmos de inferência mais gerais falham. Para corroborar essa ideia, são relatados experimentos iniciais conduzidos em bases de dados textuais.*

1. Introdução

Um dos principais desafios em aprendizado de máquina é o desenvolvimento de algoritmos que realizam inferências a partir de dados empíricos. Desde o final da década de 50, quando o algoritmo Perceptron foi proposto [Rosenblatt 1957], diversos algoritmos de aprendizado foram desenvolvidos para a realização de inferência indutiva. Em contraste, somente no final dos anos 90 foi que surgiram os primeiros algoritmos para a realização de outros tipos de inferência, como a inferência transdutiva.

A maior concentração de pesquisas em inferência indutiva é compreensível, pois, por exemplo, realizar bem inferência indutiva implica em realizar bem inferência transdutiva. Sendo assim, por que se pesquisam algoritmos de inferência transdutiva? A resposta para essa pergunta está na constatação de que realizar bem inferência transdutiva não depende de se realizar bem inferência indutiva. Desse modo, em situações nas quais algoritmos indutivos falham, ainda é possível que um algoritmo transdutivo obtenha sucesso.

Existem outros tipos de inferência empírica sugeridos na literatura que ainda não foram explorados em aprendizado de máquina. Esse é o caso da *inferência seletiva*, tema deste artigo. O motivo pelo qual esse tipo de inferência não recebeu atenção é o mesmo que causou a demora no desenvolvimento de algoritmos transdutivos: realizar bem inferência indutiva ou inferência transdutiva implica em realizar bem inferência seletiva. De maneira análoga, como será argumentado ao longo deste artigo, realizar bem inferência seletiva não depende de se realizar bem os outros dois tipos de inferência.

Com o intuito de contribuir para a exploração de inferência seletiva em aprendizado de máquina, neste artigo são identificados domínios de aplicação de amplo interesse que podem ser beneficiados pelo desenvolvimento de algoritmos seletivos. Além disso, são relatados experimentos em bases de texto que demonstram a possibilidade de se realizar bem inferência seletiva mesmo quando as inferências indutiva e transdutiva falham.

O restante deste artigo está estruturado como se segue. Na Seção 2 é destacada a importância de se desenvolver algoritmos que realizem diretamente um determinado tipo de inferência, usando como exemplo o caso da inferência transdutiva. Na Seção 3 é apresentada a definição de inferência seletiva, além de importantes domínios de aplicação e de uma formulação teórica para um caso particular. Na Seção 4 é proposto um algoritmo inspirado em inferência seletiva a fim de compará-lo experimentalmente com um algoritmo indutivo e com um algoritmo transdutivo empregados na realização de inferência seletiva. Por último, na Seção 5, são apresentadas as considerações finais.

2. Fundamentação

Nesta seção é destacada a filosofia de se realizar um determinado tipo de inferência diretamente, sem depender da realização de tipos de inferência mais gerais. Para isso, as inferências indutiva e transdutiva são definidas, e, em seguida, é mostrado como essa filosofia contribuiu no desenvolvimento de algoritmos mais apropriados para a realização de inferência transdutiva.

Em todos os tipos de inferência considerados ao longo deste artigo, assume-se a existência de uma função $f : X \mapsto Y$. A função f é desconhecida, mas é possível ter acesso a um conjunto $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ tal que $y_i = f(\mathbf{x}_i)$. Esse conjunto, denominado conjunto de treinamento, contém elementos que indicam o valor da função f em n pontos do domínio X. O objetivo geral em inferência empírica é utilizar a informação sobre f disponível no conjunto de treinamento para obter novas informações sobre f.

Diferentes tipos de inferência podem ser definidos de acordo com a magnitude das informações obtidas ao final do processo de inferência. Em *inferência indutiva*, por exemplo, o objetivo é, dado um conjunto de treinamento, encontrar uma função (ou hipótese) $h : X \mapsto Y$ que seja uma "boa aproximação" da função desconhecida f. Idealmente, o que se obtém nesse tipo de inferência é uma estimativa dos valores de f em *todos* os pontos do domínio X.

Agora considere que é necessário algo mais específico: inferir o valor da função desconhecida f somente em *alguns* pontos pré-determinados do domínio X. Em outras palavras, dados um conjunto de treinamento e um conjunto $\{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}\}$ contendo m pontos de interesse, é necessário estimar os valores $y_{n+1}, \ldots, y_{n+m}$ que a função desconhecida f assume nos m pontos de interesse. Como a magnitude das informações obtidas nesse caso é bem menor que na inferência indutiva, Vapnik [1995, p. 293] propôs diferenciar esse tipo de inferência, denominando-o *inferência transdutiva*.

A fim de se realizar inferência transdutiva, poderia-se utilizar um algoritmo indutivo[1] para obter uma função aproximadora h, que em seguida seria utilizada para estimar

[1]Entre os algoritmos de aprendizado que realizam inferência indutiva encontram-se, por exemplo, os baseados em árvores de decisão, o k-Nearest Neighbors (kNN), Support Vector Machines (SVM) e os baseados em redes neurais preditivas [Alpaydin 2004].

o valor de f nos pontos de interesse — Figura 1(a). No entanto, quando os exemplos disponíveis no conjunto de treinamento não são suficientes para se realizar bem inferência indutiva, ainda é possível que a informação disponível seja suficiente para se realizar inferência transdutiva diretamente — Figura 1(b).

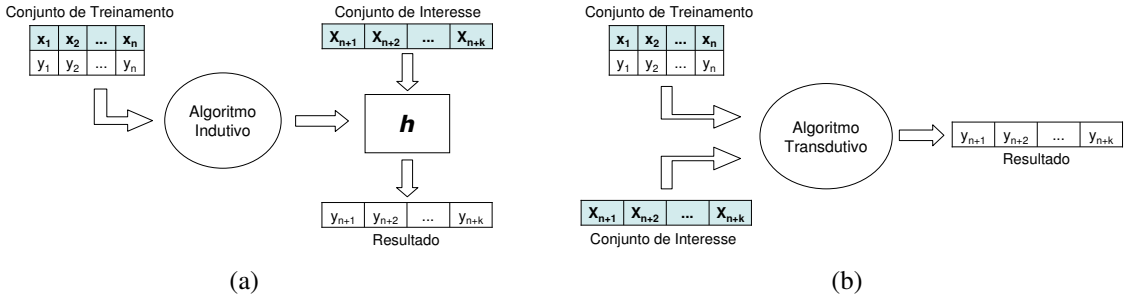


Figura 1. (a) Algoritmo indutivo realizando inferência transdutiva. (b) Algoritmo para realizar inferência transdutiva diretamente.

A busca por algoritmos que realizem inferência transdutiva diretamente é uma consequência da seguinte filosofia de inferência [Vapnik 2006, p. 477]:

> *Ao tentar resolver um problema de interesse, não tente resolvê-lo utilizando a solução de um problema mais geral. Tente obter a resposta desejada diretamente.*

No caso de algoritmos transdutivos já desenvolvidos, como, por exemplo, Transductive Support Vector Machines (TSVM) [Joachims 1999] e Spectral Graph Transducer [Joachims 2003], a vantagem obtida em relação aos algoritmos indutivos é a disponibilidade dos pontos de interesse durante a fase de treinamento. Trabalhos em diferentes áreas de aplicação de aprendizado de máquina mostram que os algoritmos transdutivos são mais eficazes que os algoritmos indutivos empregados na realização de inferência transdutiva. Essa observação é ainda mais forte nos casos em que o conjunto de treinamento contém poucos exemplos descritos em um espaço de alta-dimensionalidade [Joachims 1999, Weston et al. 2003, Audibert 2008, El-Yaniv et al. 2008].

3. Inferência Seletiva

Na seção anterior foram definidos dois tipos distintos de inferência empírica: a inferência indutiva, mais geral, e a inferência transdutiva, mais específica. Também foi destacado, no contexto da inferência transdutiva, a importância de se providenciar algoritmos que realizem um determinado tipo de inferência diretamente.

Nesta seção é apresentado um terceiro tipo de inferência, denominado inferência seletiva, para o qual ainda não se conhecem algoritmos próprios. Além disso, são listadas tarefas de aprendizado nas quais a realização de inferência seletiva deveria ser considerada. Por fim, é apresentada uma formulação teórica dada por Vapnik [2006] para um caso particular de inferência seletiva.

3.1. Definição

Suponha novamente a existência de uma função desconhecida $f : X \mapsto Y$ e de um conjunto de treinamento $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ no qual $y_i = f(\mathbf{x}_i)$. Por vezes, deseja-se selecionar alguns pontos de um conjunto pré-determinado de candidatos $\{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}\}$

de acordo com a função f. Como essa função é desconhecida, a seleção de tais candidatos não pode ser feita dedutivamente. Em *inferência seletiva*, o objetivo é selecionar os candidatos de acordo com a informação sobre f disponível no conjunto de treinamento.

Dependendo da natureza do conjunto Y e das condições que um candidato deve satisfazer para ser selecionado, pode-se obter diferentes definições particulares de inferência seletiva. Os dois casos mencionados a seguir não esgotam todas as possibilidades.

> *Caso I*. Suponha que $Y = \{\oplus, \ominus\}$. Dados um conjunto de treinamento e um conjunto contendo m exemplos candidatos, selecione p exemplos candidatos que são positivos de acordo com a função desconhecida f.
>
> *Caso II*. Suponha que $Y \subseteq \mathbb{R}$. Dados um conjunto de treinamento e um conjunto contendo m pontos candidatos, selecione o ponto candidato $\mathbf{x}$ que leva ao maior valor de $f(\mathbf{x})$ entre os pontos candidatos, isto é, selecione o ponto candidato $\mathbf{x}$ para o qual $f(\mathbf{x}) \geq f(\mathbf{x}_i), i = n+1, \ldots, n+m$.

Contrastando inferência seletiva com as inferências indutiva e transdutiva, definidas na seção anterior, percebe-se que estas são mais gerais que aquela. De fato, um algoritmo indutivo ou transdutivo pode ser utilizado na realização de inferência seletiva, mas a recíproca não é verdadeira. A Figura 2 ilustra esse argumento para o Caso II. Observe que um algoritmo transdutivo, por exemplo, pode estimar os valores da função desconhecida f nos pontos candidatos, e, em seguida, o ponto que leva ao maior valor estimado é selecionado — Figura 2(a). Contudo, o resultado de um possível algoritmo para realizar inferência seletiva diretamente não pode ser utilizado para estimar os valores de f nos pontos candidatos — Figura 2(b).

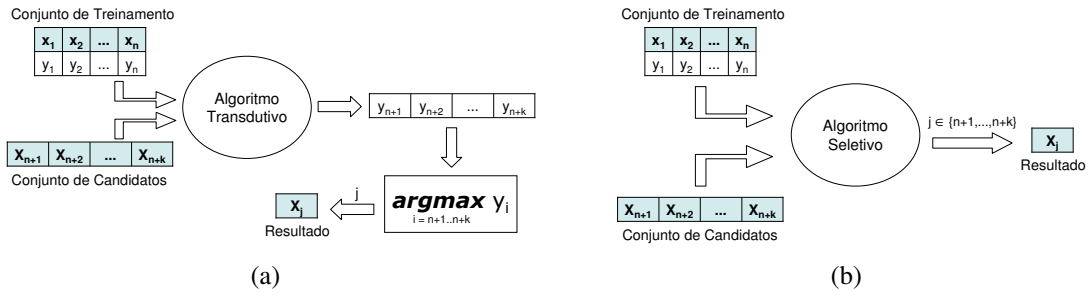


Figura 2. (a) Algoritmo transdutivo realizando inferência seletiva — Caso II. (b) Possível algoritmo para realizar inferência seletiva diretamente.

A definição de inferência seletiva foi sugerida primeiramente por Vapnik [2006, p. 468] pelos mesmos motivos que o levaram a sugerir a inferência transdutiva. Afinal, os exemplos disponíveis no conjunto de treinamento podem não ser suficientes para se realizar bem inferência indutiva ou transdutiva, mas ainda é possível que a informação disponível seja suficiente para se realizar inferência seletiva diretamente.

3.2. Aplicações

A seguir são apresentadas algumas tarefas de aprendizado de máquina nas quais a realização de inferência seletiva deveria ser considerada. Essas tarefas aparecem em domínios de aplicação de grande interesse para pesquisadores na área. As três primeiras tarefas são

exemplos da definição de inferência seletiva para o Caso I, enquanto que a última é um exemplo para o Caso II.

Desenvolvimento de fármacos. Uma etapa importante no desenvolvimento de fármacos é testar compostos químicos quanto a sua capacidade de se ligarem a um determinado receptor no corpo humano. Na tentativa de reduzir os custos envolvidos nessa etapa, os dados resultantes dos testes são analisados para que novos testes sejam feitos em compostos mais promissores. Tal análise pode ser considerada como uma tarefa de aprendizado de máquina quando um conjunto de compostos previamente testados está disponível para treinamento[2]. Dado um conjunto de compostos ainda não testados, é interessante identificar um a um qual se liga e qual não se liga ao receptor de interesse usando, por exemplo, inferência transdutiva. No entanto, a escassa quantidade de compostos de treinamento típica dessa tarefa faz com que a realização de inferência seletiva seja considerada.

Detecção de homologia entre proteínas. A detecção de homologia entre proteínas é considerada uma importante tarefa em bioinformática, pois é possível se fazer considerações sobre a estrutura desconhecida de uma proteína por meio da estrutura conhecida de uma proteína homóloga. Devido à disponibilidade crescente de algoritmos para obter características de proteínas, é possível detectar homologia entre uma proteína com estrutura desconhecida e uma com estrutura conhecida utilizando algoritmos de aprendizado[3]. Apesar de uma proteína poder ser homóloga a várias outras, pode ocorrer de somente uma ou algumas poucas já serem suficientes para a modelagem de sua estrutura. Sendo assim, não é necessário realizar um tipo de inferência mais geral que a inferência seletiva para cada proteína com estrutura desconhecida.

Descoberta de genes relevantes a uma doença. Identificar genes que desempenham papéis importantes em doenças é uma tendência na pesquisa médica. A disponibilidade de algoritmos para gerar características de genes e a tecnologia para obter níveis de expressão gênica tem colaborado para a aplicação de aprendizado de máquina nesta tarefa. O ideal seria poder identificar todos ou grande parte dos genes relevantes a uma doença dentre os genes candidatos, mas é pouca a quantidade de genes de treinamento relevantes tipicamente disponível nesta tarefa. Desse modo, os algoritmos de aprendizado são normalmente utilizados para priorizar genes para estudos futuros [Agarwal e Sengupta 2009], o que pode ser realizado com inferência seletiva.

Busca na *web*. Uma consulta realizada em sistemas de busca na *web* gera uma lista contendo páginas com variados graus de relevância, sendo necessário identificar aquelas com maior relevância para compor os primeiros resultados de busca. Essa identificação pode ser feita usando-se algoritmos de aprendizado que utilizam dados de treinamento obtidos explicitamente por meio de especialistas ou implicitamente por meio dos cliques dos usuários nos resultados de busca [Joachims 2002]. Geralmente, os usuários visualizam no máximo os 10 primeiros resultados de busca [Campos 2007]. Consequentemente, ordenar todas as páginas recuperadas, inclusive as pouco relevantes e as irrelevantes, constitui um trabalho desnecessário. Uma solução mais direta, e que pode requerer menos dados de treinamento, seria investigar as chances de cada página *web* recuperada figurar no topo dos resultados de busca, o que pode ser realizado com inferência seletiva.

[2]Veja, por exemplo, o KDD-Cup 2001: `http://pages.cs.wisc.edu/~dpage/kddcup2001/`.
[3]Veja, por exemplo, o KDD-Cup 2004: `http://kodiak.cs.cornell.edu/kddcup/tasks.html`.

No geral, a realização de inferência seletiva torna-se uma opção a ser considerada em aprendizado de máquina quando o domínio de aplicação impõe um interesse maior sobre determinados candidatos. No Caso I, esse interesse recai sobre exemplos que podem ser identificados como positivos com alta precisão, enquanto que, no Caso II, o interesse é por pontos que assumem o maior valor da função desconhecida f dentre os pontos candidatos. No entanto, se a quantidade de informação disponível no conjunto de treinamento for pouca, a realização de inferência seletiva pode se tornar a única opção, pois realizar tipos de inferência mais gerais nessa condição é mais arriscado.

3.3. Formulação Teórica

Assumindo que a realização de inferência seletiva pode ter um papel importante em aprendizado de máquina, a primeira questão que se levanta é como realizá-la sem depender de tipos de inferência mais gerais. A seguir, é apresentada uma formulação teórica para esse problema dada por Vapnik [2006, p. 341] para o Caso I de inferência seletiva, isto é, assume-se aqui que $Y = \{\oplus, \ominus\}$. Uma formulação análoga para o Caso II também foi proposta, e pode ser encontrada em [Vapnik 2006, p. 344].

Antes de prosseguir, é necessário definir o conceito de classes de equivalência de funções. Considere a existência de um espaço de funções $F \subseteq X \times Y$. Os exemplos do conjunto de treinamento e do conjunto de candidatos particionam F em l classes de equivalência $E_1, E_2, \ldots, E_l$, tal que as funções pertencentes a uma classe de equivalência E_i rotulam os exemplos de treinamento e os exemplos candidatos da mesma maneira. Formalmente, para $\forall f', f'' \in E_i$, tem-se que

$$\begin{gathered} f'(\mathbf{x}_1) = f''(\mathbf{x}_1) = y_{1,i} \\ \vdots \\ f'(\mathbf{x}_n) = f''(\mathbf{x}_n) = y_{n,i} \\ f'(\mathbf{x}_{n+1}) = f''(\mathbf{x}_{n+1}) = y_{n+1,i} \\ \vdots \\ f'(\mathbf{x}_{n+m}) = f''(\mathbf{x}_{n+m}) = y_{n+m,i} \end{gathered}$$

Agora suponha que se deseja realizar inferência seletiva para identificar um único exemplo positivo dentre os candidatos. A formulação teórica dada por Vapnik expressa a probabilidade de cada candidato $\mathbf{x}_i$ ser positivo de acordo com f dados o conjunto de treinamento e o conjunto de candidatos. O candidato que obtiver a maior probabilidade é aquele que deve ser selecionado.

No que se segue, denote o conjunto de treinamento $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ por T, o conjunto de candidatos $\{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}\}$ por C, e a sequência de candidatos obtida omitindo-se o candidato $\mathbf{x}_i$ por C^i, isto é, $C^i = \mathbf{x}_{n+1}, \ldots, \hat{\mathbf{x}}_i, \ldots, \mathbf{x}_{n+m}$. Note que a sequência C^i pode ser rotulada em duas classes de 2^{m-1} maneiras. Denote a r-ésima rotulação $y_{n+1,r}, \ldots, \hat{y}_{i,r}, \ldots, y_{n+m,r}$ por L^i_r.

Suponha que foi definida a probabilidade $P(L^i_r)$ de que a rotulação L^i_r coincidirá com a rotulação da sequência C^i dada pela função desconhecida f. Então, pela lei da probabilidade total, tem-se para um determinado candidato $\mathbf{x}_i$ que

$$P(f(\mathbf{x}_i) = \oplus | T, C) = \sum_{r=1}^{2^{m-1}} P(f(\mathbf{x}_i) = \oplus | T, C, L^i_r) P(L^i_r).$$

Para calcular essa expressão, é preciso ter mais informações sobre o problema em mãos. Considere, por exemplo, que a função desconhecida f pertença a um espaço de funções F conhecido. Nesse caso, uma das classes de equivalência $E_1, E_2, \ldots, E_l$ geradas em F rotula corretamente os exemplos de treinamento e os exemplos candidatos. Suponha *a priori* que a rotulação correta desses exemplos pode ser realizada equiprovavelmente por qualquer uma das classes de equivalência[4]. Sendo assim, os termos do lado direito da equação podem ser calculados. A probabilidade de que a rotulação L_r^i coincidirá com a rotulação da sequência C^i dada pela função desconhecida f é

$$P(L_r^i) = \frac{\#(C^i, L_r^i)}{l}$$

na qual $\#(C^i, L_r^i)$ é o número de classes de equivalência que rotulam a sequência C^i de acordo com a rotulação L_r^i. A probabilidade condicional $P(f(\mathbf{x}_i) = \oplus | T, C, L_r^i)$ é calculada como

0, se não existe uma classe de equivalência que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \oplus;$$
$\frac{1}{2}$, se existe uma classe de equivalência que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \oplus$$
e uma que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \ominus;$$
1, se existe uma classe de equivalência que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \oplus$$
e não existe uma que permita a rotulação
$$T \cup C^i L_r^i \cup \mathbf{x}_i, \ominus.$$

Uma propriedade interessante dessa formulação é que $P(f(\mathbf{x}_i) = \oplus | T, C) = \frac{1}{2}$ para qualquer candidato $\mathbf{x}_i$ quando o espaço de funções pode rotular os conjuntos de treinamento e de candidatos de todas as formas possíveis ($l = 2^{n+m}$). Isso significa que nenhuma inferência pode ser feita nesse caso, assim como já foi demonstrado para as inferências indutiva e transdutiva [Vapnik 2006, p. 152].

4. Experimento Ilustrativo

Na seção anterior foi apresentado o papel que a inferência seletiva pode desempenhar em aprendizado de máquina. Apesar de existirem tarefas de aprendizado nas quais a inferência seletiva deveria ser fortemente considerada, ainda não se conhecem algoritmos propriamente seletivos. No entanto, a formulação teórica apresentada por Vapnik [2006] pode oferecer um ponto de partida para esse desenvolvimento.

Nesta seção, com o intuito de realizar os primeiros experimentos com inferência seletiva, é proposto um algoritmo seletivo preliminar tendo como base a formulação teórica apresentada na seção anterior. Esse algoritmo é comparado experimentalmente com um algoritmo indutivo e um algoritmo transdutivo empregados na realização de inferência seletiva em nove bases de textos. Para simular uma situação de escassez de informação, os experimentos também são conduzidos em conjuntos de treinamento reduzidos artificialmente. Os resultados demonstram que um algoritmo seletivo pode obter sucesso mesmo quando os algoritmos indutivos e transdutivos falham.

[4]Também é possível levar em consideração a distribuição de exemplos positivos e negativos no problema para obter uma distribuição não uniforme [Vapnik 2006, p. 344].

4.1. Algoritmo

A formulação teórica apresentada na seção anterior expressa a probabilidade de cada exemplo candidato ser positivo de acordo com a função desconhecida f, sendo que o candidato que obtém a maior probabilidade é aquele que deve ser selecionado (Caso I com $p = 1$). Contudo, o cálculo dessas probabilidades impõe sérios problemas computacionais, pois há um somatório que envolve um número de parcelas igual a 2^{m-1}, sendo m o número de exemplos candidatos. Portanto, o algoritmo mais simples para calcular essas probabilidades será exponencial no número de exemplos candidatos.

Com o intuito de realizar os primeiros experimentos com inferência seletiva, é proposto um algoritmo seletivo preliminar baseado no caso em que a formulação teórica é aplicada a um único exemplo candidato. Consequentemente, $r = 1$, $P(L_r^i)$ fica indefinido e $P(f(\mathbf{x}_i) = \oplus|T, C) = P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i)$. A probabilidade condicional $P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i)$ é calculada como

- 0, se não existe uma classe de equivalência que permita a rotulação $T \cup \mathbf{x}_i, \oplus$;
- $\frac{1}{2}$, se existe uma classe de equivalência que permita a rotulação $T \cup \mathbf{x}_i, \oplus$ e uma que permita a rotulação $T \cup \mathbf{x}_i, \ominus$;
- 1, se existe uma classe de equivalência que permita a rotulação $T \cup \mathbf{x}_i, \oplus$ e não existe uma que permita a rotulação $T \cup \mathbf{x}_i, \ominus$.

No final, são selecionados todos os candidatos para os quais $P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i) = 1$.

No Algoritmo 1 apresentado a seguir, os exemplos candidatos para os quais $P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i) = 1$ são selecionados. Para isso, é utilizado o procedimento $SVM(T', \alpha, K)$ que executa o algoritmo indutivo Support Vector Machines [Burges 1998] sobre um conjunto de treinamento T', parâmetro de generalização α e *kernel* K. Os parâmetros α e K definem o espaço de funções F para o algoritmo SVM. Caso esse espaço de funções não acomode um determinado exemplo do conjunto de treinamento, a função obtida aplicada a esse exemplo assume um valor maior que -1 para exemplos negativos e menor que 1 para exemplos positivos. Assim, para que um exemplo candidato $\mathbf{x}_i$ obtenha $P(f(\mathbf{x}_i) = \oplus|T, \mathbf{x}_i) = 1$, é necessário que, ao ser inserido como positivo no conjunto de treinamento, a função obtida $f_\oplus$ assuma um valor maior ou igual a 1 em $\mathbf{x}_i$, e, ao ser inserido como negativo, a função obtida $f_\ominus$ assuma um valor maior que -1.

É importante ressaltar que, embora o algoritmo indutivo SVM esteja sendo utili-

Algoritmo 1: Algoritmo seletivo preliminar

Entrada:
- conjunto de treinamento $T = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$
- conjunto de candidatos $C = \{\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}\}$
- parâmetro de generalização α
- *kernel* K

$S \leftarrow \emptyset$;
para cada $\boldsymbol{x}_i \in C$ **faça**
  $f_\oplus \leftarrow SVM(T \cup \{\mathbf{x}_i, \oplus\}, \alpha, K)$;
  se $f_\oplus(\boldsymbol{x}_i) \geq 1$ **então**
    $f_\ominus \leftarrow SVM(T \cup \{\mathbf{x}_i, \ominus\}, \alpha, K)$;
    se $f_\ominus(\boldsymbol{x}_i) > -1$ **então**
      $S \leftarrow S \cup \{\mathbf{x}_i\}$;
    fim
  fim
fim
Saída: conjunto de exemplos selecionados S

zado dentro do algoritmo seletivo proposto, os exemplos candidatos são incluídos um a um no conjunto de treinamento, além de ser necessário avaliar duas funções para decidir se um exemplo será selecionado ou não. Portanto, tal utilização de SVM não pode ser considerada como uma maneira de se realizar inferência indutiva.

4.2. Desenho Experimental

Para a realização dos experimentos com inferência seletiva, foram utilizadas coleções textuais provenientes de fontes variadas, como páginas da *web*, mensagens eletrônicas em grupos de discussão, artigos científicos e críticas de filmes. Essas coleções foram estruturadas em nove tabelas atributo-valor, utilizando unigramas para a descrição dos documentos (exemplos)[5]. As informações relativas às bases utilizadas e às respectivas tabelas atributo-valor estão na Tabela 1. Nessa tabela estão descritos, para cada base de dados: o domínio de origem; o número de documentos (#Doc.); o número original de atributos (#Atributos); o número de atributos após remoção de *stopwords*, *stemming* e corte por frequência (#Atrib. pós-corte); a classe positiva ($\oplus$); e a porcentagem de exemplos positivos na base($\oplus\%$).

Tabela 1. Descrição das bases de dados utilizadas

Base	Domínio	#Doc.	#Atributos	#Atrib. pós-corte	$\oplus$	$\oplus\%$
CBR	Artigos científicos	675	21545	5072	CBR	48
CS	Artigos científicos	823	68047	13336	IA	61
EC	Páginas *web*	1199	12621	3597	Computers	50
ET	Páginas *web*	1069	12941	3941	Transport	50
HARDWARE	Mensagens de *newsgroups*	1943	13398	3958	Mac	50
MOVIE	Críticas de filmes	2000	25302	10669	Positive review	50
SCIENCE	Artigos científicos	797	74739	15580	Physics	50
SPORTS	Mensagens de *newsgroups*	1993	14254	5741	Baseball	50
VEHICLES	Mensagens de *newsgroups*	1984	14048	5362	Motorcycles	50

Fixada uma base, o processo de avaliação é realizado com 10-*fold cross-validation* estratificado, gerando-se dez pares (T, C) para a base, sendo que os conjuntos de candidatos gerados são mutualmente disjuntos. Em cada iteração do processo de *cross-validation*, é realizada inferência seletiva com um dos dez pares de conjuntos (T, C), e a precisão da inferência é avaliada. A precisão da inferência na iteração k é definida como

$$pr_k = \frac{\#\ de\ exemplos\ positivos\ selecionados}{\#\ de\ exemplos\ selecionados} \quad \text{ou} \quad pr_k = \tfrac{1}{2}, se\ \#\ de\ ex.\ sel. = 0$$

O resultado final para uma determinada base é a média de pr_k em todas as iterações.

O processo de avaliação descrito é repetido três vezes, cada vez utilizando um algoritmo diferente para realizar inferência seletiva, porém mantendo-se os mesmos dez pares (T, C). Um dos algoritmos é o algoritmo indutivo Support Vector Machines (SVM). Esse algoritmo obtém uma função que é usada para classificar todos os exemplos candidatos. Nesse caso, todos os exemplos candidatos classificados como positivos são selecionados. O segundo algoritmo é o algoritmo transdutivo Transductive Support Vector Machines (TSVM). Nesse segundo caso, todos os exemplos candidatos rotulados como positivos são selecionados. O terceiro algoritmo é o algoritmo seletivo preliminar proposto anteriormente. O software UniverSVM[6] foi utilizado para executar SVM, TSVM e o procedimento que chama SVM no algoritmo proposto.

[5] As bases podem ser encontradas em `http://www.icmc.usp.br/~igorab/msc/datasets/`.

[6] `http://3t.kyb.tuebingen.mpg.de/bs/people/fabee/universvm.html`

Pelo fato dos três algoritmos nessa avaliação terem parâmetros a serem ajustados, é necessário executá-los variando-se esses parâmetros. O único parâmetro fixo em todas as execuções dos três algoritmos é o *kernel*, que nesta avaliação, foi fixada no *kernel* linear, o qual é tradicionalmente utilizado em bases de texto. Para SVM e o algoritmo seletivo proposto, foi realizada uma execução para cada um dos seguintes valores de α: 10^{-4}, 10^{-3}, ..., 10^{0}, ..., 10^{2}, 10^{3}. Para TSVM, que tem mais parâmetros a serem ajustados, foi realizada uma execução para os mesmos valores de α mencionados, enquanto os outros parâmetros foram mantidos em seu valor padrão. O valor médio de pr_k relatado em cada base para um determinado algoritmo é dado pelo maior valor médio de pr_k atingido utilizando os diferentes valores de α. Por exemplo, se o algoritmo SVM ao ser executado na base CBR atinge o maior valor médio de pr_k para $\alpha = 10^{-}4$, então esse é o valor médio de pr_k que será relatado.

Além de avaliar os três algoritmos usando todos os exemplos da base de dados, foi realizada uma simulação de escassez de informação para verificar como se comportam os algoritmos nessa situação. Para isso, todo o processo mencionado anteriormente foi repetido usando-se 10%, 5% e 1% dos exemplos de treinamento, enquanto o número de exemplos em C era mantido constante. Por exemplo, considere que a base tem um total de 1000 exemplos, então cada um dos *folds* possui 100 exemplos. Nesse caso, cada uma das $k = 10$ iterações com 100% dos exemplos de treinamento considera $|T| = 900$ (9 *folds*) e $|C| = 100$ (1 *fold*). Entretanto, nas $k = 10$ iterações com 10% dos exemplos de treinamento, essa porcentagem de exemplos é extraída aleatoriamente dos 9 *folds* anteriores, ou seja, $|T| = 90$ e $|C| = 100$. Analogamente, $|T| = 45$ e $|T| = 9$ quando, respectivamente, 5% e 1% dos exemplos de treinamento são considerados. Em todos os casos, no entanto, é mantido $|C| = 100$. Os exemplos que integram os conjuntos de treinamento reduzidos foram escolhidos previamente à execução dos algoritmos, a fim de que os experimentos fossem pareados.

4.3. Resultados e Análise

Os resultados obtidos são apresentados na Tabela 2. Em cada base de dados, o comportamento dos algoritmos indutivo (SVM), transdutivo (TSVM) e seletivo (preliminar) podem ser observados quando 100%, 10%, 5% e 1% dos exemplos do conjunto de treinamento estão disponíveis. Em cada um desses casos, é mostrado o valor médio de pr_k atingido quando os algoritmos são executados com o melhor valor do parâmetro α.

Como pode ser observado na Tabela 2, o algoritmo seletivo proposto sempre tem desempenho superior ou igual aos algoritmos indutivo e transdutivo nas faixas de 100%, 10% e 5%. Quando 100% do conjunto de treinamento é utilizado, o algoritmo seletivo supera com boa margem os outros dois algoritmos nas bases EC, HARDWARE e MOVIE. Nas faixas de 10% e 5%, o algoritmo seletivo se destaca, superando os outros dois algoritmos com boa margem em quase todas as bases. Nessas faixas é visível, principalmente nas bases CS e HARDWARE, que um algoritmo seletivo pode obter bons resultados mesmo quando os algoritmos indutivo e transdutivo não obtêm bons resultados.

Na faixa de 1%, no entanto, o algoritmo seletivo só aprensenta resultados melhores nas bases SPORTS e VEHICLES. Nas bases EC, HARDWARE e MOVIE, os resultados são semelhantes. Nas bases CBR, CS, ET e SCIENCE, o algoritmo seletivo obtém piores resultados que os outros dois algoritmos. Nessas quatro bases, o algoritmo seletivo

Tabela 2. Comportamento dos algoritmos indutivo (SVM), transdutivo (TSVM) e seletivo (preliminar) usando 1%, 5%, 10% e 100% do conjunto de treinamento de cada base.

(a) CBR

Alg.	1%	5%	10%	100%
Ind.	0,95	**1,00**	**1,00**	**1,00**
Trans.	**0,98**	**1,00**	**1,00**	**1,00**
Sel.	0,84	**1,00**	**1,00**	**1,00**

(b) CS

Alg.	1%	5%	10%	100%
Ind.	**0,73**	0,87	0,91	0,95
Trans.	0,66	0,81	0,84	0,88
Sel.	0,52	**0,95**	**1,00**	**1,00**

(c) EC

Alg.	1%	5%	10%	100%
Ind.	0,58	0,68	0,73	0,91
Trans.	0,54	0,65	0,72	0,91
Sel.	**0,59**	**0,78**	**0,86**	**0,99**

(d) ET

Alg.	1%	5%	10%	100%
Ind.	**0,65**	0,83	0,98	0,98
Trans.	0,61	0,77	0,87	0,98
Sel.	0,61	**0,93**	**0,98**	**1,00**

(e) HARDWARE

Alg.	1%	5%	10%	100%
Ind.	0,80	0,80	0,91	0,93
Trans.	0,70	0,80	0,87	0,93
Sel.	**0,81**	**0,98**	**0,98**	**1,00**

(f) MOVIE

Alg.	1%	5%	10%	100%
Ind.	**0,72**	0,77	0,89	0,84
Trans.	0,58	0,67	0,74	0,83
Sel.	**0,72**	**0,82**	**0,91**	**0,96**

(g) SCIENCE

Alg.	1%	5%	10%	100%
Ind.	0,84	0,96	0,96	**1,00**
Trans.	**0,88**	0,96	0,97	**1,00**
Sel.	0,60	**0,99**	**1,00**	**1,00**

(h) SPORTS

Alg.	1%	5%	10%	100%
Ind.	0,74	0,93	0,97	0,99
Trans.	0,73	0,94	0,98	0,99
Sel.	**0,87**	**0,98**	**1,00**	**1,00**

(i) VEHICLES

Alg.	1%	5%	10%	100%
Ind.	0,81	0,87	0,94	**1,00**
Trans.	0,87	0,94	0,95	0,95
Sel.	**0,90**	**1,00**	**1,00**	**1,00**

se recusou a selecionar exemplos positivos em diversas iterações do processo de *cross-validation*, o que fez a média de pr_k para o algoritmo seletivo ter uma queda brusca em comparação com os resultados usando 5% dos exemplos de treinamento. Uma correlação importante nesse caso é o fato de que as quatro bases são as menores bases entre as nove consideradas — veja a Tabela 1. Dessa forma o número absoluto de exemplos é muito baixo para essas bases na faixa de 1%, indo de 6 a 10 exemplos de treinamento.

O comportamento do algoritmo seletivo na faixa de 1% precisa ser melhor estudado. No entanto, é importante lembrar que esse algoritmo considera cada exemplo candidato isoladamente. Portanto, assim que seja desenvolvido um algoritmo seletivo que consiga trabalhar com todos os exemplos candidatos conjuntamente, haverá uma quantidade maior de exemplos disponíveis na faixa de 1%.

5. Considerações Finais

Neste artigo foi apresentado um tipo de inferência ainda não explorado em aprendizado de máquina, denominado inferência seletiva. O desenvolvimento de algoritmos de inferência seletiva pode beneficiar tarefas de aprendizado em importantes domínios de aplicação, como a bioinformática e a busca na *web*. Com o intuito de realizar os primeiros experimentos com inferência seletiva em aprendizado de máquina, foi proposto um algoritmo seletivo preliminar baseado em uma formulação teórica dada na literatura.

Os resultados de experimentos em nove bases de texto mostram uma boa vantagem do algoritmo seletivo proposto sobre um algoritmo indutivo (SVM) e um algoritmo transdutivo (TSVM) empregados na realização de inferência seletiva. A vantagem se mostrou maior quando o número de exemplos de treinamento foi reduzido a 10% e a 5% do número de exemplos de treinamento obtidos originalmente com 10-*fold cross-validation*. No entanto, quando o número de exemplos de treinamento foi reduzido a 1%, o algoritmo seletivo obteve piores resultados que os outros dois algoritmos nas quatro menores bases.

Como o algoritmo proposto considera cada exemplo candidato isoladamente, os trabalhos futuros focalizarão o desenvolvimento de um algoritmo seletivo que integre de uma só vez todos os exemplos candidatos. Para isso, serão investigadas novas formulações teóricas para inferência seletiva, a fim de se evitar o somatório exponencial no número de candidatos. Além disso, pretende-se continuar investigando novas aplicações para inferência seletiva em aprendizado de máquina.

Agradecimentos À FAPESP, pelo apoio financeiro a este trabalho.

Referências

Agarwal, S. e Sengupta, S. (2009). "Ranking genes by relevance to a disease". Em *CSB '09: Proceedings of the 8th Annual International Conference on Computational Systems Bioinformatics*, páginas 1–10.

Alpaydin, E. (2004). *Introduction to Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press.

Audibert, J. Y. (2008). Transductive learning and computer vision. NIPS 2008 Workshop on Learning with Data-dependent Concept Spaces (abstract and invited talk).

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

Campos, L. F. B. (2007). Increase of precision on the top of the list of retrieved web documents using global and local link analysis. *Webology*, 4(3). `http://www.webology.ir/2007/v4n3/a44.html`.

El-Yaniv, R., Pechyony, D., e Vapnik, V. (2008). Large margin vs. large volume in transductive learning. *Machine Learning*, 72(3):173–188.

Joachims, T. (1999). "Transductive inference for text classification using support vector machines". Em *ICML '99: Proceedings of the 16th International Conference on Machine Learning*, páginas 200–209.

Joachims, T. (2002). "Optimizing search engines using clickthrough data". Em *KDD '02: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, páginas 133–142.

Joachims, T. (2003). "Transductive learning via spectral graph partitioning". Em *ICML '03: Proceedings of the 20th International Conference on Machine Learning*, páginas 290–297.

Rosenblatt, F. (1957). The perceptron: a perceiving and recognizing automaton. Relatório Técnico 85-460-1, Cornell Aeronautical Laboratory.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag.

Vapnik, V. (2006). *Estimation of Dependences Based on Empirical Data*. Information Science and Statistics. Springer-Verlag, 2^a edição.

Weston, J., Perez-Cruz, F., Bousquet, O., Chapelle, O., Elisseeff, A., e Scholkopf, B. (2003). Feature selection and transduction for prediction of molecular bioactivity for drug design. *Bioinformatics*, 19(6):764–771.

Accelerating reinforcement learning by reusing abstract policies

Yannick Plaino Bergamo [1], Tiago Matos [1], Valdinei Freire da Silva [1], Anna Helena Reali Costa [1]

[1]Laboratório de Técnicas Inteligentes (LTI/EPUSP)
Escola Politécnica, Universidade de São Paulo
São Paulo, SP, Brazil

{yannick,tiago.matos}@usp.br, valdinei.freire@gmail.com, anna.reali@poli.usp.br

Abstract. *Reinforcement learning (RL) provides a general approach for developing intelligent agents that are able to optimize their behaviors in stochastic environments. Unfortunately, most work in RL is based on propositional representations, making it difficult to apply it to more complex real-world tasks in which states and actions are more naturally represented in relational form. Moreover, most work in RL does not take into account existing solutions to similar problems when learning a policy to solve a new problem, and consequently solves the new problem from scratch, what can be very time consuming. In this article we explore the powerful possibilities of using relational descriptions so that we can learn abstract policies, and in reusing these policies to improve initial performance of an RL learner in a similar new problem. Experiments carried out attest the effectiveness of our proposal.*

1. Introduction

Reinforcement learning (RL) techniques have demonstrated great success in the development of intelligent agents with the ability to optimize their behavior in stochastic environments. However, RL methods still suffer from some important problems, among which we highlight the slow convergence and the inability to reuse previously acquired knowledge in similar tasks.

Several solutions have been proposed to reuse knowledge – previously learned in the resolution of a similar problem or given by a specialist – to improve initial performance of an RL learner in a new problem. Some works propose the use of a heuristic function properly chosen, which is used to advise appropriate actions to be performed to guide the state space exploration during the learning process [Bianchi et al. 2007, Burkov and Chaib-draa 2007, Bianchi et al. 2008, Knox and Stone 2010]. Others propose to aggregate similar states in macro-states and use macro-actions to transit between macro-states. Then, macro-states and macro-actions are used to solve similar tasks [Drummond 2002]. There are also authors who have proposed to divide the task into subtasks, and find policies that solve those subtasks. These policies represented in an abstract form are then used to solve similar problems, together with a hierarchical representation of the aggregation states of the problem [Uther and Veloso 2002].

There is still work towards representing abstract states and actions in a relational form. Relational representations facilitate formulating broad collections of related tasks as a single domain, yielding natural generalization between these related tasks. Taylor and Stone [Taylor and Stone 2009] provide a thorough survey on the transfer of learning

using propositional descriptions of the states. Cocora et al [Kersting et al. 2007] propose to learn relational decision trees as abstract navigation strategies from example paths, and then the navigation policy learned in one environment is directly applied to unknown environments. Based on samples experienced by the agent when acting optimally in the source task, in [Madden and Howley 2004] propositional symbolic learners generalize the optimal policy. In [Sherstov and Stone 2005] action relevance is used in order to reduce the action set to be explored in the target task.

In this paper we also explore the intuition that generalization from closely related, but solved, problems may often produce policies that make good decisions in many states of a new unsolved problem. Our proposal consists in abstracting policies from solved problems and then using these abstractions to significantly enhance the performance of the learning process in a new problem not yet solved.

The paper is organized as follows. Section 2 briefly reviews basic concepts of Markov decision processes (MDP), a sound theoretical framework to deal with uncertainty, and reinforcement learning, a method to learn policies by trial-and-error in problems modeled as MDPs. In Section 3 we provide a brief overview of relational reinforcement learning, so that we can define more precisely the concept of an abstract policy, which is the structure we use here to capture generalizations from problems. Then, in Section 4 we present our proposals to enable knowledge transfer to improve initial performance of an RL agent learning in a new problem. Section 5 reports an experiments carried to validate ours proposals, and section 6 summarizes our conclusions.

2. Markov decision processes

We will consider that our task can be modelled as a Markov Decision Process (MDP) [Puterman 1994]. At each (discrete) time step an agent observes the state of the system, chooses an action causing the system to evolve to a new state, returning a reward to the agent. Specifically an MDP is a tuple $\langle \mathcal{S}, \mathcal{A}, T, r \rangle$, where:

- $\mathcal{S}$, a discrete set of *states*;
- $\mathcal{A} = \bigcup_{\sigma \in \mathcal{S}} \mathcal{A}_\sigma$, a discrete set of *actions*, where $\mathcal{A}_\sigma$ is the set of allowable actions in state $\sigma \in \mathcal{S}$;
- $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \mapsto [0, 1]$, a *transition function* such that $T(\sigma, \alpha, \sigma') = \mathbb{P}(s_{t+1} = \sigma' | s_t = \sigma, a_t = \alpha)$;
- $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \mapsto \mathbb{R}$, a *reward function*, such that $r_{t+1} = r(\sigma, \alpha, \sigma')$ is the reward received when reaching, at time $t + 1$, state σ' having previously chosen action α at state σ;

the task of the agent is to find a *policy*, i.e. a sequence of decision rules, $\pi : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$, such that $\pi(\sigma, \alpha) = \mathbb{P}(a_t = \alpha | s_t = \sigma)$. An *optimal policy* π^* is one that maximizes some function R_t of the future rewards $r_{t+1}, r_{t+2}, \ldots$ A common definition, which will be used for our discussion, is to consider the sum of *discounted rewards* $R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$, where $0 < \gamma < 1$ is the discount factor.

In our discussion we will be especially interested in a subclass of MDPs that are *episodic*, i.e. the reinforcement function implicitly defines a set $\mathcal{G}$ of one or more goal states, when the goal is reached the environment is restarted in some initial state which could be chosen according to some probability distribution. In this case we

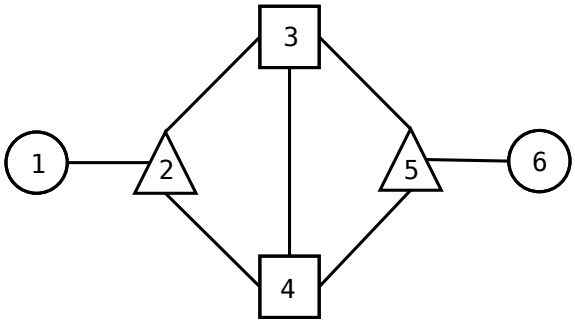


Figure 1. An example environment with 6 states.

also define [do Lago Pereira et al. 2008] a *probabilistic planning domain* to be the tuple $\mathcal{D} = \langle \mathcal{S}, \mathcal{A}, T \rangle$, and a *probabilistic planning problem* as the tuple $\mathcal{P} = \langle \mathcal{D}, \mathcal{G} \rangle$.

Example 1. Consider the environment represented on figure 1. Each geometric shape is a location that the agent can occupy, and if two locations are connected by an edge it means that the agent can reach one coming from the other. The state of the environment is represented by the location occupied by the agent[1]. We can define $\mathcal{S} = \{\sigma_1, \ldots, \sigma_6\}$, where for example σ_2 represents the configuration in which the agent is on the triangle with the number 2. The set of actions $\mathcal{A} = \bigcup_{i=1}^{6} A_{\sigma_i} = \bigcup_{i=1}^{6} \{\alpha_{ij}, j \in \{1, \ldots, 6\}\}$, represents a choice of transition for the agent, i.e. α_{12} means that the agent chooses to go to the triangle 2 when on state 1. Note that we also considered the possibility for the agent to stay in the state, denoted by α_{ii}. Let's assume for simplicity that the environment is deterministic, then $T(\sigma_i, a, \sigma_j)$ is equal to 1 if $a = \alpha_{ij}$ and zero otherwise. Our planning domain $\mathcal{D}$ is fully specified. Now suppose that we would like to learn how to reach state σ_6. We could define $r(s, a, s')$ to be equal to 0 if $s = s' = \sigma_6$ and -1 otherwise; with the reinforcement so defined, the agent maximizes R_t if he reaches σ_6 in the smallest number of steps and stays there afterwards. Thus the reinforcement function implicitly defines $\mathcal{G} = \{\sigma_6\}$, and our planning problem $\mathcal{P}$ is also specified. Since σ_2 contains two actions that are optimal, there are several possibilities for the optimal policy. We could have for example $\pi^*(\sigma_4, \alpha_{45}) = 1$, $\pi^*(\sigma_2, \alpha_{21}) = 0$ and $\pi^*(\sigma_2, \alpha_{23}) = \pi^*(\sigma_2, \alpha_{24}) = \frac{1}{2}$.

2.1. Reinforcement learning

Several methods are available for solving MDPs, in the present work we will consider Reinforcement Learning [Sutton and Barto 1998], which is particularly useful when T is not known to the agent. Within this framework it is common to define a *state action value function* $Q^\pi : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ that for a given policy π gives an estimate of the expected value $\mathbb{E}_\pi[R_t | s_t = \sigma, a_t = \alpha]$. This function is learned by direct interaction with the environment: at each time step the agent performs an action, observes the outcome and uses some strategy to update the estimate of the Q function; we refer to the book by Sutton and Barto for the details. With a good estimate of the Q function, the agent can maximize R_t by choosing $a_t = \arg\max_\alpha Q(\sigma, \alpha)$. It should be pointed out that both Q and π should be considered functions indexed by the time step t; this happens since Q is updated at (possibly) each time step, and π might change by, for example, taking into

[1]Note that there is a difference between a geometric shape that represents a location and a state: the state is a configuration of the environment, where the agent is at some location. This distinction will be particularly important on section 3.2

consideration the updated values of the Q function. In order to avoid heavy notation we won't make this dependence explicit.

Two problems arise from this approach. First, the agent must choose between exploring more the environment, in order to improve its estimate of Q, and exploiting his current knowledge to maximize his reward; second, if the goal, defined by the reinforcement function, is changed, the agent must learn everything from scratch. Our concern will be to alleviate the latter problem by abstracting a previously learned policy for a similar problem, using a relational language, to guide the initial exploration of the environment. By this we would like to provide a better initial performance for the agent, that would otherwise need to start exploring the environment randomly, while gradually replacing the previously learned policy with the one induced by the Q function he is currently learning.

Example 2. Back to the example of figure 1 and with $\mathcal{P}$ defined as in example 1, assuming the discounted reward model we can easily see that for example $Q^{\pi^*}(\sigma_4, \alpha_{45}) = -1 - \gamma$ and $Q^{\pi^*}(\sigma_4, \alpha_{43}) = -1 - \gamma - \gamma^2$. Note that α_{43} is not an optimal action; this is not a problem since Q^{π^*} tells us the value of R_t that we expect to obtain by choosing the action in the argument and then follow the optimal policy. As an illustration of the problem we face when the goal is changed, suppose that now we want to reach state σ_1. The symmetry of the figure makes it very clear how the problem is very similar to the one we solved before, but the reinforcement function has now changed to $r(s, a, s') = 0$ if $s = s' = \sigma_2$ and $r(s, a, s') - 1$ otherwise. We need to learn the Q^{π^*} from scratch since now for example $Q^{\pi^*}(\sigma_4, \alpha_{45}) = -1 - \gamma - \gamma^2 - \gamma^3$.

3. Relational representation

We begin this section with some definitions from first order logic. Then we provide a brief overview of relational reinforcement learning. With these concepts we finally make the concept of an abstract policy more precise.

3.1. Relational logic

In this subsection we present some definitions and notations from first order logic. The exposition is tailored to our needs; for a more formal treatment we refer to e.g. [Lloyd 1987].

A *relational alphabet* $\Sigma = P \cup C$ is composed of a set of predicates P and a set of *constants* C. If $\mathtt{t}_1, \ldots, \mathtt{t}_n$ are *terms*, i.e. each one is a variable or a constant, and if $\mathtt{p/n}$ is a predicate symbol with arity $n \geq 0$, then $\mathtt{p}(\mathtt{t}_1 \ldots \mathtt{t}_n)$ is an *atom*. A *conjunction* is a set of atoms; in our discussion each variable in a conjunction will be implicitly assumed to be existentially quantified. We denote by $vars(A)$ the set of variables of a conjunction A. *Background Knowledge* BK is a set of Horn Clauses. A *substitution* θ is a set $\{\mathtt{X}_1/\mathtt{t}_1, \ldots, \mathtt{X}_n/\mathtt{t}_n\}$, binding each variable $\mathtt{X}_i$ to a term $\mathtt{t}_i$; it can be applied to a term, atom or conjunction. If A and B are conjunctions and there is a substitution θ such that $B\theta \subseteq A$, we say that A is θ-*subsumed* by B, denoted by $A \preceq_\theta B$. A term is called *ground* if it contains no variables; in a similar way we define ground atoms and ground conjunctions. The *Herbrand base* B_Σ is the set of all possible ground atoms that can be formed with the predicates and constants in Σ.

3.2. Relationally factored Markov Decision Processes

A relationally factored Markov Decision Process (RMDP) [van Otterlo 2004] is a tuple $\langle \Sigma, BK, T, r \rangle$, where $\Sigma = D \cup P \cup A$, is a relational alphabet such that:

- D is a set of constants representing the objects of the environment;
- P is a set of predicates used to describe relations and properties among objects;
- A is a set of action predicates.

The set of states $\mathcal{S}$ of the RMDP is the set of all $\sigma \subseteq B_{P \cup D}$ satisfying the integrity constraints imposed by BK. The set of actions is $\mathcal{A} = B_{A \cup D}$. With $\mathcal{S}$ and $\mathcal{A}$ defined, T and r have the same meaning as described in section 2.

From the definition we can see that an RMDP is an MDP where the states and the actions are represented (factored) through a relational language. All definitions from section 2 are still valid here, and so are the methods of solution. The advantage of working with such a representation is the possibility of aggregating sets of states and actions by using abstract states. Specifically, an *abstract state* $\hat{\sigma} \in \widehat{\mathcal{S}}$ is a conjunction over P, D and BK; an *abstract action* $\hat{\alpha} \in \widehat{\mathcal{A}}$ is an atom over A and D. We can also define $\mathcal{S}_{\hat{\sigma}}$, the set of ground states covered by $\hat{\sigma}$, as $\mathcal{S}_{\hat{\sigma}} = \{\sigma \in \mathcal{S} | \sigma \preceq_\theta \hat{\sigma}\}$. Similarly we define $\mathcal{A}_{\hat{\alpha}} = \{\alpha \in \mathcal{A} | \alpha \preceq_\theta \hat{\alpha}\}$

Example 3. Let's review example 1 by using an RMDP model. There are many ways we can choose Σ; as an example let's consider $D = \{\texttt{c1}, \texttt{t2}, \texttt{s3}, \texttt{s4}, \texttt{t5}, \texttt{c6}\}$; $P = \{\texttt{in/1}, \texttt{circle/1}, \texttt{square/1}, \texttt{triangle/1}, \texttt{connected/2}\}$; $A = \{\texttt{goto/2}, \texttt{stay/1}\}$. The meaning of each predicate should be intuitive, we give one example to make it clearer: if the agent is occupying location 1 in the map, which we are representing as the object `c1`, we can describe the state of the environment as $\sigma_1 = \{\texttt{in(c1)}, \texttt{circle(c1)}, \texttt{connected(c1,t2)}, \texttt{triangle(t2)}\}$. If the agent makes a choice to go from location `c1` to location `t2`, we write this by `goto(c1,t2)`. We can also consider some abstract states, for example: $\hat{\sigma}_1 = \{\texttt{in(X)}, \texttt{connected(X,Y)}, \texttt{circle(Y)}\}$ represents all states in which the agent is in a location connected to a circle, i.e. $\mathcal{S}_{\hat{\sigma}_1} = \{\sigma_2, \sigma_5\}$; $\hat{\sigma}_2 = \{\texttt{in(X)}, \texttt{circle(X)}\}$, represents all states in which the agent is in a location that is a circle, i.e. $\mathcal{S}_{\hat{\sigma}_2} = \{\sigma_1, \sigma_6\}$. In $\hat{\sigma}_1$ the agent could consider the action `goto(X,Y)`, and in $\hat{\sigma}_2$ he could consider to stay on that set, i.e. the action `stay(X)`.

3.3. Abstract policies

We can now define an *abstract policy* as a list of abstract action rules [van Otterlo 2004] $\hat{\sigma} \xrightarrow{p_{\hat{\sigma},i}} \hat{\alpha}_i$, for $\hat{\sigma} \in \widehat{\mathcal{S}}$, $\hat{\alpha}_i \in \widehat{\mathcal{A}}$ satisfying $vars(\hat{\alpha}_i) \subseteq vars(\hat{\sigma})$; $p_{\hat{\sigma},i}$ is the probability of choosing $\hat{\alpha}_i$ when on a state covered by $\hat{\sigma}$.

There are several way of obtaining such a list. One way would be to solve the RMDP in the abstract setting [van Otterlo 2004, Kersting et al. 2004]. Another possibility, that would avoid giving the dynamics of the environment in a relational language, could be to learn an abstract policy by inducting a logical decision tree [Blockeel and De Raedt 1997] from the examples generated by applying the learned optimal policy for a specific problem: the set of abstract states is the set made by the union of all conjunctions in a path from the root to a leaf, and the set of abstract actions is the set of all abstract actions contained in the leafs. An abstract policy given by a teacher could also be an option.

An abstract policy induces a policy in the ground MDP, which we will name π_a.

Given a state σ, we find the first[2] $\hat{\sigma}$ such that $\sigma \preceq_\theta \hat{\sigma}$. We then have a set of abstract actions $\hat{\alpha}_i$, and associated probabilites. We could use those ones to assign the probability $\pi_a(\sigma, a)$, where $a \in \{\alpha \in \mathcal{A}_\sigma | \alpha \preceq_\theta \hat{\alpha}_i\}$. It should be pointed out that depending on how the abstract policy was obtained there could be no $\hat{\sigma}$ in the list for a given σ. In this case we could select randomly from $\mathcal{A}_\sigma$.

Example 4. Considering the RMDP model of example 3, one abstract policy could be:

$$\texttt{in(X),circle(X)} \xrightarrow{1.0} \texttt{stay(X)} \quad (1)$$

$$\texttt{in(X),connected(X,Y),circle(Y)} \xrightarrow{1.0} \texttt{goto(X,Y)} \quad (2)$$

$$\texttt{in(X),connected(X,Y),triangle(Y)} \xrightarrow{1.0} \texttt{goto(X,Y)} \quad (3)$$

Note how the order is important, since rule (1) applied to state σ_1 would suggest action α_{11}, while rule (3) would suggest α_{12}.

4. Using the abstract policy to guide initial exploration

4.1. Initial considerations

Let's start by recalling that we are dealing with a probabilistic planning domain, and that we want to provide a better initial performance by using some policy that uses previously acquired knowledge, while gradually learning the optimal one. We briefly add some notation to make our argument more compact. Since we are dealing with planning problems, we can consider that each task is represented by a tuple $\langle \sigma_0, \sigma_g \rangle$, where $\sigma_0 \in \mathcal{S}$ and $\sigma_g \in \mathcal{G}$. Let's also define $\mathcal{T}(\pi, \langle \sigma_0, \sigma_g \rangle) = \{s_1 s_2 \ldots s_n$ for some $n | s_1 = \sigma_0, s_n = \sigma_g, T(s_i, a_i, s_{i+1}) \neq 0$ for some a_i such that $\pi(s_i, a_i) \neq 0\}$, i.e. the set of all possible transition histories that start on state σ_0, and, by following policy π, reach at some time step the goal state σ_g.

Example 5. With $\mathcal{P}$ and π^* defined as in example 1, we have $\mathcal{T}(\pi^*, \langle \sigma_3, \sigma_6 \rangle) = \{\sigma_3\sigma_5\sigma_6\}$ and $\mathcal{T}(\pi^*, \langle \sigma_2, \sigma_6 \rangle) = \{\sigma_2\sigma_3\sigma_5\sigma_6, \sigma_2\sigma_4\sigma_5\sigma_6\}$.

In the following discussion π^* is the optimal policy for the task $\langle \sigma_0, \sigma_g \rangle$; π_r is a random policy that at each state σ chooses an action with uniform distribution from the set $\mathcal{A}_\sigma$; π_a is the policy induced by the abstract policy, as defined on section 3.3. Also let's define π_Q to be the policy that acts greedily with respect to the Q function, i.e. $\pi_Q(\sigma, \alpha^*) = 1$ if $\alpha^* = \arg\max_\alpha Q(\sigma, \alpha)$.

4.2. Using the abstract policy

The usual approach in reinforcement learning is to start learning the Q function estimate by random exploration of the environment, while gradually making use of the Q function to choose actions according to π_Q; algorithms such as ϵ-greedy and softmax action selection work in this way [Sutton and Barto 1998]. In our method we are going to make use of the previously learned knowledge contained in π_a for the initial exploration. However, for reasons that we will later explain, we are not going to fully replace π_r by π_a.

[2]Note that the abstract policy was defined as a list of action rules, i.e. order matters. In the case of a tree this mapping is also unambiguous.

We start by giving a better explanation as to why π_a should be better than π_r in the initial learning phase. If, facing a new task $\langle \sigma_0, \sigma_g \rangle$, the agent starts using π_r, he will consider all transitions in $\mathcal{T}(\pi_r, \langle \sigma_0, \sigma_g \rangle)$. On the other hand, we have $\mathcal{T}(\pi_a, \langle \sigma_0, \sigma_g \rangle) \subset \mathcal{T}(\pi_r, \langle \sigma_0, \sigma_g \rangle)$, and if we assume that π_a was learned considering a similar task, the actions that will be taken into consideration by the agent have the property they were good to solve a similar problem before. Therefore he should be able to obtain a better initial performance, i.e. a higher value of R_t in the first episodes when compared to the value he would obtain by only considering π_r.

Example 6. Let's go back to example 1, and let's consider π_a induced by the abstract policy of example 4. As previously hinted on example 2, if we change the goal from σ_6 to σ_2 the agent needs to learn everything from scratch. However note that for example $\mathcal{T}(\pi_a, \langle \sigma_3, \sigma_1 \rangle) = \{\sigma_3\sigma_2\sigma_1, \sigma_3\sigma_5\sigma_6\}$, while $\{\sigma_3\sigma_2\sigma_1, \sigma_3\sigma_5\sigma_6, \sigma_3\sigma_4\sigma_2\sigma_1, \sigma_3\sigma_5\sigma_4\sigma_2\sigma_1\} \subset \mathcal{T}(\pi_r, \langle \sigma_3, \sigma_1 \rangle)$, which is a countably infinite set.

As to why the agent should still consider π_r and gradually change to π_Q, we should note that not necessarily we have $\mathcal{T}(\pi^*, \langle \sigma_0, \sigma_g \rangle) \subset \mathcal{T}(\pi_a, \langle \sigma_0, \sigma_g \rangle)$, i.e. π_a could lead to non-optimal actions, or could not be able to solve the task. Therefore the agent still needs to explore actions not suggested by π_a, thereby justifying the use of π_r. Since proper convergence of the Q functions guarantees that $\pi_Q = \pi^*$ [Sutton and Barto 1998], in the long run the agent should act only according to π_Q to maximize his reward.

To make things more formal, we will consider that at each time step t the agent uses one of the policies in the set $\{\pi_Q, \pi_r, \pi_a\}$ probabilistically, i.e., we assume that we have two functions $\rho_t : \mathcal{S} \mapsto [0, 1]$ and $\gamma_t : \mathcal{S} \mapsto [0, 1]$ such that:

$$\pi(\sigma, \alpha) = \begin{cases} \pi_Q(\sigma, \alpha), & \text{with probability } \rho_t(\sigma) \\ \pi_r(\sigma, \alpha), & \text{with probability } [1 - \rho_t(\sigma)]\gamma_t(\sigma) \\ \pi_a(\sigma, \alpha), & \text{with probability } [1 - \rho_t(\sigma)][1 - \gamma_t(\sigma)] \end{cases} \tag{4}$$

From what we explained, both γ_t and ρ_t should converge to 1, in the limit, with a faster convergence rate for γ_t in order to guarantee that the environment was properly explored before using exclusively π_Q. As it can be seen, if we take γ_t to be identically equal to 1, we go back to the usual methods in reinforcement learning that we mentioned in the beginning of this subsection

5. A navigation problem

As an example of what we expect to obtain, consider the map on figure 2. We assigned some reference points represented by geometrical shapes, each having a meaning with respect to the semantic of the map, e.g. a square is the center of a room. We then modeled the map relationally, in a way similar to that of example 3.

Specifically, we define P = {`isCorridor/1`, `isRoom/1`, `isCenter/1`, `isNearDoor/1`, `in/1`, `isConnected/2`}; and A = {`gotoRDRD/2`, `gotoCDCD/2`, `gotoRCRD/2`, `gotoRDRC/2`, `gotoRDCD/2`, `gotoCDRD/2`, `gotoCCCD/2`, `gotoCDCC/2`}. To make notation more compact, we are considering the following abbreviations: `cd(X)` = {`isCorridor(X)`, `isNearDoor(X)`}; `cc(X)` =

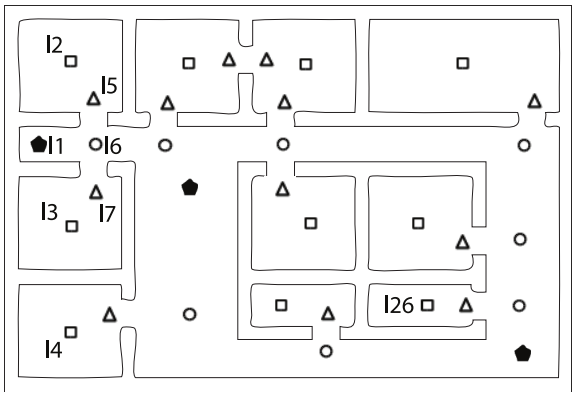


Figure 2. Map used in our navigation problem.

$\{$`isCorridor(X)`,`isCenter(X)`$\}$; `rc(X)` $=$ $\{$`isRoom(X)`,`isCenter(X)`$\}$; `rd(X)` $=\{$`isRoom(X)`,`isNearDoor(X)`$\}$.

The meaning of each predicate should be intuitive, we give one example to make it clearer: if the agent is occupying location 1 in the map, which we are representing as the object `l1` $\in$ D, we can describe the state as $\sigma_{\texttt{l1}} = \{$`in(l1)`, `isCorridor(l1)`, `isCenter(l1)`, `isConnected(l1,l6)`, `isCorridor(l6)`, `isNearDoor(l6)`$\}$. If the agent chooses to go from location $\sigma_{\texttt{l1}}$ in the map of Figure 2 to location $\sigma_{\texttt{l6}}$ we write this action by `gotoCCCD(l1,l6)`.

We started by learning the optimal policy to reach center room $\sigma_{\texttt{l2}}$ in the map, and then generated some examples by applying the optimal policy in a few tasks, where the goal state was kept fixed and the starting state was selected randomly. We then obtained an abstract policy by inducting a first order logical decision tree (FOLDT), using the TILDE algorithm [Blockeel and De Raedt 1997]. In the FOLDT each path from the root to a leaf represents an abstract state, and each leaf is a set of abstract actions. Recall from section 3.3 that each abstract action rule is of the form $\hat{\sigma} \xrightarrow{p_{\hat{\sigma},i}} \hat{\alpha}_i$; in our case we considered $p_{\hat{\sigma},i}$ to be the uniform distribution. To show the resulting abstract policy we converted the FOLDT into a list by doing a depth-first search; the result is shown below:

$$\texttt{in(X)},\texttt{cd(X)},\texttt{cc(Y)} \xrightarrow{0.33} \texttt{gotoCDCD(X,Z)}$$
$$\texttt{in(X)},\texttt{cd(X)},\texttt{cc(Y)} \xrightarrow{0.33} \texttt{gotoCDCC(X,Y)}$$
$$\texttt{in(X)},\texttt{cd(X)},\texttt{cc(Y)} \xrightarrow{0.33} \texttt{gotoCDRD(X,Z)}$$
$$\texttt{in(X)},\texttt{cd(X)} \xrightarrow{1.0} \texttt{gotoCDCD(X,Z)}$$
$$\texttt{in(X)},\texttt{connected(X,Y)},\texttt{rc(Y)} \xrightarrow{0.33} \texttt{gotoRDCD(X,Z)}$$
$$\texttt{in(X)},\texttt{connected(X,Y)},\texttt{rc(Y)} \xrightarrow{0.33} \texttt{gotoRDRC(X,Y)}$$
$$\texttt{in(X)},\texttt{connected(X,Y)},\texttt{rc(Y)} \xrightarrow{0.33} \texttt{gotoRDRD(X,Z)}$$
$$\texttt{in(X)},\texttt{rc(Z)} \xrightarrow{1.0} \texttt{gotoRCRD(X,W)}$$
$$\texttt{in(X)} \xrightarrow{1.0} \texttt{gotoCCCD(X,W)}$$

In order to take full advantage of all available information in the abstract policy, we further

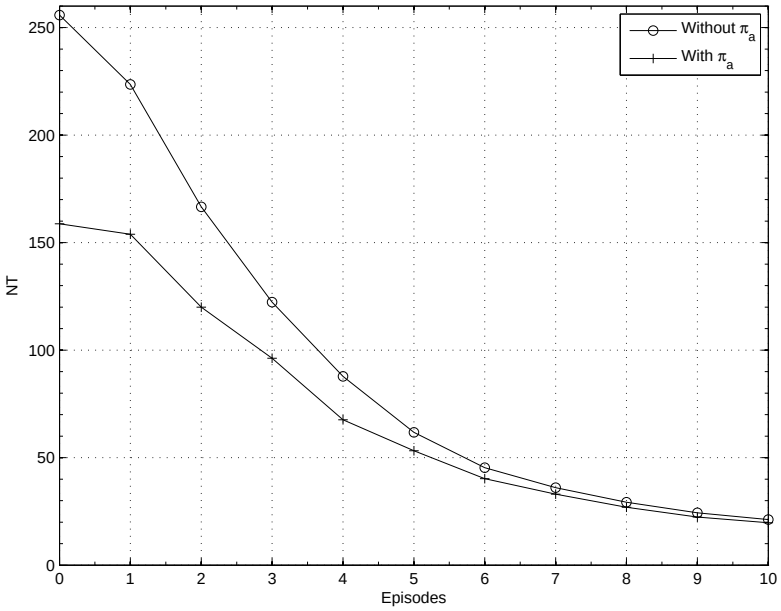


Figure 3. Average number of transitions NT required to reach the goal, with and without the use of the abstract policy π_a.

considered that a ground action suggested by the abstract policy could not be suggested again by π_a in the same episode, i.e. if during some episode the agent is at state σ and if the abstract policy suggests action $\alpha \in \mathcal{A}_\sigma$, then the next time, during the same episode, being the agent at state σ and using π_a for choosing the action to be taken, the set of possible ground actions suggested by π_a is reduced to $\mathcal{A}_\sigma - \{\alpha\}$. By episode we mean a sequence of steps that takes the robot from an initial location to the goal state, as discussed in section 2.

We assumed that the abstract policy could guide the robot in a sub-optimal way to the goal state, so that we initialized $\rho_0 = 0$ and $\gamma_0 = 0$, and at each episode we incremented ρ by 0.05 and γ by 0.06, according to what we discussed in section 4.2.

After this initial setup, we used the abstract policy to solve a new task. Specifically, we considered task $\langle \sigma_{11}, \sigma_{126} \rangle$, and we looked at the average number NT of transitions (steps) required to reach the goal at each episode. We repeated this task 1000 times and took the average value for each episode. The result for the first 10 episodes can be seen on Figure 3.

The strategy that uses π_a starts with a smaller average number of transitions and gradually decreases this value due to the learning process. The strategy that uses only π_r starts with a higher average number of steps. This difference in the beginning of the curves shows that the abstract policy can transfer knowledge between problems, thus obtaining a better initial performance as claimed.

During the learning process the distance between the curves is reduced (approximately in the fifth episode). This is due to the fact that π_a is gradually replaced by π_r during the learning process (by increasing γ_t) to guarantee the convergence of the Q function estimate.

6. Conclusions

In this work we have proposed a method to take into account existing solutions to similar problems when learning a policy to solve a new problem, so that we could improve initial performance of an RL learner. We explore the powerful possibilities of using relational representations, which facilitate policy abstraction and generalization between collections of related tasks. Our proposal consisted in abstracting policies from solved problems and then using these abstractions in a combined way in an RL learner, in order to significantly enhance the initial performance of the learning process in a new problem not yet solved.

Empirical evaluation of our approach in the robotic navigation problems were carried out. Experimental results showed that the performance of the learning algorithm can be improved even using a very simple abstract policy.

An important topic to be investigated in future works is concurrently to improve the abstract policy using experiences conducted in the current learning process so that this policy could generalize a wider class of similar problems and improve the current and future learning processes with better policy directions.

Acknowledgments

We are grateful to FAPESP (grants 10/02379-9, 09/04489-9, 09/14650-1, 8/03995-5) and to CNPQ (grant 475690/2008-7) for the support.

References

Bianchi, R. A. C., Ribeiro, C. H. C., and Costa, A. H. R. (2007). Heuristic selection of actions in multiagent reinforcement learning. In *IJCAI*, pages 690–695.

Bianchi, R. A. C., Ribeiro, C. H. C., and Costa, A. H. R. (2008). Accelerating autonomous learning by using heuristic selection of actions. *Journal of Heuristics*, 14(2):135–168.

Blockeel, H. and De Raedt, L. (1997). Top-down induction of logical decision trees. In *Artificial Intelligence*.

Burkov, A. and Chaib-draa, B. (2007). Adaptive play q-learning with initial heuristic approximation. In *ICRA*, pages 1749–1754.

do Lago Pereira, S., de Barros, L., and Cozman, F. (2008). Strong probabilistic planning. In Gelbukh, A. and Morales, E., editors, *MICAI 2008: Advances in Artificial Intelligence*, volume 5317 of *Lecture Notes in Computer Science*, pages 636–652. Springer Berlin / Heidelberg.

Drummond, C. (2002). Accelerating reinforcement learning by composing solutions of automatically identified subtasks. *Journal of Artificial Intelligence Research*, 16:59–104.

Kersting, K., Otterlo, M. V., and Raedt, L. D. (2004). Bellman goes relational. In *In ICML*, pages 465–472. ACM.

Kersting, K., Plagemann, C., Cocora, A., Burgard, W., and Raedt, L. D. (2007). Learning to transfer optimal navigation policies. *Advanced Robotics: Special Issue on Imitative Robots*, 21(13):1565—1582.

Knox, W. B. and Stone, P. (2010). Combining manual feedback with subsequent MDP reward signals for reinforcement learning. In *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*.

Lloyd, J. W. (1987). *Foundations of Logic Programming, 2nd Edition*. Springer Verlag.

Madden, M. G. and Howley, T. (2004). Transfer of experience between reinforcement learning environments with progressive difficulty. *Artif. Intell. Rev.*, 21:375–398.

Puterman, M. L. (1994). *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition.

Sherstov, A. A. and Stone, P. (2005). Improving action selection in MDP's via knowledge transfer. In *Proceedings of the Twentieth National Conference on Artificial Intelligence*.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning)*. The MIT Press.

Taylor, M. E. and Stone, P. (2009). Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(1):1633–1685.

Uther, W. T. B. and Veloso, M. M. (2002). Ttree: Tree-based state generalization with temporally abstract actions. In *In Proceedings of SARA-2002*, pages 260–290.

van Otterlo, M. (2004). Reinforcement learning for relational MDPs. In Nowe, A., Lenaerts, T., and Steenhaut, K., editors, *Proceedings of the Machine Learning Conference of Belgium and the Netherlands, BeNeLearn '04*, pages 138–145, Brussels. Brussels.

Space D*: Um Algoritmo para Path Planning Multi-Robôs

Renan Maffei[1], **Silvia Botelho**[1], **Luan Silveira**[1], **Paulo Drews Jr.**[1], **Nelson Duarte Filho**[1], **Alessandro Bicho**[1], **Felipe Almeida**[1], **Matheus Longaray**[1]

[1]Centro de Ciências Computacionais (C3)
Universidade Federal do Rio Grande (FURG) – Rio Grande – RS – Brazil

silviacb@furg.br

Abstract. *This paper describes a new method of path planning for multiple robots in unknown environments. The method, called Space D*, is based on two algorithms: the D*, which is a incremental graph search algorithm and the Space Colonization algorithm, previously used to simulate crowd behaviors. The major contribution of the proposed method is the focus on generating paths in spacious environments to help facilitate the control of robots, and thus presenting itself in a viable way for use in real environments. The results obtained validates the approach and show the advantage in relation to D* method.*

Resumo. *Este trabalho descreve um novo método de path planning para múltiplos robôs em ambientes desconhecidos. O método, chamado Space D*, é baseado em dois algoritmos: o D*, que é um algoritmo de busca em grafos incremental, e o algoritmo de Colonização do Espaço, utilizado previamente para a simulação do comportamento de multidões. A maior contribuição do método proposto é o enfoque sobre a geração de caminhos por ambientes espaçosos, visando facilitar o controle dos robôs, e, dessa forma, apresentando-se de forma viável para a utilização em ambientes reais. Os resultados obtidos validam a abordagem e mostram a vantagem em relação ao método D*.*

1. Introdução

Planejar movimentos livres de colisão para robôs autônomos situados em ambientes com obstáculos é um dos principais problemas da robótica [Latombe 1991] [LaValle 2006].

Uma área do planejamento de caminhos bastante estudada é a de ambientes com múltiplos robôs independentes. Para esse tipo de problema, dois tipos de abordagens podem ser efetuadas: a abordagem centralizada e a desacoplada. Na abordagem centralizada, o planejamento é feito considerando-se um grande e único robô formado por vários segmentos (cada um dos robôs do ambiente), situado dentro de um grande espaço de configuração (correspondente à união dos espaços de configuração de cada robô), onde cada segmento tem uma configuração final desejada (o objetivo de cada robô). O problema deste tipo de abordagem é que o espaço de configuração composto pelos robôs cresce exponencialmente com o aumento do número deles, o que se torna altamente inviável.

Em contra partida, apesar de incompleta, a abordagem desacoplada reduz o problema em planejar a movimentação de cada robô individualmente e posteriormente adequá-las. Além disso, uma alternativa viável são técnicas desacopladas distribuídas, considerando-se a facilidade de uso de técnicas de processamento distribuído para a implementação de tais abordagens. Nessas técnicas, cada robô planeja a sua

movimentação baseando-se no seu conhecimento local e nas interações com outros robôs. A diferença principal das abordagens distribuídas para as técnicas somente desacopladas é que a etapa de adequação do caminho deve ser feita de forma distribuída e em tempo real. Técnicas distribuídas apresentam maior robustez, pois aceitam melhor as falhas e incertezas na atuação de cada robô, ou seja, um único robô defeituoso não interrompe o funcionamento do todo.

Outra complexidade associada ao problema de planejamento de caminhos refere-se ao nível de conhecimento sobre o ambiente, isto é, a certeza sobre a presença de obstáculos no cenário. Uma estratégia de navegação utilizada quando não se possui o conhecimento exato do ambiente é o chamado *path planning* baseado em sensoriamento [Choset and Burdick 1994]. Neste tipo de técnica planeja-se o menor caminho baseado na configuração atual do ambiente e conforme novos obstáculos são detectados, novos caminhos devem ser planejados.

Um dos algoritmos mais populares para o tratamento deste tipo de problema é o algoritmo D* (Focussed Dynamic A*) [Stentz 1995], que adapta a otimalidade do A* para a dinamicidade do ambiente, mesclando heurísticas com buscas incrementais, e alcançando consideráveis ganhos sobre execuções repetidas do A*.

Diversos trabalhos foram propostos baseando-se no D* e, posteriormente, no D* Lite [Koenig and Likhachev 2002], que implementa a mesma estratégia do D*, porém de forma simplificada e com um grau de eficiência igual ou até maior do que o D*. Em [Ferguson and Stentz 2005], foi proposta uma extensão para o D* e D* Lite utilizando interpolação linear para produzir caminhos mais suaves, contornando, assim, a questão de poucas possibilidades de transição entre as células. Outra extensão do D* Lite foi proposta em [Likhachev et al. 2005], onde o algoritmo foi adaptado para sistemas com escassez de tempo. Neste tipo de técnica, como em [van den Berg et al. 2006], o planejamento é feito incrementalmente e conforme se dispõe de mais tempo, refina-se a solução obtida.

O trabalho atual propõe uma extensão do algoritmo D* para ambientes multi-robôs, visando criar uma forma de colaboração entre os robôs e assim facilitar o processo de planejamento de caminhos. O algoritmo, denominado Space D*, combina o D* com o algoritmo de Colonização do Espaço, projetado originalmente para a simulação da modelagem de plantas [Runions et al. 2005], e posteriormente adaptado para a simulação do comportamento de multidões [Bicho 2009]. A principal característica do algoritmo de Colonização do Espaço é a preferência de deslocamento por espaços livres, o que evita o risco de colisão dado a incerteza e a escalabilidade de obstáculos fixos e móveis.

Por focar-se na criação de caminhos por ambientes com maior espaço livre, o método visa facilitar o controle dos robôs, apresentando-se de forma viável para a utilização em ambientes reais. Métodos que simplesmente buscam o menor caminho tendem a se aproximar dos obstáculos, e isto implica em reduções na velocidade dos robôs, e consequentemente, aumento no tempo total de execução (apesar de obterem menores distâncias totais percorridas). Por outro lado, alguns métodos que buscam ambientes livres, como campos potenciais, não garantem a obtenção de caminhos (caindo em mínimos locais), diferentemente do Space D*, que herda esta característica do D*.

Além disso, através da troca de informações, busca-se reduzir as distâncias per-

corridas por cada robô, visto que, ao adquirir conhecimento maior sobre o ambiente, o planejamento de caminhos torna-se mais preciso.

2. Trabalhos Relacionados

Grande parte dos trabalhos sobre planejamento de caminhos de múltiplos robôs, como [van den Berg et al. 2009] [Clark et al. 2003], baseiam-se em ambientes que devem estar totalmente determinados. Desta forma, a aplicação de técnicas como planejamento priorizado e coordenação de caminhos fixos em ambientes desconhecidos pode ser realizada desde que juntamente com alguma abordagem para tratamento de incerteza.

Por sua vez, os algoritmos de planejamento de caminhos em ambientes desconhecidos, como [Stentz 1995] [Koenig and Likhachev 2002] [Ferguson and Stentz 2005], são desenvolvidos para robôs independentes. No entanto, podem claramente ser aplicados em problemas com múltiplos robôs considerando os outros robos no cenário como obstáculos dinâmicos.

Porém, existem alguns trabalhos que se aproveitam da presença de múltiplos robôs com a capacidade de se comunicar e desenvolvem técnicas colaborativas para o planejamento de caminhos lidando com a incerteza. A grande maioria dos trabalhos de planejamento colaborativo de caminhos para múltiplos robôs situados em ambientes desconhecidos focam-se no problema de exploração do ambiente. Em [Rekleitis et al. 1997], o cenário é separado em faixas que são exploradas por times de robôs, sendo que dentro de cada time somente um robô se move de cada vez, visando reduzir erros de odometria. Em [López de Màntaras et al. 1997], robôs exploram aleatoriamente o ambiente e trocam informações sobre obstáculos quando se encontram.

Entretanto, existem alguns trabalhos que visam o planejamento de caminhos em si. Em [Pereira et al. 2003], utilizando funções de navegação, os robôs mantêm-se próximos uns dos outros para poderem trocar informações, enquanto rumam em direção aos seus objetivos. Em [Chen and Li 2005], robôs que precisam ir a um destino comum, são coordenados por um robô específico, enquanto os restantes responsabilizam-se por coletar as informações do ambiente. Em [Okada et al. 2007], é proposto um algoritmo baseado em *Roadmaps* Probabilísticos (PRM), onde os robôs possuem diferentes prioridades, requisitadas para prevenção de colisões, e são capazes de trocarem informações sobre suas trajetórias. Em [Pallottino et al. 2007], são utilizadas regras sociais para realizar o planejamento local em robôs, evitando a ocorrência de colisões de forma totalmente descentralizada. Já em [Jung et al. 2010], robôs com objetivos em comum trocam informações sobre o mapa quando se cruzam, para evitar que áreas sejam revisitadas.

3. Técnicas Utilizadas

O Space D* conforme mencionado anteriormente tem como base dois algoritmos que serão comentados brevemente a seguir: o primeiro é o D* e o segundo é o algoritmo de Colonização do Espaço.

3.1. D*

Conforme citado anteriormente, os algoritmos D* [Stentz 1995] e D* Lite [Koenig and Likhachev 2002] são uma das abordagens mais utilizadas e eficientes para

o problema de planejamento de caminhos em ambientes desconhecidos. De fato, atualmente, o D* Lite passou a ser utilizado em lugar do D* pela grande maioria dos pesquisadores, e é nele que o Space D* se baseia.

O comportamento básico de ambos algoritmos consiste em recalcular o menor caminho para o destino sempre que os custos de alguma célula variarem (algum obstáculo ou espaço livre for descoberto). Isto pode ser visualizado na Figura 1, onde a posição atual do robô é o nó marcado com um ponto, o destino é o nó marcado com um "x" e o caminho computado são os nós hachurados. Os nós preto representam obstáculos já descobertos.

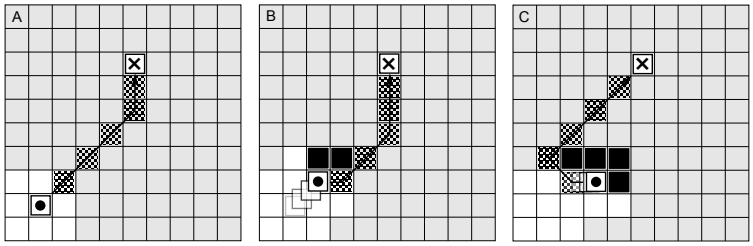


Figure 1. Funcionamento do D*/D* Lite. a)O robo sente o ambiente e calcula o caminho mínimo. b) Move-se para o próximo nodo no caminho, le os sensores, detecta um obstáculo e recalcula o caminho. c) Vai para a direita e executa o mesmo procedimento de (b).

O funcionamento do D* Lite é inspirado no A*, sendo que a cada atualização de uma célula, apenas as células que forem influenciadas por essa mudança serão recalculadas. Após esse processo, o menor caminho até o objetivo é calculado. Mais informações sobre o algoritmo podem ser encontradas em [Koenig and Likhachev 2002];

3.2. Algoritmo de Colonização do Espaço

Baseado em estudos da biologia, que atribuíram o desenvolvimento das nervuras a um hormônio chamado auxina presente nas folhas, o algoritmo de Colonização do Espaço foi desenvolvido em [Runions et al. 2005] para simular o crescimento das nervuras de uma folha em relação à distribuição das auxinas nesta folha.

O algoritmo é composto basicamente por três processos. O primeiro deles é o crescimento da lâmina da folha. Paralelamente a ele ocorrem os outros dois: o desenvolvimento do padrão de nervuras, que é influenciado diretamente pela distribuição das fontes de auxinas na folha (representando espaços livres), e a geração de novas auxinas, que por sua vez é afetada pelo desenvolvimento das nervuras.

O processo de crescimento da nervura começa através da associação de cada auxina ao nodo da nervura mais próximo. Em seguida, os nodos se expandem na direção das auxinas a que se associaram. Conforme o padrão cresce e se aproxima das auxinas, estas vão sendo removidas, visto que o espaço que elas representam deixa de ser livre. Depois, a lâmina da folha cresce e novas auxinas são geradas aleatoriamente, distribuídas de forma uniforme pelo espaço livre. Então, o ciclo recomeça com a associação auxinas-nodos e continua até a folha atingir o tamanho máximo. Como é possível perceber, o algoritmo é caracterizado como uma competição dos nodos da nervura pelas auxinas, representantes de espaços livres, visto que quanto mais auxinas um nodo consegue se associar, mais espaço ele tem para se expandir.

Em [Bicho 2009], o algoritmo de Colonização do Espaço foi adaptado para a simulação do comportamento de multidões. Os conceitos de nodos da nervura e auxinas foram transformados em agentes e marcadores de espaços livres, ou seja, o algoritmo de Colonização do Espaço passou a ser utilizado para simular a competição de agentes pelos espaços livres do ambiente.

Ainda em [Bicho 2009], algumas modificações sobre o algoritmo foram propostas. A primeira é que os marcadores se tornaram persistentes, isto é, não são mais "consumidos" pelos agentes, mas sim alocados quando algum agente se aproxima e liberados quando se afasta. Outra modificação é que apenas os marcadores dentro de certa região ao redor de um agente podem influenciá-lo (antes qualquer auxina podia influenciar qualquer nodo da folha, desde que este fosse o mais próximo dela). Uma terceira modificação é que a direção de movimento também passa a ser guiada pela intenção de se alcançar um destino específico e não apenas por grandes disponibilidades de marcadores. E uma última é que a velocidade de movimentação dos agentes passa a variar conforme o espaço livre que ele dispõe (no algoritmo original a velocidade era uniforme).

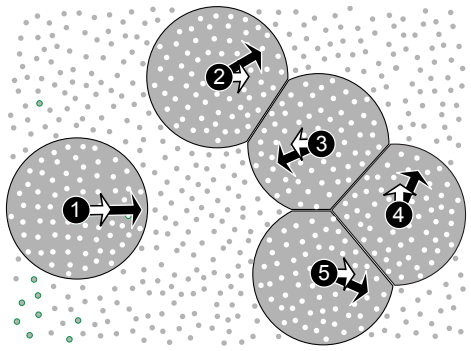


Figure 2. Algoritmo de Colonização do Espaço para simulação do comportamento de multidões.

Um exemplo de execução do algoritmo pode ser observado na Figura 2. Os agentes são os círculos pretos numerados e as setas brancas indicam a direção de seus destinos. As áreas maiores são as regiões de marcadores que cada agente foi capaz de alocar. Como se pode observar, as restrições nas regiões alocadas influenciam diretamente o vetor de movimento de cada agente, representado pela seta preta. Cada agente busca a direção do seu destino, mas evitando a proximidade com outros agentes ou obstáculos. Contudo, o algoritmo de Colonização do Espaço sozinho não resolve o problema de planejamento de caminhos em ambientes com obstáculos, mas a sua característica de busca por espaços livres motivou a mesclagem do algoritmo com o D* Lite para o planejamento de caminhos para múltiplos robôs em ambientes desconhecidos.

4. Space D*

O algoritmo desenvolvido neste trabalho para o planejamento de caminhos de múltiplos robôs em ambientes desconhecidos relaciona o algoritmo D* ao algoritmo de Colonização do Espaço e por esse motivo recebeu o nome de Space D*. O funcionamento resumido do método é o seguinte:

- O ambiente é representado por um *grid* formado por marcadores, que indicam os custos para um robô atravessá-lo (se está livre ou se é um obstáculo). No início, o

mapa é vazio e conforme obstáculos são detectados, o mapa é preenchido.
- O algoritmo D* Lite é aplicado por cada robô sobre o respectivo conjunto de marcadores, atualizando as distâncias para o destino de cada um;
- A navegação do robô é feita usando uma modificação do algoritmo de Colonização do Espaço. O robô aloca todos os marcadores livres dentro de sua área de alcance e com as distâncias para o destino desses marcadores calcula a sua direção de movimento, que sempre se dará dentro da zona alocada;
- Quando os robôs se cruzam, são trocadas informações sobre seus respectivos mapas, permitindo a atualização dos custos dos marcadores. Além disso, o algoritmo de Colonização do Espaço busca evitar colisões entre eles, pois cada robô aloca apenas os marcadores mais próximos de si;

Uma das principais contribuições do método proposto é a adição, ao D*, da preferência por ambientes mais espaçosos, o que conduz a uma robustez maior. Na sua definição original, o D* gera caminhos que tendem a passar muito perto de obstáculos, a fim de diminuir a distância total percorrida, no entanto, isso acaba exigindo um controle muito preciso na hora de executar a movimentação dos robôs.

Outra vantagem do método proposto é a troca de informações entre os robôs, enfoque de alguns trabalhos relacionados, como [Jung et al. 2010]. Como cada robô precisa mapear o ambiente por onde passa (para realizar o cálculo das distâncias para o destino), as informações sobre a presença de obstáculos podem ser trocadas quando robôs se encontrarem. Isto gera a possibilidade de se evitar caminhos que estejam bloqueados e que só futuramente o robô perceberia.

Contudo as mesmas restrições existentes no D*, continuam presentes no Space D*. Há a necessidade de discretização do mundo, podendo esta ter uma densidade variável(número de células no mapa). Por ser uma algoritmo de busca em grafo, quanto maior a densidade, mais demorada será a sua execução.

Outra importante restrição do método, é que, em sua atual implementação, a etapa de colonização do espaço considera os robôs como infinitesimais, ou seja, os seus corpos são considerados pontos. No entanto, para o correto funcionamento do algoritmo deve-se garantir que todos os robôs nunca saiam das respectivas regiões de marcadores alocados e que, conforme essas regiões diminuem, os seus deslocamentos também diminuam. Para robôs reais, o que o algoritmo garante é que o centróide do robô nunca sairá da região delimitada, porém isto pode ocasionar problemas quando a área alocada de um robô for muito pequena.

4.1. Funcionamento do Space D*

O Algoritmo 1 descreve o funcionamento do Space D*. À seguir cada uma das etapas é explicada.

Mapeamento do Ambiente: A etapa inicial do processo consiste no mapeamento do ambiente, onde cada robô percebe a presença de obstáculos e define os custos dos marcadores associados a cada posição.

Algoritmo 1: Space-D*

```
Mapeia o ambiente.
se Detectar Outros Robos então
    Troca informações com os outros robôs.
fim se
Atualiza os custos dos marcadores.
Calcula D* Lite.
Aloca os marcadores mais próximos.
Calcula a direção de movimento.
```

Detecção de outros robôs e troca de informações: Em seguida cada robô detecta a existência de robôs ao seu redor. Para isso ser possível cada robô possui um identificador único, o que faz com que quando um robô se aproximar de outro, ele possa perceber qeu o obstáculo é, na verdade, um robô e ainda qual robô. Quando dois robôs se encontrarem, ambos param por um momento e trocam informaços correspondentes às suas descrições do mapa.

Atualização dos custos dos marcadores: Após o mapeamento e troca de informações entre os robôs, deve-se atualizar os custos dos marcadores. Vale destacar que na atual implementação, todos os obstáculos são estáticos (excetuando-se os próprios robôs), assim a atualização dos custos dos marcadores consiste em verificar se no mapa recebido existem novos obstáculos que no mapa atual ainda não foram descobertos.

Cálculo do D* Lite: Na sequência são atualizadas as distâncias para o destino partindo de cada marcador de interesse do robô. Com as distâncias atualizadas calcula-se o menor caminho para o destino, segundo o algoritmo D* Lite [Koenig and Likhachev 2002].

Alocação de marcadores mais próximos Na próxima etapa, os robôs alocam os marcadores mais próximos de si. Esta etapa é descentralizada, pois cabe a cada robô alocar o conjunto dos marcadores situados dentro de seu alcance, considerando apenas aqueles que se encontram mais perto de si do que de outros robôs. Uma restrição reside sobre o tamanho do raio da região de alocação. Para que não ocorram colisões entre os robôs é necessário que o alcance do sensor de percepção de robôs seja sempre maior que o raio da região de marcadores alocados. O ideal é que o alcance do sensor seja no mínimo o dobro do raio de alocação, pois assim garante-se que no exato momento em que dois robôs se encontrarem, cada um estará alocando apenas os marcadores mais próximos de si.

Cálculo da direção de movimento Na última etapa, para calcular a sua direção de movimentação, cada robô utilizará as distâncias para o destino dos marcadores de sua região de interesse. O cálculo do vetor de movimento é uma soma vetorial ponderada dos vetores que ligam o robô a cada marcador associado. Porém, diferentemente de [Bicho 2009], onde o tamanho de cada vetor era relacionado à direção do destino, agora os módulos dos vetores baseiam-se na distância entre marcador e destino, de forma que

quanto menor a distância, maior deve ser o módulo do vetor. A definição da função f que determina o módulo do vetor de um marcador s associado a um robô r é:

$$f(s) = max_{p \in S(r)}(g(p)) - g(s)$$

Ou seja, o tamanho de cada vetor é a subtração da maior distância ($g(p)$) de um marcador pertencente ao conjunto $S(r)$ do robô, pelo valor de g de cada marcador.

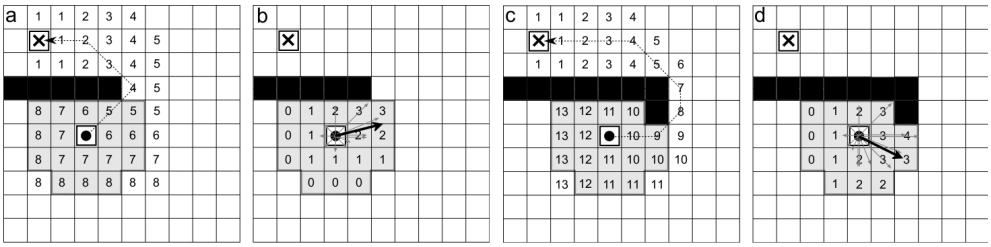


Figure 3. Exemplo de execução do Space D*. A subfigura (a) demonstra os passos 5 e 6 do processo. O passo 7 é mostrado na subfigura (b). (c) e (d) demonstram os mesmos passos na próxima iteração do algoritmo, depois de encontrar outros obstáculos.

Na Figura 3, pode ser observado um exemplo da execução do Space D*. A posição do robô é o nó indicado com um círculo, o destino é o nó indicado com um "x" e os obstáculos são os nós pretos. Em (a), a área demarcada representa a região alocada pelo robô, com as distâncias para o destino de cada marcador (em pontilhado está o menor caminhado calculado pelo D* Lite). Em (b), são gerados os vetores na direção dos marcadores com o módulo calculado por f, com os quais se obtém o vetor de movimento (vetor maior). Em (c), após deslocar-se na direção indicada anteriormente, o robô percebe novos obstáculos e recalcula as distâncias para o destino. Em (d), um novo vetor de movimento é gerado.

5. Testes e Validação

O algoritmo Space D* foi validado junto a um conjunto de testes simulados de ambientes multi-robôs nos softwares Player/Stage. Ambos fazem parte do Player Project [Gerkey et al. 2003]. Os mapas utilizados podem ser vistos na Figura 4.

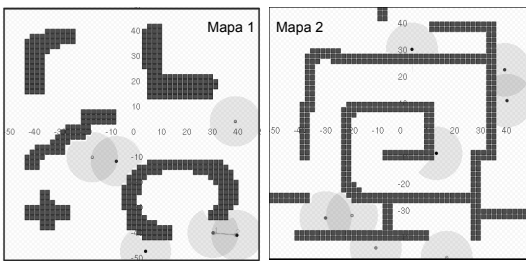


Figure 4. Mapas usados nos testes.

A validação foi realizada comparando-se as soluções obtidas na execução do Space D* com os resultados da execução do D* Lite. Verificou-se que, assim como o D* Lite, o Space D* gera trajetórias que sempre alcançam o destino sem a ocorrência de

colisões com obstáculos, mas além disso, conforme foi previsto, ele busca manter o robô afastado de qualquer obstáculo, seja dinâmico (outros robôs) ou estático (paredes).

Para o método utilizado para a comparação, a direção de movimentação dos robôs é a indicada pelo menor caminho computado pelo algoritmo D*. Como se pode deduzir, se não existirem erros e imprecisões no controle dos robôs, tal técnica funcionará perfeitamente. No entanto, muitas vezes ocorrem colisões ao se gerar caminhos próximos de obstáculos, a não ser que as velocidades de deslocamento dos robôs sejam reduzidas nessas localidades.

Na Figura 5 é mostrada uma comparação entre as trajetórias geradas na execução do Space D* e do D* no mapa 1 da Figura 4. São analisados os caminhos que três robôs dispersos aleatoriamente percorreram. É possível ver que as trajetórias obtidas com o Space D* passam longe o suficiente dos obstáculos, permitindo que a velocidade se mantenha próxima do limite estabelecido. Por outro lado, utilizando apenas o D*, a trajetória tangencia os obstáculos e, em virtude disso, em velocidade bem abaixo da máxima.

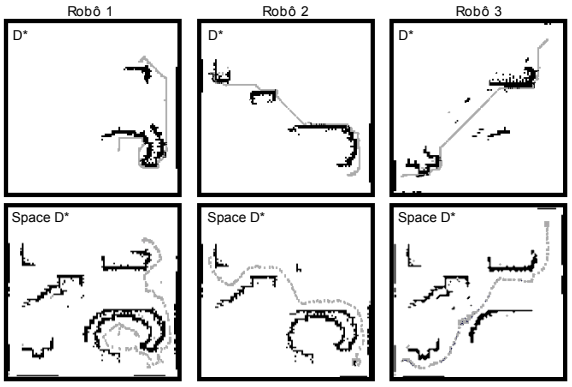


Figure 5. Comparação entre caminhos gerados pelo D* e pelo Space D* para três robôs. Os segmentos pretos são obstáculos e os cinzas são as trajetórias geradas pelos algoritmos.

Na tabela 1 foram comparados os tempos médios de execução, as distâncias totais médias e as velocidades médias dos robôs entre a simulação utilizando o Space D* e a simulação utilizando apenas o D*. Em todos os casos de teste o ambiente tinha 100x100 metros (e utilizou-se uma discretização de 100x100 marcadores) e a velocidade máxima dos robôs era de 5 m/s. Foram avaliados três conjuntos de teste (10, 15 e 20 robôs) cada um com cinco configurações aleatórias de posições iniciais e finais para cada robô.

É preciso ressaltar que em termos de tempo de execução de algoritmo, por efetuar um processamento maior, o Space D* é mais lento que o D* Lite. De fato, utilizando-se o D* Lite cada ciclo de simulação foi de aproximadamente metade do tempo do ciclo do Space D*. Contudo, por evitar a aproximação de obstáculos o Space D* permite uma taxa de atualização menor. Com isso, o que acaba influenciando majoritariamente os tempos de simulação são as velocidades médias de cada robô, associadas às distâncias percorridas.

O resultado obtido foi que com o Space D* o tempo médio de simulação se mostrou sempre menor (cerca de 30%), isto porque, apesar de o D* obter trajetórias

	Utilizando o Space D*					
Nº de Robôs	Tempo (s)		Distância (m)		Velocidade (m/s)	
	Média	DP	Média	DP	Média	DP
10	25.6	7.6	89.74	38.16	3.71	0.82
15	29.7	12.2	88.91	36.66	3.03	0.7
20	33.2	16.8	86.29	35.07	2.69	0.61
	Utilizando apenas o D*					
Nº de Robôs	Tempo (s)		Distância (m)		Velocidade (m/s)	
	Média	DP	Média	DP	Média	DP
10	33.2	10.3	83.21	29.61	2.57	0.64
15	42.4	16.0	85.68	31.07	2.13	0.58
20	50.9	21.8	86.91	33.48	1.74	0.39

Table 1. Comparação entre o Space D* e o método usando apenas o D*.

menores (como pode-se observar pelas distâncias médias), a velocidade média do Space D* é consideravelmente maior. Com isso, conclui-se que, em situações realistas, o uso do Space D* se mostrou mais aconselhável, visto que apesar de o D* tratar a incerteza sobre o espaço de configuração do ambiente, ele desconsidera a incerteza no controle dos robôs.

Na sequência, resolveu-se analisar se a existência de troca de informações entre os robôs contribui de forma significativa para o desempenho do Space D*. Para esse teste os dois mapas da Figura 4 foram utilizados, com três conjuntos de teste (10, 15 e 20 robôs) cada um com cinco configurações aleatórias de posições iniciais e finais para cada robô. Os resultados podem ser vistos na tabela 2. Para o primeiro mapa, a troca de informações não contribuiu para uma melhora no tempo de execução, pelo contrário, se mostrou menos eficiente que o método sem troca de informações para o exemplo com 10 robôs. No entanto, nos testes com 20 robôs, o método com trocas de informações obteve um melhor resultado. Entretanto, para o segundo mapa, o método com troca de informações permitiu que os robôs alcançassem o alvo de 10% à 30% mais rápido e com uma distância percorrida 20% a 50% menor.

O que pode-se inferir é que quando os obstáculos são esparsos e passíveis de serem contornados, a troca de informações não faz muita diferença. Porém, em um ambiente similar a um labirinto, a escolha do caminho a ser tomado, quando se alcança um obstáculo, é crítica. Quanto mais informações sobre o ambiente cada robô possuir, maior será a certeza na tomada dessa decisão e menor será a distância percorrida até o destino.

6. Conclusão

O algoritmo Space D* foca a geração de caminhos por ambientes espaçosos, mantendo o robô afastado de obstáculos dinâmicos e desconhecidos(paredes, obstáculos e outros robos. Dessa forma, o método se apresenta aplicável em situações realistas, onde o uso somente do D* não se mostra uma alternativa viável. Além disso, a troca de informações entre robôs apresenta ganhos no tempo de execução em ambientes onde os obstáculos formam trechos de caminhos sem saída.

Como trabalho futuro, existe a ideia de associar níveis de confiabilidade para os marcadores, bem como estender o algoritmo para o tratamento de obstáculos dinâmicos. Outra possibilidade é incluir no cálculo dos custos dos marcadores a concentração de

Mapa 1

Nº de Robôs	Método com Troca de Informações					
	Tempo (s)		Distância (m)		Velocidade (m/s)	
	Média	DP	Média	DP	Média	DP
10	25.6	7.6	89.74	38.16	3.71	0.82
15	29.7	12.2	88.91	36.66	3.03	0.7
20	33.2	16.8	86.29	35.07	2.69	0.61
	Método sem Troca de Informações					
Nº de Robôs	Tempo (s)		Distância (m)		Velocidade (m/s)	
	Média	DP	Média	DP	Média	DP
10	23.8	6.5	89.23	38.56	3.91	0.88
15	29.6	11.8	90.86	39.9	3.12	0.69
20	35.4	17.9	92.15	41.08	2.74	0.63

Mapa 2

Nº de Robôs	Método com Troca de Informações					
	Tempo (s)		Distância (m)		Velocidade (m/s)	
	Média	DP	Média	DP	Média	DP
10	37.9	18.3	126.26	63.71	3.43	0.74
15	39.4	19.0	113.05	52.12	2.89	0.66
20	40.3	19.7	99.74	44.48	2.41	0.42
	Método sem Troca de Informações					
Nº de Robôs	Tempo (s)		Distância (m)		Velocidade (m/s)	
	Média	DP	Média	DP	Média	DP
10	42.2	22.1	142.65	71.09	3.51	0.83
15	47.9	24.9	145.33	75.39	3.07	0.7
20	51.8	27.2	146.14	78.95	2.82	0.61

Table 2. Comparação entre os métodos com e sem troca de informações, para os mapas da Figura 4.

robôs em áreas do ambiente, de forma que ao trocar informações seja possível determinar os lugares com maiores concentrações de robôs, e, quem sabe, evitá-los. Além disso pode-se usar os vários robôs na realização de tarefas em cooperação.

References

Bicho, A. L. (2009). *Da modelagem de plantas à dinâmica de multidões: um modelo de animação comportamental bio-inspirado.* Doutorado em engenharia elétrica, Universidade Estadual de Campinas - UNICAMP.

Chen, J. and Li, L.-R. (2005). Path planning protocol for collaborative multi-robot systems. In *Proceedings 2005 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 721–726, Espoo, Finland.

Choset, H. and Burdick, J. W. (1994). Sensor-based planning and nonsmooth analysis. In *ICRA*, pages 3034–3041.

Clark, C. M., Rock, S. M., and Latombe, J.-C. (2003). Dynamic networks for motion planning in multi-robot space systems. In *In Intl. Symp. of Artificial Intelligence, Robotics and Automation in Space*.

Ferguson, D. and Stentz, A. T. (2005). Field d*: An interpolation-based path planner and replanner. In *Proceedings of the International Symposium on Robotics Research (ISRR)*.

Gerkey, B. P., Vaughan, R. T., and Howard, A. (2003). The player/stage project: Tools for multi-robot and distributed sensor systems. In *In Proceedings of the 11th International Conference on Advanced Robotics*, pages 317–323.

Jung, J. H., Park, S., and Kim, S.-L. (2010). Multi-robot path finding with wireless multihop communications. *Comm. Mag.*, 48:126–132.

Koenig, S. and Likhachev, M. (2002). Improved fast replanning for robot navigation in unknown terrain. In *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, volume 1, pages 968 – 975 vol.1.

Latombe, J.-C. (1991). *Robot Motion Planning*. Kluwer Academic Publishers, Norwell, MA, USA.

LaValle, S. M. (2006). *Planning Algorithms*. Cambridge University Press, Cambridge, U.K. Also available at http://planning.cs.uiuc.edu/.

Likhachev, M., Ferguson, D. I., Gordon, G. J., Stentz, A., and Thrun, S. (2005). Anytime dynamic a*: An anytime, replanning algorithm. In *Proceedings of the Fifteenth International Conference on Automated Planning and Scheduling (ICAPS 2005), June 5-10 2005, Monterey, California, USA*, pages 262–271. AAAI.

López de Màntaras, R., Amat, J., Esteva, F., López, M., and Sierra, C. (1997). Generation of unknown environment maps by cooperative low-cost robots. In *Proceedings of the first international conference on Autonomous agents*, AGENTS '97, pages 164–169, New York, NY, USA. ACM.

Okada, T., Beuran, R., Nakata, J., Tan, Y., and Shinoda, Y. (2007). Collaborative motion planning of autonomous robots. In *Proceedings of the 2007 International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 328–335, Washington, DC, USA. IEEE Computer Society.

Pallottino, L., Scordio, V. G., Frazzoli, E., and Bicchi, A. (2007). Decentralized cooperative policy for conflict resolution in multi-vehicle systems. *IEEE Trans. on Robotics*, 23(6):1170–1183.

Pereira, G. A. S., Das, A. K., Kumar, V., and Campos, M. F. M. (2003). Decentralized motion planning for multiple robots subject to sensing and communication constraints. In *in Proceedings of the Second MultiRobot Systems Workshop*, pages 267–278. Kluwer Academic Press.

Rekleitis, I. M., Dudek, G., and Milios, E. E. (1997). Multi-robot exploration of an unknown environment, efficiently reducing the odometry error. In *Proceedings of the Fifteenth international joint conference on Artifical intelligence - Volume 2*, pages 1340–1345, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Runions, A., Fuhrer, M., Lane, B., Federl, P., Rolland-Lagan, A.-G., and Prusinkiewicz, P. (2005). Modeling and visualization of leaf venation patterns. *ACM Trans. Graph.*, 24(3):702–711.

Stentz, A. T. (1995). The focussed d* algorithm for real-time replanning. In *Proceedings of the International Joint Conference on Artificial Intelligence*.

van den Berg, J., Snoeyink, J., Lin, M., and Manocha, D. (2009). Centralized path planning for multiple robots: Optimal decoupling into sequential plans. In *Proceedings of Robotics: Science and Systems*, Seattle, USA.

van den Berg, J. P., Ferguson, D., and Kuffner, J. (2006). Anytime path planning and replanning in dynamic environments. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, May 15-19, 2006, Orlando, Florida, USA*, pages 2366–2371.

Aceleração do Aprendizado por Reforço em Problemas com Múltiplos Objetivos

Helen C. de Mattos Senefonte[1], Reinaldo A. C. Bianchi[2], Carlos H. C. Ribeiro[3]

[1]Universidade Estadual de Londrina (UEL)
Londrina, PR – Brazil

[2]Centro Universitário da FEI (FEI)
São Bernardo do Campo, SP – Brazil

[3]Instituto Tecnológico de Aeronáutica (ITA)
São José dos Campos, SP – Brazil

helen@uel.br, carlos@ita.br, rbianchi@fei.edu.br

Abstract. *We present a new variation of an action-selection approach to be used in Multi Objective Reinforcement Learning problems, in order to speed up the learning process. To do so, we propose to use the action selection method of the Heuristically Accelerated Q-learning algorithm (HAQL), in which a heuristic function $\mathcal{H}$ is used to influence the choice of actions during the learning, in traditional multi objective RL algorithms. This proposal was evaluated using a traditional research task: the predator-prey problem, and the results indicate that the use of heuristics is able to provide the desired learning acceleration.*

Resumo. *Apresentamos neste artigo uma nova variação de uma abordagem para a seleção de ações em problemas de Aprendizado por Reforço com Múltiplos Objetivos, visando à diminuição do tempo necessário para a realização do aprendizado. Para tanto, é proposta a utilização do método de seleção de ações do algoritmo Q-learning Acelerado por Heurísticas (HAQL), no qual uma função heurística $\mathcal{H}$ é usada para influenciar o agente na escolha de suas ações durante o aprendizado, em algoritmos tradicionais de RL multiobjetivos. A proposta foi avaliada utilizando um domínio tradicional em pesquisas com múltiplos objetivos: o problema Presa-Predador, e os resultados obtidos indicam que o uso de heurísticas é capaz de proporcionar a aceleração do aprendizado desejada.*

1. Introdução

Para permitir a utilização de algoritmos de Aprendizado por Reforço (*Reinforcement Learning* – RL) [Sutton e Barto, 1998] em problemas onde existem múltiplos objetivos (*Multiple Objective Reinforcement Learning* – MORL), por vezes concorrentes, diversos autores têm proposto soluções modularizadas para as tarefas, decompondo o problema original em sub-tarefas, preferencialmente independentes entre si, cada uma representada por um Processo Decisório de Markov (*Markov Decision Process* – MDP). Essa divisão traz consigo uma questão a ser tratada: como selecionar as ações a serem executadas, considerando que cada MDP propõe uma ação distinta a cada instante de tempo para solucionar cada sub-tarefa.

Estudos recentes apresentam diferentes técnicas para a solução deste problema: Barrett e Narayanan [2008] propõe um método que garante encontrar a política ótima para qualquer função de valor linear; já outra abordagem é apresentada por Vamplew et al. [2008] que projeta problemas MORL como um conjunto de soluções baseada no princípio de Pareto, e mostra que o uso de reforços escalares – ao invés de um reforço vetorial em que cada elemento do vetor é o reforço de cada objetivo do problema MORL – tem limitações fundamentais, quando usada para encontrar políticas ótimas de Pareto.

Este trabalho propõe a utilização do método de seleção de ações do algoritmo Q-learning Acelerado por Heurísticas (HAQL) [Bianchi et al., 2008], uma variante do Q-learning [Watkins e Dayan, 1992] que incorpora a utilização de heurísticas aproveitando indícios existentes no processo de aprendizagem da estrutura do domínio, em problemas de RL com múltiplos objetivos. A análise desta proposta é feita para problemas em que um único agente deve aprender simultaneamente várias sub-tarefas independentes, utilizando um domínio tradicional em pesquisas com múltiplos objetivos: o problema Presa-Predador.

O artigo está estruturado da seguinte maneira: na Seção 2 são apresentados definições básicas de RL; a Seção 3 apresenta algoritmos utilizados no problema de seleção de ações em MORL. A seção 4 descreve o uso de heurísticas no aprendizado por reforço. A Seção 5 apresenta algumas características do ambiente experimental, juntamente com a descrição da heurística utilizada. Resultados obtidos na etapa experimental são apresentados na Seção 6. A Seção 7 apresenta as conclusões do artigo.

2. MDPs e o Aprendizado por Reforço

O Aprendizado por Reforço (RL) é uma técnica muito atraente para solucionar uma variedade de problemas de controle e planejamento quando não existem modelos disponíveis *a priori*, já que seus algoritmos têm a convergência para uma situação de equilíbrio garantida [Sutton e Barto, 1998], além de permitirem o aprendizado de estratégias de controle adequadas. No RL o aprendizado acontece por meio da interação direta entre o agente e o ambiente. Infelizmente, a convergência dos algoritmos de RL só pode ser atingida após uma extensiva exploração do espaço de estados-ações, que é geralmente demorada.

No RL, um agente aprendiz interage com o ambiente em intervalos de tempos discretos em um ciclo de percepção-ação. A maneira mais tradicional para formalizar o RL é utilizando o conceito de Processo Markoviano de Decisão (*Markov Decision Process* – MDP), que pode ser definido formalmente [Mitchell, 1997] pela quádrupla $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$, onde: $\mathcal{S}$ é um conjunto finito de estados do ambiente; $\mathcal{A}$ é um conjunto finito de ações que o agente pode realizar; $\mathcal{T} : \mathcal{S} \times \mathcal{A} \mapsto P_{s,a}(\mathcal{S})$ é a função de transição de estados ($P_{s,a}(\mathcal{S})$ é uma distribuição de probabilidades sobre o conjunto de estados $\mathcal{S}$, e portanto $\mathcal{T}(s_t, a_t, s_{t+1})$ define a probabilidade de realizar a transição do estado s_t para o estado s_{t+1} quando se executa a ação a_t); e $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \Re$ é a função de reforço ($\Re$ denota o conjunto dos números reais).

Assim, no ciclo percepção-ação, o agente aprendiz observa, a cada passo de iteração, o estado corrente s_t do ambiente e escolhe a ação a_t para realizar. Ao executar esta ação a_t – que altera o estado do ambiente – o agente recebe um sinal escalar de reforço (penalização ou recompensa) $r(s_t, a_t)$. Resolver um MDP consiste em computar a política $\pi : \mathcal{S} \mapsto \mathcal{A}$ que maximiza (ou minimiza) o valor esperado de um funcional

dos reforços, geralmente uma soma ponderada destes ao longo do tempo. Na formulação mais comum, o objetivo do agente de aprendizado é encontrar uma política de ações estacionária $a_t^* = \pi^*(x_t)$ que maximize a função de valor esperado da Equação 1 para cada estado x_0:

$$V^{\pi}(x_0) = \lim_{M \to \infty} \mathrm{E}\left[\sum_{t \to 0}^{M} \gamma^t r(x_t, \pi(x_t))\right] \tag{1}$$

O fator de desconto $0 < \gamma < 1$ define se os reforços recentes são mais importantes ou não. A política ótima estacionária é dada por $\pi^*(x) = \arg\max_{\pi} V^{\pi}(x), \forall x$.

Provavelmente mais popular algoritmo de RL, o algoritmo *Q–Learning* foi proposto como uma maneira de aprender iterativamente a política ótima π^* quando o modelo do sistema não é conhecido [Watkins e Dayan, 1992]. O algoritmo *Q–Learning* propõe que o agente aprenda uma função Q de soma de recompensas esperadas com descontos, conhecida como função valor–ação. O algoritmo opera aproximando iterativamente $\hat{Q}$ – a estimativa de $Q^*(s, a)$ no instante t – da seguinte maneira: estando o agente num determinado estado x, ele escolhe uma ação a disponível neste estado usando uma dada estratégia de exploração, executa essa ação a, recebe o respectivo reforço r, e visita o estado resultante y. No estado y é feita uma busca, entre as ações disponíveis, para encontrar a ação b que produza o maior valor esperado. Assim, o valor do par estado-ação é atualizado, segundo a Equação 2 [Watkins e Dayan, 1992]:

$$Q(x,a) \;=\; Q(x,a) \;+\; \alpha \;\left[r \;+\; \gamma \;\max_{b} Q(y,b) \;-\; Q(x,a)\right]. \tag{2}$$

O parâmetro α é o fator de aprendizado ($0 < \alpha \leq 1$), frequentemente definido pela Equação 3:

$$\alpha = \frac{1}{1 + visitas(x,a)} \tag{3}$$

Sendo $visitas(x, a)$ o número de visitas já realizadas ao estado x, com ação a escolhida e realizada.

Uma propriedade importante deste algoritmo é que as ações usadas durante o processo iterativo de aproximação da função Q podem ser escolhidas usando qualquer estratégia de exploração (ou explotação). Uma estratégia para a escolha das ações bastante utilizada em implementações do *Q–Learning* é a exploração aleatória $\epsilon - Greedy$, na qual o agente executa a ação com o maior valor de Q com probabilidade $1 - \epsilon$ e escolhe uma ação aleatória com probabilidade ϵ. Neste caso, a transição de estados é dada pela seguinte equação:

$$\pi(s_t) = \begin{cases} a_{random} & \text{se } q \leq \epsilon, \\ \arg\max_{a_t} \hat{Q}_t(s_t, a_t) & \text{caso contrário,} \end{cases} \tag{4}$$

onde: q é um valor escolhido de maneira aleatória com distribuição de probabilidade uniforme sobre [0,1] e ϵ ($0 \leq \epsilon \leq 1$) é o parâmetro que define a taxa de exploração: quanto menor o valor de ϵ, menor a probabilidade de se fazer uma escolha aleatória e a_{random} é uma ação aleatória selecionada entre as ações possíveis de serem executadas no estado s_t.

3. Métodos de Seleção de Ações

As técnicas para a seleção de que ação executar quando o aprendizado do agente envolve uma única tarefa é um tema bem estudado, com resultados teóricos e técnicas conhecidas [Sutton e Barto, 1998]. Em contrapartida, no aprendizado por reforço com múltiplos objetivos estes métodos possuem uma alta complexidade e ainda apresentam desafios.

Partindo da idéia de que nos sistemas multiobjetivos cada MDP corresponde apenas a uma determinada sub-tarefa, surge a necessidade de desenvolver um mecanismo que considere o conhecimento obtido em cada um dos MDPs para encontrar uma política ótima (ou pelo menos adequada) conjunta, durante o processo de aprendizagem.

Os métodos de seleção de ações são mecanismos que têm como objetivo escolher uma ação (idealmente ótima para uma dada função de custo definida *ex-ante*) entre as ações possíveis de ser executadas a cada instante, num estado x, em cada um dos MDPs. Neste artigo foram considerados dois métodos propostos por Humphrys [1997]: "Maximiza a Maior Satisfação" e "*W(x,a)-learning*".

Os métodos de seleção de ações são baseados em uma competição entre os agentes, utilizando-se uma variável W que avalia a importância das possíveis ações. Considera-se vitoriosa a ação que tiver o maior valor W, definido de acordo com o método de seleção considerado e a partir dos valores $Q(x, a)$ obtidos independentemente por cada agente, atuando em seu MDP.

3.1. Maximiza a Maior Satisfação

Por conveniência notacional, denotemos a_i como sendo a ação preferida para o agente que opera no MDP i (ou seja, $a_i = \arg\max_{a \in A_i} Q_i(x, a)$, onde A_i é o conjunto de ações disponíveis para o MDP i). O método Maximiza a Maior Satisfação é baseado na maior satisfação individual, e ignora eventuais penalizações que a escolha de uma ação possa causar aos demais MDPs. O valor W_i para um agente operando no seu MDP i é obtido como $W_i(x) = Q_i(x, a_i) = \max_{a \in A_i} Q_i(x, a)$, e o objetivo é escolher a ação que maximiza este valor entre todos os agentes, de acordo com a Equação 5:

$$a_k = \arg\max_{a \in A'} W_i(x) = \arg\max_{a \in A'} [\max_{a \in A_i} Q_i(x, a)] \tag{5}$$

Onde $A' = A_1 \cup A_2 \cup \ldots \cup A_n$ é o conjunto de ações disponíveis no estado x, n é o número de MDPs e k identifica o MDP cuja ação preferida é a_k e que venceu a competição pela escolha da ação no estado x, nesse instante. Note que a ação a_k pode produzir baixos valores $Q_i(x, a_k)$ para MDPs não vitoriosos.

3.2. W(x,a)-learning

Este método considera uma atualização dos valores W e uma expressão mais fidedigna destes como indicadores de preferência, pois considera-se não apenas a satisfação individual: o valor W_i representa a diferença entre o valor que seria recebido se o MDP i fosse o vencedor e o valor recebido pela execução da ação vencedora. Sousa [2006] propõe, de modo a se evitarem problemas causados pela estocasticidade da atualização dos valores Q em MDPs, a armazenagem de valores $W_i(x, a)$ e não $W_i(x)$, tal como originalmente proposto por Humphrys [1997]. A atualização é feita de acordo com a Equação 6:

$$W_i(x, a) = (1 - \alpha)\, W_i(x, a) + \alpha\, [Q_i(x, a_i) - (r_i + \gamma \max_{a \in A_i} Q_i(y, a))] \tag{6}$$

e a ação é então escolhida a cada iteração como $a_k = \arg\max_{a \in A'} W_i(x, a)$.

4. Q-Learning Acelerado por Heurísticas (HAQL)

Por ser o mais popular algoritmo de RL e possuir uma grande quantidade de dados na literatura para a realização de uma avaliação comparativa, o primeiro algoritmo que utiliza heurísticas a ser implementado foi uma extensão do algoritmo Q-Learning, denominado Q-Learning Acelerado por Heurísticas -– HAQL [Bianchi et al., 2008]. Este algoritmo faz uso de uma função heurística como uma maneira de usar o conhecimento sobre a política de um sistema para acelerar o aprendizado. Este conhecimento pode ser derivado diretamente do domínio ou construído a partir de indícios existentes no próprio processo de aprendizagem.

No HAQL a função heurística é usada somente na regra de transição de estados que escolhe a ação a a ser tomada no estado x. A estratégia para a escolha das ações utilizada no HAQL é a exploração aleatória $\epsilon - Greedy$, na qual, além da estimativa das funções de probabilidade de transição $\mathcal{T}$ e da recompensa $\mathcal{R}$, a função $\mathcal{H}$ também é considerada. A regra de transição de estados aqui proposta é dada pela Equação 7:

$$\pi(x) = \begin{cases} a_{random} & \text{se } q \leq p; \\ \arg\max_a \left[Q(x,a) + \xi \, H(x,a)^\beta \right] & \text{caso contrário,} \end{cases} \tag{7}$$

onde:

- q é um valor escolhido de maneira aleatória com distribuição de probabilidade uniforme sobre $[0, 1]$ e p $(0 \leq p \leq 1)$ é o parâmetro que define a taxa de exploração: quanto menor seu valor, menor a probabilidade de se fazer uma escolha aleatória,
- a_{random} é uma ação selecionada de maneira aleatória entre as ações executáveis no estado x,
- ξ e β são variáveis reais usadas para ponderar a influência da função heurística, e
- $H(x, a)$ é definido pela Equação 8:

$$H(x,a) = \begin{cases} \max_a \left[Q(x,a) \right] - Q(x,a) + \eta & \text{se } a = \pi^H(x); \\ 0 & \text{caso contrário.} \end{cases} \tag{8}$$

onde η é um valor pequeno e $\pi^H(x)$ é a heurística obtida usando um método apropriado.

Uma heurística pode ser definida como uma técnica que melhora, no caso médio, a eficiência na solução de um problema [Bianchi et al., 2008]. Funções heurísticas são a forma mais comum de se aplicar um conhecimento adicional do problema a um algoritmo de busca [Russell e Norvig, 1995].

A hipótese central deste trabalho é que a aceleração do aprendizado por reforço em problemas com múltiplos objetivos pode ser alcançada a partir do uso de heurísticas para escolha de ações nos métodos apresentados na seção anterior. Para tanto, a próxima seção apresenta como se pode fazer uso de heurísticas nos métodos de seleção de ações "Maximiza a Maior Satisfação" e "*W(x,a)-learning*".

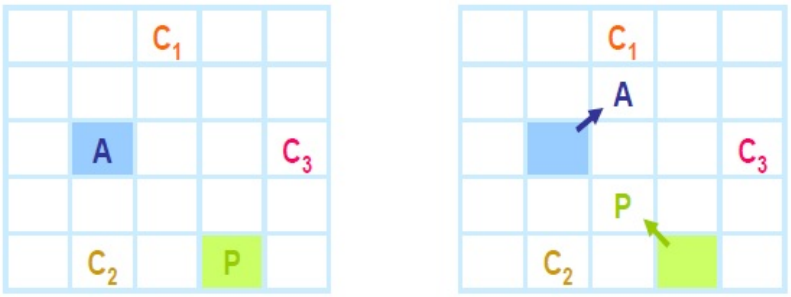


Figura 1. Ambiente padrão de testes.

5. Uso de heurísticas em métodos de seleção de ações

Para fazer uso de heurísticas para escolha de ações nos métodos apresentados na Seção 3, é necessário introduzir a função heurística nas equações de seleção de ação. Assim, propomos que as funções de seleção de ações usadas passem a ser, em analogia com a modificação introduzida no algoritmo HAQL para incluir a função heurística:

- Para o método "Maximiza a Maior Satisfação":

$$a_k = \arg\max_{a \in A'} \{ \max_{a \in A_i} \left[Q_i(x,a) + \xi H(x,a)^\beta \right] \} \tag{9}$$

- E para o método "*W(x,a)-learning*":

$$a_k = \arg\max_{a \in A'} \left[W_i(x,a) + \xi \, H(x,a)^\beta \right] \tag{10}$$

Definir o ambiente onde os testes foram realizados facilita a descrição de como a heurística $H(x,a)$ é computada. Apresentamos portanto o domínio considerado, a seguir.

5.1. O problema da Presa-Predador

O domínio para testes implementado é baseado no conhecido problema da Presa-Predador, introduzido por Singh e Cohn [1998] e utilizado por autores (Sprague e Ballard [2003]; Sousa [2006]), que realizaram testes com diversos métodos de seleção de ações.

No problema da Presa-Predador, um agente A e um predador P se movem num mundo representado por uma matriz 5×5. O ambiente possui três células de reforço positivo C1, C2 e C3 estáticas, conforme apresentado na Figura 1. O agente move-se em 8 direções possíveis (N, NE, E, SE, S, SO, O e NO), e tem por objetivo procurar as células de reforço positivo presentes no ambiente, e ao mesmo tempo, evitar o predador. O agente encontra um reforço positivo ou é apanhado pelo predador quando partilha a mesma posição da matriz. Quando o agente executa uma ação que o leva contra os limites do ambiente, permanece na mesma posição.

Foram desenvolvidos três ambientes experimentais baseados nesse ambiente, denominados (em concordância com Sousa [2006]) R1110, R2340 e R2343. Os nomes referem-se aos reforços recebidos pelo agente quando encontra as células de reforço positivo e o predador, de acordo com a Tabela 1.

O desempenho do agente é medido através do reforço médio $R_m = \frac{1}{100} . \sum_{t=1}^{100} r_t$ recebido ao longo de 100 passos simulados, correspondendo a um episódio. Para geração dos resultados de desempenho foram executados testes a intervalos regulares, intercalando períodos de treinos. As diferenças entre treino e teste são:

Ambientes	Célula 1	Célula 2	Célula 3	Predador
R1110	+1	+1	+1	0
R2340	+2	+3	+4	0
R2343	+2	+3	+4	-3

Tabela 1. Tabela de reforços para ambientes experimentais testados.

1. Treinos: há exploração para aumentar o grau de conhecimento de cada MDP do ambiente; período em que ocorre a extração de estrutura para a construção da heurística e atualização dos valores Q.
2. Testes: Não há exploração e sim a execução da melhor política encontrada até então; período em que são obtidos os resultados dos experimentos, ou seja, são gerados os valores de R_m.

5.2. Uso de métodos "Heurísticas a partir de X"

A função heurística H pode ser definida de maneira *ad hoc* ou pode ser extraída durante o processo de aprendizado. Para a extração automática iterativa das heurísticas, Bianchi et al. [2008] propõe uma classe de métodos chamados "Heurísticas a partir de X", que funcionam em dois estágios: Extração de Estrutura, que retira da estimativa da função de valor informações sobre a estrutura do domínio; e Composição de Heurística, que encontra a heurística para a política usando as informações extraídas no estágio anterior.

5.2.1. Extração de Estrutura

Segundo Bianchi et al. [2008], pode-se extrair uma heurística útil a partir do momento em que o agente está recebendo reforços do ambiente. A técnica de extração utilizada para os métodos de seleção de ações baseia-se na seguinte descrição.

O ambiente experimental desse estudo possui algumas características importantes: assumindo que o algoritmo *Q-Learning* já quase convergiu, o Predador só alcança o Agente se este escolher uma ação que forme um par estado-ação de perigo. Tal situação é caracterizada quando o Agente tenta executar um movimento inválido (para fora do ambiente), aumentando suas chances de ser capturado, pois continuará na mesma posição e o Predador caminhará um passo em sua direção.

Considerando o ambiente ilustrado na Figura 2, o número total de pares estado-ação é ($=$ Posições possíveis do Agente $\times$ Posições possíveis do Predador $\times$ Ações $= 25 \times 25 \times 8 = 5.000$). A análise individual de cada posição do ambiente experimental permite a identificação das posições da grade que correspondem a estados com características distintivas, ainda conforme a Figura 2. Considerando:

- Identificador: identificação da posição que está sendo analisada e calculada na grade na Figura 2,
- QOG: quantidade de ocorrências do respectivo identificador na grade,
- QPP: quantidade de posições em que o predador pode causar perigo caso o agente execute uma ação que o leve aos limites do ambiente,
- QAL: quantidade de ações que levam o agente aos limites do ambiente,

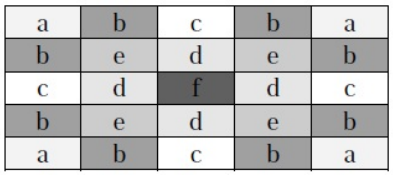


Figura 2. Identificação das posições do ambiente experimental.

- QPQ: quantidade de posições em que o predador pode estar e causar perigo em sua próxima movimentação,

tem-se:

Identificador = QOG x [(QPP x QAL) + QPQ]
a = 4 x [(4 x 5) + 5] = 100
b = 8 x [(6 x 3) + 6] = 192
c = 4 x [(6 x 3) + 9] = 108
d = 4 x [(0 x 0) + 11] = 44
e = 4 x [(0 x 0) + 7] = 28
f = 1 x [(0 x 0) + 16] = 16
Total = 488 pares estado-ação

O agente deverá evitar as ações que levam aos limites do ambiente, e a informação a ser extraída da estrutura durante os treinos é definir portanto quais ações evitam que o agente seja levado a estes limites. Essas ações são anotadas em uma estrutura $A^H(x)$, inicializada com valores 1. Para cada ação que evita que o agente seja levado aos limites, o par estado-ação continua com o valor 1, caso ele tome alguma ação que o leve aos limites, recebe o valor 0.

5.2.2. Composição de Heurística

As heurísticas são extraídas durante o treino, e é nesse período que a estrutura $A^H(x)$ é construída. Com as informações da estrutura $A^H(x)$, a heurística é aplicada durante as fases de teste, influenciando na decisão do agente e permitindo que a aceleração seja observada durante todo o estágio do aprendizado. Na aplicação da heurística, $A^H(x)$ é analisada, e se o valor respectivo do par estado-ação for igual a 1, indica que a ação não é perigosa. Então, $H(x, a)$ é atualizada de acordo com a Equação 11.

$$H(x,a) = \begin{cases} \max_a [Q(x,a)] - Q(x,a) + \eta & \text{se } A^H(x) = 1; \\ 0 & \text{caso contrário.} \end{cases} \quad (11)$$

As ações não-perigosas são selecionadas seguindo as regras dos métodos de seleção de ações.

6. Experimentos

Para gerar os resultados foram executados testes sobre 1.000 episódios intercalados com períodos de treinos de 10.000 episódios. Os dados demonstrados nos gráficos são gerados nos períodos de testes. No eixo x os valores são multiplicados por 10^5, por exemplo,

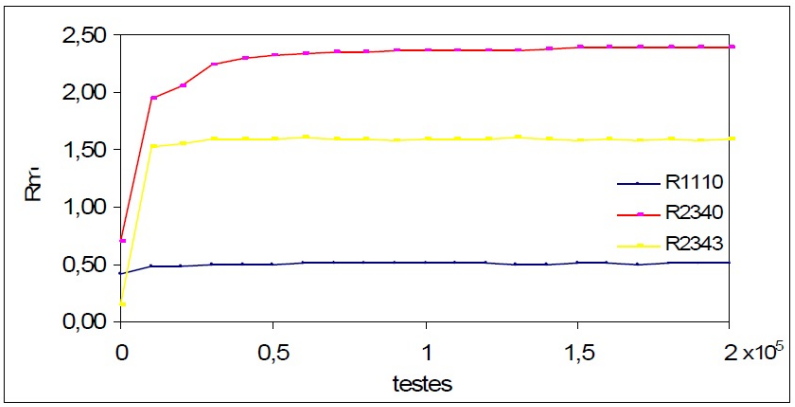


Figura 3. Maximiza Maior Satisfação.

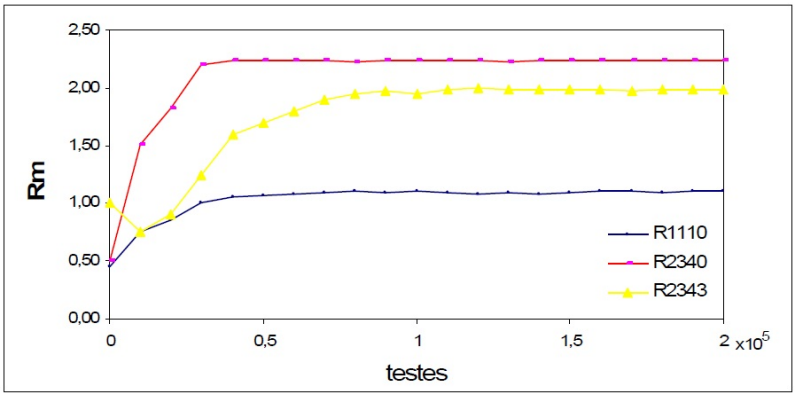


Figura 4. *W(x,a)-Learning.*

$1,5$ equivale a 150.000 testes e o eixo y apresenta os valores do reforço médio R_m. Os parâmetros utilizados para a atualização dos valores-Q foram α (conforme Equação 3), $\gamma = 0,9$ e ε variando linearmente de $0,4$ a 0 durante a primeira metade dos episódios de cada período de treino. Os valores de ξ e β foram definidos como sendo iguais a 1.

A Figura 3 apresenta os testes de desempenho com o método Maximiza Maior Satisfação nos ambientes R1110, R2340 e R2343. O desempenho nos testes com o método *W(x,a)-Learning* é visualizado na Figura 4.

A Figura 5 apresenta a comparação do desempenho do método Maximiza a Maior Satisfação com e sem heurística, no ambiente R2340. Pode-se notar que o reforço médio gerado é sensivelmente maior com a utilização de heurística. A comparação entre os testes realizados com o método *W(x,a)-Learning* com e sem heurística no ambiente R2340 é apresentada na Figura 6. Com a utilização da heurística, também foram obtidos reforços médios maiores. Tal resultado se verificou também em testes comparativos realizados nos outros dois ambientes (R1110 e R2343), com melhoria similar de desempenho causado pelo uso da heurística na seleção de ações.

O aumento nos valores de reforço médio obtidos com a utilização de heurística comprova que a heurística é adequada e que as informações retiradas do domínio são

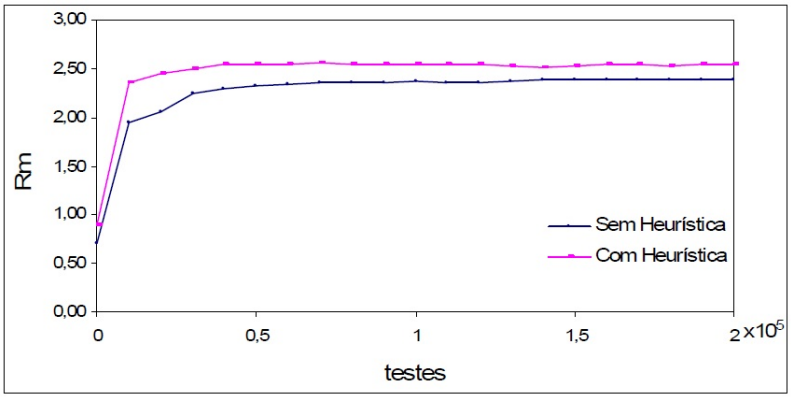


Figura 5. Maximiza a Maior Satisfação, com e sem heurística.

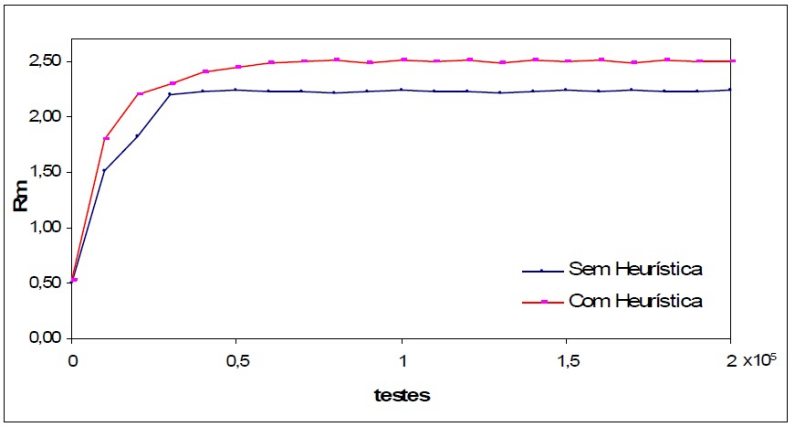


Figura 6. *W(x,a)-Learning* com e sem heurística.

suficientes para acelerar o aprendizado. O teste estatístico T de Student [Spiegel, 1998] foi utilizado nos resultados obtidos, e comprovou que os desempenhos obtidos com e sem heurística diferem com nível de confiança maior que 90%.

7. Conclusões

Esse estudo propôs a aceleração do aprendizado por reforço em sistemas com múltiplos objetivos, aplicando uma heurística para seleção de ações em dois métodos de seleção de ações conhecidos. A heurística aplicada explora características retiradas do ambiente durante o processo de aprendizado, e os resultados mostraram que os métodos que a utilizaram apresentaram melhores resultados que os métodos originais, confirmando a afirmação em Bianchi et al. [2008] que, mesmo com a utilização de heurísticas adequadas simples, o aprendizado por reforço pode ser acelerado.

Deve-se porém considerar que a heurística considerada nesse estudo explorou um problema que abrange apenas os pares estado-ação de perigo, que equivalem a 10% dos pares estado-ação possíveis do ambiente considerado. Isso sugere que resultados ainda melhores podem ser obtidos com a análise e aplicação de heurísticas mais sofisticadas, cujo estudo é sugerido como trabalho futuro.

Referências

Barrett, L. e Narayanan, S. (2008). Learning all optimal policies with multiple criteria. In *Proceedings of the 25th International Conference on Machine learning*, ICML '08, pages 41–47, New York, NY, USA. ACM.

Bianchi, R. A. C., Ribeiro, C. H. C. e Costa, A. H. R. (2008). Accelerating autonomous learning by using heuristic selection of actions. *Journal of Heuristics*, 14(2):135–168.

Humphrys, M. (1997). *Action selection methods using reinforcement learning*. PhD thesis, University of Cambridge, Trinity Hall, Cambridge, UK.

Mitchell, T. (1997). *Machine Learning*. McGraw Hill, New York.

Russell, S. e Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Prentice Hall, Upper Saddle River, NJ.

Singh, S. e Cohn, D. (1998). How to dynamically merge markov decision processes. In *Proceedings of the 1997 conference on Advances in neural information processing systems 10*, NIPS '97, pages 1057–1063, Cambridge, MA, USA. MIT Press.

Sousa, C. A. O. (2006). *Aprendizagem por Reforço de Sistemas com Múltiplos Objectivos: o Problema da Selecção de Acções*. Tese de Doutorado, Universidade Técnica de Lisboa, Instituto Superior Técnico, Lisboa.

Spiegel, M. R. (1998). *Statistics*. McGraw-Hill.

Sprague, N. e Ballard, D. (2003). Multiple-goal reinforcement learning with modular sarsa(o). In *Proceedings of the 18th international joint conference on Artificial intelligence*, pages 1445–1447, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Sutton, R. e Barto, A. (1998). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA.

Vamplew, P., Yearwood, J., Dazeley, R. e Berry, A. (2008). On the limitations of scalarisation for multi-objective reinforcement learning of pareto fronts. In *Proceedings of the 21st Australasian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence*, AI '08, pages 372–378, Berlin, Heidelberg. Springer-Verlag.

Watkins, C. e Dayan, P. (1992). Q-learning. *Machine Learning*, 8:279–292.

Jogos Markovianos Alternados com Probabilidades Imprecisas dadas por Conjuntos Credais

Fábio de O. Franco[1], Leliane N. de Barros[1], Karina V. Delgado[2], Fábio G. Cozman[3]

[1]Instituto de Matemática e Estatística – Universidade de São Paulo (USP)
Caixa Postal 66281 – 05508-090 – São Paulo – SP – Brazil

[2]Escola de Artes, Ciências e Humanidades – USP – São Paulo – SP – Brazil

[3]Escola Politécnica – Universidade de São Paulo – São Paulo – SP – Brazil

{ffranco,leliane}@ime.usp.br, {kvd,fgcozman}@usp.br

Abstract. *Alternate Markov Games (AMGs) are used to represent sequences of decisions with probabilistic effects and are particularly useful to model multi-agent interactions. However, in practice it is often impossible to obtain a precise value for all transition probabilities. A variant of AMGs has been recently proposed, where imprecision in probability values is represented by probability intervals. In this paper we present a more general variant of AMGs that allows for probabilistic imprecision to be expressed by credal sets (sets of probability distributions). We also define an algorithm that achieves the equilibrium value for the proposed model.*

Resumo. *Jogos Markovianos Alternados (AMGs) são usados para representar sequencias de decisões com efeitos probabilísticos e são particularmente úteis para modelar interações por múltiplos agentes. No entanto, na prática é frequentemente impossível obter um valor preciso para cada probabilidade de transição. Uma variante dos AMGs foi proposta recentemente, onde imprecisão em valores de probabilidades é representada por intervalos de probabilidade. Neste artigo apresentamos uma variante mais geral de AMGs que permite que a imprecisão seja expressa por conjuntos credais (conjuntos de distribuições de probabilidades). Também definimos um algoritmo que encontra o valor de equilíbrio para o modelo proposto.*

1. Introdução

A teoria dos jogos é amplamente utilizada em uma variedade de áreas, da economia à política, da ciência da computação ao entretenimento [Ferguson 2011]. Na pesquisa em técnicas para inteligência artificial, a teoria de jogos é uma ferramenta poderosa para analisar e sintetizar sistemas multi-agentes, em que a decisão de um agente pode afetar diretamente o comportamento de todos os outros agentes.

Normalmente, um jogo é caracterizado pelo número de jogadores que interagem entre si e com o ambiente, executando ações sob condições de incerteza e recebendo alguma recompensa (positiva ou negativa). Uma definição ampla e informal para jogos desse tipo é dada por [Russell and Norvig 2003]:

– um conjunto de jogadores que tomam decisões (ações);
– um conjunto de estados em que os jogadores podem se encontrar num determinado estágio do jogo;
– uma coleção de conjuntos de ações definidas, um conjunto para cada jogador;
– uma função de recompensa que dá a utilidade de possíveis combinações de ações.

Um jogador deve selecionar e executar uma *política*: uma política *determinística* especifica uma ação para cada estado, enquanto que uma política *probabilística* especifica uma distribuição de probabilidades sobre ações para cada estado.

Uma *solução* para um jogo prescreve a melhor política para cada jogador. Frequentemente procura-se uma solução na qual as políticas estão em equilíbrio de Nash, isto é, nenhum jogador se beneficia em mudar sua própria política se todos os outros jogadores mantém intactas as suas próprias políticas de equilíbrio. Esse equilíbrio pode ser visto como um ótimo local no espaço de políticas.

De acordo com a definição acima, os jogadores estão todos em algum estado conhecido antes da execução de suas ações e, após executá-las, se movem para um outro estado. A transição entre os estados pode ser modelada por uma distribuição de probabilidades. Um jogo markoviano é um jogo em que a função de transição para o próximo estado é determinada apenas pelo estado atual e as ações selecionadas neste estado [Fudenberg and Tirole 1991].

Um tipo importante de jogo markoviano é o *jogo markoviano alternado* (*Alternate Markov Game* – AMG) [Littman 1996], no qual:

1. os jogadores alternam nas escolhas de suas ações (apenas um jogador realiza uma ação em cada estado), e
2. o jogador conhece as jogadas realizadas pelos outros jogadores neste jogo, ou seja, o jogo é de informação perfeita [Parthasarathy and Raghavan 1971].

Permitir que seja inserida incerteza na função de transição probabilística pode trazer mais realismo e flexibilidade a um AMG, bem como a capacidade de compactar jogos muito grandes (jogos envolvendo um número muito grande de estados). Podemos representar as transições entre os estados com intervalos de probabilidade ou conjuntos de distribuições de probabilidade em duas situações: quando houver crenças incompletas e ambíguas sobre as probabilidades de transição ou quando o jogo for muito grande e para resolvê-lo é preciso fazer a agregação de estados. Um modelo que permite representar as crenças com intervalos de probabilidades para lidar com essas duas limitações foi formulado em [Chang 2006]. Os resultados de Chang são voltados para a análise de sensibilidade e soluções aproximadas de jogos markovianos.

O objetivo deste artigo é generalizar o trabalho de Chang de modo que as transições sejam expressas através de conjuntos de distribuições de probabilidade, denominados *conjuntos credais* (*credal sets*) [Cozman 2000]. A idéia vem de trabalhos anteriores sobre Processos de Decisão de Markov com Probabilidades Imprecisas (*Markov Decision Process with Imprecise Probabilities* – MDPIPs) [Delgado et al. 2009]. Um MDPIP é simplesmente uma extensão de um Processo Markoviano de Decisão [Puterman 1994] em que as probabilidades de transição podem ser imprecisas, ou seja, ao invés de uma medida de probabilidade sobre o espaço

de estados, temos um conjunto de medidas de probabilidade [Delgado et al. 2011]. O objetivo é criar um modelo análogo para jogos markovianos alternados, que chamaremos de *jogos markovianos alternados com probabilidades imprecisas dadas por conjuntos credais* (*Alternate Markov Games with Imprecise Probabilities* – AMG-IP).

Este artigo está organizado da seguinte forma. Na Seção 2 fazemos uma breve revisão sobre jogos markovianos alternados de dois jogadores de soma zero com probabilidades precisas. Na Seção 3 discutimos modelos gerais de incerteza para jogos markovianos alternados. Na Seção 4, apresentamos um novo modelo de jogo markoviano alternado com probabilidades imprecisas dadas por conjuntos credais. Na Seção 5, apresentamos uma solução para o modelo proposto e descrevemos um algoritmo. Na Seção 6, discutimos os trabalhos relacionados. Na Seção 7, concluímos e sugerimos possíveis trabalhos futuros.

2. Jogos markovianos

Um jogo markoviano (ou jogo de Markov) [Owen 1982] é definido por um conjunto de k jogadores, um conjunto de estados S, uma coleção de conjuntos de ações $A_1, A_2, ..., A_k$, um para cada jogador, e definições para as funções de transições de estados e funções de recompensas. A transição de estado depende probabilisticamente do estado atual e da ação selecionada por cada jogador, isto é, $p(s'|s, a_1, a_2, ..., a_k)$ é a probabilidade do jogador ir para o estado s', dado que ele está em s e $a_1, a_2, ..., a_k$ são as ações escolhidas pelos jogadores $1, 2, ..., k$, respectivamente. É definido para cada jogador i uma função de recompensa $R_i(s, a_1, a_2, ..., a_k)$, que depende do estado e das ações tomadas por cada jogador. O objetivo de cada jogador é maximizar o valor esperado de suas recompensas descontadas, $E\{\sum_{j=0}^{\infty}\gamma^j r_{t+j}^i\}$, em que r_{t+j}^i é a recompensa recebida pelo jogador i em j passos para o futuro, sendo $\gamma \in]0, 1[$ um fator de desconto. O horizonte T é o número de estágios restantes do jogo. Assim, a recompensa obtida em t estágios para o futuro é descontada por um fator γ^t.

Definição 1 (**Jogo markoviano alternado**). *Um jogo markoviano alternado de dois jogadores de soma zero é definido pela tupla* $\langle S,A_1,A_2,P,R,\gamma\rangle$ *de tal forma que: o conjunto de estados* S *é decomposto em dois conjuntos,* Y *(estados do jogador 1) e* Z *(estados do jogador 2), em que* $Y \cap Z = \emptyset$ *e* $Y \cup Z = S$*. A função de transição de estado é definida por* $P : S \times A_1 \times A_2 \to PD(S)$ *em que* $PD(S)$ *representa o conjunto de distribuições de probabilidades sobre* S*. Chamamos de* $A_i(s)$ *o conjunto finito de ações executáveis em* s *para o jogador* i*. Para cada jogador* i*, existe uma ação* $noop \in A_i$ *que pode ser executada em qualquer estado do jogo e cujo efeito é nulo, ou seja, se todos os agentes executarem suas ações* $noop$ *no estado* s*, o jogo permanecerá no mesmo estado. Os jogadores fazem suas jogadas alternadamente dado que* $P(y'|y, a_1, a_2) = 0$ *para todos os* $y, y' \in Y$*,* $a_1 \in A_1(y)$*,* $a_2 \in A_2(y)$ *e* $P(z'|z, a_1, a_2) = 0$ *para todos os* $z, z' \in Z$*,* $a_1 \in A_1(z), a_2 \in A_2(z)$*. Em cada estado* $s \in (Y \cup Z)$ *somente um jogador tem uma ou mais ações executáveis em* s*. A função de recompensa é dada por* $R : S \times A_1 \times A_2 \to \mathbb{R}$ *que representa as recompensas instantâneas do jogador. Como se trata de um jogo de soma zero, essa mesma função recompensa é positiva para o jogador 1 e negativa para o jogador 2. Assim, chamaremos o jogador 1 de maximizador e o jogador 2 de minimizador (ou de adversário).* $\gamma \in]0, 1[$ *é o fator de desconto, como descrito anteriormente.* □

Chamamos de política estacionária uma política em que a ação especificada para cada estado independe do estágio do jogo. Definimos $\pi : S \to A_1(S)$ como a política estacionária para o maximizador e por Π o conjunto de todas as políticas estacionárias do maximizador. Da mesma forma, Φ é o conjunto de todas as políticas estacionárias para o minimizador.

O valor de um jogo que segue as políticas π e ϕ, isto é, as políticas do maximizador e o minimizador, respectivamente, a partir de um estado inicial $s \in S$, é dado por:

$$V(\pi, \phi)(s) = R(s, \pi(s), \phi(s)) + \gamma \sum_{s' \in S} p(s'|s, \pi(s), \phi(s)) V(\pi, \phi)(s'). \tag{1}$$

Ou seja, o valor de um estado s quando os jogadores 1 e 2 estão seguindo as políticas π e ϕ, respectivamente, é a soma da recompensa atual e o valor esperado dos próximos estados.

A solução de um jogo é frequentemente vista como um equilíbrio (ou *saddle point*), isto é, as políticas π e ϕ são escolhidas de forma que os jogadores não podem obter uma recompensa maior mudando suas ações.

Definição 2 (**Políticas de equilíbrio**). *Um par da políticas $\pi^* \in \Pi$ e $\phi^* \in \Phi$ é um par de políticas (ótimas) de equilíbrio se não existe uma política $\phi \in \Phi$ tal que [Osborne and Rubinstein 1994]:*

$$V(\pi^*, \phi)(s) < V(\pi^*, \phi^*)(s), \quad s \in S, \tag{2}$$

e não há uma política $\pi \in \Pi$ tal que

$$V(\pi^*, \phi^*)(s) < V(\pi, \phi^*)(s), \quad s \in S. \quad \square \tag{3}$$

Seja $Q(s, a_1, a_2)$ o valor da recompensa esperada para cada estado $s \in S$, quando as ações $a_1 \in A_1$ e $a_2 \in A_2$ são selecionadas, dada por [Littman 1996]:

$$Q(s, a_1, a_2) = R(s, a_1, a_2) + \gamma \sum_{s' \in S} p(s'|s, a_1, a_2) V^*(s'), \tag{4}$$

Assim, a solução de um AMG de dois jogadores de soma zero é encontrar um par de políticas ótimas de equilíbrio π^* and ϕ^* que produzem o valor *maximin* em cada estado, ou seja:

$$V(\pi^*, \phi^*)(s) = V^*(s) = \max_{a_1 \in A_1} \min_{a_2 \in A_2} (Q(s, a_1, a_2))$$

$$= \min_{a_2 \in A_2} \max_{a_1 \in A_1} \left(R(s, a_1, a_2) + \gamma \sum_{s' \in S} p(s'|s, a_1, a_2) V^*(s') \right). \tag{5}$$

Note que as políticas ótimas, π^* e ϕ^*, são definidas calculando-se as funções $\arg\max$ e $\arg\min$, respectivamente, ou seja:

$$(\pi^*, \phi^*)(s) = \arg\max_{a_1 \in A_1} \arg\min_{a_2 \in A_2} \left(R(s, a_1, a_2) + \gamma \sum_{s' \in S} p(s'|s, a_1, a_2) V^*(s') \right). \quad (6)$$

Veja ainda que na Equação (5) a ordem dos cálculos de *min* e *max* não altera o valor do jogo. Esta propriedade foi provada por [Shapley 1953], que também provou que a solução *minimax* ou *maximin* converge, isto é, a Equação (5) encontra o valor de equilíbrio de um jogo AMG de dois jogadores de soma zero.

Note que com a formulação da Equação (5) temos uma única função valor de equilíbrio, embora possa haver diferentes pares de políticas que satisfaçam a Equação (6). Como veremos nas próximas seções, no caso do AMG-IP, a função valor de equilíbrio não é única [Chang 2006].

3. Incerteza sobre a função de transição de estado

Como dissemos anteriormente, em geral, é difícil ou impossível obter de forma precisa as probabilidades de transição de estados para jogos markovianos. Existem várias maneiras de abordar tais situações, entre elas:

- Associar um intervalo de probabilidades para a transição do estado s para o estado s' dado por $[\underline{p}(.|s, a_1, a_2), \overline{p}(.|s, a_1, a_2)]$. A Figura 1a ilustra um conjunto de distribuições de probabilidades. Note que uma distribuição de probabilidade é definida escolhendo-se um valor de probabilidade dentro de cada intervalo, tal que a soma da distribuição de probabilidade seja igual a 1.
- Podemos impor restrições sobre os valores das probabilidades de transição a partir do estado s para estado s'. Estas restrições podem ser definidas como intervalos, mas também podem ser mais gerais como na Figura 1b que ilustra um exemplo de transições probabilísticas parametrizadas por p, q e r. Uma distribuição de probabilidades é escolhida de tal forma que satisfaça as restrições sobre p, q e r.

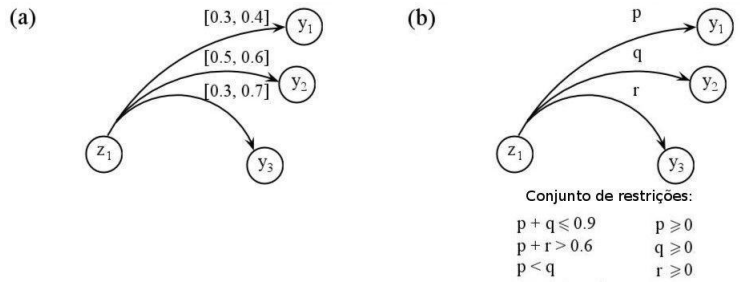


Figura 1. Duas maneiras de representar a imprecisão nas probabilidades de transição em um AMG.

Um *conjunto credal* é definido como um conjunto de distribuições de probabilidades. Por exemplo, a Figura 1a tem como conjunto de probabilidades todas as distribuições limitadas pelos intervalos que definem um conjunto credal. Na Figura 1b, o conjunto de distribuições de probabilidades que satisfazem o conjunto de restrições também define um conjunto credal.

Um conjunto credal que define as distribuições condicionais sobre o próximo estado s', dado um estado s e ações a_1 e a_2 é referido como um conjunto credal de transição (*transition credal set*) e denotado por $K(s'|s, a_1, a_2)$. Assumimos que todos os conjuntos credais são fechados, conexos e convexos [Walley 1991].

Exemplo 1. *Suponha que dado* $S = \{s_0, s_1, s_2\}$ *e as escolhas de ações* $a_1 \in A_1$ *e* $a_2 \in A_2$*, o conjunto de parâmetros de probabilidade* $p(s'|s, a_1, a_2) \in K(s'|s, a_1, a_2)$*, nomeados por* p_0*,* p_1 *e* p_2*, em que*

$$p_0 = p(s_0|s_0, a_1, a_2),$$
$$p_1 = p(s_1|s_0, a_1, a_2),$$
$$p_2 = p(s_2|s_0, a_1, a_2),$$

são definidos pelo seguinte conjunto de restrições:

$$C = \{p_0 \leq 2/3, \quad p_2 \leq 2/3, \quad 2p_0 \geq p_1\}.$$

A região bi-dimensional de todas as medidas de probabilidade que satisfazem C *é mostrada na região cinza da Figura 2, que corresponde ao conjunto credal para o exemplo dado.*

$\Diamond$

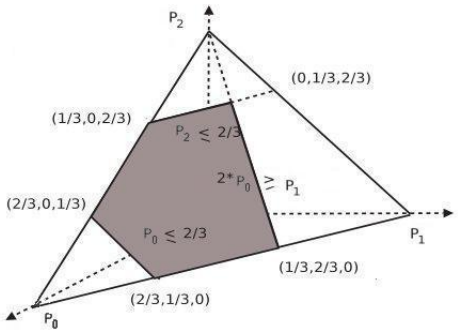


Figura 2. Um exemplo de um conjunto credal definido pelos valores dos parâmetros de probabilidade p_0, p_1 e p_2 que satisfazem o conjunto de restrições C (região cinza).

4. AMG-IP

Num AMG de dois jogadores de soma zero com a imprecisão nas probabilidades dadas por conjuntos credais, o maximizador e o minimizador selecionam e executam suas ações com o conhecimento do estado $s \in S$ em que eles se encontram. O jogo faz uma transição com uma escolha de probabilidades $p(s'|s, a_1, a_2) \in K(s'|s, a_1, a_2)$. Interpretamos essa escolha como uma escolha da Natureza. Como resultado o jogador maximizador recebe uma recompensa de $R(s, a_1, a_2)$ que depende somente do estado em que o jogo está e das escolhas de ações dos jogadores.

Definição 3 (AMG-IP). *Um jogo markoviano alternado de dois jogadores de soma zero com probabilidades imprecisas dadas por conjuntos credais (Alternate Markov Game with Imprecise Probabilities – AMG-IP) é dado pela tupla* $\langle S, A_1, A_2, K, R, \gamma \rangle$*, em que:* S *é um conjunto discreto e finito de estados completamente observáveis do*

jogo. S é composto de dois subconjuntos disjuntos Y e Z; A_1 é o conjunto finito de ações disponíveis para o jogador 1; A_2 é o conjunto finito de ações disponíveis para o jogador 2; $K(s'|s, a_1, a_2)$ define um conjunto de medidas de probabilidade de transição válidas, isto é, um conjunto credal de transição. $K(s'|s, a_1, a_2)$ pode ser implicitamente representado por um conjunto de probabilidades de transição consistente com um conjunto de restrições lineares; $R(s, a_1, a_2) \in \mathbb{R}$ é a função recompensa e; $\gamma \in]0, 1[$ é o fator de desconto.

A Equação (1) pode ser usada para calcular $V(\pi, \phi)$ num AMG-IP considerando que $p(s'|s, \pi(s), \phi(s)) \in K(s'|s, \pi(s), \phi(s))$, e assumindo que K é convexo, conexo e fechado.

Há vários critérios que podem ser usados para definir o valor ótimo de um jogo AMG-IP. Isto é, podemos usar diferentes critérios para escolher quais políticas os jogadores devem usar diante das escolhas da Natureza. Podemos usar, por exemplo, o critério sob a suposição que a Natureza seleciona $p(s'|s, a_1, a_2)$ adversariamente, isto é, minimizando a função valor. Assim, a Equação (5) deve ser modificada da seguinte forma:

$$V^*(s) = \max_{a_1 \in A_1} \min_{a_2 \in A_2} \left(R(s, a_1, a_2) + \gamma \min_{p \in K} \sum_{s' \in S} p(s'|s, a_1, a_2) V^*(s') \right), \tag{7}$$

ou seja, dadas as escolhas $a_1 \in A_1$ e $a_2 \in A_2$ dos dois jogadores, $p \in K$ é selecionado pela Natureza de forma a minimizar a recompensa esperada nos estágios futuros.

Outro possível critério é assumir que a Natureza selecionará $p \in K$ para maximizar a função valor. Assim, a Equação (5) é substituída por:

$$V^*(s) = \max_{a_1 \in A_1} \min_{a_2 \in A_2} \left(R(s, a_1, a_2) + \gamma \max_{p \in K} \sum_{s' \in S} p(s'|s, a_1, a_2) V^*(s') \right). \tag{8}$$

Note que (7) define o limite mínimo para $V^*(s)$ (a melhor escolha no pior caso), que chamaremos de $\underline{V}^*(s)$, enquanto que (8) define o limite superior para $V^*(s)$ (a melhor escolha no melhor caso), que chamaremos de $\overline{V}^*(s)$.

O melhor compromisso entre esses dois critérios de escolha da Natureza é garantir a melhor escolha no pior caso. Uma vez que existe um conjunto Δ de pares de políticas (π^*, ϕ^*) que satisfaz a Equação (7), podemos agora escolher o melhor par de políticas do conjunto Δ usando a Equação (8). Ou seja, sem o risco de diminuir a recompensa esperada, é possível assumir uma Natureza que coopera com o maximizador e escolhe a melhor política em Δ.

Portanto, para encontrar uma solução segundo o critério descrito acima, calculamos primeiro:

$$\underline{V}^*(s) = \max_{a_1 \in A_1} \min_{a_2 \in A_2} \left(R(s, a_1, a_2) + \gamma \min_{p \in K} \sum_{s' \in S} p(s'|s, a_1, a_2) \underline{V}^*(s') \right), \tag{9}$$

e em seguida, utilizamos as ações que alcançam $\underline{V}^*(s)$ para finalmente calcular o valor ótimo (V^*) assumindo a Natureza cooperativa, ou seja:

$$V^*(s) = \max_{a_1 \in A_1[\underline{V}^*](s)} \min_{a_2 \in A_2[\underline{V}^*](s)} \left(R(s,a_1,a_2) + \gamma \max_{p \in K} \sum_{s' \in S} p(s'|s,a_1,a_2) V^*(s') \right), \quad (10)$$

em que $A_1[\underline{V}^*](s)$ e $A_2[\underline{V}^*](s)$ são os conjuntos de ações do jogador 1 e 2, respectivamente, que alcançam o valor ótimo quando a Natureza é adversária do jogador 1.

Em [Chang 2006] foi provado que esse critério de compromisso entre escolhas no pior caso (Natureza min) e escolhas no melhor caso (Natureza max) define a função valor de equilíbrio para um AMG com probabilidades imprecisas dadas por intervalos, ou seja, $p \in [\underline{p}(.|s,a_1,a_2), \overline{p}(.|s,a_1,a_2)]$. É possível provar que isso também vale quando $p \in K(.|s,a_1,a_2)$, ou seja, para imprecisões nas probabilidades definidas por conjuntos credais se assumirmos que eles sejam fechados, conexos e convexos.

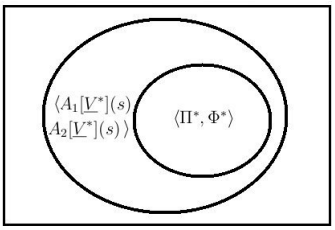


Figura 3. Pares de políticas de equilíbrio em um AMG-IP.

A Figura 3 ilustra o critério usado para encontrar os pares de políticas de equilíbrio num jogo AMG-IP. Os conjuntos de pares de políticas ótimas $\langle \Pi^*, \Phi^* \rangle$ são subconjuntos das políticas formadas pelos conjuntos $A_1[\underline{V}^*](s)$ e $A_2[\underline{V}^*](s)$.

Políticas de equilíbrio para AMG-IPs Como foi dito na seção anterior, dado um AMG-IP e considerando os conjuntos de distribuições de probabilidades $K(.|s,a_1,a_2)$, com $s \in S, a_1 \in A_1, a_2 \in A_2$, fechados, conexos e convexos, o limite inferior $\underline{V}^*(s)$ da função valor pode ser calculado pela Equação 9.

Estendemos os resultados de Chang (2006) que considera intervalos reais da forma $[\underline{p}(.|s,a_1,a_2), \overline{p}(.|s,a_1,a_2)]$ gerando um conjunto credal fechado, conexo e convexo de conjuntos de probabilidades, fazendo essa mesma suposição sobre $K(.|s,a_1,a_2)$ gerado a partir de um conjunto de restrições lineares. Com isso, garantimos encontrar a função valor ótima $\underline{V}^*(s)$ para o caso da Natureza adversária. Como Chang (2006) provou, a condição *maximin* nas escolhas de ações da Equação 9 pode ser invertida para *minimax*, garantindo assim que as políticas ótimas geradas por essa estratégia (que chamaremos de políticas conservadoras) sejam políticas de equilíbrio, independente das escolhas da Natureza.

Finalmente, o par de políticas π^* e ϕ^* que alcançam V^* (Equação 10) também estão em equilíbrio uma vez que elas pertencem ao conjunto de políticas conservadoras, que por sua vez, são política de equilíbrio A escolha de π^* e ϕ^* leva em

consideração a possibilidade da Natureza agir de maneira cooperativa. Isso só pode aumentar a utilidade para ambos os jogadores, mas nunca diminui-la. Portanto, π^* e ϕ^* também são políticas de equilíbrio.

5. Iteração de valor para AMG-IP

Podemos implementar as Equações (9) e (10) usando programação dinâmica, por exemplo, generalizando o algoritmo de Iteração de valor. O algoritmo Value-Iteration-AMG-IP (Algoritmo 1) devolve a função valor ótima para um AMG-IP e tem como parâmetros de entrada o AMG-IP e o número N de iterações desejado. Para isso, Value-Iteration-AMG-IP faz chamadas a dois outros algoritmos, SolveOne (Algoritmo 2) e SolveTwo (Algoritmo 3).

O algoritmo SolveOne implementa a Equação (9) devolvendo os conjuntos $A_1[\underline{V}^*](s)$ e $A_2[\underline{V}^*](s)$. De forma semelhante, o algoritmo SolveTwo implementa a Equação (10) e devolve V^*. Estes dois algoritmos são semelhantes ao algoritmo de iteração por valor de AMGs [Littman 1996], porém o cálculo da recompensa total descontada em cada estado $s \in S$ é realizado por Regress (Algoritmo 4).

A seqüencia de execução do algoritmo Value-Iteration-AMG-IP é definida da seguinte forma:

1. O algoritmo Value-Iteration-AMG-IP chama o algoritmo SolveOne, passando os mesmos parâmetros recebidos na entrada.
2. O algoritmo SolveOne, em cada iteração, encontra para todos os estados do jogo os pares de políticas $\pi^*(s)$ e $\phi^*(s)$ que alcançam $\underline{V}^*$. Para isso o algoritmo SolveOne chama o algoritmo Regress afim de calcular:
$$Q(s, a_1, a_2) = R(s, a_1, a_2) + \gamma \min_{p \in K} \sum_{s' \in S} p(s'|s, a_1, a_2)\underline{V}^*(s').$$
Note que Regress precisa chamar um otimizador para poder calcular Q. Após ocorrerem as N iterações, SolveOne devolve os conjuntos $A_1[\underline{V}^*]$ e $A_2[\underline{V}^*]$ para Value-Iteration-AMG-IP.
3. O algoritmo SolveTwo é chamado para calcular $V^*(s)$ passando como parâmetros o AMG-IP e o número N de iterações, porém, ao invés de A_1 e A_2 são passados os novos conjuntos de ações $A_1[\underline{V}^*]$ e $A_2[\underline{V}^*]$.
4. O algoritmo SolveTwo, em cada iteração, encontra para todos os estados os pares de políticas $\pi(s)^*$ e $\phi(s)^*$ que alcançam V^*. Para isso o algoritmo SolveTwo chama o algoritmo Regress afim de calcular:
$$Q(s, a_1, a_2) = R(s, a_1, a_2) + \gamma \max_{p \in K} \sum_{s' \in S} p(s'|s, a_1, a_2)V^*(s')$$

Algoritmo 1: Value-Iteration-AMG-IP(AMG-IP,*maxIter*)

input : AMGP-IP (given by $\langle S, A_1, A_2, R, K, \gamma \rangle$),
maxIter (maximum number of iterations)
output: V^* (t-state-to-go value function)
begin
$\langle A_1[\underline{V}^*], A_2[\underline{V}^*] \rangle$ = SolveOne($\langle S, A_1, A_2, R, K, \gamma \rangle$,*maxIter*);
V^* = SolveTwo($\langle S, A_1[\underline{V}^*], A_2[\underline{V}^*], R, K, \gamma \rangle$,*maxIter*);
return V^* ;

Algoritmo 2: SolveOne(AMG-IP,$maxIter$)

input : AMGP-IP (given by $\langle S, A_1, A_2, R, K, \gamma\rangle$),
$maxIter$ (maximum number of iterations)
output: $\langle A_1[\underline{V}^*], A_2[\underline{V}^*]\rangle$ (players set actions that achieves $\underline{V}^*$)

```
begin
    V^0 = 0;
    t = 0;
    //build new actions set for the players
    A_1[V*] = ∅;
    A_2[V*] = ∅;
    //construct t-stage-to-go value functions V^t until termination condition is met
    while i < maxIter do
        t = t + 1;
        foreach s ∈ S do
            V^t(s) = 0;
            //update V^t with Q^t
            foreach a_1 ∈ A_1(s) do
                V^t_min(s) = ∞;
                foreach a_2 ∈ A_2(s) do
                    Q^t = Regress(V^{t-1},s,S,a_1,a_2,R,K,γ,true);
                    φ(s) = argmin(V^t_min(s),Q^t);
                    V^t_min(s) = min(V^t_min(s),Q^t);
                π(s) = argmax(V^t(s),V^t_min(s));
                V^t(s) = max(V^t(s),V^t_min(s));
            Add π(s) in A_1[V*](s);
            Add φ(s) in A_2[V*](s);
    return ⟨A_1[V*], A_2[V*]⟩;
```

Algoritmo 3: SolveTwo(AMG-IP,$maxIter$)

input : AMGP-IP (given by $\langle S, A_1[\underline{V}^*], A_2[\underline{V}^*], R, K, \gamma\rangle$),
$maxIter$ (maximum number of iterations)
output: V^t (t-state-to-go value function)

```
begin
    V^0 = 0;
    t = 0;
    //construct t-stage-to-go value functions V^t until termination condition is met
    while i < maxIter do
        t = t + 1;
        foreach s ∈ S do
            V^t(s) = 0;
            //update V^t with Q^t
            foreach a_1 ∈ A_1[V*](s) do
                V^t_min(s) = ∞;
                foreach a_2 ∈ A_2[V*](s) do
                    Q^t = Regress(V^{t-1},s,S,a_1,a_2,R,K,γ,false);
                    φ(s) = argmin(V^t_min(s),Q^t);
                    V^t_min(s) = min(V^t_min(s),Q^t);
                π(s) = argmax(V^t(s),V^t_min(s));
                V^t(s) = max(V^t(s),V^t_min(s));
    return V^t ;
```

Algoritmo 4: REGRESS(V,s,S,a_1,a_2,R,K,γ,isMin)

input : V (value function vector), s (currently state), S (states set), a_1 (action choose by Player 1 in state s), a_2 (action choose by Player 2 in state s), R (reward for the players), K (credal transition set), γ (discount factor), isMin (flag)
output: Q (discounted total reward)

```
begin
    if isMin = true then
        //Choose p ∈ K such as
        Q = R(s,a_1,a_2) + γ min_{p∈K} Σ_{s'∈S} p(s'|s,a_1,a_2)V(s');
    else
        //Choose p ∈ K such as
        Q = R(s,a_1,a_2) + γ max_{p∈K} Σ_{s'∈S} p(s'|s,a_1,a_2)V(s');
    return Q ;
```

Após ocorrerem as N iterações, SolveTwo devolve V^* para Value-Iteration-AMG-IP.

5. Finalmente, Value-Iteration-AMG-IP retorna o valor de equilíbrio V^* do jogo.

6. Trabalhos relacionados

Jogos markovianos alternados de dois jogadores de soma zero são amplamente estudados na literatura de teoria dos jogos, sendo que os resultados fundamentais da área foram obtidos a partir de trabalhos em MDPs (*Markov Decision Processes*) [Puterman 1994]. [Kurano et al. 1998] generalizou o modelo MDP em que a probabilidade de transição varia a cada estágio e demonstrou que seu modelo converge para uma solução ótima. O modelo proposto por Kurano está relacionado a um MDP-IP (*Markov Decision Process with Imprecise Probabilities*) [Satia and Lave Jr. 1973, White III and Eldeib 1994, Delgado et al. 2011] e a um BMDP (*Bounded-parameter Markov Decision Processes*) [Givan et al. 2000]. [Chang 2006] estendeu o trabalho de Kurano para resolver AMGs de dois jogadores de soma zero com imprecisão nas probabilidades dadas por intervalos e provou que é possível encontrar políticas de equilíbrio para esse jogo. A nossa extensão foi inspirada no modelo MDP-IP [Delgado et al. 2011]. Sendo este um modelo mais geral que os modelos MDP e BMDP, ao estendermos o MDP-IP para dois jogadores, definimos um tipo mais geral de AMG, que chamamos de AMG-IP. Um trabalho que também inclui imprecisão em um jogo de dois jogadores é [Quaeghebeur and de Cooman 2009] em que o principal objetivo é fazer com que o jogador 1 aprenda a estratégia do jogador 2 (adversário) usando o modelo de Dirichlet impreciso para representar e atualizar as crenças do jogador 1. Esse trabalho trata apenas jogos simultâneos com políticas probabilísticas, enquanto que estamos interessados em jogos alternados com políticas determinísticas.

7. Conclusão

A teoria dos jogos interpreta qualquer ambiente multi-agente como um jogo, desde que o impacto de cada agente sobre os outros seja significativo, independentemente dos agentes serem cooperativos ou competitivos [Russell and Norvig 2003]. Assim, este trabalho está relacionado a área de sistemas multi-agentes, visto que, propõe uma variante de Jogos Markovianos Alternados com dois jogadores de soma zero (AMGs) que permite tratar imprecisão em probabilidades de transição expressa por conjuntos credais.

Para esse novo modelo, definimos uma solução que estende a solução proposta por Chang [Chang 2006] e mostramos como encontrar políticas de equilíbrio diante da imprecisão sobre a função de transição probabilística. Além disso, definimos um algoritmo, Value-Iteration-AMG-IP, que implementa a solução proposta.

Como trabalho futuro, pretendemos construir uma solução fatorada para o modelo AMG-IP (assim como foi construído no trabalho de [Delgado 2010] para o modelo MDP-IP), que permita resolver de maneira eficiente AMG-IPs com um grande número de estados.

Agradecimentos

Este trabalho recebeu apoio financeiro do CNPq e FAPESP (projeto temático 2008/03995-5).

Referências

Chang, H. S. (2006). Perfect information two-person zero-sum Markov games with imprecise transition probabilities. In *Mathematical Methods of Operations Research*, pages (64)335–351. Springer-Verlag.

Cozman, F. G. (2000). Credal networks. In *AI Journal*, pages 120(2):199–233.

Delgado, K. V. (2010). *Processos de decisão Markovianos fatorados com probabilidades imprecisas.* PhD thesis, IME-USP, Brasil.

Delgado, K. V., de Barros, L. N., Cozman, F. G., and Shirota, R. (2009). Representing and solving factored Markov decision processes with imprecise probabilities. In *ISIPTA*, pages 169–178. Durham, United Kingdom.

Delgado, K. V., Sanner, S., and de Barros, L. N. (2011). Efficient solutions to factored MDPs with imprecise transition probabilities. In *AI Journal*. Accepted 03 January 2011.

Ferguson, T. S. (Accesso em 18 fevereiro de 2011.). *Game Theory.* http://www.math.ucla.edu/t̃om/Game_Theory/ Contents.html.

Fudenberg, D. and Tirole, J. (1991). *Game Theory.* Cambridge, MA.

Givan, R., Leach, S., and Dean, T. (2000). Bounded-parameter Markov decision processes. In *AI Journal*, pages (39)71–109.

Kurano, M., Song, J., Hosaka, M., and Huang, Y. (1998). Controlled Markov set-chains with discounting. In *J. Appl. Prob.*, pages (35)293–302.

Littman, M. L. (1996). *Algorithms for Sequential Decision Making.* PhD thesis, Department of Computer Science - Brown University.

Osborne, M. J. and Rubinstein, A. (1994). *A Course in Game Theory.* Cambridge, MA.

Owen, G. (1982). *Game Theory: Second Edition.* Orlando, Florida.

Parthasarathy, T. and Raghavan, T. E. S. (1971). *Some topics in two-person games.* New York.

Puterman, M. L. (1994). *Markov Decision Processes.* New York.

Quaeghebeur, E. and de Cooman, G. (2009). Learning in games using the imprecise Dirichlet model. In Inc., E. S., editor, *Int. J. Approx. Reasoning*, pages (50)243–256. New York, NY, USA.

Russell, S. J. and Norvig, P. (2003). *Artificial Intelligence - A modern Approach.* Prentice-Hall, New Jersey.

Satia, J. K. and Lave Jr., R. E. (1973). Markovian decision processes with uncertain transition probabilities. In *Operations Research*, pages (21)728–740.

Shapley, L. S. (1953). Stochastic games. In *Proceedings of the National Academy of Sciences*, pages 39 (10):1095–1100.

Walley, P. (1991). *Statistical Reasoning with Imprecise Probabilities.* London.

White III, C. C. and Eldeib, H. K. (1994). Markov decision processes with imprecise transition probabilities. In *Operations Research*, pages (42)739–749.

A formal system for quantum communication environments*

Anderson de Araújo[1], **Marcelo Finger**[1]

[1]Department of Computer Science
Institute of Mathematics and Statistics (IME)
University of Sao Paulo (USP)
CEP 05508 090 – São Paulo – SP – Brazil

aaraujo@gmail.com, mfinger@ime.usp.br

Abstract. *The present paper provides a complete first-order system for quantum communication environments. These environments are static reliable synchronous systems of quantum communication among agents and the formal system defined combines probability and knowledge operators to describe them.*

1. Introduction

In these days quantum computing is one of the main lines of research in computer science [Nielson and Chuang 2000]. The interest basically relies on results that show, first, that quantum computers can efficiently perform tasks for which nowadays we do not know efficient classical algorithm [Shor 1994] and, second, on results that indicates the total security of quantum communication protocols [Mayers 2001]. Although some scientists are suspicious about the viability of quantum computers (see, for instance, [Landauer 1995]), the majority of them beliefs that quantum computation and information are physically implementable and they will be a reality in the future (see, for instance, [Preskill 1998]).

In particular, according to [Ying 2010], quantum computing will have a strong interplay with artificial intelligence, mainly with respect to distributed computation and communication systems, because the physical implementation of functional quantum computers is difficult and an important alternative is the distributed implementation [Yimsiriwattana and Lomonaco 2004]. Nonetheless, the possible physical implementation of quantum distributed systems still has foundational problems, associated to the quantum decoherence and the accumulation and propagation of errors in quantum computers. These problems are summarized in the question raised by Landauer in [Landauer 1996]: how can coding prevent our quantum gates from making small errors, if the code and the device have no way of knowing what gate we are trying to implement?

Landauer's question indicates two central features of computing at the level of quantum systems: probability and knowability. From the one hand, it is necessary to have sophisticated devices to control the probability of errors during computations and, from the other hand, the postulates of quantum mechanics establish that measurements change these probability, that is to say, in a certain sense we have knowledge limitations about quantum systems. In order to make explicit these features in the case of distributed computing, the present paper we will provide a logical analysis of quantum communication systems.

In section 2, the main previous works related to the theme of this paper are presented. In section 3, a formal description of quantum communication environments will

*This work was supported by Fapesp Thematic Project 2008/03995-5 (LOGPROB).

be given. Only static synchronous communication of quantum messages among agents in distributed environments is analyzed, which means that the time evolution of the messages and adversaries are not considered. In section 4, it will be associated a first-order language to quantum communication environments that is able to express facts about probability and knowability in these quantum systems. In section 5, first-order multimodal structures with probability operator are defined, which provides semantics for the language defined. In section 6, a complete axiomatization of the quantum communication structures will be given.

2. Previous works

Quantum communication systems were analyzed for the first time in [van der Meyden and Patra 2003a], where it is proposed a modal logic for knowledge and time in quantum protocols. No axiomatization was presented in that work, its description of the quantum distributed systems will be, however, the main reference for our definition of quantum communication environment. Our perspective is logical oriented and restricted to communication systems, but an overview about others approaches to distributed quantum system can be found in [Denchev and Pandurangan 2008].

It is important to emphasize that our approach to quantum communication systems, differently from [van der Meyden and Patra 2003a], is based on density operators. According to [Cohen-Tannoudji et al. 1977], the formalization of quantum mechanics in terms of density operators is more convenient than the formalization using the state vector language for thinking about scenarios whose states are not completely known, although both being mathematically equivalent. Communication environments are just such a kind of scenario. For this reason, it seems to us that this approach is preferable.

Many logical system have been defined to axiomatize the characteristics of quantum probability; the main references can be found in [K. Engesser and Lehmann 2009]. We mention in special the paper [van der Meyden and Patra 2003b] which provides a clear axiomatization for measurement probabilities in quantum systems. In what follows we will incorporate some of the strategies of [van der Meyden and Patra 2003b]; the differences will be indicated when we reach at the description of our axiomatic.

With respect to knowledge operators in distributed quantum system, the main works are again [van der Meyden and Patra 2003a] and [Baltag and Smets 2010]. Our treatment of knowability presented bellow is similar to that in [Baltag and Smets 2010]. Nevertheless, the axiomatic presented here is essentially different from the previous approaches, because we will relate knowability to quantum communication. In fact, the axiomatics defined here for the knowledge operator in communication environments permits to express many properties that the previous ones are not able to express. Unfortunately, there is no space to show this expressive power in the present paper. In section 7, we will indicate some future works, and to analyse the expressive power of the system outlined in this paper is indeed one of these works.

In what follows we will presuppose knowledge about quantum mechanics, as presented for instance in [Cohen-Tannoudji et al. 1977, Nielson and Chuang 2000], and about modal logics, for instance [Carnielli and Pizzi 2008].

3. Quantum communication environments

We will define quantum communication environments in terms of the concepts of *agents* and *informational states*. This analysis is restricted to static reliable synchronous communication among agents.

Definition 3.1. *An* agent g *is a finite set of* propositions*, i.e.,* $g = \{p_1, \ldots, p_n\}$ *where each* p_i *is a proposition. A* group of agents *is a non-empty finite set of agents. The notion of proposition is primitive.*

To model the communication among agents, we assume that for each agent g in a group G there is a codification cod of the propositions of g as sequences of bits.

Definition 3.2. *Let* $G = \{g_1, \ldots, g_n\}$ *be a group of agents. The* message assignment *is the function* $msg : \{1, 2, \ldots, n\} \rightarrow cod_i$ *where* $cod_i : g_i \rightarrow \{0, 1\}^m$ *is a function which associates a code of length* m *to each proposition of agent* g_i *(this code is a sequence of bits with length* m*). Each* $msg(i)$ *will be called a* message*, and we will write* M_G *to denote the set of all messages of* G*.*

By the first postulate of quantum mechanics [Cohen-Tannoudji et al. 1977, Nielson and Chuang 2000], to the agents' messages there is a corresponding Hilbert space that represents these messages as physical entities.

Definition 3.3. *Let* $G = \{g_1, \ldots, g_n\}$ *be a group of agents. The set of* possible messages H_G *of* G *is the complex Hilbert space generated by* M_G*, i.e.,* H_G *is a set of normalized vectors* $|\ \rangle : M_G \rightarrow \mathbb{C}$*, equipped with the inner product, such that* $\sum_{m \in M_G} ||m\rangle|^2 < \infty$*.*

As we said in the previous section, in communications environments the states of the systems may be not completely known. Then, to formalize the messages while quantum states, we will use the density operator language.

Definition 3.4. *Given a set of vectors* $|msg_1\rangle, \ldots, |msg_k\rangle$ *in* H_G *associated to the messages in* M_G *with respective probability amplitudes* $p_1, \ldots, p_k$*, the density operator for the system* H_G *is the operador* ρ_G *defined by the equation*

$$\rho_G \stackrel{def}{=} \sum_{i=1}^{k} p_i |msg_i\rangle\langle msg_i|,$$

where $|msg_i\rangle\langle msg_i|$*, the outer product of* $msg_i \in M_G$*, is called a* quantum message.

Due to one of the postulates of quantum mechanics [Cohen-Tannoudji et al. 1977, Nielson and Chuang 2000], any vector in H_G can be represented as sequences of *quantum bits* (*qubit* for short), which are represented by the two dimensional Hilbert space $\overline{H}_G$, with a preferred orthonormal basis given by the two vectors $|0\rangle$ and $|1\rangle$. More precisely, an m-qubit Hilbert space is a space $\overline{H}_G^m$ of dimension 2^m with the form

$$\overline{H}_G \overbrace{\otimes \cdots \otimes}^{m} \overline{H}_G,$$

where $\otimes$ is the tensor product. We write $|v_1, \ldots, v_m\rangle$ for the vector $|v_1\rangle \otimes \cdots \otimes |v_m\rangle$ in $\overline{H}_G^m$, where each v_j is either 0 or 1. The set of vectors $|v_1, \ldots, v_m\rangle$ is a basis of $\overline{H}_G^m$, called the *computational basis*. We will identify H_G with $\overline{H}_G^m$ and will just write H_G.

In a quantum passing message system, each message is thought of as being in the possession of some agent, but this agent may change from time to time, as an agent can send some of its message to another. Following [van der Meyden and Patra 2003a], we define a function to denote the location of each message.

Definition 3.5. *Let H_G be the Hilbert space of the group of agents $G = \{g_1, \ldots, g_n\}$. The* m-location assignment *is the function $loc : H_G \to \{1, 2, \ldots, n\}$ such that $loc(msg(i))$ denotes that agent g_i has the message $msg(i)$ of H_G.*

Note that the letter m in the expression "m-location assignment" is to remember that the codes are sequences of qubits with length m. These sequences of qubits represent in quantum terms the previous codes of the messagens in M_G. Now we can model the reliable synchronous communication of quantum messages among the agents as the transmission of quantum messages from one agent to another agent in a group.

Definition 3.6. *Let H_G be the Hilbert space of group of agents $G = \{g_1, \ldots, g_n\}$. The* channel assignment *is the function $chan : \{1, 2, \ldots, n\}^2 \to H_G$ such that $chan(i, j) = |msg\rangle$ means that the quantum message $|msg\rangle$ has been transmitted from agent g_i to agent g_j.*

Suppose that $loc^{-1}(i) = \{i_l, \ldots, i_k\}$ is the set of indices of the quantum messages located at agent i. Then agent i is able to perform a general measurement on these k messages. We represent a quantum operation on k messages by a finite sequence of operators $M = (M_1, \ldots, M_l)$ with each M_j operating on H_G. Suppose the agents simultaneously perform the quantum measurements $(M_1, \ldots, M_l)$ where each M_i is a measurement on the $k_i = |loc^{-1}(i)|$ quantum message located at agent i. Each operation M_i products some outcome m_i, the index of some linear transformation M_j operating on H_G. We represent a combined outcome of these measurements by a new function from agents to outcomes.

Definition 3.7. *Let H_G be the set of possible messages of a group of agents $G = \{g_1, \ldots, g_n\}$ and $M = (M_1, \ldots, M_l)$ be a finite sequence of operators on H_G. The* measurement assignment *is the function $res : G \to M \times \mathbb{R}$ such that $res(i) = (M_i, m_i)$ records the measurement performed and the outcome obtained by the agent i.*

By one of postulates of quantum mechanics [Cohen-Tannoudji et al. 1977, Nielson and Chuang 2000], each measurement M_i is a self-adjoint linear operator and the outcome of a measurement is a real number. So the measurement assignment res is well-defined. In this way, we conclude our description of the quantum communication environments.

Definition 3.8. *A* quantum communication environment *is a tuple $E = (G, S)$ where G is a group of agents and $S = \{s_1, \ldots, s_m\}$ is the set of* informational states *for $s_j = (msg, loc, chan, res)$.*

Since we are confined to a static description of quantum communication environments, we will not consider the postulate of quantum mechanics that expresses which the time evolution of quantum systems is given by an unitary operator [Cohen-Tannoudji et al. 1977, Nielson and Chuang 2000].

4. Quantum communication languages

A quantum communication language is specified by defining its *alphabet*, *terms* and *formulas*.

Definition 4.1. *Let $E = (G, S)$ be a quantum communication environment. The* language *of E is the multimodal first-order language with equality $\mathcal{L}$ such that:*

1. *The alphabet of $\mathcal{L}$ has two sorts of variables, one sort for* scalars *and other for* vectors*, as well as an infinite set of* basis variables $\{\vec{v}, \vec{w}, \ldots\}$ *to represent the basis set considered for H_G[1];*
2. *For each of these basis variables $\vec{v}$, the alphabet of $\mathcal{L}$ also has a finite set of* constant vectors $\vec{v}_0, \ldots, \vec{v}_m$ *representing the vectors in the basis of H_G;*
3. *The alphabet of $\mathcal{L}$ has the* scalar constants 0 *and* 1 *and the* vector constants $\vec{0}$ *and* $\vec{1}$*;*
4. *The alphabet of $\mathcal{L}$ has the functions symbols $-$, $+$ and $\times$ for the usual operations on elements in a real-closed field, the function symbols $\neg_v$, $+_v$, $\times_s$ and $\times_v$ for the orthocomplement, matrix addition, multiplication by a scalar and matrix multiplication on vector spaces;*
5. *The alphabet of $\mathcal{L}$ has a unary* probability operator P *and a binary* basis transformation M_{ij}*;*
6. *The alphabet of $\mathcal{L}$ has* knowledge operators K_I*, one for each subgroup of agents $I \subseteq G$;*
7. *Nothing more except the symbols specified above are in the* alphabet *of $\mathcal{L}$.*

The definitions of all syntactic notions are as in [Carnielli and Pizzi 2008], including the *terms* and *formulas*, except by the following differences.

Definition 4.2. *The set of terms of $\mathcal{L}$ is such that:*

1. *The* matrix terms *of $\mathcal{L}$ are defined in the following way:*
   (a) *The constant vectors and variables for constant vectors are matrix terms;*
   (b) *If α and β are matrix terms and a is an scalar constant, then $\neg_v\alpha$, $a \times_s \alpha$, $\alpha +_v \beta$ and $\alpha \times_v \beta$ also are terms.*
2. *If α is a matrix term, then $P(\alpha)$ is a term of $\mathcal{L}$. These terms are called* probability terms*.*
3. *If $\vec{v}$ and $\vec{w}$ are basis variables, then $M_{ij}(\vec{v}, \vec{w})$ is a term of $\mathcal{L}$. These terms are called* transformation terms*.*
4. *The matrix, probability and transformation terms are just the terms of $\mathcal{L}$ defined by the conditions (1), (2) and (3) above.*

Moreover, the scalar terms *are just the terms defined in $\mathcal{L}$ from 0 and 1 using $-$, $+$ or $\times$ as well as any pseudo-term defined using these scalar constants and funtions*[2].

[1] Since the expressiveness of many-sorted logic and first-order logic is the same [Hodges 2005, p.202-206], we will omit the reference to sorts.

[2] Pseudo-terms are formulas that defines elements is the structure considered. For example, terms for denoting real numbers and complex numbers can be defined in $\mathcal{L}$ as pseudo-terms.

In quantum mechanics probability has a crucial role. Hence it is important to designate some special formulas that express facts about probabilities in quantum communication environments.

Definition 4.3. *In the set of formulas of $\mathcal{L}$ a* linear probability atom *is an expression of the form $a_1 \times P^1(\alpha_1) + \cdots + a_k \times P^1(\alpha_k) = a$, where each a_i is a scalar term as well as a, and each α_i is a matrix term.*

As the language $\mathcal{L}$ has knowledge operators K_I for subgroups of agents, it can state facts about distributed knowledge in communication environments, such as in [Baltag and Smets 2010]. In particular, the description of the knowledge of an individual agent i is described by $K_{\{i\}}$, which we abbreviate by K_i. Besides, $\mathcal{L}$ has probability and transformation operators similar to [van der Meyden and Patra 2003b], and so it can also state properties about quantum distributed knowledge. Concepts like entanglement of information, phase relations, uncertainty relation, etc, also can be expressed in $\mathcal{L}$. This shows that the language $\mathcal{L}$ has enough expressiveness with respect to quantum communication environments. For instance, the statement that agent i knows the message α with probability $\frac{1}{\sqrt{2}}$ can be expressed by the sentence $K_i P(\alpha) = \frac{1}{\sqrt{2}}$ of the language $\mathcal{L}$[3].

5. Quantum communication structures

Quantum communication environments are distributed systems, so we need a semantics for the language $\mathcal{L}$ which permits us to consider the transmission of quantum messages among the agents. For such an aim, we define the notion of *informational range*.

Definition 5.1. *Let $E = (G, S)$ be a communication environment. The* informational range *R_G of the group G is a family of binary relations $\approx_I$ on H_G^2, one for each $I \subseteq G$, i.e,*

$$R_G = \{\approx_I \subseteq H_G^2 : I \subseteq G\}.$$

Intuitively, an informational range shows how the information is available among the subgroups of agents. From this concept, we can define equivalence relations among the information associated to the agent's messages.

Definition 5.2. *Let $E = (G, S)$ be a communication environment and R_G be the informational range of G. An E-*quantum communication frame *is a multi-modal frame (H_G, R_G) such that:*

Equivalence *: For each $I \subseteq G$, $\approx_I$ is an equivalence relation;*
Observability *: For all $s, r, \in H_G$ and $I \subseteq G$, if $s = r$ then $s \approx_I r$;*
Monotonicity *: For all $I, J \subseteq G$, if $I \subseteq J$ then $\approx_J \subseteq \approx_I$;*
Vacuousness *: For all $s, r, \in H_G$, $s \approx_\emptyset r$.*

The equivalence condition establishes that if $s \approx_I r$ then the subgroup I cannot distinguish the informational state s from r, that is to say, s and r are *indistinguishable* to I. One the other hand, by the observability condition, if r and s are equals then every

[3]The expression $\frac{1}{\sqrt{2}}$ is a pseudo-term that is definable using the symbols for field expression of the language $\mathcal{L}$.

group of agents cannot distinguish between r and s. The monotonicity expresses that if some informational states are indistinguishable for a group of agents I then these states are also indistinguishable for any subgroup J of I. Finally, the vacuouness says that the empty group cannot distinguish between any two informational states, which is true for it is unable to make any observation.

Definition 5.3. *Let $E = (G, S)$ be a communication environment and (H_G, R_G) be an E-quantum communication frame. A* quantum communication structure *$\mathcal{A}$ for the language $\mathcal{L}$ is the tuple $\mathcal{A} = (H_G, R_G, I_G)$ in which I_G is an interpretation function from $\mathcal{L}$ to H_G such that:*

1. *$I_G(0)$ is the number zero and $I_G(1)$ is the number one in the complex field $\mathbb{C}$ of H_G;*
2. *$I_G(\vec{0})$ is the null matrix and $I_G(\vec{1})$ is the identity matrix of H_G;*
3. *For each state constant $\vec{v}_i$ of $\mathcal{L}$, $I_G(\vec{v}_i)$ is the matrix $|v_i\rangle\langle v_i|$ where $|v_i\rangle$ is the i-th vector of the computational basis of H_G;*
4. *$I_G(-)$, $I_G(+)$ and $I_G(\times)$ are the functions inverse, plus and times, respectively, on the complex field $\mathbb{C}$ of H_G;*
5. *$I_G(\neg_b)$ is the projection operator $\perp$ projecting onto the orthogonal complement of the image of H_G under the state considered, $I_G(+_v)$ is the matrix addition of H_G, $I_G(\times_s)$ is matrix multiplication by a scalar and $I_G(\times_v)$ is the matrix multiplication of H_G[4];*
6. *$I_G(P)$ is the unary operator on H_G such that $I_G(P)(|v_i\rangle)$ is the trace of the matrix $|v_i\rangle\langle v_i||v_i\rangle\langle v_i|$, i.e., $I_A(P)(|v_i\rangle) := \mathit{Tr}(|v_i\rangle\langle v_i||v_i\rangle\langle v_i|) := \sum_j(|v_i\rangle\langle v_i||v_i\rangle\langle v_i|)_{jj} = |\langle v_i|v_i\rangle|^2$;*
7. *$I_G(M_{i,j})$ is the binary operator on H_G such that $I_G(M_{i,j})(|v\rangle, |w\rangle)$ is the element v_{ij} such that $M = (v_{ij})$ is the $m \times m$ unitary complex matrix for which $M|v\rangle = |w\rangle$.*

From the notion of quantum communication structure, we define the *satisfatibility relation* $\models_s$ as in [Carnielli and Pizzi 2008]. We only emphasize the case for modal formulas:

$$\mathcal{A} \models_s K_I\phi \text{ if, and only if, for all } r \in H_G \text{ such that } r \approx_I s, \mathcal{A} \models_r \phi.$$

6. Quantum communication axiomatics

In this section we will define a theory $\mathcal{T}$ for quantum communication environments. We will presuppose a *derivability relation* $\vdash$ defined according to some classical calculus for first-order logic, for instance the one in [Carnielli and Pizzi 2008], and we will only specify the new axioms and rules of $\mathcal{T}$. The axiomatization of $\mathcal{T}$ consists of four parts, each dealing with one aspect of these communication systems.

The first part of $\mathcal{T}$ has axioms to express that the set of possible messages is an m-dimensional Hilbert space, where m is the maximum length of the messages in M_G, and that the Hilbert space is a orthocomplemented lattice.

(A1) $(\vec{v}_1 \vee \cdots \vee \vec{v}_n) \wedge (\neg i = j \rightarrow \neg(\vec{v}_i \wedge \vec{v}_j))$

[4]We will omit $\times_v$ when it is clear that we are considering the matrix multiplication.

(A2) $(\alpha \times_v \neg_v \alpha = \vec{0}) \wedge (\alpha +_v \neg_v \alpha = \vec{1}) \wedge (\neg_v \neg_v \alpha = \alpha)$
(A3) $(\alpha \leq \beta \rightarrow \neg_v \beta \leq \neg_v \alpha) \wedge (\alpha \times_v (\neg_v \alpha +_v (\alpha \times_v \beta)) \leq \beta)$

The second part of $\mathcal{T}$ has axioms to state the properties of the quantum probability operator for quantum communication environments.

(A4) $0 \leq P(\alpha) \wedge P(\alpha) \leq 1$.
(A5) $P(\alpha +_v \neg_v \beta) = 1$
(A6) $P(\alpha \times_v \beta) + P(\alpha \times_v \neg_v \beta) = P(\alpha)$
(A7) $\alpha = (a_1 \times_s \vec{v}_1) +_v \cdots +_v (a_n \times_s \vec{v}_n) \rightarrow (P(\beta) = |a_1|^2 + \cdots + |a_m|^2 \wedge \beta = (\sqrt{P(\beta)})^{-1} \times ((a_1 \times_s \vec{v}_1) +_v \cdots +_v (a_m \times_s \vec{v}_m)))$ if the measurement was done on the vectors in $\{\vec{v}_i\}_{i \leq n}$.

Note that these axioms are different from those given in [van der Meyden and Patra 2003b]. They explicitly formalize the main property of measuments in quantum mechanics. Besides, in our approach the distributivity of probability is the sentence $\alpha = \beta \rightarrow P(\alpha) = P(\beta)$, which is an immediate theorem due to the Leibniz's law for equality.

The third part of $\mathcal{T}$ has axioms for basis transformation, which corresponds to the identity matrix when the basis is not changed and consecutive basis transformations correspond to matrix multiplication.

(A8) $M_{ij}(\vec{v}, \vec{w}) = M^*_{ij}(\vec{w}, \vec{v})$
(A9) $(i = j \rightarrow M_{ij}(\vec{v}, \vec{v}) = 1) \wedge (\neg i = j \rightarrow M_{ij}(\vec{v}, \vec{v}) = 0)$
(A10) $M_{ij}(\vec{v}, \vec{x}) = (M_{i1}(\vec{v}, \vec{w}) \times M_{i1}(\vec{w}, \vec{x})) + \cdots + (M_{im}(\vec{v}, \vec{w}) \times M_{im}(\vec{w}, \vec{x}))$

Differently from [van der Meyden and Patra 2003b], we have fixed the computational basis for the transformations, which is in accordance with the fact that quantum measurements in other bases can be carried out by combining unitary transformation and measurements on the computational basis [Cohen-Tannoudji et al. 1977, Nielson and Chuang 2000]. In this way, we can define the transition probability operator T (a quantum analogue of conditional probabilities) defined in [van der Meyden and Patra 2003b] stating that $T(\vec{v}_i, \vec{w}_j) = |M_{i,j}(\vec{v}, \vec{w})|^2$. Note, moreover, that from axioms $A8$ and $A10$, it is possible to derive unitarity of the transformations $((M_{i1}(\vec{v}, \vec{w}) \times M^*_{j1}(\vec{v}, \vec{w})) + \cdots + (M_{im}(\vec{v}, \vec{w}) \times M^*_{jm}(\vec{v}, \vec{w}))) = 1$.

The fourth, and last, part of $\mathcal{T}$ has axioms for knowledge operators. These axioms are the usual axioms of the epistemic logic [Baltag and Smets 2010], but in the context of the general framework for distributed knowledge developed here.

(A11) If $\mathcal{T} \vdash \phi$ then $\mathcal{T} \vdash K_G \phi$.
(A12) If $I \subseteq J$ then $\mathcal{T} \vdash K_I \phi \rightarrow K_J \phi$.
(A13) $K_G(\phi \rightarrow \psi) \rightarrow (K_G \phi \rightarrow K_G \psi)$
(A14) $K_G \phi \rightarrow K_G K_G \phi$
(A15) $\neg K_G \phi \rightarrow K_G \neg K_G \phi$

In this way, we end the definition of $\mathcal{T}$.

Definition 6.1. *Let $E = (G, S)$ be a quantum communication environment. The* theory *of E is the first-order theory $\mathcal{T}$ with the axioms and rules $A1$-$A15$ defined above plus a complete axiomatization for algebraically closed fields.*

Since Tarski showed in [Tarski 1951] that the algebraically closed fields can be completely axiomatized, the theory $\mathcal{T}$ is well-defined. Furthermore, we can prove the adequacy of $\mathcal{T}$ with respect to quantum communication structures $\mathcal{A}$.

Theorem 6.1. *For all sentence ϕ of $\mathcal{L}$, $\mathcal{T} \vdash \phi$ if, and only if, $\mathcal{A} \vDash \phi$.*

Proof sketch: The soundness, $\mathcal{T} \vdash \phi$ implies $\mathcal{A} \vDash \phi$, is straightforward. The completeness, $\mathcal{A} \vDash \phi$ implies $\mathcal{T} \vdash \phi$, is a non-trivial task. It comprises an appropriate combination of the completeness proof given in [van der Meyden and Patra 2003b], for the probability and transformation part of $\mathcal{T}$, and the completeness proof for the calculus of orthocomplemented lattice, for the algebraic part of $\mathcal{T}$, with a construction of canonical models for multi-modal logics, for the epistemic part of $\mathcal{T}$[5]. □

7. Future works

The completeness of the theory $\mathcal{T}$ with respect to quantum communication structures shows the adequacy of our approach from a logical point of view. Nonetheless, we have not explored the expressiveness of this framework with respect to distributed quantum systems. An interesting development of our work is to examine that expressiveness.

In our approach we have not considered the time evolution of quantum systems. An interesting approach would be to extend our axiomatic with modalities for time. In [van der Meyden and Patra 2003a], a first attempt was realized, but it is to much restricted and no axioms were exhibited. We believe that a more compelling approach would be to use the logic CTL with probability, for instance as that showed in [Brázdil et al. 2008] or as in [Baltazar et al. 2008].

In quantum mechanics, tensor product has a crucial significance. We can already handle this to some extent simply by applying our language to the case where the dimension of the Hilbert space H_G is a restricted product, but it is desirable to have the tensor product as a more integral part of the language. In [Mateus and Sernadas 2006], it was showed a quantum logic capable to expresses tensor product, but such a logical system is so complicated and we cannot foresee how to combine it with multi-modal logic, as we need in quantum communication environments.

To conclude, it is important to mention that we have not analyzed questions of computational complexity. As we work with probability operators in quantum systems, it would be desirable to analyze the *probabilistic satisfiability problem* [Georgakopoulos et al. 1988] in the context of quantum communication environments. In fact it is the quantum version of the satisfiability problem [Bravyi 2006] which is in question, for quantum systems have quantum probability and not the classical one. We do not know if the logical system presented in this paper is able to express the quantum satisfiability problem. If it can, then we can analyze open problems about quantum satisfiability using our logical system, from the one hand, and to examine the complexity of quantum communication in distributed systems, for the other.

[5]The complete proof will be presented in the extended version of this paper.

References

Baltag, A. and Smets, S. (2010). Correlated knowledge: an epistemic-logic view on quantum entanglement. *International Journal of Theoretical Physhics*, 49:3005–3021.

Baltazar, P., Chadha, R., and Mateus, P. (2008). Quantum computation tree logic - model checking and complete calculus. *International Journal of Quantum Information*, 6(2):219–236.

Bravyi, S. (2006). Efficient algorithm for a quantum analogue of 2-sat. ArXiv Quantum Physics e-prints.

Brázdil, T., Forejt, V., and Kucera, J. K. A. (2008). The satisfiability problem for probabilistic ctl. In *LICS 2008*, volume 1043-6871/08 of *IEEE Computer Science Society*, pages 391–402.

Carnielli, W. and Pizzi, C. (2008). *Modalities and Multimodalities*. Logic, Epistemology, and the Unity of Science. Springer Verlag, Berlin.

Cohen-Tannoudji, C., Diu, B., and Laloë, F. (1977). *Quantum Mechanics*. Wiley, New York.

Denchev, V. and Pandurangan, G. (2008). Distributed quantum computing: A new frontier in distributed systems or science fiction? *ACM SIGACT News*, 39(3):77–95.

Georgakopoulos, G., Kavvadias, D., and Papadimitriou, C. H. (1988). Probabilistic satisfiability. *Journal of Complexity*, 4(1):1–11.

Hodges, W. (2005). *Model Theory*. Encyclopedia of Mathematics and Applications. Cambridge University Press, Cambridge.

K. Engesser, D. M. G. and Lehmann, D. (2009). *Handbook of quantum logic and quantum structures*. North-Holland. Elsevier, Amsterdan.

Landauer, R. (1995). Is quantum mechanics useful? *Philosophical Transactions: Physical Sciences and Engineering*, 353(1703):367–376.

Landauer, R. (1996). The physical nature of information. *Physics Letters A*, 217:188–193.

Mateus, P. and Sernadas, A. (2006). Weakly complete axiomatization of exogenous quantum propositional logic. *Information and Computation*, 204:771–794.

Mayers, D. (2001). Unconditional security in quantum cryptography. *Journal of the ACM*, 48(3):351–406.

Nielson, M. A. and Chuang, I. L. (2000). *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge.

Preskill, J. (1998). Quantum computing: Pro and con. In *Quantum Coherence and Decoherence*, volume 454 of *Mathematical, Physical and Engineering Sciences*, pages 469–486. The Royal Society.

Shor, P. (1994). Algorithms for quantum computation: Discrete log and factoring. In Goldwasser, S., editor, *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 124–134. IEEE Computer Society Press, Los Alamitos.

Tarski, A. (1951). *A Decision Method for Elementary Algebra and Geometry*. 2nd ed. University of California Press, Berkeley.

van der Meyden, R. and Patra, M. (2003a). Knowledge in quantum systems. In Tennenholtz, M., editor, *Theoretical Aspects of Rationality and Knowledge*, volume 6433, pages 104–117. ACM.

van der Meyden, R. and Patra, M. (2003b). A logic for probability in quantum systems. In Baaz, M. and Makowsky, J., editors, *Computer Science Logic*, volume 2803 of *Lecture Notes in Computer Science*, pages 427–440, Berlin. Springer.

Yimsiriwattana, A. and Lomonaco, S. J. (2004). Distributed quantum computing: A distributed shor algorithm. In Donkor, E., Pirich, A. R., and Brandt, H. E., editors, *Quantum Information and Computation II*, volume 5436 of *MSPIE*, pages 360–372.

Ying, M. (2010). Quantum computation, quantum theory and ai. *Artificial Intelligence*, 174:162–176.

Reduzindo o Custo do Teste de Integração com Algoritmos Evolutivos Multiobjetivos e Diferentes Medidas de Acoplamento

Wesley Klewerton Guez Assunção[1], **Thelma Elita Colanzi**[1 2], **Aurora Trinidad Ramirez Pozo**[1], **Silvia Regina Vergilio**[1]

[1]Departamento de Informática, Programa de Pós-Graduação em Informática, Universidade Federal do Paraná (UFPR)– Curitiba, PR – Brasil

[2]Departamento de Informática
Universidade Estadual de Maringá(UEM) – Maringá, PR – Brasil

{wesleyk, thelmae, aurora, silvia}@inf.ufpr.br

Resumo. *Para realizar o teste de software orientado a objetos é necessário determinar uma ordem para integrar e testar as classes que permita minimizar o número de stubs necessários para simular o comportamento de classes ainda não disponíveis durante a integração. O uso de algoritmos multiobjetivos permite considerar vários fatores e medidas que afetam a criação de stubs, e que por serem geralmente conflitantes impedem a obtenção de uma única solução. Em estudos realizados comparando diferentes algoritmos, o NSGA-II, um algoritmo evolutivo multiobjetivo (MOEA), tem obtido os melhores resultados. No entanto, existem outros MOEAs que não foram aplicados neste contexto e que podem melhorar o desempenho do NSGA-II. Além disso, os trabalhos existentes costumam utilizar somente duas medidas de acoplamento: número de métodos e de atributos. Considerando estas limitações, este artigo explora além do NSGA-II, o MOEA SPEA2 para resolver o problema em sistemas reais, considerando quatro medidas de acoplamento. Duas outras medidas são introduzidas: número de diferentes tipos de retorno e de diferentes tipos de parâmetros, em adição às duas medidas tradicionalmente utilizadas. Os resultados permitem a comparação entre os algoritmos, e mostram que ambos são eficientes para resolver o problema, já que encontram soluções de custo de teste mínimo para alguns sistemas, mesmo considerando um número maior de objetivos.*

1. Introdução

O teste de software é uma atividade considerada essencial para garantia de qualidade. Ele geralmente é realizado em etapas. A etapa de integração tem como objetivo testar a interação entre módulos, que são geralmente as classes de um software Orientado a Objetos(OO). Nesta etapa, é comum que algumas classes necessitem de recursos de outras classes que ainda estão em desenvolvimento, o que leva a necessidade de construção de *stubs*, aumentando a complexidade e o custo do teste. *Stubs* são pseudoimplementações que simulam recursos necessários, mas não disponíveis, que são indispensáveis para a execução da classe. Para reduzir o custo associado à construção dos *stubs* é necessário determinar uma ordem de desenvolvimento e teste das classes que minimize o número de *stubs* a serem gerados. O problema de determinar esta ordem ideal é conhecido na

literatura como problema de Ordenação e Teste de Integração de Classes (CITO - *Class and Integration Test Order*) [Abdurazik and Offutt 2006].

Este problema tem sido tema de vários trabalhos na literatura. Geralmente, as estratégias de solução são baseadas em um grafo direcionado chamado *Object Relation Diagrams* (ORDs) [Kung et al. 1995], no qual os vértices representam as classes, e as arestas os seus relacionamentos de dependência. Quando não existem ciclos de dependências no ORD, uma solução que não necessite de *stubs* é encontrada por uma simples ordenação topológica inversa do grafo, porém a maior parte dos programas desenvolvidos em Java apresentam ciclos de dependência [Melton and Tempero 2007], o que aumenta a complexidade para se encontrar uma solução e torna a resolução do problema CITO não trivial.

As estratégias baseadas em grafos [Kung et al. 1995, Tai and Daniels 1997, Traon et al. 2000, Briand and Labiche 2003, Bansal et al. 2009] possuem diversas limitações. Elas visam à minimização do número de ciclos que devem ser quebrados, mas entretanto, existem situações em que quebrar dois ciclos pode exigir menor custo do que quebrar somente um. Nestas situações, estas estratégias encontram apenas soluções subótimas [Briand et al. 2002]. Além disso, uma outra limitação é que estas estratégias são muito difíceis de serem adaptadas para considerar diversos fatores associados ao custo da construção de *stubs*, tais como: acoplamento por métodos e/ou por atributos, e outras restrições contratuais e/ou relacionadas ao desenvolvimento de software.

Estas limitações motivaram a utilização de algoritmos bioinspirados [Briand et al. 2002, Cabral et al. 2010, Pozo et al. 2011]. O primeiro trabalho [Briand et al. 2002] propôs a utilização de um Algoritmo Genético (AG) com uma função de *fitness* baseada em uma agregação de dois objetivos: o número de métodos e o número de atributos a serem simulados na construção dos *stubs*. Mas devido à dificuldade de ajustar os pesos correspondentes a estes objetivos, trabalhos mais recentes têm apresentado melhores resultados utilizando algoritmos multiobjetivos [Cabral et al. 2010, Pozo et al. 2011]. Em um destes estudos [Pozo et al. 2011] foram explorados três diferentes algoritmos multiobjetivos: NSGA-II, MTabu e PACO. O NSGA-II, um algoritmo evolutivo multiobjetivo (*Multi-objective Otimization Evolutionary Algorithm* - MOEA), foi o algoritmo que melhor se ajustou ao problema.

Apesar de estes estudos serem bastante promissores, algumas questões relacionadas ao problema CITO ainda permanecem em aberto. A primeira delas diz respeito aos objetivos utilizados. Nestes trabalhos apenas duas medidas, número de atributos e de métodos foram utilizadas, mas o problema é de fato multiobjetivo e diversas outras medidas de acoplamento entre classes que influenciam a complexidade de construir *stubs* precisam ser também consideradas. Estes estudos utilizam apenas sistemas pequenos, com poucas classes, o que gera um questionamento sobre o comportamento destas estratégias para sistemas grandes e complexos. Além disso, diferentes MOEAs existem, e não há registros de estudos de avaliação do desempenho de diferentes MOEAs para o problema em questão.

Para responder a estas questões este trabalho tem por objetivo analisar o comportamento de dois diferentes MOEAs, o NSGA-II e o SPEA2 quando aplicados ao problema CITO com quatro medidas de acoplamento associadas ao custo de construção de *stubs*. Duas outras medidas são introduzidas: número de diferentes tipos de retorno e número

de diferentes tipos de parâmetros, em adição às duas medidas tradicionalmente utilizadas: número de métodos e de atributos. Desta maneira foi possível, além da comparação dos resultados alcançados pelo NSGA-II e pelo SPEA2, analisar o comportamento dos MOEAs para resolver o problema utilizando vários objetivos, casos nos quais comumente o desempenho dos MOEAs pode ser prejudicado. O experimento conduzido utilizou quatro sistemas reais, maiores e mais complexos que aqueles utilizados nos trabalhos relacionados [Briand et al. 2002, Cabral et al. 2010, Pozo et al. 2011]. Os resultados são comparados utilizando-se três indicadores de qualidade: Distância Geracional (GD), Distância Geracional Inversa (IGD) e Distância Euclidiana a Solução Ideal (ED).

Este artigo está organizado da seguinte maneira: na Seção 2 são brevemente descritos os algoritmos multiobjetivos utilizados e na Seção 3 são discutidos aspectos da sua aplicação ao problema CITO. Na Seção 4 é dada uma descrição do estudo experimental realizado cujos resultados são apresentados e analisados na Seção 5. Finalmente, a Seção 6 contém as conclusões.

2. Algoritmos multiobjetivos

Problemas com objetivos múltiplos, ou multiobjetivo, estão presentes na maioria das disciplinas. Estes problemas são influenciados por diferentes fatores que determinam certos objetivos a serem, por exemplo, minimizados ou maximizados e para os quais não é possível encontrar uma solução única. Geralmente as funções objetivo são ditas conflitantes, e um conjunto de soluções ótimas surge. Para avaliar e determinar o grupo dessas boas soluções, utiliza-se o conceito de dominância de Pareto [Pareto 1927]. A ideia é descobrir soluções que não são dominadas por nenhuma outra no espaço de busca. Dado um conjunto de possíveis soluções para o problema, uma solução A domina uma solução B, se e somente se, A é melhor que B em satisfazer pelo menos um dos objetivos, sem ser pior que B em nenhum dos outros objetivos restantes. O conjunto de soluções não-dominadas formam a fronteira de Pareto, que fornece informações sobre o problema e é útil para resolver problemas reais, tais como os problemas da Engenharia de Software. Em muitas aplicações encontrar a fronteira de Pareto real é NP-difícil [Knowles et al. 2006], então o desafio é encontrar a fronteira mais próxima possível da fronteira de Pareto.

Algoritmos evolutivos multiobjetivos são extensões de AGs para problemas multiobjetivos que incorporam os conceitos de dominância de Pareto para criar estratégias distintas para evoluir e diversificar as soluções. Neste trabalho foram selecionados os MOEAs NSGA-II [Deb et al. 2002] e SPEA2 [Zitzler et al. 2001]. Ambos algoritmos utilizam uma população que evolui geração a geração aplicando os operadores de seleção, cruzamento e mutação.

O NSGA-II (*Non-dominated Sorting Genetic Algorithm*) [Deb et al. 2002] ordena, a cada geração, os indivíduos das populações pai e filha considerando a relação de não-dominância. Ele cria diversas fronteiras (linhas 10 e 11 da Figura 1(a)) e, depois da ordenação, as soluções com mais baixa dominância são descartadas. As fronteiras representam a estratégia de elitismo adotada pelo NSGA-II. Além disso, este algoritmo usa um operador de diversidade (*crowding distance*) que ordena os indivíduos de acordo com a sua distância em relação aos vizinhos da fronteira para cada objetivo, visando garantir maior espalhamento das soluções.

Ambas ordenações, de fronteiras e de distância de multidão, são usadas pelo ope-

rador de seleção (linhas 6 e 16 da Figura 1(a)) para determinar os indivíduos que sobrevivem para a próxima geração. O NSGA-II utiliza a seleção por torneio, selecionando soluções de fronteiras com maior dominância e, em caso de empate na dominação, a distância de multidão serve como critério de desempate.

```
   Entrada: N', g, f_k(X)
1  Inicializar população P'
2  Gerar população aleatória - Tamanho N'
3  Avaliar valores dos objetivos
4  Atribuir rank baseado na dominância de Pareto - Ordenação
5  Gerar população filho
6      Seleção por torneio binário
7      Recombinação e Mutação
8  para i=1 até g faça
9      para cada Pai e Filho na População faça
10         Atribuir rank baseado na dominância de Pareto - Ordenação
11         Gerar fronteiras não dominadas
12         Determinar a distância de multidão para cada solução das fronteiras e
           Percorrer todas as fronteiras adicionando para a próxima geração do
           primeiro ao N' indivíduo
13     fim
14     Selecionar indivíduos das melhores fronteiras e com maior distância de
       multidão
15     Gerar população filho
16         Seleção por torneio binário
17         Cruzamento e Mutação
18 fim
```

(a) NSGA-II

```
   Entrada: N', N̄, g, f_k(X)
1  Inicializar população P' - Tamanho N'
2  Criar um arquivo vazio E'
3  para i=1 até g faça
4      Calcular o fitness para cada indivíduo de P' e E'
5      Copiar todos os indivíduos não dominados de P' e E' para E'
6      se Tamanho de E' maior que N̄ então
7          Usar operador de eliminação de soluções de E'
8      senão se E' menor que N̄ então
9          Usar soluções dominadas de P' para completar E'
10     fim
11     Executar seleção por torneio binário com substituição
12     Aplicar Cruzamento e Mutação
13 fim
```

(b) SPEA2

Figura 1. Pseudocódigos dos algoritmos adaptados de [Coello et al. 2006]

O algoritmo SPEA2 (*Strength Pareto Evolutionary Algorithm*) [Zitzler et al. 2001], cujo pseudocódigo é mostrado na Figura 1(b), além de ter sua população regular, mantém um arquivo externo que armazena soluções não-dominadas encontradas a cada geração. Para cada solução que está no arquivo e na população é calculado um valor de *strength* que também é usado no cálculo do *fitness* do indivíduo. O valor de *strength* de uma solução *i* corresponde a quantidade de indivíduos *j*, pertencentes ao arquivo e a população, dominados por *i*. Este valor é utilizado durante o processo de seleção. O *fitness* de uma solução é a soma dos valores de *strength* de todas as soluções dominadas por *i*, tanto do arquivo externo como da população regular (linha 4 da Figura 1(b)). No processo evolutivo, o arquivo externo é utilizado para complementar a nova geração com soluções não dominadas. Como o tamanho do arquivo é fixo, se este tamanho for excedido, um algoritmo de agrupamento é utilizado a fim de reduzi-lo [Coello et al. 2006] (linhas 6 e 7 da Figura 1(b)).

3. MOEAs aplicados ao problema CITO

Para aplicar os MOEAs a fim de resolver o problema CITO é preciso encontrar uma boa representação para o problema. E esta representação influencia na implementação de todos os estágios do algoritmo. Como o problema CITO é relacionado à permutação de classes, as quais formam as ordens de teste, o cromossomo é representado por um vetor cujas posições contém inteiros que representam as classes. O tamanho do vetor é igual ao número de classes de cada sistema. Então, se cada classe é representada por um inteiro, uma solução válida para um problema de 8 classes seria $(3, 5, 4, 2, 1, 7, 6, 8)$. Neste exemplo, a primeira classe a ser testada e integrada seria a classe representada pelo número 3. Esta representação é a mesma utilizada em [Cabral et al. 2010].

Outra questão relacionada aos MOEAs é a escolha das funções objetivo. Como anteriormente mencionado, podem ser usados no problema CITO várias medidas e fa-

tores como acoplamento e restrições de tempo. Neste trabalho foram usadas quatro funções baseadas em diferentes medidas de acoplamento: acoplamento de atributos, acoplamento de métodos, número de diferentes tipos de retorno [Abdurazik and Offutt 2006], número de diferentes tipos de parâmetros [Abdurazik and Offutt 2006]. As duas primeiras medidas foram usadas na maioria dos trabalhos relacionados encontrados na literatura [Ré and Masiero 2007, Briand et al. 2002, Cabral et al. 2010]. Então, considerando-se que c_i e c_j são duas classes acopladas, essas medidas de acoplamento são definidas como segue:

- Acoplamento de atributos (A) = Número de atributos localmente declarados em c_j quando referências ou apontadores para uma instância de c_j aparecem na lista de argumentos de alguns métodos de c_i, como o tipo de seu valor de retorno, na lista de atributos de c_i, ou como parâmetros locais de métodos de c_i (adaptado de [Briand et al. 2002]). Assim como no experimento realizado por [Briand et al. 2002], no caso de herança o número de atributos herdados das classes ancestrais não foi contado.
- Acoplamento de Métodos (M) = Número de métodos (inclusive construtores) localmente declarados em c_j que são invocados por métodos de c_i (adaptado de [Briand et al. 2002])[1].
- Número de tipos de retorno diferentes (Rd) = Número de tipos de retorno distintos dos métodos localmente declarados em c_j que são invocados por métodos de c_i [Abdurazik and Offutt 2006]. Retornos do tipo *void* não são contados como tipo de retorno uma vez que este tipo representa a ausência de retorno[1].
- Número de tipos de parâmetros diferentes (Pd) = Número de tipos de parâmetros distintos dos métodos localmente declarados em c_j que são invocados por métodos de c_i [Abdurazik and Offutt 2006]. Quando há sobrecarga de métodos, o número de parâmetros equivale ao total de tipos de parâmetros diferentes entre todas as implementações de cada método sobrecarregado. Desta forma, considera-se o pior caso representado por situações nas quais o acoplamento se daria pela chamada de todas as opções de sobrecarga de um dado método[1].

Os dados de entrada de cada MOEA são formados por cinco matrizes que são lidas de um arquivo de texto e são: uma (1) matriz de dependência entre as classes e outras quatro correspondentes a cada uma das métricas descritas acima; (2) matriz de acoplamento de atributos (métrica A); (3) matriz de acoplamento de métodos (métrica M); (4) matriz de diferentes tipos de retornos (métrica Rd); e (5) matriz de diferentes tipos de parâmetros (métrica Pd). A matriz de dependência é representada por um vetor de duas dimensões que contém o tipo de dependência entre as classes. Com base nessa matriz são definidas as restrições para cada um dos algoritmos. A restrição que deve ser considerada durante o processamento do algoritmo é não quebrar as dependências relacionadas a herança[2] [Briand et al. 2002]. As métricas de acoplamento A, M, Rd e Pd são usadas para calcular o *fitness* de cada solução, sendo que a soma das dependências entre as classes de cada métrica corresponde a um objetivo, e para reduzir o custo dos *stubs* deve-se minimizar todos os objetivos (métricas).

[1]Nas medidas M, Rd e Pd, em caso de herança, também são contados os métodos invocados herdados das classes ancestrais, já que a sobrecarga de métodos pode influenciar a complexidade de criação de *stubs*.

[2]Diferente do trabalho de Briand et al [Briand et al. 2002], consideramos que dependências relacionadas a Agregação podem ser quebradas e, por isso, elas não são consideradas restrições.

4. Descrição do Estudo Experimental

O estudo experimental realizado envolve a aplicação dos MOEAs NSGA-II e SPEA2, que foram implementados usando as versões disponíveis no *framework* JMetal [Durillo et al. 2010].

Foram utilizados quatro sistemas OO na realização do estudo experimental: MyBatis, JHotDraw, BCEL e JBoss. A Tabela 1 contém informações sobre os sistemas como número de classes e de linhas de código (LOC). MyBatis é um *framework* de mapeamento de classes de aplicações OO em base de dados relacionais. JHotDraw é um *framework* de gráficos bidimensionais para editores de desenho escritos em Java. Neste trabalho foi utilizado somente o pacote org.jhotdraw.draw da versão 7.5.1. JBoss AS é um servidor de aplicações Java. Neste trabalho foi utilizado somente o subsistema Management da versão 6.0.0M5. O último sistema é o BCEL (*Byte Code Engineering Library*), uma biblioteca que possibilita aos seus usuários analisar, criar e manipular binários Java. Somente o pacote org.apache.bcel.classfile da versão 5.0 foi utilizado no experimento.

Tabela 1. Sistemas utilizados no estudo experimental

Software	Versão	Classes	Dependências	LOC	Disponível em:
BCEL	5.0	45	289	2999	http://archive.apache.org/dist/jakarta/bcel/old/v5.0/
JBoss	6.0.0M5	150	367	8434	http://www.jboss.org/jbossas/downloads.html
JHotDraw	7.5.1	197	809	20273	http://sourceforge.net/projects/jhotdraw/
MyBatis	3.0.2	331	1271	23535	http://code.google.com/p/mybatis/downloads/list

Considerando a dificuldade em obter documentação arquitetural de sistemas complexos para usar no experimento, optou-se por desenvolver um *parser* para realizar a engenharia reversa do código dos sistemas obtidos a fim de identificar os tipos de relacionamentos existentes entre as classes. O *parser* desenvolvido teve como base o software AJATO[3] (*AspectJ and Java Assessment Tool*).

Para determinar os parâmetros a serem utilizados nos dois MOEAs utilizou-se uma estratégia de calibragem formada por três passos que incluem a definição: (i) do tamanho da população e do número máximo de avaliações; (ii) da taxa de cruzamento; e (iii) da taxa de mutação. Em todos os passos, os MOEAs foram executados quatro vezes com os mesmos parâmetros a fim de ter uma visão mais completa do comportamento de cada algoritmo. Durante esse processo foram utilizadas as matrizes de dependência obtidas para o sistema BCEL [Briand et al. 2002] que foi utilizado previamente em um contexto similar [Briand et al. 2002, Cabral et al. 2010]. A Tabela 2 mostra os valores adotados para o estudo experimental. O número de avaliações de *fitness* foi utilizado como critério de parada. Com isso, é possível comparar as soluções obtidas com custo computacional similar. Além disso, eles foram executados em um mesmo computador servidor e o tempo de execução foi registrado.

A fim de conhecer o comportamento de cada algoritmo para o problema CITO, cada MOEA foi executado 30 rodadas para cada sistema. Em cada rodada, cada algoritmo encontra um conjunto aproximado de soluções denominado PF_{approx}. Além disso, para cada MOEA, obtém-se o conjunto PF_{known} que é formado pelas soluções encontradas por cada algoritmo nas 30 rodadas, eliminando-se as repetidas e dominadas. Como

[3]AJATO disponível em: http://www.teccomm.les.inf.puc-rio.br/emagno/ajato/

Tabela 2. Parâmetros para o NSGA-II e SPEA2

Parâmetro	NSGA-II	SPEA2
Tamanho da População	300	300
Número de avaliações de *fitness*	20000	20000
Taxa de mutação	0,02	0,02
Taxa de Cruzamento	0,95	0,95
Tamanho do arquivo	-	250

neste caso, o conjunto Pareto real (PF_{true}) não é conhecido, no experimento o PF_{true} foi obtido através da união de todos os PF_{approx} encontrados por cada MOEA, eliminando-se as soluções dominadas e repetidas [Zitzler et al. 2003]. Os conjuntos PF_{true} e PF_{known} foram utilizados na análise e comparação dos algoritmos NSGA-II e SPEA2, realizadas por meio de três indicadores de qualidade: Distância Geracional(GD), Distância Geracional Inversa(IGD) e Distância Euclidiana a Solução Ideal(ED).

O indicador GD [van Veldhuizen and Lamont 1999] é usado para calcular a distância entre um conjunto de Pareto encontrado, o Pareto Conhecido (PF_{known}) em relação ao PF_{true}, ou seja, é uma medida de erro na qual verifica-se o quão distante está um ponto encontrado do seu correspondente mais próximo no conjunto PF_{true}. O IGD [Radziukyniene and Zilinskas 2008] é baseado no GD, porém com objetivo de calcular a distância do conjunto PF_{true} em relação ao conjunto PF_{known}, ou seja, observa-se o inverso da GD. Após obter os resultados dos indicadores GD e IGD, foi executado o teste pareado de Wilcoxon [Gárcia et al. 2009] de modo a verificar se os MOEAs são considerados estatisticamente iguais ($p - value \geq 0,05$).

O indicador Distância Euclidiana a Solução Ideal (ED) consiste em determinar os melhores valores para cada um dos objetivos entre todas as soluções de PF_{true}, encontrando um ponto considerado como uma solução ideal para o problema, então calcula-se a partir deste ponto a distância euclidiana em relação a todas as soluções de PF_{known}. Este indicador tem como base a técnica de classificação *Compromise Programming* [Cochrane and Zeleny 1973] que é usada em otimização multiobjetivo como técnica de apoio ao tomador de decisão para selecionar uma solução entre as várias encontradas.

5. Resultados e Análise

A Tabela 3 apresenta alguns resultados do experimento como a cardinalidade de PF_{true}, o número de soluções encontradas por cada um dos MOEAs e seu tempo de execução. A segunda coluna contém a cardinalidade de PF_{true} após todas as 30 execuções dos MOEAs. Na quarta coluna apresenta-se a cardinalidade média de PF_{approx} e entre parênteses o total de diferentes soluções de PF_{approx} retornadas por cada algoritmo. A partir destes resultados é possível estimar a complexidade do problema CITO para cada sistema. JBoss e BCEL são mais simples, uma vez que para estes sistemas, os dois algoritmos encontraram todas as soluções em PF_{true} em quase todas as execuções. Entretanto, JHotDraw é mais complexo, pois para ele, os algoritmos encontraram algumas das soluções em PF_{true} em cada execução e o SPEA2 encontrou um maior número de diferentes soluções do que o NSGA-II. MyBatis tem uma PF_{true} com maior cardinalidade que os outros sistemas e novamente o SPEA2 encontrou um maior número de diferentes soluções do que o NSGA-II. Apesar disso, é possível perceber que o SPEA2 requer um maior tempo

de execução com um maior desvio padrão do que o NSGA-II (quinta e sexta colunas da Tabela 3).

Os dois algoritmos encontraram uma única e igual solução para o sistema JBoss com os seguintes valores para cada objetivo: A = 10, M = 6, Rd = 2 e Pd = 9. Então, esta solução envolve criar *stubs* para simular dependências de 10 atributos e 6 métodos com 9 diferentes tipos de retorno e 2 diferentes tipos de parâmetro. Apesar de alcançar uma solução ótima, as soluções iniciais geradas pelos algoritmos multiobjetivos não são ótimas. Inicialmente os valores de *fitness* são altos e ao longo das gerações eles conseguem baixar o *fitness* até alcançar a solução supracitada.

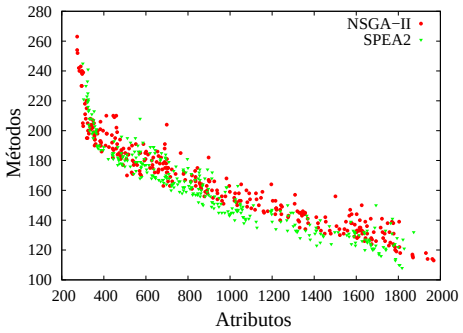
Tabela 3. Número de soluções não dominadas

Sistema	# PF_{true}	MOEA	Número de Soluções	Tempo de Execução(ms)	Desvio Padrão
BCEL	37	NSGA-II	36,8 (37)	1925,40	76,06
		SPEA2	36,4 (37)	18102,23	2184,92
JBoss	1	NSGA-II	1,0 (1)	8023,53	194,16
		SPEA2	1,0 (1)	269933,13	60335,63
JHotDraw	15	NSGA-II	8,0 (8)	13386,83	230,28
		SPEA2	7,6 (14)	22259,20	2790,60
MyBatis	612	NSGA-II	190,2 (315)	34558,13	404,42
		SPEA2	175,8 (326)	49626,30	486,18

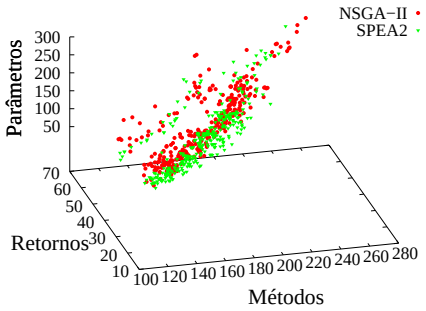
O número de classes, dependências e ciclos influencia a cardinalidade da PF_{Approx} obtida pelos algoritmos. Por exemplo, JBoss tem 150 classes, 367 dependências e 8 ciclos, e para ele os dois MOEAs alcançaram uma única solução. Por outro lado, NSGA-II e SPEA2 obtiveram 37 soluções para o BCEL que tem 45 classes, 289 dependências e 416.091 ciclos. Neste caso, o número de ciclos tem a maior influência sobre o número de soluções encontradas. Mas é possível notar que os dois sistemas que possuem mais relações de dependências, JHotDraw e MyBatis, são mais complexos para serem resolvidos. Entretanto, a respeito do JHotDraw, parece que as métricas não são altamente interdependentes e conflitantes já que o número de soluções alcançadas pelos MOEAs é menor. Pode ser que os métodos do JHotDraw sejam menos acoplados ou mais simples que os métodos do MyBatis.

Na Figura 2 são apresentados os gráficos que representam as soluções no espaço de busca para o sistema MyBatis. Apesar de as soluções apresentarem quatro dimensões, correspondentes aos quatro objetivos, somente é possível apresentar duas ou três dimensões, portanto decidiu-se por agrupar as medidas mais relacionadas: A e M na Figura 2(a) e M, Rd e Pd na Figura 2(b).

Para calcular os indicadores GD e IGD utiliza-se: o conjunto PF_{known} para cada algoritmo e o conjunto PF_{true}, formado por todas as soluções encontradas pelos dois MOEAs nas 30 rodadas, eliminando-se as repetidas e dominadas. Pela Tabela 4 pode-se observar que os algoritmos apresentam resultados bem próximos para os indicadores GD e IGD. Inclusive o teste de Wilcoxon retornou os $p - values$: 0.5229 para o BCEL, 0.9824 para o JHotDraw, e 0.2601 para o MyBatis, referente ao GD, o que indica que não existe diferença significativa para nenhum sistema. Da mesma forma, no indicador IGD, os $p - values$ retornados pelo teste de Wilcoxon não indicam diferença significativa entre o NSGA-II e o SPEA2 (BCEL= 0.08778, JHotDraw= 0.7117 e MyBatis= 0.1304).



(a) Métricas A e M



(b) Métricas M, Rd e Pd

Figura 2. Soluções no espaço de busca para o sistema MyBatis

Tabela 4. Média e Desvio Padrão dos indicadores GD e IGD

Indicador	Sistema	NSGA-II		SPEA2	
		Média	Desvio Padrão	Média	Desvio Padrão
GD	BCEL	0,0049	0,0013	0,0047	0,0010
	JBoss	0	0	0	0
	JHotDraw	0,5414	0,3912	0,5813	0,5512
	MyBatis	0,0086	0,0015	0,0092	0,0019
IGD	BCEL	0,0051	0,0017	0,0045	0,0018
	JBoss	0	0	0	0
	JHotDraw	0,6442	0,8970	0,6106	0,5378
	MyBatis	0,0138	0,0027	0,0122	0,0034

Para o cálculo da ED o custo da solução ideal para cada sistema é apresentado na segunda coluna da Tabela 5. Esta tabela também apresenta a menor distância encontrada e seu respectivo *fitness* para os sistemas usados no experimento. A menor distância alcançada encontra-se destacada em negrito. O NSGA-II alcançou as soluções que tem a menor ED a partir das soluções ideais para o JHotDraw e o MyBatis. Para os outros dois sistemas ambos MOEAs alcançaram a mesma ED.

Tabela 5. Custo da Solução Ideal e menor ED

Sistema	Custo da Solução Ideal	MOEA	Menor ED	*Fitness* da solução de menor ED
BCEL	45, 24, 0, 96	NSGA-II	**32,186**	64, 39, 15, 111
		SPEA2	**32,186**	64, 39, 15, 111
JBoss	10, 6, 2, 9	NSGA-II	**0**	10, 6, 2, 9
		SPEA2	**0**	10, 6, 2, 9
JHotDraw	28, 11, 1, 13	NSGA-II	**5,099**	28, 12, 1, 18
		SPEA2	5,744	30, 13, 1, 18
MyBatis	221, 100, 19, 83	NSGA-II	**195,0**	319, 195, 49, 219
		SPEA2	202,309	330, 214, 45, 207

A solução de menor ED alcançada pelo NSGA-II para o sistema JHotDraw tem o menor valor para dois dos quatro objetivos (A e Rd), enquanto a solução de menor ED gerada pelo SPEA2 obtém o valor mínimo somente para o objetivo Rd. É interessante observar que no caso do BCEL e do MyBatis nenhum dos MOEAs alcançou os valores mínimos para todos os objetivos em suas soluções. Em relação ao JBoss, os dois algoritmos alcançaram todos os objetivos mínimos já que encontraram a solução ideal.

NSGA-II tem soluções mais perto da solução ideal do que o SPEA2 para os sistemas JHotDraw e MyBatis. Apesar disso, não há diferença entre eles em termos de distribuição das soluções, exceto para o JHotDraw, no qual o SPEA2 é melhor. O gráfico da Figura 3 mostra o número de soluções encontradas por ambos MOEAs em relação a ED para o JHotDraw. Nota-se que, apesar de o NSGA-II ter obtido a solução de menor ED, o SPEA2 é o melhor já que ele tem mais soluções com baixas EDs.

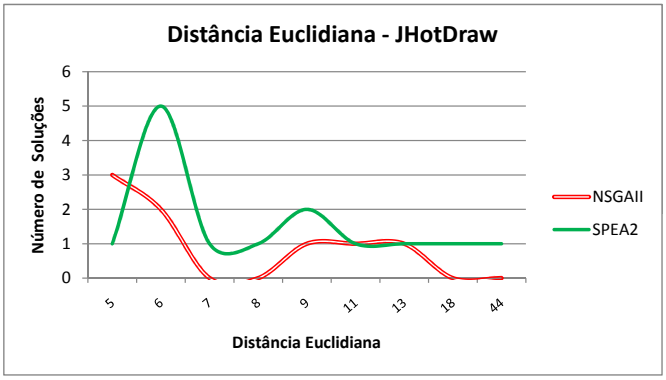


Figura 3. Número de soluções X Distância Euclidiana - JHotDraw

Logo, NSGA-II e SPEA2 tem um desempenho bem parecido para os sistemas BCEL e MyBatis neste indicador de qualidade. Mas, apesar de o NSGA-II ter alcançado as soluções de menor ED para os outros dois sistemas, a distribuição de soluções em termos de ED aponta que o SPEA2 tem o melhor desempenho para o JHotDraw.

Em termos de aplicação prática, as soluções representam um bom compromisso entre as medidas de acoplamento utilizadas. Isso pode ser ilustrado analisando o custo das soluções S1 (326, 220, 54, 286) e S2 (326, 221, 51, 275), para as métricas (A, M, Rd, Pd), obtidas pelo SPEA2 para o MyBatis. O uso das medidas tradicionais A e M indicaria que S1 é melhor que S2 já que M é menor. No entanto, o uso das métricas Rd e Pd mostra que, apesar de S2 precisar de 1 método a mais, ela requer 3 tipos de retorno e 11 tipos de parâmetros a menos que S1, implicando em um menor esforço para a criação de *stubs*.

Quando os MOEAs são usados, o testador pode usufruir da variedade de soluções encontradas de acordo com suas preferências e necessidades. O testador pode conduzir o teste de integração priorizando alguma das métricas de acoplamento e a decisão sobre a priorização de objetivos pode ser influenciada por diferentes fatores, tais como restrições do negócio, econômicas ou contratuais.

6. Conclusões

Neste trabalho foi analisado o emprego de dois MOEAs, NSGA-II e SPEA2, para resolver o problema CITO. Foi realizado um estudo experimental para minimizar a criação de *stubs* em quatro sistemas reais: BCEL, JBoss, JHotDraw e MyBatis. Além disso foram utilizados quatro objetivos que aumentam a complexidade da criação de *stubs*: acoplamento de atributos, acoplamento de métodos, número de diferentes tipos de retorno e número de diferentes tipos de parâmetros.

Os dois MOEAs alcançaram uma única solução para o sistema JBoss, indicando que neste caso os objetivos não estão em conflito. No caso do BCEL e do MyBatis, os

três indicadores apontaram equivalência entre os dois MOEAs. Quando se considera o sistema JHotDraw, os dois MOEAs são equivalentes em termos de GD e IGD, mas o SPEA2 obteve a melhor distribuição de soluções no indicador ED.

Considerando todos os sistemas, não é possível destacar um dos MOEAs como o melhor. Os dois se mostraram adequados e eficazes para os sistemas do experimento. A partir dos resultados é possível afirmar que o SPEA2 tem uma convergência ligeiramente melhor que a do NSGA-II. O SPEA2 cobre toda a fronteira PF_{Approx} e também apresenta uma boa distribuição de soluções nas regiões próximas à solução ideal. E como frequentemente os tomadores de decisão preferem soluções próximas à solução ideal, neste caso, poderia-se escolher o SPEA2. Futuramente, pode-se utilizar outros indicadores na tentativa de evidenciar a superioridade de algum dos algoritmos.

A resposta para a questão de pesquisa é sim, os dois algoritmos são eficazes para tratar o problema CITO usando quatro objetivos, já que eles alcançaram resultados similares apesar de usarem diferentes estratégias na resolução de problemas multiobjetivos. Os resultados indicam que a abordagem multiobjetivo apresenta uma variedade de boas soluções especialmente em sistemas complexos.

Futuramente, pretende-se realizar experimentos utilizando outros sistemas a fim de confirmar as evidências identificadas neste trabalho. Além disso, seria interessante analisar o desempenho de outros MOEAs nesse mesmo contexto.

7. Agradecimentos

Os autores agradecem a Edison Klafke Fillus por sua contribuição no experimento. Este trabalho foi apoiado pela CAPES/Reuni e pelo CNPq.

Referências

Abdurazik, A. and Offutt, J. (2006). Coupling-based Class Integration and Test Order. In *Proceedings of The 2006 International Workshop on Automation of Software Test*, pages 50–56.

Bansal, P., Sabharwal, S., and Sidhu, P. (2009). An investigation of strategies for finding test order during integration testing of object oriented applications. In *Proceedings of International Conference on Methods and Models in Computer Science 2009*, pages 1–8. IEEE.

Briand, L. and Labiche, Y. (2003). An investigation of graph-based class integration test order strategies. *IEEE Transactions on Software Engineering*, 29(7):594–607.

Briand, L. C., Feng, J., and Labiche, Y. (2002). *Experimenting with Genetic Algorithms and Coupling Measures to Devise Optimal Integration Test Orders*. Carleton University, Technical Report SCE-02-03.

Cabral, R., Pozo, A., and Vergilio, S. (2010). A Pareto Ant Colony Algorithm Applied to the Class Integration and Test Order Problem. In *22nd IFIP International Conference on Testing Software and Systems (ICTSS'10)*. Springer.

Cochrane, J. and Zeleny, M. (1973). *Multiple Criteria Decision Making*. University of South Carolina Press, Columbia.

Coello, C. A. C., Lamont, G. B., and Veldhuizen, D. A. V. (2006). *Evolutionary Algorithms for Solving Multi-Objective Problems (Genetic and Evolutionary Computation)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation*, 6(2):182 –197.

Durillo, J., Nebro, A., and Alba, E. (2010). The jMetal framework for multi-objective optimization: Design and architecture. In *2010 IEEE Congress on Evolutionary Computation(CEC)*, pages 4138–4325, Barcelona, Spain.

Gárcia, S., Molina, D., Lozano, M., and Herrera, F. (2009). A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC'2005 Special Session on Real Parameter Optimization. *Journal of Heuristics*, 15(6):617–644.

Knowles, J., Thiele, L., and Zitzler, E. (2006). A tutorial on the performance assessment of stochastic multiobjective optimizers. Technical report, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Switzerland. Revised version.

Kung, D. C., Gao, J., Hsia, P., Lin, J., and Toyoshima, Y. (1995). Class firewal, test order and regression testing of object-oriented programs. *J. of Object-Oriented Progr.*, 8(2).

Melton, H. and Tempero, E. (2007). An empirical study of cycles among classes in Java. *Empirical Software Engineering*, 12:389–415.

Pareto, V. (1927). *Manuel D'Economie Politique*. Ams Press, Paris.

Pozo, A., Bertoldi, G., Árias, J., Cabral, R., and Vergilio, S. (2011). Multi-objective optimization algorithms applied to the class integratio n and test order problem. *Software Tools for Technology Transfer*. Submitted.

Radziukyniene, I. and Zilinskas, A. (2008). Evolutionary Methods for Multi-Objective Portfolio Optimization. In *Proceedings of the World Congress on Engineering 2008 Vol II*.

Ré, R. and Masiero, P. C. (2007). Integration testing of aspect-oriented programs: a characterization study to evaluate how to minimize the number of stubs. In *Brazilian Symposium on Software Engineering*, pages 411–426.

Tai, K.-C. and Daniels, F. J. (1997). Test order for inter-class integration testing of object-oriented software. In *21st International Computer Software and Applications Conference*, pages 602–607. IEEE Computer Society.

Traon, Y. L., Jéron, T., Jézéquel, J.-M., and Morel, P. (2000). Efficient object-oriented integration and regression testing. *IEEE Transactions on Reliability*, pages 12–25.

van Veldhuizen, D. A. and Lamont, G. B. (1999). Multiobjective evolutionary algorithm test suites. In *Proceedings of the 1999 ACM symposium on Applied computing*, SAC '99, pages 351–357, New York, NY, USA. ACM.

Zitzler, E., Laumanns, M., and Thiele, L. (2001). SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Technical Report 103, Gloriastrasse 35, CH-8092 Zurich, Switzerland.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and da Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7:117–132.

Programação Genética Linear com Inspiração Quântica

Douglas Mota Dias[1], **Marco Aurélio C. Pacheco**[1]

[1]Departamento de Engenharia Elétrica
Pontificia Universidade Católica do Rio Janeiro (PUC-Rio)
Rio de Janeiro – RJ – Brasil

{douglasm,marco}@ele.puc-rio.br

Abstract. *Quantum-inspired evolutionary algorithms (QIEAs) exploit principles of quantum mechanics to improve the performance of classical evolutionary algorithms. This paper presents a new model of QIEA ("Quantum-Inspired Linear Genetic Programming" – QILGP) to evolve machine code programs. Symbolic regression problems and the current more efficient model to evolve machine code (AIMGP) are used in comparative tests. QILGP outperforms it, obtaining better solutions with fewer evaluations, parameters and operators. One conclude that the quantum inspiration can be a competitive approach to evolve programs more efficiently.*

Resumo. *Algoritmos evolutivos com inspiração quântica (AEIQ) aproveitam princípios da mecânica quântica para melhorar o desempenho de algoritmos evolutivos clássicos. Este artigo apresenta um novo modelo de AEIQ ("Programação Genética Linear com Inspiração Quântica" – PGLIQ) para evoluir programas em código de máquina. Nos testes comparativos são utilizados problemas de regressão simbólica e o modelo atual mais eficiente na evolução de código de máquina (AIMGP). A PGLIQ apresenta desempenho superior, obtendo melhores soluções com menos avaliações, parâmetros e operadores. Conclui-se que a inspiração quântica pode ser uma abordagem competitiva para se evoluir programas mais eficientemente.*

1. Introdução

Em sua definição mais simples, a Programação Genética (PG) é uma técnica de Computação Evolutiva (CE) que permite a computadores resolverem problemas sem que sejam explicitamente programados para tal [Koza 1992]. Ou seja, a PG parte de uma declaração de alto nível sobre "o que se necessita ser feito" e cria automaticamente um programa de computador para resolver o problema.

Diferentes tipos de PG existentes evoluem diferentes tipos de programas e os representam por diferentes tipos de estruturas. Este artigo se baseia na PG Linear (PGL), que possui esta denominação por representar seus programas em estruturas lineares (listas). A PGL é utilizada para a evolução de programas em linguagens imperativas (p.ex. C [Brameier and Banzhaf 2007] e linguagem de montagem [Dias et al. 2006]) ou em código de máquina.

Desde sua concepção e formalização [Koza 1992], a PG vem sendo utilizada para resolver uma ampla variedade de problemas práticos, produzindo resultados competitivos

com os de especialistas humanos, incluindo novas descobertas científicas e invenções patenteáveis [Koza 2010].

Entretanto, os Algoritmos Evolutivos (AE) apresentam características que podem prejudicar seu desempenho. Como estes algoritmos necessitam avaliar diversas vezes a qualidade das soluções, problemas onde cada avaliação seja computacionalmente custosa podem tornar proibitivo seu uso. Neste sentido, os AE com inspiração quântica (AEIQ) representam um dos mais recentes avanços na CE. Estes algoritmos se baseiam em ideias inspiradas na mecânica quântica, apresentando melhor desempenho em diversos tipos de aplicações. Mais especificamente, um AEIQ usando representação binária foi utilizado com sucesso em problemas de otimização combinatória em [Han and Kim 2002], apresentando resultados superiores em relação aos AGs convencionais, por necessitar de menos avaliações para obter resultados melhores. Em [Abs da Cruz et al. 2010] é apresentado um AEIQ para otimização numérica, inspirado no princípio de múltiplos universos da física quântica e com representação de números reais, que apresenta um tempo de convergência menor para problemas *benchmark* quando comparado com algoritmos convencionais.

Sendo assim, o principal objetivo deste artigo é apresentar um novo modelo de AEIQ, inspirado no conceito de superposição de estados da mecânica quântica, para a evolução de programas, denominado "Programação Genética Linear com Inspiração Quântica" (PGLIQ). Desta forma, também se pretende investigar a capacidade deste paradigma de melhorar o desempenho da PG, mais especificamente, em problemas de regressão simbólica. Uma versão preliminar e simplificada deste modelo foi proposta pelos autores em [Dias and Pacheco 2009], apenas como prova de conceito.

Optou-se pela evolução de programas em código de máquina para o modelo por esta ser a abordagem mais rápida existente quanto à velocidade total de execução [Poli et al. 2008], uma vez que dispensa etapas como interpretação ou compilação dos indivíduos. O modelo de PGL convencional mais bem sucedido na evolução de programas em código de máquina denomina-se AIMGP (*Automatic Induction of Machine Code by Genetic Programming*) [Nordin 1998] e, portanto, é utilizado aqui como referência para fins de comparação.

2. Computação Quântica e Algoritmos com Inspiração Quântica

Em um computador clássico, o *bit* é a menor unidade de informação, podendo assumir os valores 0 ou 1. Em um computador quântico, a unidade de informação básica, chamada de *qubit*, pode assumir os estados $|0\rangle$, $|1\rangle$ ou uma superposição dos dois estados. Esta superposição de dois estados é uma combinação linear dos estados $|0\rangle$ e $|1\rangle$ e pode ser representada por $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$, onde $|\psi\rangle$ é o estado do *qubit* e $|\alpha|^2$ e $|\beta|^2$ são as probabilidades de que os estados 0 ou 1 acarretem da observação (ou medição) do *qubit*, sendo $|\alpha|^2 + |\beta|^2 = 1$, com α e β números complexos.

A superposição de estados oferece aos computadores quânticos um grau de paralelismo incomparável que, se devidamente explorado, permite que estes computadores realizem tarefas impraticáveis em computadores clássicos. Um exemplo é a fatoração inteira de grandes números, onde se deseja encontrar os dois números primos cujo produto seja igual a um número fornecido. Esta tarefa pode levar milhões de anos para ser resolvida em computadores clássicos usando os algoritmos mais sofisticados conhecidos. Em

um computador quântico, no entanto, a mesma tarefa levaria apenas alguns segundos para ser concluída [Lavor et al. 2003].

A dinâmica de um sistema quântico deve levar o sistema de um estado para outro de uma forma que preserve a norma. Por sua vez, qualquer transformação unitária de um espaço de estados quânticos é uma transformação quântica legítima e vice versa. Ou seja, pode-se encarar as transformações como sendo rotações de um vetor complexo. Na computação quântica, os conjuntos de transformações primitivas de estados quânticos são denominados "portas quânticas". Para computadores clássicos existem conjuntos de portas lógicas que são universais, no sentido que qualquer computação clássica pode ser efetuada usando-se uma combinação dessas portas. De forma similar, existem conjuntos universais de portas quânticas para a computação quântica.

De fato, a maioria das abordagens da computação quântica emprega *qubits* codificados em sistemas quânticos de dois níveis. No entanto, os sistemas que codificam informação quântica normalmente têm uma estrutura física muito mais complexa, com diversos graus de liberdade diretamente acessíveis (p.ex. átomos, fótons etc.). Recentemente, começou-se a considerar sistemas quânticos multiníveis, que utilizam o *qudit* como unidade de informação, o qual pode assumir qualquer um de d valores, ou uma superposição de d estados [Lanyon et al. 2008].

Embora a computação quântica ofereça uma boa promessa em termos de capacidade de processamento, atualmente dois problemas a impedem que se torne uma ferramenta útil: a dificuldade de se implementar um computador quântico e a dificuldade de se criar algoritmos que tirem proveito.

No entanto, em [Moore and Narayanan 1995] foi proposta uma nova abordagem. Ao invés de se buscar desenvolver algoritmos para computadores quânticos ou de se tentar viabilizar o uso destes, a ideia de "computação com inspiração quântica" é apresentada. O objetivo desta nova abordagem é criar algoritmos clássicos, ou seja, executáveis em computadores clássicos, que tirem proveito de paradigmas da mecânica quântica, de forma a melhorar o desempenho dos mesmos na resolução de problemas. Ou seja, assim como os AEIQ já citados [Moore and Narayanan 1995, Han and Kim 2002, Abs da Cruz et al. 2010], o modelo aqui proposto se inspira em um fenômeno da mecânica quântica, neste caso, a superposição de estados quânticos, para melhorar o desempenho de um algoritmo clássico: a Programação Genética.

3. Programação Genética Linear com Inspiração Quântica

3.1. Plataforma

A PGLIQ evolui programas para a plataforma Intel x86 [Int 1997], utilizando algumas instruções da sua Unidade de Ponto Flutuante (*Floating Point Unit* – FPU), as quais operam com dados da memória principal (m) e/ou dos 8 registradores da FPU ($ST(i)$, onde $i = 0, 1, \ldots, 7$). Neste modelo, são utilizadas instruções para adição, subtração, multiplicação, divisão e transferência de dados, assim como instruções aritméticas e trigonométricas. O modelo representa internamente estas instruções por um "*token* de função" (TF), que pode assumir valores inteiros de 0 a $(f - 1)$, de forma a representar de maneira única cada uma das f funções da PGLIQ, as quais compõem o conjunto de funções para os estudos de caso deste artigo. A tabela 1 mostra as instruções citadas, suas respectivas

funcionalidades, o respectivo argumento de cada uma, caso haja, e o valor do *token* destas instruções.

Tabela 1. Descrição das instruções.

Instrução	Descrição	Argumento	*token*
NOP	Nenhuma operação	-	0
FADD m	$ST(0) \leftarrow ST(0) + m$	m	1
FADD ST(0), ST(i)	$ST(0) \leftarrow ST(0) + ST(i)$	i	2
FADD ST(i), ST(0)	$ST(i) \leftarrow ST(i) + ST(0)$	i	3
FSUB m	$ST(0) \leftarrow ST(0) - m$	m	4
FSUB ST(0), ST(i)	$ST(0) \leftarrow ST(0) - ST(i)$	i	5
FSUB ST(i), ST(0)	$ST(i) \leftarrow ST(i) - ST(0)$	i	6
FMUL m	$ST(0) \leftarrow ST(0) \times m$	m	7
FMUL ST(0), ST(i)	$ST(0) \leftarrow ST(0) \times ST(i)$	i	8
FMUL ST(i), ST(0)	$ST(i) \leftarrow ST(i) \times ST(0)$	i	9
FXCH ST(i)	$ST(0) \leftrightarrows ST(i)$ (troca conteúdos)	i	A
FDIV m	$ST(0) \leftarrow ST(0) \div m$	m	B
FDIV ST(0), ST(i)	$ST(0) \leftarrow ST(0) \div ST(i)$	i	C
FDIV ST(i), ST(0)	$ST(i) \leftarrow ST(i) \div ST(0)$	i	D
FABS	$ST(0) \leftarrow \lvert ST(0) \rvert$	-	E
FSQRT	$ST(0) \leftarrow \sqrt{ST(0)}$	-	F
FSIN	$ST(0) \leftarrow \operatorname{sen} ST(0)$	-	G
FCOS	$ST(0) \leftarrow \cos ST(0)$	-	H

Um programa em código de máquina evoluído pela PGLIQ representa uma solução e lê os dados de entrada na memória principal, os quais são compostos pelas variáveis de entrada do problema e, opcionalmente, constantes fornecidas pelo usuário. Estas entradas podem ser representadas por um vetor, como por exemplo:

$$I = (V[0], V[1], 1, 2, 3), \tag{1}$$

onde $V[0]$ e $V[1]$ contêm os valores das duas entradas do problema exemplificado, e onde 1, 2 e 3 são os valores das três constantes. Basicamente, o modelo AIMGP é implementado da mesma forma pelo *software* comercial Discipulus™ [Francone 2010], que é utilizado, portanto, para a execução dos experimentos comparativos deste artigo.

3.2. Representação

A PGLIQ é baseada nas seguintes entidades, da seguinte forma: o cromossomo de um "indivíduo quântico", que representa a superposição de todos os programas possíveis no espaço de busca predefinido, é observado para gerar o cromossomo de um "indivíduo clássico" (IC), a partir do qual um programa em código de máquina é finalmente gerado. Uma vez que as instruções utilizadas têm apenas um ou nenhum argumento, todas as funções têm apenas um terminal que é representado por um "*token* de terminal" (TT). No caso de uma função sem terminal, o seu *token* de terminal correspondente é ignorado pelo modelo.

Um programa é internamente representado pelo cromossomo do IC, representado por uma estrutura com $(L \times 2)$ *tokens*, conforme mostrado na porção direita da figura 1,

onde: L é o comprimento máximo do programa (em número de instruções); cada linha representa uma instrução i ($1 \leq i \leq L$) e é definida como um "gene"; a coluna da esquerda contém os valores dos *tokens* de função (TF_i) e a da direita, os valores dos seus respectivos *tokens* de terminal (TT_i). A ordem de execução do programa é da primeira até a última linha.

Apesar do cromossomo do IC possuir comprimento fixo, o programa efetivo que ele representa possui comprimento variável, assim como na PGL e na AIMGP. No caso da PGLIQ, esta variação é obtida pela inclusão da instrução `NOP` no conjunto de instruções. Ocorre que o processo de geração de código ignora qualquer gene no qual um `NOP` esteja presente, ou seja, todo gene cujo valor do seu *token* de função valha zero.

O modelo aqui apresentado é inspirado em sistemas quânticos multiníveis [Lanyon et al. 2008]. Portanto, conforme mostrado na seção 2, a unidade básica de informação adotada pela PGLIQ é o *qudit*. Esta informação pode ser descrita por um vetor de estado em um sistema mecânico quântico de d níveis, o qual equivale a um espaço vetorial d-dimensional, onde d é o número de estados nos quais o *qudit* pode ser medido. Isto é, d representa a cardinalidade do *token* que terá seu valor determinado pela observação do seu respectivo *qudit*. O estado de um *qudit* é uma superposição linear dos d estados e pode ser representado por $|\psi\rangle = \sum_{i=0}^{d-1} \alpha_i |i\rangle$, onde $|\alpha_i|^2$ é a probabilidade de que o *qudit* seja observado no estado i. A normalização unitária garante: $\sum_{i=0}^{d-1} |\alpha_i|^2 = 1$.

Tome-se como exemplo um conjunto de funções composto pelas 11 primeiras instruções da tabela 1. Ao se observar um *qudit* de função cujo estado é dado pela equação 2, tal que:

$$|\psi\rangle = \frac{1}{\sqrt{5}} |0\rangle + \frac{1}{\sqrt{4}} |1\rangle + \frac{1}{\sqrt{10}} |2\rangle + \cdots + \frac{1}{\sqrt{8}} |\mathrm{A}\rangle , \tag{2}$$

a probabilidade de se medir a instrução `NOP` (estado '0') é $(1/5)^2 = 0,200$, para `FADD m` (estado '1') é $(1/4)^2 = 0,250$, para `FADD ST(i)` (estado '2') é $(1/10)^2 = 0,100$ e assim por diante, até que finalmente a probabilidade de se medir a instrução `FXCH ST(i)` (estado 'A') é $(1/5)^2 = 0,125$.

O cromossomo do IQ, denominado "cromossomo quântico", é representado por uma lista de estruturas denominadas "genes quânticos". Um gene quântico é composto por um *qudit* de função (QF), que representa a superposição de todas as funções predefinidas pelo conjunto de funções. Também possui dois *qudits* de terminal (QT), uma vez que as funções podem utilizar dois tipos diferentes de terminais. Um dos *qudits* de terminal representa os registradores (QT_{Reg}) e o outro, as posições de memória (QT_{Mem}), sendo que estes registradores e posições de memória pertencem a um conjunto de terminais predefinido. Por exemplo, a instrução `FADD ST(0),ST(i)` utiliza um *qudit* de terminal que representa a superposição dos índices i dos registradores $ST(i)$, enquanto que o *qudit* de terminal da instrução `FADD m` representa a superposição das posições m da memória. Como cada gene quântico é observado para gerar um gene do IC (uma instrução completa), o cromossomos de ambos os indivíduos, quântico (IQ) e clássico (IC), possuem o mesmo comprimento, conforme mostra a figura 1.

Pode-se ilustrar o processo de criação de um gene pela observação de um gene quântico, a partir de um exemplo baseado na equação 2 e no vetor de caso $I = (V[0], V[1], 1, 2, 3)$, definido em (1), conforme os três seguintes passos básicos:

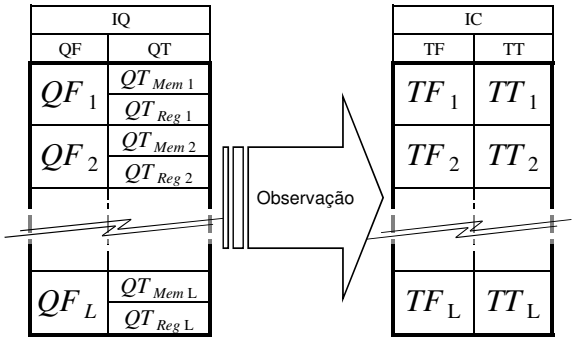


Figura 1. Criação de um indivíduo clássico (IC) pela observação de um indivíduo quântico (IQ).

1. O *qudit* de função (QF) é observado e o valor resultante (p.ex. 7) é atribuído ao *token* de função (TF) do gene.
2. O valor do *token* de função determina o *qudit* de terminal a ser observado, uma vez que cada instrução demanda um tipo diferente de terminal dentre dois: registrador ou memória.
3. O *qudit* de terminal (QT) determinado pelo valor do *token* de função é observado e o valor resultante (p.ex. 1) é atribuído ao *token* de terminal (TT) do gene.

Portanto, neste exemplo, a instrução observada é `FMUL V[1]`, uma vez que '7' é o valor do *token* de função para esta instrução (tabela 1) e '1' é o valor do *token* de terminal que representa $V[1]$ no vetor de entrada I, definido em (1). Neste caso, portanto, os valores de um *token* de terminal de '0' a '4' indicam que o argumento é $V[0]$, $V[1]$, 1, 2 ou 3, respectivamente. Entretanto, se o *token* de terminal for de uma instrução cujo argumento seja um registrador, o valor do *token* de terminal indica diretamente qual dos 8 registradores é o argumento, sendo que apenas os 4 primeiros são utilizados neste exemplo: $ST(0)$ a $ST(3)$.

Cada programa em código de máquina evoluído pela PGLIQ, assim como na AIMGP, é composto pelos três seguintes segmentos:

- Cabeçalho (*header*): inicia a FPU e carrega o valor zero em todos os registradores.
- Corpo: é o código evoluído em si (único segmento afetado pela evolução).
- Rodapé (*footer*): transfere o conteúdo de $ST(0)$ para $V[0]$ (saída predefinida do programa) e reinicia a FPU. Em seguida, executa a instrução de retorno para encerrar a execução do programa e retornar ao fluxo principal do algoritmo evolutivo.

3.3. Avaliação de um Indivíduo Clássico

O processo se inicia com a geração de um programa em código de máquina a partir do IC a ser avaliado, onde seu cromossomo é percorrido sequencialmente, gene por gene, *token* por *token* (de função e de terminal), gerando serialmente o código de máquina do corpo do programa relativo àquele IC, cuja avaliação compreende a execução do seu código de máquina para todos os casos de aptidão do problema (todos os dados de um conjunto de avaliação). Quando instruções `FDIV` incorrem em divisão por zero ou instruções `FSQRT` incorrem em raiz quadrada de número negativo, o que é detetado quando $ST[0]$ apresenta um valor inválido, atribui-se o valor nulo a $V[0]$ como resultado da avaliação daquele caso de aptidão (mesma abordagem adotada pela AIMGP).

3.4. Operador Quântico

O operador quântico aqui proposto atua diretamente nas probabilidades p_i de um *qudit*, satisfazendo a condição de normalização: $\sum_{i=0}^{d-1} |\alpha_i|^2 = 1$, onde d é a cardinalidade do *qudit* e $|\alpha_i|^2 = p_i$. Sendo assim, este operador, aqui denominado "operador P", representa a funcionalidade de uma porta quântica, efetuando rotações no vetor que representa o estado $|\psi\rangle$ de um *qudit* em um espaço vetorial d-dimensional. O operador P funciona em dois passos básicos. Primeiramente, o operador incrementa uma dada probabilidade do *qudit*, da seguinte maneira: $p_i \leftarrow p_i + s(1 - p_i)$, onde s é um parâmetro denominado "tamanho de passo", que pode assumir qualquer valor real entre 0 e 1. O segundo passo é o ajuste dos valores de todas as probabilidades do *qudit* de forma a satisfazer a condição de normalização. Ou seja, o valor da probabilidade p_i de um *qudit* é incrementado pelo operador, de um valor diretamente proporcional ao tamanho de passo s. Sendo assim, o que diferencia um indivíduo quântico de outro são os valores das probabilidades de seus *qudits*, os quais variam distintamente ao longo da evolução.

3.5. Estrutura e Algoritmo Evolutivo

O diagrama na figura 2 ilustra a estrutura do modelo e seu funcionamento básico. A PGLIQ possui uma população composta por uma população quântica e outra clássica, ambas possuindo M indivíduos. Também possui M ICs auxiliares C_i^{obs}, que resultam das observações dos IQs Q_i, onde $1 \leq i \leq M$. No exemplo da figura 2, $M = 4$.

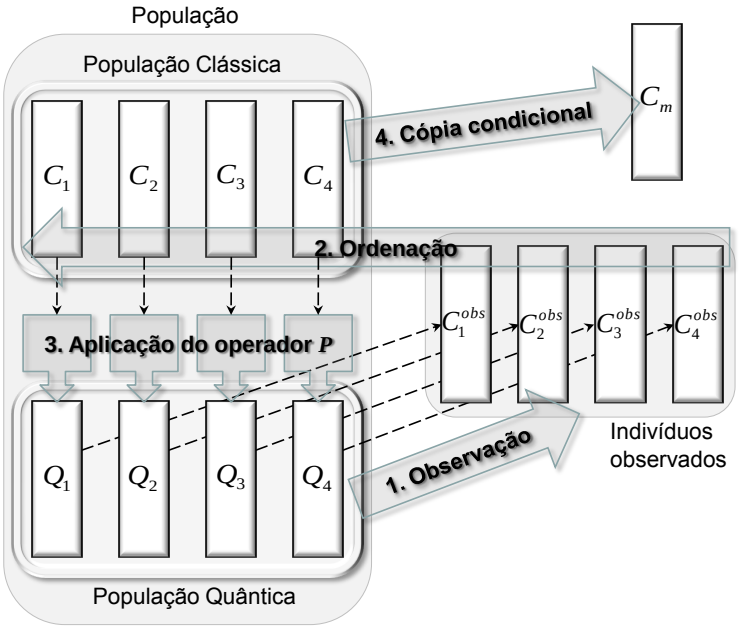


Figura 2. Diagrama descritivo básico do modelo PGLIQ.

A figura 2 também enumera as 4 etapas básicas que caracterizam uma "geração" do modelo. Na etapa 1, cada um dos M IQs é observado uma vez, resultando nos M ICs C_i^{obs}. Na etapa 2, os indivíduos da população clássica e os indivíduos observados são ordenados conjuntamente quanto às suas avaliações. Como resultado, os M melhores ICs dentre os $2M$ avaliados são mantidos na população clássica, ordenados do melhor para o pior, de C_1 para C_M. Na etapa 3, O operador P é aplicado a cada indivíduo quântico Q_i, tomando como referência seus indivíduos C_i correspondentes na população clássica. Na

aplicação do operador em um IQ, as probabilidades dos *qudits*, de função e de terminal, de cada um dos seus L genes GQ_g são ajustadas segundo seus respectivos genes GC_g de um IC (vide seção 3.4), onde $1 \leq g \leq L$. É nesta etapa que ocorre efetivamente a evolução, uma vez que, a cada nova geração, a aplicação do operador aumenta a probabilidade de que a observação dos IQs gerem ICs mais parecidos com os melhores encontrados até então. E na etapa 4, caso algum dos indivíduos da população clássica, avaliados na geração atual, seja melhor que o melhor IC avaliado até então, uma cópia do mesmo é armazenada em C_m, o melhor IC obtido até então.

Cada IQ deve ser iniciado antes de uma execução, iniciando-se cada um dos seus L genes quânticos. Cada *qudit* de terminal é iniciado pela atribuição do mesmo valor $1/t$ a todas as suas probabilidades p_i, onde $0 \leq i < t$, e t é o número de terminais representados pelo *qudit*. A iniciação de um *qudit* de função atribui um valor predeterminado $p_{0,0}$ a p_0, de forma a fornecer um controle de comprimento inicial sobre os primeiros programas evoluídos, uma vez que p_0 é a probabilidade de se observar a instrução NOP, ignorada durante a criação de um programa a partir de um IC. Assim, as demais probabilidades p_i $(1 \leq i < f)$ são iniciadas com o mesmo valor $(1 - p_{0,0})/(f - 1)$, onde f é o número de instruções representadas pelo *qudit*.

De forma a evitar uma convergência prematura, quando uma execução atinge o valor predeterminado do parâmetro $gsm_{máx}$ (número máximo de gerações sem melhoria de aptidão), as populações quântica e clássica são reiniciadas. Este processo elimina o conteúdo dos ICs e reinicia as probabilidades de todos os IQs. Em seguida, C_m é copiado para C_1 como uma semente. Finalmente, o operador P é aplicado uma vez a Q_1, tomando C_1 como referência. Entretanto, o valor de s utilizado aqui é diferente, sendo determinado pelo parâmetro s_r (tamanho do passo de reinício), cujo valor é algumas ordens de grandeza maior que o de s. Este último processo pode ser considerado uma "semeadura quântica".

4. Experimentos

Uma vez que os estudos de caso aqui abordados são de regressão simbólica, a base de dados é composta por três conjuntos. O conjunto de treinamento é o que efetivamente conduz a evolução, enquanto que o de validação serve para verificar constantemente a qualidade da generalização das melhores soluções encontradas durante a evolução. O conjunto de teste, que não é apresentado ao algoritmo durante a evolução, é aplicado ao melhor indivíduo obtido em uma execução completa do algoritmo. Cada experimento é composto por 100 execuções dos modelos, de forma a se obter a média das aptidões dos melhores indivíduos por geração, cujo valor é a média obtida por treinamento e validação, assim como as aptidões dos melhores indivíduos dentre as 100 execuções.

A aptidão de um indivíduo é determinada pela computação do seu erro absoluto médio (EAM) sobre os n casos de aptidão, definido como

$$\text{EAM} = \frac{1}{n} \sum_{i=1}^{n} |t_i - V[0]_i|, \tag{3}$$

onde t_i é o valor esperado (real) do caso i e $V[0]_i$ é o valor da saída do indivíduo para o mesmo caso (aproximação).

4.1. Configuração dos Modelos

A configuração da AIMGP adotada nos experimentos é descrita na tabela 2. Os parâmetros recebem seus valores padrão, conforme recomendado pelo manual do Discipulus™ [Francone 2010]. O número de gerações por execução e o tamanho da população foram escolhidos com base em faixas de valores típicos de experimentos com PG [Brameier and Banzhaf 2007, Poli et al. 2008].

Tabela 2. Configuração do modelo AIMGP para os experimentos.

Parâmetro	Valor
Número de gerações (G)	1.000
Tamanho da população (M)	1.000
Número de *demes*	10
Taxa de migração entre *demes*	1%
Comprimento máximo inicial do programa	20
Comprimento máximo do programa	128
Frequência de mutação	95%
– Taxa de mutação de bloco	30%
– Taxa de mutação de instrução	30%
– Taxa de mutação de dados	40%
Frequência de cruzamento	50%
– Taxa de cruzamento homólogo	95%
Número de registradores da FPU	8

Todos os estudos de caso são executados com e sem o uso de 10 *demes* (ou subpopulações) com 100 indivíduos cada (vide [Francone 2010]). Uma vez que a PGLIQ possui apenas uma população, composta por IQs e ICs, pode-se concluir que a comparação de desempenho entre os modelos é mais neutra ao se considerar a AIMGP sem *demes*, ou seja, também com uma única população.

A configuração da PGLIQ é mostrada na tabela 3, onde os parâmetros da segunda à sexta linha recebem seus melhores valores, determinados a partir de experimentos preliminares.

Tabela 3. Configuração do modelo PGLIQ para os experimentos.

Parâmetro	Valor
Número de gerações (G)	300.000 a 350.000
Tamanho da população (M)	6
Número máximo de gerações sem melhoria ($gsm_{máximo}$)	20.000
Probabilidade inicial de NOP ($p_{0,0}$)	0,9
Tamanho do passo de reinício (s_r)	1,0
Tamanho do passo do operador P (s)	0,004
Comprimento máximo do programa	128
Número de registradores da FPU	8

4.2. Resultados

O estudo de caso "Distância", baseado em um problema utilizado como *benchmark* em [Brameier and Banzhaf 2007], requer que a PG gere um programa que calcule a distância

euclideana entre dois pontos a e b no espaço tridimensional (6 variáveis de entrada), conforme mostra a equação:

$$f_{dist}\left(a_x, b_x, a_y, b_y, a_z, b_z\right) = \sqrt{\left(a_x - b_x\right)^2 + \left(a_y - b_y\right)^2 + \left(a_z - b_z\right)^2}. \quad (4)$$

Os conjuntos de dados são compostos por 300 casos de aptidão cada, onde as coordenadas dos pontos ($\in \mathbb{R}^3$) são aleatórios em $[0,\ 1]$. Diferentemente dos demais estudos de caso, este utiliza um conjunto de funções reduzido, composto pelas instruções: `FADD`, `FSUB`, `FMUL`, `FXCH`, `FABS` e `FSQRT`.

A tabela 4 mostra média e desvio padrão σ relativos às aptidões dos melhores indivíduos por geração, para as 100 execuções de cada modelo. Também mostra o número de indivíduos que, dentre as 100 execuções, representam a solução exata do problema ("Acertos"). As últimas colunas à direita ("Variação") mostram os valores percentuais das reduções dos erros médios obtidos pela PGLIQ sobre a AIMGP. A última coluna, em particular, mostra em quantas vezes o número de acertos obtidos pela PGLIQ supera os obtidos pela AIMGP.

Tabela 4. Comparação do desempenho dos modelos para "Distância".

	AIMGP			PGLIQ			Variação (%)		
Demes	Média	σ	Acertos	Média	σ	Acertos	Média	σ	Acertos
Não	0,111	0,049	2	0,068	0,047	17	-38,8	-4,0	750 (8,5 $\times$)
Sim	0,100	0,050	3	0,069	0,048	17	-30,7	-3,8	470 (5,7 $\times$)

O estudo de caso "Composição Química" tem origem em dados laboratoriais de uma composição química (saída) que, apesar do elevado custo de obtenção, são significativamente ruidosos. Portanto, o objetivo da PG é gerar um programa que relacione estes dados a 57 medições de processo de baixo custo, tais como temperaturas, pressões e vazões (entradas). Baseia-se em dados de uma aplicação industrial real da empresa Dow Chemical, tendo sido utilizado em uma competição de Regressão Simbólica no congresso "EvoStar 2010". A base de dados original [Kordon 2010] é composta por um conjunto de treinamento com 747 amostras e um de teste com 319 amostras. Para os experimentos aqui conduzidos, o conjunto de treinamento original foi dividido: 374 amostras para treinamento e 373 para validação.

A tabela 5 compara o desempenho médio dos modelos, e a tabela 6, o desempenho dos melhores indivíduos obtidos por ambos, mostrando a aptidão dos indivíduos com relação aos três conjuntos de dados. As duas últimas colunas à direita mostram que a PGLIQ é superior à AIMGP para validação e teste, apresentando soluções com erros menores em 7,1 a 11,0%.

Tabela 5. Comparação do desempenho médio para "Composição Química".

	AIMGP		PGLIQ		Variação (%)	
Demes	Média	σ	Média	σ	Média	σ
Não	0.189	0.025	0.174	0.021	-8.1	-14.7
Sim	0.185	0.023	0.176	0.021	-5.2	-11.0

Tabela 6. Comparação dos melhores indivíduos para "Composição Química".

	AIMGP			PGLIQ			Variação (%)		
Demes	Trein.	Valid.	Teste	Trein.	Valid.	Teste	Trein.	Valid.	Teste
Não	0,126	0,131	0,140	0,116	0,121	0,124	-7,9	-7,1	-11,0
Sim	0,120	0,139	0,143	0,126	0,126	0,130	1,9	-9,7	-9,2

O estudo de caso "Pontos Quânticos" se baseia em [Singulani et al. 2008], onde são utilizadas redes neurais para inferir o comportamento da altura média de pontos quânticos obtidos (saída) em função de diferentes parâmetros de síntese. Pontos quânticos são estruturas nanométricas que confinam os elétrons nas três dimensões, acarretando na quantização completa dos níveis eletrônicos de energia. O desempenho de diversos dispositivos optoeletrônicos pode ser significativamente beneficiado por esta quantização. Entretanto, seu desempenho depende fortemente do crescimento de estruturas com alta densidade de pontos, com pontos de tamanho pequeno e que apresentem baixa dispersão de tamanho dos pontos. A base de dados com os parâmetros de síntese possui 67 amostras, obtidas por experimentos realizados no laboratório de semicondutores (LABSEM) da PUC-Rio. Os 6 parâmetros (entradas) são: vazão do elemento In no reator, temperatura de crescimento, tempo de deposição, espessura da camada sobre a qual os pontos são crescidos e concentrações de In e Al da camada. Os conjuntos de treinamento, validação e teste contêm, respectivamente, 47, 14 e 6 amostras, como no trabalho de referência.

A tabela 7 compara o desempenho médio dos modelos de PG. A PGLIQ também alcança um desempenho superior ao das redes neurais de [Singulani et al. 2008] para o conjunto de teste. O melhor indivíduo evoluído pela PGLIQ apresenta erro percentual absoluto médio (MAPE) de = 5,2%, contra 8,3% da melhor rede neural, o que representa a redução do erro em 37,3%.

Tabela 7. Comparação do desempenho médio para "Pontos Quânticos" por PG.

	AIMGP		PGLIQ		Variação (%)	
Demes	Média	σ	Média	σ	Média	σ
Não	1.775	0.189	1.359	0.168	-23.5	-11.1
Sim	1.747	0.262	1.402	0.172	-19.8	-34.3

5. Conclusões e Trabalhos Futuros

Este artigo apresentou um novo AEIQ, cujo objetivo é a síntese de programas em código de máquina, o qual obteve desempenho superior ao modelo de referência em testes com problemas de regressão simbólica, ao necessitar de menos avaliações, parâmetros e operadores para alcançar resultados melhores. Estes resultados incentivam seu aperfeiçoamento e futura aplicação em outros tipos de problema. Uma possível melhoria de desempenho pode advir da inclusão de *demes* e de estratégias de migração entre eles (em desenvolvimento e com resultados promissores). É conhecido o benefício dos *demes* sobre convergência prematura e mínimos locais em AE em geral [Poli et al. 2008].

Referências

(1997). *Intel Architecture Software Developer's Manual*. Intel Corporation.

Abs da Cruz, A., Vellasco, M., and Pacheco, M. (2010). Quantum-inspired evolutionary algorithms applied to numerical optimization problems. In *IEEE Congress on Evolutionary Computation (CEC 2010)*, pages 1–6.

Brameier, M. and Banzhaf, W. (2007). *Linear Genetic Programming*. Number XVI in Genetic and Evolutionary Computation. Springer, Boston, MA, USA.

Dias, D. M. and Pacheco, M. A. C. (2009). Toward a quantum-inspired linear genetic programming model. In Tyrrell, A., editor, *IEEE Congress on Evolutionary Computation*, pages 1691–1698, Trondheim, Norway. IEEE Press.

Dias, D. M., Pacheco, M. A. C., and Amaral, J. F. M. (2006). Automatic synthesis of microcontroller assembly code through linear genetic programming. In Nedjah, N., Abraham, A., and de Macedo Mourelle, L., editors, *Genetic Systems Programming: Theory and Experiences*, volume 13 of *Studies in Computational Intelligence*, pages 195–234. Springer, Germany.

Francone, F. D. (2010). *Discipulus Owner's Manual*. Register Machine Learning Technologies. `https://www.rmltech.com/doclink/Owners%20Manual/Discipulus_Owners_Manual.pdf`.

Han, K.-H. and Kim, J.-H. (2002). Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *Evolutionary Computation, IEEE Transactions on*, 6(6):580–593.

Kordon, A. (2010). Symbolic regression competition – EvoStar Conference. `http://casnew.iti.upv.es/index.php/evocompetitions/105-symregcompetition`. Dow Chemical (USA).

Koza, J. R. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA.

Koza, J. R. (2010). Human-competitive results produced by genetic programming. *Genetic Programming and Evolvable Machines*, 11(3-4):251–284.

Lanyon, B., Barbieri, M., Almeida, M., Jennewein, T., Ralph, T., Resch, K., Pryde, G., O'Brien, J., Gilchrist, A., and White, A. (2008). Quantum computing using shortcuts through higher dimensions. *Arxiv preprint arXiv:0804.0272*.

Lavor, C., Manssur, L., and Portugal, R. (2003). Shor's algorithm for factoring large integers. *Arxiv preprint quant-ph/0303175*.

Moore, M. and Narayanan, A. (1995). Quantum-inspired computing. *Dept. Comput. Sci., Univ. Exeter*.

Nordin, P. (1998). AIMGP: A formal description. In Koza, J. R., editor, *Late Breaking Papers at the Genetic Programming 1998 Conference*, University of Wisconsin, Madison, Wisconsin, USA. Stanford University Bookstore.

Poli, R., Langdon, W. B., and McPhee, N. F. (2008). *A field guide to genetic programming*. Published via `http://lulu.com`. (With contributions by J. R. Koza).

Singulani, A., Vilela Neto, O., Pacheco, M., Vellasco, M., Pires, M., and Souza, P. (2008). Computational intelligence applied to the growth of quantum dots. *Journal of Crystal Growth*, 310(23):5063–5065.

A Importância da Informação Heurística Visibilidade para Algoritmos Baseados em Otimização por Colônia de Formigas Aplicados a Domínios Contínuos*

Cassio Rodrigo Conti[1], **Mauro Roisenberg**[1]

[1]Departamento de Informática e Estatística – Universidade Federal de Santa Catarina (UFSC)
Caixa Postal 476 – 88.040-900 – Florianópolis – SC – Brazil

{cassio,mauro}@inf.ufsc.br

Abstract. *Ant Colony Optimization (ACO) is an optimization metaheuristic based in the foraging behavior of ants. This metaheuristic was originally proposed to find good solutions to discrete combinatorial problems. Many extensions of the ACO for continuous domains have been proposed, but even those with close similarity with classical (discrete domain) ACO do not use the heuristic information called visibility commonly used in the original ACO algorithm. In this paper we show the importance of the visibility in ACO and propose its implementation for a continuous domain ACO algorithm. The improvement in convergence speed is shown in results of experiments when the visibility heuristic is used.*

Resumo. *Otimização por Colônia de Formigas (sigla em inglês, ACO) é uma meta-heurística baseada no comportamento das formigas na busca por alimento e foi originalmente desenvolvida para encontrar boas soluções em problemas de otimização combinatória (domínio discreto). Extensões do ACO para trabalhar diretamente no domínio contínuo têm surgido, entretanto as propostas mais similares à ideia clássica não usam a informação heurística chamada "visibilidade", geralmente presente em algoritmos de ACO discreto. Neste trabalho é mostrada a importância da visibilidade em domínio discreto e proposta sua implementação em domínio contínuo. Resultados dos experimentos mostram melhora na velocidade de convergência com o uso da visibilidade.*

1. Introdução

A otimização é uma área de pesquisa que apresenta rápido crescimento e onde o objetivo dos problemas é encontrar o melhor conjunto/combinação de elementos que compõem uma solução válida. Nestes problemas esse conjunto de valores para as variáveis, ou seja, essa solução, tem um custo numérico associado a cada possível conjunto e a complexidade destes problemas cresce exponencialmente à medida que cresce o número de variáveis a serem otimizadas. Em alguns casos, quando há um grande número de variáveis, o tempo necessário para encontrar a melhor solução por meio de uma busca exaustiva pode ser grande o suficiente para ser considerado inviável [Vannimenus and Mézard 1984].

Técnicas de otimização são propostas para resolver esta classe de problemas em uma quantidade de tempo viável. Estas técnicas normalmente não garantem que a melhor

*Este trabalho foi parcialmente financiado pela CAPES e pelo termo de cooperação PETROBRAS / UFSC 0050.0061468.10.9.

solução seja encontrada, mas, ao menos, uma boa solução. *Ant Colony Optimization* (ACO) é uma meta-heurística de otimização proposta por [Dorigo 1992] e que se baseia no comportamento de forrageamento das formigas. Esta técnica foi aplicada a vários problemas e mostrou ser um eficaz processo de otimização para problemas de otimização discreta [Dorigo et al. 1991] [Dorigo et al. 1996].

Na área da otimização, existem problemas em que os valores para as variáveis estão em um domínio contínuo. Para aplicar algoritmos de otimização discreta nestes problemas, as variáveis precisam ser discretizadas, o que não é sempre conveniente, especialmente se o intervalo de valores possíveis das variáveis é grande ou se é necessária uma precisão muito alta [Socha and Dorigo 2006].

Métodos baseados na meta-heurística de ACO para resolverem problemas em domínios contínuos foram propostos em diversos trabalhos, como por exemplo, [Bilchev and Parmee 1995], [Huang and Hao 2006], [Kong and Tian 2006], [Madadgar and Afshar 2008] e [Mathur et al. 2000]. Entretanto, de acordo com [Socha and Dorigo 2006], estas propostas não mantêm uma estreita semelhança com o ACO clássico ou não o seguem precisamente. Em [Socha 2004] foi proposta uma técnica de otimização baseada em ACO para domínio contínuo chamada $ACO_{\mathbb{R}}$. Embora essa implementação seja muito parecida com as definições do ACO clássico, ela não usa uma característica importante dos algoritmos de ACO; um termo conhecido como *visibilidade*. A visibilidade é qualquer informação do ambiente que possa guiar os agentes (formigas) para regiões melhores[1] do espaço de busca, a fim de aumentar a velocidade de convergência em bons resultados.

Neste artigo, é mostrada a importância do termo visibilidade para um ACO clássico e estendido o conceito dessa informação heurística a um ACO de domínio contínuo. É descrita uma possível interpretação da visibilidade em domínio contínuo e a respectiva implementação é aplicada a alguns experimentos com a finalidade de mostrar melhora na velocidade de convergência quando esse recurso é usado.

2. Trabalhos Relacionados

2.1. Otimização por Colônia de Formigas

Introduzido por Marco Dorigo em sua tese de doutorado [Dorigo 1992], ACO é um método probabilístico que tem sido amplamente utilizado em uma vasta classe de problemas de otimização combinatória. Esta meta-heurística tenta explorar o comportamento das formigas no processo de forrageamento em que as formigas caminham entre o ninho e a fonte de alimento, utilizando caminhos diferentes, cada caminho com sua chance própria de ser escolhido e, no decorrer do tempo, cada caminho pode tornar-se mais ou menos desejável (menores caminhos ganham mais chances de serem escolhidos). Este comportamento emerge porque as formigas depositam uma substância chamada feromônio no caminho por onde passam e a quantidade desta substância pode ser notada por outras formigas que, probabilisticamente, escolhem o caminho a seguir usando essa informação. As formigas que utilizam os caminhos mais curtos vão e voltam em menos tempo e acumulam mais feromônio naquele percurso do que aquelas que utilizam caminhos mais longos. Mais feromônio significa mais chances de um caminho ser escolhido.

[1] Áreas do domínio onde os valores para as variáveis implicam em uma saída numericamente desejável para a função objetivo.

[Dorigo et al. 1996] explica com mais detalhes como o menor caminho emerge do comportamento das formigas e faz a analogia com as formigas artificiais que são utilizadas na meta-heurística ACO.

Existe outra importante característica dos algoritmos de ACO, que é o uso de um termo heurístico chamado *visibilidade* ou *atratividade*. Este termo é qualquer informação do ambiente que reflita uma expectativa da medida de custo sobre quão boa determinada escolha provavelmente será durante a construção de uma solução e é uma informação que melhora a busca das formigas artificiais. Em problemas como do Caixeiro Viajante (*Traveling Salesman Problem* - TSP) a distância entre as cidades é comumente usada em trabalhos na literatura como sendo essa informação heurística (por exemplo, [Dorigo et al. 1996]) e dessa forma cidades mais próximas da cidade em que a formiga se encontra recebem maiores chances de serem escolhidas, além da chance definida pelo feromônio. O mesmo ocorre para o Problema do Menor Caminho (*Shortest Path Problem*) onde a distância entre os nós/vértices é usada como informação heurística. No Problema da Mochila (*Knapsack Problem*) a relação custo/benefício de cada item pode ser usada como a visibilidade.

A escolha de cada componente da solução que está sendo construída é feita pela fórmula introduzida na Equação 1, onde p_{ij} é a probabilidade do componente j ser escolhido, dado que i foi escolhido na etapa anterior, f_{ij} é o feromônio referente a aresta/caminho ij, v_{ij} é a visibilidade[2], α e β controlam a importância do feromônio e da visibilidade respectivamente, e $permitidos$ é o conjunto de componentes que podem ser escolhidos pela formiga[3].

$$p_{ij} = \begin{cases} \frac{f_{ij}^{\alpha} \cdot v_{ij}^{\beta}}{\sum_{p \in permitidos} f_{ip}^{\alpha} \cdot v_{ip}^{\beta}} & \text{, se } j \in permitidos. \\ 0 & \text{, caso contrário.} \end{cases} \quad (1)$$

Se for colocada muita importância na visibilidade e pouca no feromônio, a busca se comportará como uma busca gulosa (*greedy search*) [Cormen et al. 2009]. Por outro lado, muita importância para o feromônio e pouca para a visibilidade implica que a informação heurística não influencia no algoritmo. Para mostrar a importância da visibilidade, o algoritmo *Ant System* (AS) [Dorigo et al. 1991] ACO foi implementado e aplicado ao TSP. Foram selecionadas algumas instâncias de problemas da TSPLIB[4]. A Tabela 1 mostra o percurso mais curto encontrado pelas formigas quando a visibilidade é usada e quando não é. Os valores estão em médias de 50 simulações independentes para cada caso e são os melhores percursos encontrados dentro de um limite de 1000 iterações. Os valores entre parênteses são o número médio de iterações necessárias até o melhor percurso da simulação ser encontrado e os valores utilizados para α e β foram 1 e 5 respectivamente, como sugerido em [Dorigo et al. 1991]. Note que a visibilidade melhorou tanto o comprimento do melhor caminho encontrado quanto o número de iterações

[2]No TSP, ij é o caminho entre as cidades i e j. Quanto maior é a distância entre as cidades, menor é o valor de v_{ij} implicando em uma menor chance de ser escolhido.

[3]No TSP, $permitidos$ é o conjunto de cidades que ainda não foram visitadas.

[4]TSPLIB é uma biblioteca de instâncias de exemplo para o Problema do Caixeiro Viajante. Pode ser acessada em http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/.

necessárias para encontrá-lo. Quanto mais complexo é o problema, mais significante é a influência da visibilidade para acelerar a convergência e melhorar os resultados.

Tabela 1. Médias dos valores de menor caminho encontrado na simulação e da iteração em que o menor caminho foi encontrado para o AS ACO aplicado ao TSP.

Problema	Sem visibilidade	Com visibilidade
oliver30.tsp	898,72 [559,58]	416,28 [373,92]
eil51.tsp	1208,58 [536,28]	443,14 [373,68]
a280.tsp	28938,58 [575,34]	2892,92 [490,2]

É importante ressaltar que o mesmo número de avaliações por iteração é feito quando a visibilidade é usada e quando não é. Os valores dos demais parâmetros do algoritmo (número de agentes, taxa de evaporação, etc.) utilizados nestas simulações (Tabela 1) são os sugeridos em [Dorigo et al. 1991].

2.2. Otimização por Colônia de Formigas para Domínios Contínuos

Em problemas de otimização contínua existem variáveis que podem assumir qualquer valor real dentro de um determinado intervalo e o objetivo é maximizar ou minimizar a saída da função/problema. Algumas técnicas de otimização baseadas em ACO foram desenvolvidas visando a sua utilização diretamente em problemas de otimização contínua. [Bilchev and Parmee 1995] propuseram uma versão contínua do ACO (CACO), em que é representado um número finito de direções (vetores) a partir de um ponto base (ninho) e que evoluem com o tempo de acordo com o percurso das formigas. [Huang and Hao 2006] implementaram uma versão do ACO com uma codificação discreta das variáveis contínuas (CACO-DE). [Mathur et al. 2000] propuseram algumas alterações ao modelo CACO básico. Outros modelos foram propostos por [Monmarchè et al. 2000] (API), [Dréo and Siarry 2002] (CIAC) e por [Kong and Tian 2005] (BAS). No entanto, [Socha 2004] afirma que todas essas técnicas não representam de forma muito similar o ACO clássico e então ele propõe um modelo chamado *Extended ACO* ($\text{ACO}_{\mathbb{R}}$). O feromônio nesse modelo está relacionado a combinações de diferentes distribuições de probabilidade para cada variável [Kong and Tian 2006].

Outra extensão é proposta neste trabalho, com base no modelo desenvolvido por [Socha 2004] [Socha and Dorigo 2006] incluindo a visibilidade a fim de guiar o algoritmo de otimização para áreas promissoras do espaço de busca mais rapidamente, reduzindo o número de iterações e avaliações necessárias para alcançar boas soluções, como mostrado na seção anterior para o domínio discreto. O $\text{ACO}_{\mathbb{R}}$ é apresentado a seguir e a visibilidade proposta neste trabalho é detalhada na Seção 3.

2.2.1. $\text{ACO}_{\mathbb{R}}$

Na técnica $\text{ACO}_{\mathbb{R}}$ proposta por [Socha 2004] existe um vetor de soluções chamado *arquivo população* que é mantido durante a execução do algoritmo e que contém um número determinado de k soluções para o problema a se otimizar. Esse vetor com caminhos (soluções) das formigas é uma estrutura análoga a *matriz de feromônio* do ACO

de otimização discreta e representa o feromônio no ambiente. Além disso, o *arquivo população* é mantido ordenado o tempo todo pela qualidade da solução e as melhores soluções (que estão nas primeiras posições do vetor) são preferencialmente escolhidas, assim como acontece com os melhores caminhos encontrados em um ACO clássico.

No início do algoritmo o *arquivo população* é inicializado com k soluções s_j ($j = 1, 2, \ldots, k$), cada uma gerada com valores aleatórios para as n variáveis s_j^i ($i = 1, 2, \ldots, n$) que compõem a solução do problema. Estes valores aleatórios estão entre os valores mínimo e máximo permitidos para cada variável (restrição de domínio). Durante a execução, as soluções no arquivo são usadas para gerar novas soluções na etapa de construção e podem ser substituídas por soluções melhores construídas pelas formigas durante as iterações seguintes. A ideia desse *arquivo população* é ter uma distribuição de probabilidade diferente para cada variável do problema, como representado na Figura 1, que mostra as probabilidades individuais de cinco soluções e uma outra que é a combinação delas (núcleo) e que representa o conhecimento da colônia.

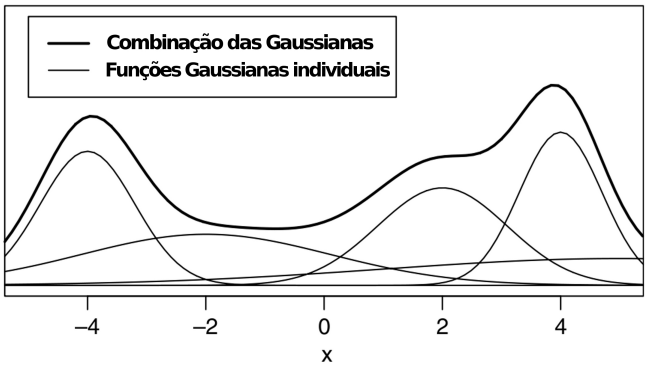


Figura 1. Exemplo de cinco funções Gaussianas e a distribuição combinada delas. Adaptado de [Socha and Dorigo 2006].

O processo de construção de novas soluções pode ser dividido em duas etapas: a primeira em que a formiga escolhe uma solução s_l ($l = 1, 2, \ldots, k$) que servirá de base para todo o processo de amostragem de uma nova solução X construída por essa formiga; e a segunda etapa que é a utilização dos valores das variáveis que compõem s_l como médias na amostragem de cada variável de X. Assim, a cada iteração, cada formiga escolhe uma solução s_l do *arquivo população*. Na Equação 2, p_l é a probabilidade da solução s_l ser escolhida e os valores ω_l e ω_j, que são definidos na Equação 3, representam a quantidade de feromônio no caminho (solução) s_l e s_j respectivamente[5]. A variável q é um parâmetro do algoritmo onde baixos valores de q implicam que as melhores soluções serão preferencialmente escolhidas pois as primeiras posições do vetor terão altos valores de ω_j, por outro lado, altos valores para q implicam que as soluções serão escolhidas com uma probabilidade mais uniforme, porque os valores de cada ω_j terão uma menor diferença entre eles.

Da solução s_l, escolhida para ser a solução base para essa formiga nessa iteração, pode-se obter um valor de média e um de desvio padrão para cada uma das n dimensões do problema, onde a média é o valor s_l^i ($i = 1, 2, \ldots, n$), que é o valor da i-ésima variável

[5] l e j são os índices das soluções no *arquivo população* e dois valores são necessários pelo fato do feromônio de cada solução ser comparado com todas as outras no arquivo.

da solução s_l e o desvio padrão é definido pela Equação 4, onde σ_l^i é o valor do desvio padrão da i-ésima variável da solução s_l e ξ é um parâmetro do algoritmo, onde um alto valor de ξ implica em uma baixa velocidade de convergência da população no algoritmo.

$$p_l = \frac{\omega_l}{\sum_{j=1}^{k} \omega_j}. \quad (2) \qquad \omega_j = \frac{1}{qk\sqrt{2\pi}} e^{-\frac{(j-1)^2}{2q^2k^2}}. \quad (3) \qquad \sigma_l^i = \xi \sum_{j=1}^{k} \frac{\left|s_j^i - s_l^i\right|}{k-1}. \quad (4)$$

Na construção de uma nova solução X, para cada variável $x_i \in X$ $(i = 1, 2, \ldots, n)$, o processo de amostragem usa, como descrito acima, o valor s_l^i como média de uma função Gaussiana (função de distribuição normal) juntamente com o valor de desvio padrão da Equação 4, para amostrar o valor de x_i. Quando todos os valores desta nova solução $X = x_1, x_2, \ldots, x_n$ forem amostrados, X é avaliado e mantido em um *arquivo população* auxiliar/temporário. Só no final da iteração, ou seja, quando todas as formigas tiverem construído suas soluções, estas novas soluções serão movidas do arquivo auxiliar para o *arquivo população*.

3. Algoritmo Proposto

Na Subsecção 2.2.1 foi mostrado que o $ACO_{\mathbb{R}}$ escolhe a solução no processo de construção utilizando apenas a informação do feromônio (Equação 2) e como foi mostrado na Subseção 2.1, o uso da visibilidade juntamente com o feromônio (Tabela 1) implica em uma significativa melhora da técnica. Nesta seção é proposta uma possível interpretação para visibilidade em domínio contínuo, com base na premissa de que a informação heurística tem grande importância na meta-heurística de ACO para acelerar a velocidade de convergência em boas soluções. A técnica apresentada nesta seção para a visibilidade utiliza a representação de feromônio proposta por [Socha 2004] para o feromônio e as implementações são comparadas na Seção 4.

3.1. Visibilidade para ACO em Domínio Contínuo

A visibilidade proposta é baseada na ideia de que, para cada solução construída pelas formigas, existe uma solução vizinha que parece ser mais promissora do que a original. Essa direção de pesquisa promissora é obtida por uma busca local na vizinhança da solução construída por cada formiga, como exemplificado na Figura 2 que apresenta uma possível evolução do algoritmo. A Figura 2a, mostra um exemplo em que as três soluções s_i (i = 1,2,3) são soluções geradas pelas formigas. A Figura 2b mostra uma possível distribuição de probabilidade de cada solução s_i referente a chance de que ela seja escolhida em uma próxima iteração para que seus valores sirvam de média em uma nova amostragem (construção de uma nova solução) no algoritmo $ACO_{\mathbb{R}}$ básico. Note que s_3 tem a maior probabilidade de ser escolhida apesar de que essa solução não se encontra em uma área promissora, pois já está muito próxima do ponto de máximo local. Voltando na Figura 2a, existem as soluções s_i' que são vizinhas geradas pela busca local na região das soluções s_i. A Figura 2c mostra a nova distribuição de probabilidade após essa informação heurística, baseada em quão promissora uma solução parece ser, ser adicionada no *arquivo população*. Agora, as chances de s_1' e s_2' serem escolhidas como valores de média para novas amostragens pelo algoritmo se tornam maiores. A ideia

dessa proposta é inserir essa informação heurística (visibilidade), apresentada no exemplo da Figura 2, no *arquivo população* a fim de acelerar a velocidade de convergência em boas soluções.

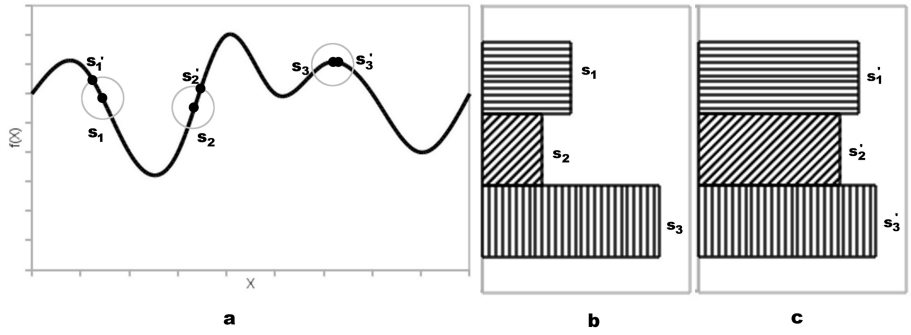


Figura 2. Probabilidade da solução ser escolhida antes e depois da influência da visibilidade.

Para aplicar ao algoritmo o comportamento descrito acima, ao final do processo de construção da solução X realizado pela formiga, uma cópia de X é feita e a busca local é realizada na vizinhança, sendo armazenada nessa cópia (X'). Esta nova solução X' é tratada como se fosse uma solução regular, gerada pelas formigas, ou seja, ela é adicionada a um arquivo auxiliar/temporário juntamente com a solução X de onde será movida para o *arquivo população* somente no final da iteração.

Em um algoritmo de ACO discreto, a visibilidade influencia principalmente nas primeiras iterações, quando o feromônio ainda não acumulou em uma determinada região do espaço de busca. A visibilidade leva os agentes a explorarem uma boa região no início e, a partir dali, as formigas acabam encontrando excelentes soluções devido ao acumulo de feromônio. De forma análoga, a visibilidade contínua dessa proposta influencia principalmente nas primeiras iterações e, após atingir um determinado limiar, ela é desligada. Desligar a visibilidade é necessário devido ao método demandar mais avaliações que o $ACO_{\mathbb{R}}$ original e também pelo fato de que, a partir de um certo ponto, a influência não é mais significativa. Deixar a visibilidade ligada só implicaria em um maior número de avaliações, consequentemente um maior tempo de processamento/execução. O desligamento da visibilidade é descrito na Subseção 3.2.

O comportamento descrito acima influenciará as soluções do algoritmo para que caminhem para boas áreas do domínio mais rapidamente e garante que essa área será explorada. Essa influência implica em um comportamento similar a influência da visibilidade no ACO clássico e, apesar de parecer simples, essa interpretação para a visibilidade apresenta resultados significativamente melhores que serão apresentados na Seção 4.

3.2. Algoritmo para a Visibilidade

A busca local na vizinhança, aplicada nas soluções geradas pelas formigas na proposta de visibilidade para que boas regiões sejam encontradas mais rapidamente, é definida no Algoritmo 1. X é a solução gerada pela formiga, X' é o melhor vizinho de X, X'' é uma solução temporária/auxiliar, X'_i ($i = 1, 2, \ldots, n$) é a i-ésima variável da solução X', Δx é o valor absoluto de variação no valor original da variável que será verificado se é melhor, $random()$ retorna um número real aleatório entre 0 e 1 e o método $melhor\text{-}entre\ ()$ testa

se a saída da solução modificada $f(X'')$ é melhor que a já conhecida $f(X')$. Ao final do algoritmo, X' será o melhor vizinho de X.

Algoritmo 1 Visibilidade - Busca local na vizinhança.

$X' \leftarrow X$
$i \leftarrow 1$
enquanto $i \leq n$ **faça**
  $X'' \leftarrow X'$
  $\Delta x \leftarrow \sigma_l^i$
  se $random() < 0{,}5$ **então**
    $X_i'' \leftarrow X_i' + \Delta x$
  se não
    $X_i'' \leftarrow X_i' - \Delta x$
  fim se
  $X' \leftarrow$ *melhor-entre*(X', X'')
  $i \leftarrow i + 1$
fim enquanto

A utilização do método $random()$ no Algoritmo 1 implica que, para cada variável, o valor de Δx será ou adicionado ou subtraído, sendo feita apenas uma alteração por variável cada vez que a visibilidade é chamada. Esta nova solução X' (melhor vizinho de X), retornada no final do algoritmo, será considerada uma solução comum, ou seja, receberá o mesmo tratamento que uma solução construída pelas formigas e competirá com todas as outras soluções do *arquivo população* para ser a solução base e seus valores servirem de média em novas amostragens.

Como citado na Subseção 3.1, a visibilidade é desligada em determinado ponto. Esse ponto ocorre quando todos os valores de desvio padrão são menores que um $limiar$ (parâmetro do algoritmo). Durante a construção da solução X pela formiga, os n valores de desvio padrão referentes as variáveis da solução base escolhida s_l são calculados e se todos eles forem menores que o $limiar$, o método para a busca local não é chamado. Além da economia no número de avaliações, consequentemente no tempo de processamento, refletida por essa atitude, o fato de que um valor de desvio padrão muito baixo implica que a busca local explorará a vizinhança com baixos valores de Δx, sendo que a própria amostragem, por ser uma distribuição normal, acabará explorando essa pequena vizinhança.

Ao final da iteração, quando todas as m formigas tiverem construído suas soluções e outras m soluções resultantes da busca local nas soluções das formigas também tiverem sido geradas, todas essas $2m$ soluções são movidas do arquivo auxiliar para o *arquivo população*. Estas soluções são então ordenadas e as $2m$ piores são descartadas da população. Dessa forma, o algoritmo sempre mantem a quantia de k soluções no *arquivo população*.

4. Testes e Resultados

Como mostrado na Subseção 2.1, o termo visibilidade é muito importante para algoritmos de ACO no domínio discreto. Para provar que este comportamento se estende a domínios

contínuos, é proposto um conjunto de experimentos para mostrar as melhoras no algoritmo usando a visibilidade. Nestes experimentos foram comparadas a proposta de visibilidade (apresentada na Subseção 3.1) com o $ACO_{\mathbb{R}}$ original (introduzido na Subseção 2.2.1).

Os problemas usados para comparar o $ACO_{\mathbb{R}}$ básico com a versão com visibilidade são todos problemas conhecidos na literatura por serem geralmente usados por autores de novas técnicas para mostrar a qualidade e permitir a comparação entre diferentes propostas. Além disso, estas mesmas funções/problemas foram utilizadas por [Socha and Dorigo 2006] e, devido a limitação de espaço, suas fórmulas são omitidas neste artigo mas podem ser encontradas no trabalho [Socha and Dorigo 2006]. Os nomes dos problemas usados, juntamente com suas respectivas dimensões e domínios são apresentados junto com as tabelas dos resultados, Tabelas 2 e 3.

Para que fosse possível a comparação com os valores divulgados por [Socha and Dorigo 2006], utilizou-se os mesmos critérios de parada, que são os definidos pelas Equações 5 e 6. A Equação 5, utilizada nos problemas *Cigar*, *Ellipsoid* e *Tablet*, têm como parâmetros f que é a saída da função objetivo para uma solução construída pelas formigas, f^* que é a melhor saída da função (valor ótimo[6]) e ϵ_1 que é o erro admitido e foi utilizado com o valor $\epsilon_1 = 10^{-10}$. A Equação 6, utilizada nos demais problemas, têm ϵ_2 e ϵ_3 que foram utilizados com o mesmo valor $\epsilon_2 = \epsilon_3 = 10^{-4}$. Todos esses valores vieram de [Socha and Dorigo 2006] e os critérios de parada implicam que o algoritmo pare quando certa precisão do valor encontrado pela solução das formigas em relação ao valor ótimo da função seja atingido, assim as soluções encontradas nas simulações realizadas neste trabalho são equivalentes as realizadas em [Socha and Dorigo 2006].

$$|f - f^*| < \epsilon_1. \qquad (5) \qquad\qquad |f - f^*| < \epsilon_2 \cdot f + \epsilon_3. \qquad (6)$$

Os valores para os parâmetros utilizados no algoritmo são os mesmos usados por [Socha and Dorigo 2006] em suas simulações: $m = 2$, $k = 50$, $\xi = 0{,}85$ e $q = 0{,}0001$. Além disso, os autores escolheram estes valores porque outras técnicas de otimização na literatura os usam, assim, seria possível uma comparação de desempenho com outras propostas sem que houvesse a necessidade de replicar os algoritmos e experimentos. Sobre os parâmetros, m é o número de formigas, k é o número de soluções no *arquivo população*, ξ influência na velocidade de convergência da população (Equação 4) e q reflete nas melhores soluções sendo preferencialmente escolhidas como solução base ou se essa probabilidade será mais uniforme (Equação 3). O parâmetro adicional desta proposta é o uso de $limiar = 1{,}5$, onde $limiar$ é um parâmetro da visibilidade que controla quando ela deve ser desligada, como explicado na Subseção 3.1, e seu valor foi definido após uma série de simulações de teste com o objetivo de encontrar um valor que retornasse o comportamento mais desejado possível para as saídas dos problemas. O valor não é o melhor para todos os problemas, mas, ao definir um único valor para todos, esse foi o que apresentou os melhores resultados. A escolha desse parâmetro pode ser relacionada a escolha de valores para α e β para definir a importância do feromônio e da visibilidade em um ACO discreto.

[6]O valor é conhecido por se tratarem de problemas muito explorados na literatura.

As Tabelas 2 e 3 mostram, respectivamente, o número de iterações e número de avaliações necessárias até o critério de parada ser atingido. Os valores referentes ao $ACO_{\mathbb{R}}$ original foram retirados de [Socha and Dorigo 2006]. Já os valores da versão com visibilidade são todos médias de 50 simulações independentes para a implementação proposta utilizando os parâmetros definidos para o experimento. É importante ressaltar que os valores do número de iterações para o $ACO_{\mathbb{R}}$ original não aparecem de forma explícita no trabalho de [Socha and Dorigo 2006], entretanto, a partir do número de agentes m e do número de soluções geradas aleatoriamente no inicio do algoritmo k, esse valor é deduzível[7]. Os valores entre parenteses são o desvio padrão para o $ACO_{\mathbb{R}}$ + visibilidade para o número de iterações na Tabela 2 e para o número de avaliações na Tabela 3. Os valores de desvio padrão referentes ao $ACO_{\mathbb{R}}$ original não são apresentados em [Socha and Dorigo 2006].

Tabela 2. Comparação do número de iterações sem e com a visibilidade.

Função, dimensão, intervalo	$ACO_{\mathbb{R}}$*	$ACO_{\mathbb{R}}$ + visibilidade
B2, $n = 2$, $[-100, 100]$	247	103 (11,92)
Branin RCOS, $n = 2$, $[-5, 15]$	403,75	100,3 (31,93)
Cigar, $n = 10$, $[-3, 7]$	2663	872 (90,51)
De Jong, $n = 3$, $[-5{,}12, 5{,}12]$	171	73,02 (10,19)
Easom, $n = 2$, $[-100, 100]$	361	115,22 (13,98)
Ellipsoid, $n = 10$, $[-3, 7]$	5760	660,26 (77,08)
Goldstein and Price, $n = 2$, $[-2, 2]$	167	78,64 (10,22)
Martin and Gaddy, $n = 2$, $[-20, 20]$	147,5	73,98 (12,52)
Sphere model, $n = 6$, $[-5{,}12, 5{,}12]$	365,5	142,92 (19)
Tablet, $n = 10$, $[-3, 7]$	1258,5	674,46 (75,63)
Zakharov, $n = 2$, $[-5, 10]$	121,25	59,88 (10,19)

* Valores desta coluna foram divulgados por [Socha and Dorigo 2006].

Tabela 3. Comparação do número de avaliações sem e com a visibilidade.

Função, dimensão, intervalo	$ACO_{\mathbb{R}}$*	$ACO_{\mathbb{R}}$ + visibilidade
B2, $n = 2$, $[-100, 100]$	544	394,88 (29,57)
Branin RCOS, $n = 2$, $[-5, 15]$	857,5	391,88 (132,43)
Cigar, $n = 10$, $[-3, 7]$	5376	2168 (222,81)
De Jong, $n = 3$, $[-5{,}12, 5{,}12]$	392	240,32 (23,72)
Easom, $n = 2$, $[-100, 100]$	772	573,8 (67,39)
Ellipsoid, $n = 10$, $[-3, 7]$	11570	1983,72 (188,64)
Goldstein and Price, $n = 2$, $[-2, 2]$	384	207,28 (20,44)
Martin and Gaddy, $n = 2$, $[-20, 20]$	345	298,28 (26,54)
Sphere model, $n = 6$, $[-5{,}12, 5{,}12]$	781	467,36 (49,08)
Tablet, $n = 10$, $[-3, 7]$	2567	2028,52 (172,87)
Zakharov, $n = 2$, $[-5, 10]$	292,5	226,56 (23,87)

* Valores desta coluna foram divulgados por [Socha and Dorigo 2006].

Analisando os resultados das Tabelas 2 e 3 é possível perceber que, apesar do número de avaliações ter diminuído para todos os problemas quando a visibilidade foi

[7]Cálculo: $iterações = \frac{avaliações - k}{2}$

usada, alguns alcançaram uma redução maior que outros. Esse comportamento pode ser relacionado ao ACO discreto, onde foi visto que dependendo da complexidade do problema, havia diferente variação da influência da visibilidade no resultado. Além disso, o número de iterações teve significante redução para todos os problemas testados com reduções de, no mínimo, 46,4%.

Embora a visibilidade proposta demande algumas avaliações a mais nas primeiras iterações em relação ao $ACO_{\mathbb{R}}$ original devido a busca local, como a visibilidade é desligada em determinado ponto durante a execução e como número de iterações é reduzido, o numero total de avaliações também é reduzido. A consequência disso é um menor tempo de execução do algoritmo. Outro fator que é importante citar é que, para alguns problemas é mais rápido realizar algumas avaliações extras do que executar uma nova iteração que amostra valores para cada variável e todos os demais passos já explicados. Esse é o caso dos problemas que foram testados, pois apesar de alguns problemas não apresentarem significativa redução do número de avaliações, o tempo de execução foi reduzido significativamente devido ao menor número de iterações.

A partir dos resultados, também pode-se concluir que a visibilidade, apesar de acelerar a convergência do algoritmo, demanda avaliações extras que, utilizando a comparação pelo número de avaliações, pode não parecer tão significativa. Entretanto, combinando essa redução com a redução do número de iterações reflete-se a importância do uso da visibilidade em algoritmos ACO de domínio contínuo, assim como acontece no domínio discreto, que é guiar o algoritmo mais rapidamente para as melhores soluções do domínio. Em todos os problemas testados a visibilidade apresentou melhores resultados que a versão original, tanto no número de avaliações, quanto no número de iterações.

5. Conclusão

Neste artigo foi apresentado como o ACO pode ser usado em ambos domínios, discreto e contínuo, e como a visibilidade pode melhorar a velocidade de convergência e os resultados encontrados pelos algoritmos.

Entre as propostas de implementações para o ACO em domínio contínuo, o $ACO_{\mathbb{R}}$ foi escolhido pela similaridade com os conceitos e aspectos originais da meta-heurística. Mesmo assim, esta técnica não inclui a visibilidade no algoritmo. Foi então proposta uma interpretação de como a visibilidade poderia ser em um domínio contínuo.

A proposta foi implementada e foram feitos testes com os mesmos problemas de otimização já apresentados e aplicados no trabalho original do $ACO_{\mathbb{R}}$. Os resultados mostraram melhora na velocidade de convergência refletida em um menor número de iterações e de avaliações necessárias para alcançar bons resultados.

Pesquisas sobre outras possíveis interpretações da visibilidade e do método de busca local que encontra o melhor vizinho estão em desenvolvimento e serão o assunto de futuras publicações.

Referências

[Bilchev and Parmee 1995] Bilchev, G. and Parmee, I. C. (1995). The ant colony metaphor for searching continuous design spaces. *AISB Workshop on Evolutionary Computing*, 993:25–39.

[Cormen et al. 2009] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Greedy Algorithms*, chapter 16. MIT Press, 3rd edition.

[Dorigo 1992] Dorigo, M. (1992). *Optimization, Learning and Natural Algorithms*. PhD thesis, Dip. Elettronica e Informazione, Politecnico di Milano, Italy.

[Dorigo et al. 1991] Dorigo, M., Maniezzo, V., and Colorni, A. (1991). Positive feedback as a search strategy. Technical report, Dipartimento di Elettronica - Politecnico di Milano - Italy.

[Dorigo et al. 1996] Dorigo, M., Maniezzo, V., and Colorni, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, 26(1):29–41.

[Dréo and Siarry 2002] Dréo, J. and Siarry, P. (2002). A new ant colony algorithm using the heterarchical concept aimed at optimization of multiminima continuous functions. In Dorigo, M., Di Caro, G., and Sampels, M., editors, *Ant Algorithms*, volume 2463 of *Lecture Notes in Computer Science*, pages 216–221. Springer Berlin / Heidelberg.

[Huang and Hao 2006] Huang, H. and Hao, Z. (2006). Aco for continuous optimization based on discrete encoding. *Ant Colony Optimization and Swarm Intelligence*, pages 504–505.

[Kong and Tian 2005] Kong, M. and Tian, P. (2005). A binary ant colony optimization for the unconstrained function optimization problem. In Hao, Y., Liu, J., Wang, Y.-P., Cheung, Y.-m., Yin, H., Jiao, L., Ma, J., and Jiao, Y.-C., editors, *Computational Intelligence and Security*, volume 3801 of *Lecture Notes in Computer Science*, pages 682–687. Springer Berlin / Heidelberg.

[Kong and Tian 2006] Kong, M. and Tian, P. (2006). A direct application of ant colony optimization to function optimization problem in continuous domain. *Ant Colony Optimization and Swarm Intelligence*, pages 324–331.

[Madadgar and Afshar 2008] Madadgar, S. and Afshar, A. (2008). An Improved Continuous Ant Algorithm for Optimization of Water Resources Problems. *Water Resources Management*, 23(10):2119–2139.

[Mathur et al. 2000] Mathur, M., Karale, S. B., Priye, S., Jayaraman, V. K., and Kulkarni, B. D. (2000). Ant colony approach to continuous function optimization. *Industrial & Engineering Chemistry Research*, 39(10):3814–3822.

[Monmarchè et al. 2000] Monmarchè, N., Venturini, G., and Slimane, M. (2000). On how pachycondyla apicalis ants suggest a new search algorithm. *Future Generation Computer Systems*, 16(9):937–946.

[Socha 2004] Socha, K. (2004). Aco for continuous and mixed-variable optimization. *Ant Colony Optimization and Swarm Intelligence*, pages 25–36.

[Socha and Dorigo 2006] Socha, K. and Dorigo, M. (2006). Ant colony optimization for continuous domains. *European Journal of Operational Research*, 185:1155–1173.

[Vannimenus and Mézard 1984] Vannimenus, J. and Mézard, M. (1984). On the statistical mechanics of optimization problems of the travelling salesman type. *Journal de Physique Lettres*, 45(24):1145–1153.

Algoritmo de Enxame de Partículas Híbrido Aplicado a Clusterização de Dados

Gustavo H. A. Sousa[1], **Ahmed A. A. Esmin**[1]

[1] Departamento de Ciência da Computação – Universidade Federal de Lavras (UFLA)
Caixa Postal 3037 – 37.200-000 – Lavras – MG – Brazil

{sousagh,aesmin}@gmail.com

Abstract. *Clustering is an important data mining task and has been explored extensively by a number of researchers for different application areas such as text application and bio-informatics data. The Particle Swarm Optimization (PSO) is a technique based on social behavior that has been successfully applied to several types of problems. In this paper we propose the use of a novel algorithm for clustering data that we call Hybrid Particle Swarm Optimization with Mutation (HPSOM) which is based on PSO. The HPSOM basically uses PSO and by incorporating the mutation process often used in Genetic Algorithms to allow the search to escape from local optima. It is shown how the PSO/HPSOM can be used to find the centroids of a user specified number of clusters. The new algorithm is evaluated on real clustering benchmark data. The proposed method is compared with k-means clustering technique and standard Particle Swarm clustering algorithm. The results show that the algorithm is efficient and produces compact clusters.*

Resumo. *Clusterização é uma importante ferramenta de mineração de dados e tem sido extensivamente explorada por diversos pesquisadores de diferentes áreas de aplicação, tais como mineração de texto e bio-informática. O Particle Swarm Optimization (PSO) é uma técnica baseada no comportamento social que vem sendo aplicada com sucesso a diversos tipos de problemas. Neste trabalho, propomos a utilização de um novo algoritmo para agrupamento de dados que chamamos de Hybrid Particle Swarm Optimization with Mutation (HPSOM), que é baseado no PSO. O HPSOM utiliza basicamente PSO e incorpora o processo de mutação, muitas vezes usados em Algoritmos Genéticos, para escapar de ótimos locais. É mostrado como o PSO/HPSOM pode ser usado para localizar o centroides de um número de clusters especificado pelo usuário. A nova proposta de método é avaliada em dados reais de referência clustering (benchmarks). O método proposto é comparado com a técnica de agrupamento K-means, com o algoritmo Particle Swarm Optimization padrão, e com um algoritmo híbrido que usa o algoritmo K-means para preencher o enxame inicial do PSO. Os resultados mostram que o algoritmo é eficiente e produz clusters compactos.*

1. Introdução

Clustering, também conhecido como Agrupamento ou Clusterização, é uma das técnicas mais utilizadas na descoberta de conhecimento a partir de bancos de dados. Este método

consiste em encontrar grupos nos quais os seus elementos são similares entre si e diferentes dos elementos de outros grupos [Cohen and Castro 2006]. Clusterização de dados é o processo que visa reunir vetores multidimensionais em grupos, tendo em vista obter a maior similaridade dentro de cada grupo [Merwe and Engelbrecht 2003].

O *Particle Swarm Optimization* (PSO) é uma técnica de otimização estocástica proposta inicialmente por [Kennedy and Eberhart 1995] inspirado pelo comportamento social de cardumes de peixes e bandos de aves. Kennedy e Eberhart descobriram o algoritmo através da simulação de um modelo social simplificado, onde a intenção original foi simular a graciosa, porém imprevisível, coreografia de um bando de pássaros.

O *Particle Swarm Optimization* é um algoritmo relativamente novo e muitas vezes é aplicado com sucesso a diversos problemas de otimização. Recentemente surgiram na literatura algumas abordagens do PSO para resolver o problema de Clusterização de dados com resultados satisfatórios. Já que a aplicação do PSO em *Clustering* é recente, suas abordagens ainda podem ser trabalhadas a fim de encontrar melhores resultados.

Hybrid Particle Swarm Optimization with Mutation é um algoritmo proposto por [Esmin et al. 2006] que se que integra ao PSO com mutação, muitas vezes utilizada em Algoritmos Genéticos, objetivando evitar os mínimos locais e solução do problema de estagnação apresentado pelo PSO padrão.

Este artigo tem como objetivo o estudo de *Hybrid Particle Swarm Optimization with Mutation* para resolver o problema de Clusterização de dados, além de propor um novo algoritmo baseado em HPSOM no qual o algoritmo *K-means* é usado para preencher o enxame inicial.

Este trabalho está organizado da seguinte forma: a Seção 2 apresenta o algoritmo *K-means*. O algoritmo PSO é descrito na Seção 3. A Seção 4 explica como o PSO pode ser utilizado para Clusterização de dados. O algoritmo *Hybrid Particle Swarm Optimization with Mutation* é apresentado na Seção 5. Duas técnicas de Clusterização baseadas em PSO são discutidas na Seção 6. Na Seção 7 são expostos os resultados e a Seção 8 apresenta as conclusões deste artigo.

2. *K-means*

K-means (KM) é uma técnica de Clusterização de Particionamento proposta inicialmente por [MacQueen 1967] e tem o propósito de particionar uma população n-dimensional em k grupos em uma dada base de dados.

De acordo com [Carlantonio 2001], o algoritmo KM toma um parâmetro de entrada, k, e particiona um conjunto de n objetos em k *clusters* de modo que a similaridade *intracluster* resultante é alta, mas a similaridade entre *clusters* diferentes é baixa. A similaridade de *clusters* é medida em respeito ao valor médio dos objetos em um *cluster*, que pode ser visto como o centro de gravidade do *cluster*.

O algoritmo *k-means* trabalha da seguinte maneira. Inicialmente, o método, aleatoriamente, seleciona k objetos, cada um dos quais, inicialmente, representa os centroides, ou centros dos *clusters*. Para cada um dos objetos da base de dados, é feita a atribuição ao *cluster* ao qual o objeto é mais similar, baseado na distância entre o objeto e a média do *cluster*. Ele, então, computa o novo valor dos centroides, que é tido pela média dos valores dos objetos que estão atribuídos à cada *cluster* [Carlantonio 2001].

O método KM padrão pode ser resumido da seguinte maneira:

1. Escolha aleatoriamente k objetos da base de dados como os centros iniciais dos *clusters*;
2. **Repetir**;
   - (a) (re)atribua cada objeto ao *cluster* ao qual o objeto é mais similar, de acordo com o calor médio dos objetos no *cluster*;
   - (b) atualize as médias dos *clusters*, ou seja, calcule o calor médio dos objetos para cada cluster;

   até que se atinja o critério de parada;

O algoritmo *k-means* pode parar se exceder o máximo de iterações pré estabelecido, se a alteração nos centroides após uma iteração for insignificante, ou se não houver alteração nos membros de cada centroide.

A performance do algoritmo KM é sensível à partição inicial, gerada pela escolha aleatória dos centros iniciais [Zhang et al. 1999].

A necessidade de o usuário ter que especificar k, o número de *clusters* com antecedência é uma desvantagem. O método *k-means* não é adequado para descobrir *clusters* com formas não convexas ou grupos de dimensões muito diferentes. Além disso, ele é sensível a ruídos, visto que pequeno número de tais dados pode influenciar, substancialmente, o valor do resultado final do algoritmo [Carlantonio 2001].

3. *Particle Swarm Optimization*

Particle Swarm Optimization (PSO) é um método de otimização baseado em população inicialmente proposto por [Kennedy and Eberhart 1995] inspirado no comportamento social de bando de pássaros. A busca por alimento ou pelo ninho e a interação entre os pássaros ao longo do vôo são modelados como um mecanismo de otimização [Saramago and Prado 2005].

De acordo com [Merwe and Engelbrecht 2003], dado um problema, o algoritmo mantém uma enxame de partículas de tamanho s, em que cada partícula representa um candidato potencial para a solução do problema. Cada partícula encontra-se em uma posição em um espaço de busca multidimensional e mantém as seguintes informações:

- f – *fitness* da partícula. Este valor indica quão bem a partícula está posicionada no espaço de busca. Em outras palavras, o *fitness* de cada partícula significa quão boa ela é para resolver o problema. Quanto mais próximo de zero, melhor.
- x_i – Posição atual da partícula.
- v_i – Velocidade Atual da partícula.
- y_i – Melhor posição pessoal da partícula (melhor posição em que a partícula já esteve).

Assim como na natureza, para que as partículas circulem no espaço de busca, é necessário que cada partícula possua uma velocidade. Assim, a cada iteração a partícula tem sua velocidade ajustada de acordo com a seguinte equação:

$$v_{i,k}(t+1) = wv_{ik}(t) + c_1 r_{1,k}(t)(y_{i,k}(t) - x_{i,k}(t)) + c_2 r_{2,k}(t)(\hat{y}_k(t) - x_{i,k}(t)) \qquad (1)$$

Onde $v_{i,k}$ denota a $k^{-ésima}$ dimensão do vetor velocidade associado à $i^{-ésima}$ partícula. O valor w é um peso fornecido pelo usuário, que pode-se manter fixo ao longo de todo o algoritmo ou variar entre um w_{min} e um $w_{máx}$ de acordo com a seguinte equação:

$$\frac{w = w_{min} + iw_{máx} - w_{min}}{t} \tag{2}$$

Onde i é o numero da iteração e t é o número máximo de iterações do algoritmo.

A velocidade é modificada separadamente em cada dimensão $k \in i \ldots n$. r_1 e r_2 são valores aleatórios, $r_1 \sim U(0,1)$ e $r_2 \sim U(0,1)$, ou seja, entre zero e um, que contribuem para a natureza estocástica do algoritmo. c_1 e c_2, $0 < c_1, c_2 \leq 2$, são coeficientes de aceleração, c_1 regula o passo máximo na direção da melhor posição pessoal e c_2 na direção da posição global (*gbest*, ou *global best*) ou da vizinhança (*lbest*, ou *local best*) [van der Bergh 2001]. O termo $c_1 r_{1,k}(t)(y_{i,k}(t) - x_{i,k}(t))$ leva em conta as experiências passadas da partícula e é associado a cognição. O termo $c_2 r_{2,k}(t)(\hat{y}_k(t) - x_{i,k}(t))$ é o termo social, pois a partícula se inspira na melhor solução ao seu redor [van der Bergh 2001, Cohen and Castro 2006].

Na versão *gbest* do PSO, $\hat{y}$ representa a melhor posição que foi encontrada dentre todas as partículas do enxame. Esta versão fornece uma taxa de convergência mais rápida, mas que, no entanto, é menos robusta. Na versão *lbest*, a população é dividida em vizinhanças, e $\hat{y}_j$ representa a melhor posição encontrada na vizinhança da partícula j , para uma população de tamanho s e vizinhança de tamanho l [van der Bergh 2001]. A definição de $\hat{y}$, como é utilizado na versão *gbest*, é representada pela Equação (3).

$$\hat{y}_i(t) \in \{y_0(t), y_1(t), \ldots, y_s(t)\} \mid f(\hat{y}_i(t+1)) = min\{y_0(t), y_1(t), \ldots, y_s(t)\} \tag{3}$$

Onde f é a função objetivo do problema. A melhor posição global é aquela que possui menor valor da função objetivo.

A posição da partícula é atualizada usando o novo vetor velocidade de acordo com a seguinte equação:

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{4}$$

A melhor posição pessoal da partícula i é calculada como:

$$x(t+1) = \begin{cases} y_i(t) & \text{se } f(x_i(t+1)) \geq f(y_i(t)) \\ x_i(t+1) & \text{se } f(x_i(t+1)) < f(y_i(t)) \end{cases}$$

(5)

O algoritmo consiste em repetidas aplicações das equações de atualização de posição apresentadas (Figura 1). A ação de alterar a posição de cada partícula simula o movimento de um pássaro no espaço, que é influenciado pela própria memória da partícula, ou melhor posição pessoal (*personal best position*), e pela partícula que está melhor posicionada, ou melhor posição global (*global best position*). O algoritmo segue a heurística que a melhor resposta sempre se encontra próxima à melhor posição global encontrada por uma partícula, por isso, todas as outras tendem a procurar a melhor solução próxima ao *gbest*.

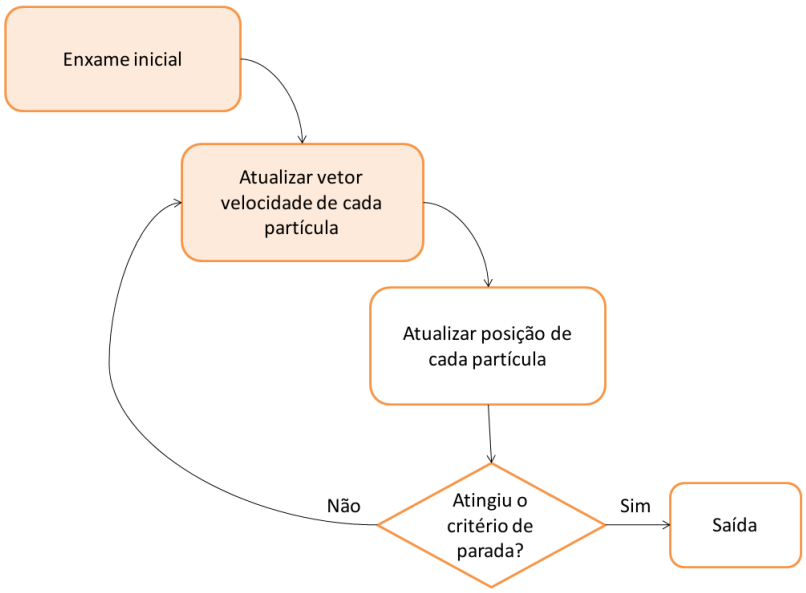


Figure 1. Fluxograma para o algoritmo PSO básico (Saramago and Prado 2005)

A inicialização da população de colônia normalmente é obtida com as partículas dispostas aleatoriamente sobre o espaço de projeto, $[-x_{max}, x_{max}]$. As velocidades $v_{i,j}$ geralmente também são inicializadas aleatoriamente em um intervalo $[-v_{max}, v_{max}]$. O critério de parada pode ser um número determinado de iterações ou outro critério dependendo do problema [van der Bergh 2001].

4. PSO Usado para Clusterização de Dados

No método proposto por [Merwe and Engelbrecht 2003], cada partícula do PSO é composta por um vetor de centroides de tamanho N_c, onde N_c é o número de grupos a serem criados, e representa uma solução completa para o problema. A cada iteração, os dados são atribuídos ao centroide ao qual estão mais próximos. As soluções são avaliadas e então atualizadas de acordo com sua melhor posição no espaço de busca e a melhor posição já encontrada por alguma partícula.

Cada partícula x_i é tido como um conjunto de *clusters*, pois, só assim, cada partícula representaria uma resposta completa da Clusterização [Merwe and Engelbrecht 2003]. Em outras palavras, no enxame do PSO, cada partícula possui um vetor de *k* centroides que são construídos da seguinte forma:

$$x_i = (m_{i1}, m_{i2}, \ldots, m_{ij}, \ldots, m_{iN_c},)$$

onde N_c é o número de *clusters* a serem criados em $m_{i,j}$ e corresponde ao $j^{-ésimo}$ centroide da $i^{-ésima}$ partícula. Assim, uma única partícula representa um candidato à solução do problema de Clusterização.

Cada partícula é avaliada através da seguinte equação denominada Função Objetivo:

$$J_e = \frac{\sum_{j=1}^{N_c}[\sum_{\forall Z_p \in c_{i,j}} d(z_p.m_{i,j})/ \mid C_{i,j} \mid]}{N_c} \tag{6}$$

Onde Z_p é o $p^{-ésimo}$ dado, $\mid C_{ij} \mid$ é o número de dados pertencentes ao *cluster* C_{ij} e d é a distância euclidiana entre Z_p e m_{ij}.

O algoritmo proposto por [Merwe and Engelbrecht 2003] pode ser descrito da seguinte forma:

1. Inicializar o valor dos centroides de cada partícula aleatoriamente
2. Para t = 1 até t_{max} faça
   - (a) Para cada partícula *i* faça
   - (b) Para cada dado z_p
     - *i*. calcule a distância Euclidiana $d(z_p, m_{ij})$ para todos os centroides m_{ij}
     - *ii*. atribua z_p ao *cluster* C_{ij}tal que $d(z_p, m_{ij}) = min_{\forall k=1...N_c}\{d(z_p, m_{ij})\}$
     - *iii*. Calcule o *fitness* da partícula utilizando a função objetivo
   - (c) Atualize o melhor global e os melhores locais
   - (d) Atualize os centroides dos *clusters*

5. *Hybrid Particle Swarm Optimization with Mutation*

O trabalho de [Esmin et al. 2006] propõe um novo motodo chamado *Hybrid Swarm Optimizer with Mutation* (HPSOM), que tem como objetivo resolver o problema de estagnação apresentado pelo PSO padrão e, desta forma impedir que as partículas fiquem presas em mínimos locais. Para resolver este problema, o HPSOM integra o processo de mutação muitas vezes utilizado em GA no PSO. A cada iteração, as partículas tem uma probabilidade, que é escolhida pelo usuário, de sofrer mutação. O processo de mutação é empregado utilizando a seguinte equação:

$$mut(p[k]) = p([k]x - 1) + \omega \tag{7}$$

Onde $p[k]$ é uma partícula escolhida aleatoriamente do enxame e ω é um valor, também aleatório, que está no intervalo $[0, 0.1x(x_{max} - x_{min})]$, que representa 10% do espaço de busca.

Simulações em uma série *benchmarks* e testes mostram que o algoritmo híbrido proposto apresenta uma melhor capacidade de encontrar o ótimo global do que algoritmo PSO padrão.

Este artigo apresenta o estudo de aplicação do método HPSOM para resolver o problema de Clusterização, o qual ainda não existe na literatura.

6. Algoritmos Híbridos

De acordo com [Merwe and Engelbrecht 2003], o método *k-means* converge para o resultado mais rapidamente que o *Particle Swarm Optimization*. Em outras palavras, o algoritmo *k-means* alcança a convergência em poucas iterações. Entretanto, na maioria dos casos, o PSO encontra grupos que possuem maior *fitness*.

Em seu trabalho, [Merwe and Engelbrecht 2003] mostram que o desempenho do algoritmo PSO para Clusterização de dados pode ser melhorado. Isto pode ser conseguido fornecendo no enxame inicial o resultado do algoritmo *k-means* a fim de obter pelo menos uma partícula próxima ao melhor particionamento. O resultado do método *k-means* é utilizado como uma das particulas, enquanto o resto do enxame é inicializado aleatoriamente e a partir daí, buscam refinar o resultado trazido pelo *k-means*. O método híbrido utilizando PSO e *k-means* foi chamado de PSOKM.

O presente trabalho faz uso da proposta de [Merwe and Engelbrecht 2003], chamando-o de PSOKM. Além disso, é proposta uma nova abordagem para a solução do problema de Clusterização, um algoritmo que usa a mesma ideia de hibridização do trabalho de [Merwe and Engelbrecht 2003] (PSOKM), mas que, no entanto, utiliza o HPSOM ao invés do PSO clássico. Esta abordagem busca a união de duas abordagens, tanto a proposta por [Esmin et al. 2006] quanto a proposta por [Merwe and Engelbrecht 2003], a fim de se obter melhores resultados que ambos separadamente. Esta combinação de métodos foi chamada neste trabalho de HPSOMKM (*Hybrid Particle Swarm Optimizatiom with Mutation and K-Means*) e é ilustrada na Figura 2.

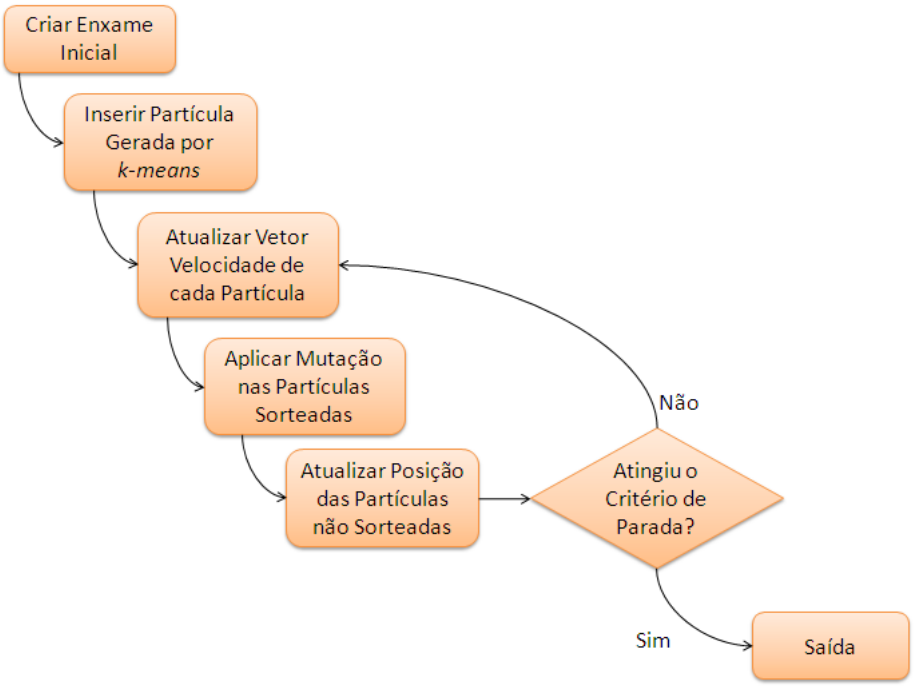


Figure 2. Fluxograma para o algoritmo HPSOMKM

Se compararmos a Figura 1 com a Figura 2 é perceptível que o algoritmo HPSOMKM utiliza a mesma ideia do PSO, com algumas diferenças básicas. No segundo passo, é introduzido uma ultima partícula que é o resultado do algoritmo *kmeans*, que

benckmark	Número de Instâncias	Número de Atributos	Número de Classes
Iris	150	4	3
Wine	178	13	3
Glass	214	9	7

Table 1. Características dos *benchmarks*

servirá como guia para as outras partículas, otimizando o algoritmo. O quarto passo, como descrito na Seção 5, executa a mutação nas partículas do enxame. Cada partícula tem uma probabilidade estabelecida arbitrariamente de sofrer mutação, que está descrita na Equação 7.

O algoritmo HPSOMKM é apresentado na Figura 2, podemos observar a introdução de uma partícula gerada a partir do resultado do *kmeans* e esta partícula tem um papel fundamental na melhoria do resultado global do enxame. A mutação também é aplicada em algumas partículas aleatoriamente selecionadas após a atualização das suas velocidades como é descrito na seção 5.

7. Resultados

Para avaliar e comparar os métodos, foram usados três *benchmarks* amplamente abordados na literatura: *Iris*, *Glass* e *Wine* [Asuncion and Newman 2007]. Todos podem ser encontrados no UCI *Repository of Machine Learning Databases* e suas características estão descritas na Tabela 1. O *benchmark Iris* apresenta cento e cinquenta instâncias da flor Iris, divididas em três classes com cinquenta instâncias cada. A primeira classe representa o tipo Setosa, a segunda ao tipo Versicolour e a terceira Virginica. As instâncias possuem quatro atributos de valores reais, *sepal length* (comprimento da sépala), *sepal width* (largura da sépala), *petal length* (comprimento da pétala), *petal width* (largura da pétala). Uma das classes (Setosa) é linearmente separável das outras duas, que não são linearmente separáveis entre si.

O *benchmark Wine* possui cento e setenta e oito instâncias de vinho. Esses dados são resultados de uma análise química realizada em vinhos da mesma região da Itália, mas vindos de diferentes cultivares. A análise determinou as quantidades de treze constituintes encontrados em cada um dos três tipos de vinho. Os atributos são: *alcohol , malic acid, ash, alcalinity of ash, magnesium, total phenols, flavanoids, noflavanoid fenols, proanthocyanins, color intensity, hue, OD280/OD315 of diluted wines, praline*. A primeira classe contém cinquenta e nove instâncias, a segunda classe contém setenta e uma e a terceira quarenta e oito.

O *benchmark Glass* apresenta duzentas e quatorze instâncias de vidros divididas em sete tipos. Setenta instâncias em *building windows float processed*, dezessete em *vehicle windows float processed*, setenta e seis em *buiding windows non-float processed*, nenhuma em *vehicle windows non-float processed*, treze em *containers*, 9 em *tableware*, vinte e nove em *headlamps*. Os atributos dos dados são: *refrative index, sodium, magnesium, aluminium, silicon, potassium, calcium, barium, iron*.

Para cada conjunto de dados, cada algoritmo foi executado por 100 vezes, com 400 iterações, 30 partículas, w variando entre 0.5 e 0.005, $c1 = 0.2$ e $c2 = 0.2$. Após uma grande quantidade de testes, foram escolhidos estes valores por obterem bons resultados

Método	*Iris*	*Wine*	*Glass*
PSO	0.08696 ± 0.00484	0.05903 ± 0.00153	0.01911 ± 0.00108
KM	0.08307	0.06155	0.01393
PSOKM	0.08210 ± 0.00203	0.05820 ± 0.00069	0.01382 ± 0.00026
HPSOM 10%	0.08203 ± 0.00289	0.00289 ± 0.00148	0.01442 ± 0.00123
HPSOM 5%	0.08295 ± 0.00320	0.05838 ± 0.00141	0.01446 ± 0.00116
HPSOMKM 10%	$0.08222 \pm$ **0.00114**	$0.05810 \pm$ **0.00074**	$0.01385 \pm$ **0.00022**
HPSOMKM 5%	**0.08164** ± 0.00250	**0.05801** ± 0.00075	**0.01377** ± 0.00037

Table 2. Resultados dos testes experimentais

em todos os métodos.

Cada execução é avaliada de acordo com o *fitness* da melhor partícula, ou seja, é calculada a média da distância entre cada elemento do conjunto de dados e o centroide ao qual ele pertence.

Para saber se determinado elemento foi agrupado de maneira correta, é preciso saber a que classe do *benchmark* o *cluster* encontrado pelo algoritmo corresponde [Esmin et al. 2008, Pereira 2007]. Dado que:

- $G = \{g_1, g_2, \ldots, g_k\}$ é o conjunto dos grupos gerados pelo algoritmo.
- $B = \{b_1, b_2, \ldots, b_k\}$ é o conjunto de classes do *benchmark*.
- r_i é a classe representada pelo grupo i.
- n_{ij} é o grupo de dados da classe *j* no grupo *i*.

O algoritmo de mapeamento de *clusters* pode ser descrito da seguinte forma:

Enquanto $G \neq$
  Encontrar maior $n_{ij} \mid g_i \in G$ e $b_j \in B$
  $r_i = j$
  $G = G - g_i$
  $B = B - b_j$

Dessa maneira, todos os dados que estiverem em c_i e também pertencerem a classe r_i estão agrupados corretamente, os outros estão agrupados incorretamente.

A Tabela 2 resume os resultados obtidos nos seis algoritmos de agrupamento para os *benchmarks* já descritos na questão da distância intracluster, ou seja, a média do valor do *fitness* da melhor partícula em todas as execuções e seu desvio padrão ($f \pm \sigma$) para cada um dos métodos descritos neste trabalho. Vale destacar que nos algoritmos que possuem mutação, foram testados dois tipos de porcentagem de mutação, 5% e 10%. Em outras palavras, os algoritmos HPSOM 5%, HPSOM 10%, HPSOMKM 5% e HPSOMKM 10% na Tabela 2 são os testes dos algorítmos HPSOM e HPSOMKM em 5% e 10% de mutação, respectivamente.

O tempo de execução médio de cada método, em 100 execuções, foi medido em milissegundos e é apresentado na Tabela 3, juntamente com o desvio padrão, para a melhor análise dos dados.

Observando a Tabela 3, pode-se perceber que o método *k-means* executa mais rapidamente que todos os outros algoritmos testados neste trabalho. Na Tabela 2, KM

Método	*Iris*	*Wine*	*Glass*
PSO	4516.5 ± 145.44	11943.43 ± 115.73	20274.5 ± 254.18
KM	$\mathbf{6.75 \pm 7.91}$	$\mathbf{33.18 \pm 7.83}$	$\mathbf{51.81 \pm 9.29}$
PSOKM	4370.06 ± 200.37	11157.18 ± 246.25	18771.5 ± 580.02
HPSOM 10%	4514.75 ± 223.71	11244.25 ± 264.12	19143.56 ± 288.55
HPSOM 5%	4526.25 ± 119.73	11217.81 ± 417.12	18730.5 ± 222.78
HPSOMKM 10%	4426.81 ± 36.77	11103.37 ± 122.08	21744.25 ± 452.76
HPSOMKM 5%	4434.62 ± 49.96	11372.06 ± 213.71	21022.56 ± 179.47

Table 3. Tempo de execução dos testes experimentais

obteve bons resultados, muito competitivos em relação ao PSO em todos os *benchmarks* testados, e sempre chega ao mesmo resultado devido a não ser um método estocástico.

O PSO obteve resultados competitivos nas duas bases *Iris* e *Wine* na comparação com o KM. No entanto, PSO apresentou resultados inferiores que o KM na base *Glass*.

Os métodos HPSOM 5% e HPSOM 10% obtiveram melhores resultados que o PSO em todos os testes e valores muito próximos entre si, com exceção dos testes feitos com *Wine*, onde o HPSOM 10% obteve resultados largamente melhores que HPSOM 5%. Também, conseguiram melhores resultados que KM no *Iris* e *Wine*. Além disso, HPSOM 5% e HPSOM 10% conseguiram menor desvio padrão que o PSO, o que significa maior confiabilidade no método.

PSOKM, como dito por [Merwe and Engelbrecht 2003], de fato, refinou o resultado do KM, ou seja, nenhum dos resultados do método *k-means* foi melhor que PSOKM. Além disso, PSOKM foi melhor que os métodos HPSOM 5% e HPSOM 10% nos testes com *Wine* e *Glass* e , também, apresentou menor desvio padrão.

Por fim, o método HPSOMKM obteve os melhores resultados deste trabalho tanto na distância intracluster quanto no desvio padrão. HPSOMKM 5% conseguiu os melhores *clusters* (Figura 3) e HPSOMKM 10% obteve o melhor desvio padrão.

8. Conclusões

Nesse trabalho, foi apresentado um estudo do problema de clusterização, uma das principais tarefas de descoberta de conhecimento em bancos de dados e aplicado em diversas áreas.

Inicialmente, foi apresentado o PSO, *Particle Swarm Optimization*, um algoritmo baseado em comportamento social que tem sido aplicado com sucesso em diversos problemas. Foram apresentados alguns métodos de resolução do problema de clusterização baseados no PSO que tem surgido na literatura nos últimos anos. Dentre eles, o método desenvolvido por [Merwe and Engelbrecht 2003]. Então, foram propostos e testados dois algoritmos híbridos, que usam a ideia de [Merwe and Engelbrecht 2003] em conjunto com *k-means* e um esquema de mutação proposto por [Esmin et al. 2006], para a clusterização de dados.

Três *benchmarks* conhecidos da área foram usados para comparar a eficiência desses métodos. O HPSOMKM demonstrou-se capaz de encontrar boas soluções para o problema, melhores que todas as outras testadas neste trabalho, principalmente em *clus-*

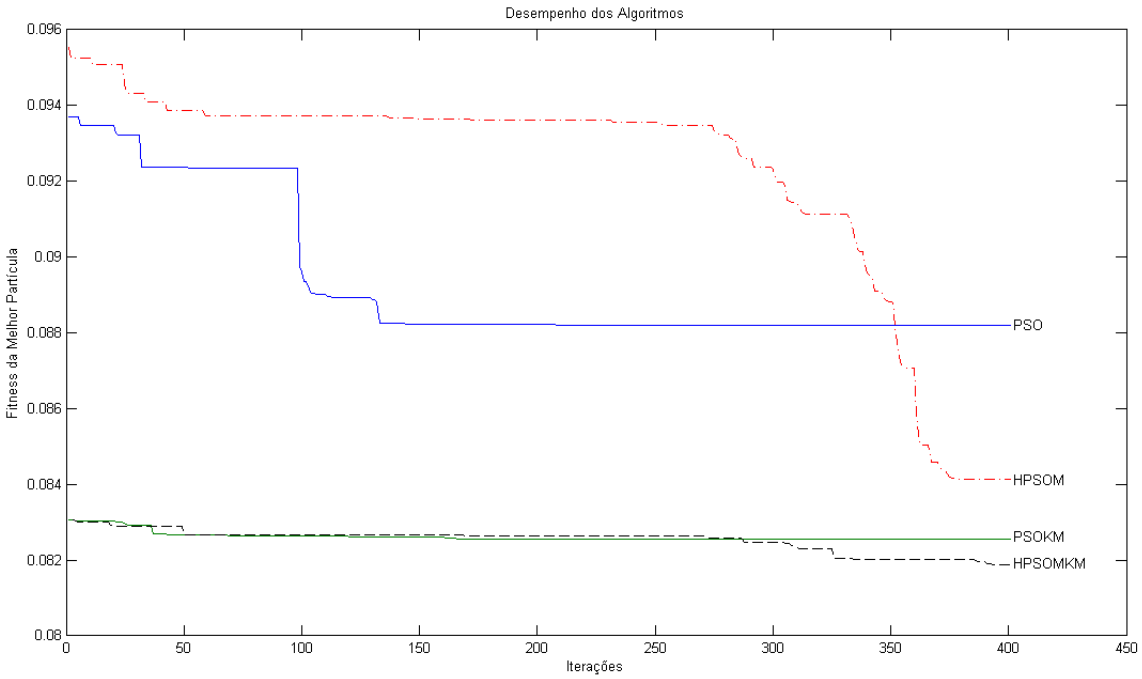


Figure 3. *Fitness* da melhor partícula ao longo das iterações (valores multiplicados por 100, para se obter melhor visibilidade)

ters com formato circular. Os resultados demonstram que é interessante utilizar técnicas híbridas para resolver problemas de Clusterização.

Em trabalhos futuros, deseja-se o desenvolvimento de um método de Clusterização, baseado no PSO, capaz de lidar com *clusters* de forma complexa e seja capaz de determinar o número ideal de clusters.

9. Agradecimentos

Os autores agradecem ao CNPq e à FAPEMIG pelo auxílio dado a esta pesquisa.

References

Asuncion, A. and Newman, D. J. (2007). Uci machine learning repository [http://www.ics.uci.edu/ mlearn/mlrepository.html]. *Irvine, CA: University of California, Department of Information and Computer Science.*

Carlantonio, L. M. (2001). Novas metodologias para clusterização de dados, coordenação de programas de pós-graduação em engenharia. Master's thesis, COPPE/UFRJ.

Cohen, S. C. M. and Castro, L. N. (2006). Data clustering with particle swarms. *Congress on Evolutionary Computation*, Proceedings of IEEE Congress on Evolutionary Computation 2006:1792–1798.

Esmin, A. A. A., Pereira, D. L., and Araújo, A. F. R. (2008). Study of different approach to clustering data by using the particle swarm optimization algorithm. In *IEEE Congress on Evolutionary Computation*, pages 1817–1822.

Esmin, A. A. A., Torres, G. L., and Alvarenga, G. B. (2006). Hybrid evolutionary algorithm based on pso and ga mutation. *6th International Conference on Hybrid Intelligent*

Systems & 4th Conference on Neuro-Computing and Evolving Intelligence. Auckland - Nova Zelândia, pages 57–60.

Kennedy, J. and Eberhart, R. C. (1995). Particle swarm optimization. *IEEE International Conference on Neural Networks, 1995, Perth. Proceedings of IEEE International Conference on Neural Networks*.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability.*, volume 1. University of California Press.

Merwe, D. W. and Engelbrecht, A. P. (2003). Data clustering using particle swarm optimization. *Congress on Evolutionary Computation, 2003. Proceedings of IEEE Congress on Evolutionary Computation*, pages 215–220.

Pereira, D. L. (2007). Estudo do pso na clusterização de dados. Monografia apresentada na UFLA (Universidade Federal de Lavras).

Saramago, S. F. and Prado, J. R. (2005). Otimização por colônia de partículas. *FAMAT em Revista (UFU)*, pages 87–103.

van der Bergh, F. (2001). *An Analysis of Particle Swarm Optimizers*. PhD thesis, (PhD in the Faculty of Natural and Agricultural Science) – University of Pretoria.

Zhang, B., Hsu, M., and Dayal, U. (1999). K-harmonic means -a data clustering algorithm. *Hewlett-Packard Research Laboratory*.

Improved Genetic Operators for the Multiobjective Generalized Assignment Problem

Robert F. Subtil[1], Eduardo G. Carrano[1], Ricardo H. C. Takahashi[2], Marcone J. F. Souza[3], Sérgio R. de Souza[1]

[1]Centro Federal de Educação Tecnológica de Minas Gerais.
Av. Amazonas, 7675
30510-000 – Belo Horizonte, MG – Brazil

[2]Universidade Federal de Minas Gerais
Av. Antônio Carlos, 6627
31270-901 – Belo Horizonte, MG – Brazil

[3]Universidade Federal de Ouro Preto
Campus Universitário
35400-000 – Ouro Preto, MG – Brazil

rsubtil@yahoo.com.br, egcarrano@dppg.cefetmg.br

taka@mat.ufmg.br, marcone@iceb.ufop.br, sergio@dppg.cefetmg.br

Abstract. *The original formulation of the Generalized Assignment Problem (GAP) consists in, given a set of n different tasks and m different agents, assigning each task to an agent in such a way that a cost function is minimized. A previous work introduced the Equilibrium Function as a new objective function in the problem formulation. The purpose of this second objective function is to minimize the maximum difference between the amount of work assigned to the agents. This allows better distributions of the tasks between the agents than the results found from the original problem, with a small increase in the cost. This paper proposes new crossover and mutation operators that produce improvements in the algorithm presented in [Subtil et al. 2010], leading to considerably better Pareto approximations than the ones obtained in the previous work, within the same number of function evaluations. The proposed operators exploit problem-specific information in a probabilistic way, performing operations that lead to objective function enhancement or feasibility enhancement with greater probability than operations that do not cause such enhancements. A statistical comparison procedure is employed for supporting such conclusions.*

1. Introduction

The Generalized Assignment Problem (GAP) is a classical $\mathcal{NP}$-Hard problem [Sahni and Gonzalez 1976, Chu and Beasley 1997] which is used for modeling several practical design problems, such as job assignment in computer networks [Balachandran 1976], parallel machine scheduling [Lenstra et al. 1990], multiprocessor scheduling [Turek et al. 1992], facility location [Ross and Soland 1977] and vehicle routing [Fisher and Jaikumar 2006].

In its usual form, the GAP is stated as follows: given a set $\mathcal{T}$ of tasks, a set $\mathcal{A}$ of agents, the cost of each task j when it is accomplished by each agent i and the amount of resources required by each agent i for completing each task j, find the optimal assignment X such that the sum of the costs spent for accomplishing all tasks is minimum, ensuring, on the other hand, that each task is assigned to a single agent and that the total of resources demanded from agent i is lower than its upper bound.

Therefore, if an optimization algorithm is employed to solve such problem, it should ideally find the less expensive assignment policy, regardless the distribution of the tasks amongst the agents. This characteristic of this optimal solution may render it unsuitable for several practical problems, for instance:

People management: If some people of a team receives much more work than other people, they tend to become dissatisfied and their efficiency may decline due to the excessive workload.

Computer networks or parallel computing: A task allocation policy that concentrates the processing in some few resources is more likely to cause bottlenecks than a more balanced distribution policy.

In practice, a more equitable distribution of the tasks can be helpful in any situation in which load balance can be relevant. In [Subtil et al. 2010] a multiobjective adaptation of the original GAP (referred as Multiobjective GAP or MoGAP) is proposed. In this new formulation, a second objective function, called Equilibrium Function, is introduced. The purpose of this function is to guide the search in order to reach solutions which provide better distribution of the tasks amongst the agents. Since two objective functions are considered, the solution of this problem is a set of efficient points, in which a solution can not be chosen without introducing any preference.

This paper proposes an extension of the integer NSGA-II proposed in [Subtil et al. 2010]. The new algorithm employs improved versions of the crossover and mutation operators, leading to considerably better Pareto approximations than the ones obtained by the former algorithm. In order to support such a comparison, a detailed statistical evaluation procedure is employed. Comparisons of the quality of the approximations achieved and the quality of a deterministically chosen solution are also performed.

This paper has the following structure: the formulation of the MoGAP is shown in section 2; the improved algorithm, which is proposed for dealing with MoGAP, is presented in section 3; finally, a statistical comparison of the two algorithms which have been considered is performed on section 4.

2. The Multiobjective Generalized Assignment Problem

Let $\mathcal{A}$ be the set of agents, $\mathcal{T}$ be the set of tasks, $c_{i,j}$ be the cost of assigning the task $j \in \mathcal{T}$ to the agent $i \in \mathcal{A}$, $a_{i,j}$ be the amount of resources required by the agent $i \in \mathcal{A}$ for performing the task $j \in \mathcal{T}$, b_i be the total resource capacity of agent $i \in \mathcal{A}$ and $x_{i,j}$ be a binary decision variable, which assumes value "1" if the task $j \in \mathcal{T}$ is assigned to agent $i \in \mathcal{A}$ or value "0", otherwise. The MoGAP can be stated as [Subtil et al. 2010, Yagiura et al. 2006]:

$$\mathcal{X}^* = \arg\min_x \begin{bmatrix} f_C(x) \\ f_E(x) \end{bmatrix} \tag{1}$$

$$\text{subject to : } \begin{cases} (G1): \sum_{j \in \mathcal{T}} a_{i,j} \cdot x_{i,j} - b_i \leq 0 & , \forall i \in \mathcal{A} \\ (G2): \sum_{i \in \mathcal{A}} x_{i,j} - 1 = 0 & , \forall j \in \mathcal{T} \\ (G3): x_{i,j} \in \{0,1\} & , \forall i \in \mathcal{A},\ \forall j \in \mathcal{T} \end{cases} \tag{2}$$

in which:

$$f_C(x) = \sum_{i \in \mathcal{A}} \sum_{j \in \mathcal{T}} c_{i,j} \cdot x_{i,j} \tag{3}$$

$$f_E(x) = \max_{i \in \mathcal{A}} \left(\sum_{j \in \mathcal{T}} a_{i,j} \cdot x_{i,j} \right) - \min_{i \in \mathcal{A}} \left(\sum_{j \in \mathcal{T}} a_{i,j} \cdot x_{i,j} \right) \tag{4}$$

In this formulation, the equations (3) and (4) are the cost and equilibrium functions, respectively. The equilibrium is modeled as the difference between the most busy agent and the most vacant one. This difference should be minimized. In (2), the constraint $g_1(x)$ ensures that the maximum resource capacity of each agent is not violated, the constraint $g_2(x)$ ensures that each task is assigned to a single agent and the constraint $g_3(x)$ defines that each variable $x_{i,j}$ is binary. An important feature of this formulation is that it cannot be solved by an integer linear programming method, since $f_E(\cdot)$ is not linear.

3. Proposed Algorithm

The characteristics of the problem, which is combinatorial, multiobjective and has a non-linear objective function, preclude the employment of most part of the deterministic optimization methods. The genetic algorithms are particularly adequate for dealing with the MoGAP, since they are suitable for handling with multiobjective optimization problems with rather arbitrary objective and constraint functions [Coello 2000, Fonseca and Fleming 1995]. The algorithm proposed here is an improved version of the one described in [Subtil et al. 2010]. It is based on the NSGA-II [Deb et al. 2002], and includes also some problem-specific solution encoding, crossover and mutation mechanisms. A general scheme for this algorithm can be seen in Algorithm 1.

In this algorithm:

- $P \leftarrow$ **new_population**(N) generates a random population with N feasible individuals;
- $A \leftarrow$ **non_dominated**(P) returns the individuals which lie in the first front of the population P;
- $\mathcal{F} \leftarrow$ **fast_non_dominated_sorting**(P) employs *fast non-dominated sorting* [Deb et al. 2002] to find the front of each solution in the population P;
- $\mathcal{C}_i \leftarrow$ **crowding_distance_assignment**$(\mathcal{F}_i)$ employs *crowding distance assignment* [Deb et al. 2002] to estimate how the solutions of front i are spread in the objective space;
- $\mathcal{F}_i \leftarrow$ **sort**$(\mathcal{F}_i, \mathcal{C}_i,$ 'descending') sorts the solutions of front i in descending order of $\mathcal{C}_i$;

Algorithm 1 Pseudocode for NSGAII

procedure NSGAII(N,N_A)
  $t \leftarrow 0$
  $P_t \leftarrow$ **new_population**(N)
  $Q_t \leftarrow \emptyset$
  $A \leftarrow$ **non_dominated**(P_t)
  while not stop criterion **do**
    $R_t \leftarrow P_t \cup Q_t$
    $\mathcal{F} \leftarrow$ **fast_non_dominated_sorting**(R_t)
    $P_{t+1} \leftarrow \emptyset$
    $i \leftarrow 1$
    while $|P_{t+1}| + |\mathcal{F}_i| \leq N$ **do**
      $\mathcal{C}_i \leftarrow$ **crowding_distance_assignment**$(\mathcal{F}_i)$
      $P_{t+1} \leftarrow P_t \cup \mathcal{F}_i$
      $i \leftarrow i + 1$
    end while
    $\mathcal{F}_i \leftarrow$ **sort**$(\mathcal{F}_i, \mathcal{C}_i,$ 'descending')
    $P_{t+1} \leftarrow P_{t+1} \cup \mathcal{F}_i[1 : (N - |P_{t+1}|)]$
    $Q_{t+1} \leftarrow$ **selection**(P_{t+1}, N)
    $Q_{t+1} \leftarrow$ **GAPCrossover**(Q_{t+1})
    $Q_{t+1} \leftarrow$ **GAPMutation**(Q_{t+1})
    $t \leftarrow t + 1$
    $A \leftarrow$ **non_dominated**$(A \cup Q_t)$
  end while
end procedure

- $Q \leftarrow$ **selection**(P, N) uses binary stochastic tournaments for performing selection of the population P. The outcome of this procedure is a population Q, with N individuals which have been selected from P with replacement.

The selection, fitness assignment, niche and elitism procedures employed in the proposed algorithm are identical to the canonical ones of NSGA-II, presented in [Deb et al. 2002]. The encoding scheme, crossover and mutation operators have been chosen in order to adapt to the problem structure. These components are described in the sequel.

3	4	4	2	1	4	3	2	2	4

Figure 1. Example of the encoding scheme. In this candidate solution, the task 5 is assigned to agent $A1$, the tasks 4, 8 and 9 are assigned to agent $A2$, the tasks 1 and 7 are assigned to agent $A3$ and the tasks 2, 3, 6 and 10 are assigned to agent $A4$.

3.1. Encoding Scheme

The encoding scheme adopted here was the same proposed in [Subtil et al. 2010]. In this representation, each candidate solution is represent by a $1 \times |\mathcal{T}|$ integer vector. Each position of this vector can assume any integer value in $\{1, |\mathcal{A}|\}$. An example of encoding for an instance with 4 agents and 10 tasks is shown in Figure 1.

3.2. GAPCrossover

The crossover operator introduced in this paper, called GAPCrossover, is a modification of the original uniform crossover, commonly employed in binary GAs. However, it carries some knowledge about the problem, performing the steps described in Algorithm 2.

Algorithm 2 GAPCrossover

Input: parent solutions p_1 and p_2

Output: offspring solutions o_1 and o_2

1: generate a random binary word (ref) of length $|\mathcal{T}|$;
2: generate a scalar d_T, at random, following an uniform distribution;
3: $o_1 \leftarrow p_1$
4: $o_2 \leftarrow p_2$
5: **for** each position j in which $ref \neq 0$ **do**
6: **if** $d_T < 0.50$ **then**
7: swap $o_1(i)$ and $o_2(i)$;
8: **else**
9: $o'_1 \leftarrow o_1$
10: $o'_2 \leftarrow o_2$
11: swap $o_1(i)'$ and $o_2(i)'$;
12: **if** $g_1(o'_1) \leq 0$ and $g_1(o'_2) \leq 0$ **then**
13: **if** $c_{o_1(i)',j} < c_{o_2(i)',j}$ **then**
14: $o_1 \leftarrow o'_1$
15: $o_2 \leftarrow o'_2$
16: **end if**
17: **else if** $g_1(o'_1) \leq g_1(p_1)$ and $g_1(o'_2) \leq g_1(p_2)$ **then**
18: $o_1 \leftarrow o'_1$
19: $o_2 \leftarrow o'_2$
20: **end if**
21: **end if**
22: **end for**

From this scheme, it is possible to note that the algorithm has 50% of probability of performing the recombination, and 50% of probability of swapping the selected positions only in the case of a improvement condition is verified. In the first case, the operator is identical to the original uniform crossover, while, in the second case, the operator only swaps the values which provide one of the following improvements:

- the solution o_1 improves the cost of parent p_1; or
- the two new solutions inflict the constraint $g_1(\cdot)$ less than their respective parents.

If none of these conditions are not observed, then the positions are kept unchanged. An example of this operator, when it reduces to the uniform crossover case, can be seen in Figure 2.

It should be noticed that this operator generates less unfeasible individuals than the original uniform crossover, since a "feasibility-oriented" mechanism is included in some conditions. In some tests, this mechanism was executed in all situations, in order to completely avoid the generation of unfeasible solutions. This version of the operator was

p_1	3	4	4	2	1	4	3	2	2	4
p_2	1	3	4	1	1	2	2	3	2	3
ref	0	1	0	0	1	1	1	0	0	1
o_1	3	3	4	2	1	2	2	2	2	3
o_2	1	4	4	1	1	4	3	3	2	4

Figure 2. Example of uniform crossover: Let $d_T < 0.50$. p_1 is copied to o_1 and p_2 is copied to o_2. Then, the values of o_1 and o_2 are swapped in the positions in which the $ref(i) = 1$ (gray columns).

discarded, because it presented lower performance than the one shown here. The authors believe that the degradation of performance can be credited to the loss of population diversity caused by such an operator.

3.3. GAPMutation2

The GAPMutation2 proposed in this paper is a small variation of the GAPMutation proposed in [Subtil et al. 2010]. Both operators are based on the flip mutation, and employ some kind of additional procedure for incorporating problem knowledge, as shown in Algorithm 3.

Algorithm 3 GAPMutation2

Input: parent solution p

Output: offspring solution o

1: choose a position $j \in \{1, |\mathcal{T}|\}$, at random;
2: generate a scalar r_D, at random, following uniform distribution;
3: $o \leftarrow p$
4: **if** $r_D \leq 0.20$ **then**
5: replace $o(j)$ by a new agent $a \neq p(j)$, at random;
6: **else if** $r_D > 0.20$ and $r_D \leq 0.50$ **then**
7: replace $o(j)$ by the agent a with the highest contribution $f_E(m)$ (least required agent) and $a^* \neq p(j)$;
8: **else**
9: replace $o(j)$ by the agent a^* in which $a^* = \arg\min_a c_{a,j}$ with $a^* \neq p(j)$, ensuring that $g_1(o) \leq 0$;
10: **end if**

p	3	4	4	2	1	4	3	2	2	4
						⇓				
o	3	4	4	2	1	1	3	2	2	4

Figure 3. Example of GAPMutation2. Let $j = 6$ and $0.20 < r_D < 0.50$ and assume that all tasks demand the same amount of resources of all agents. The task 6 has been re-assigned to the least required agent, which is $A1$.

Such as for GAPCrossover, a version of GAPMutation which ensures the feasibility of all obtained solutions had been tested. This version was not considered on further

implementations of the algorithm since it reduced the global efficiency of the method. An example of the GAPMutation2 can be found in Figure 3.

3.4. Constraint Handling

The solution encoding and the operators described earlier in this section ensure that $(G2)$ and $(G3)$ constraints (see equation 2) are always valid. However, some additional mechanism is necessary to employ for handling $(G1)$, since it is not possible to *a priori* ensure that this constraint is valid for all individuals of the algorithm. Following the same methodology proposed in [Subtil et al. 2010], an exponential penalty function was employed in order to penalize the solutions which do not comply with such a constraint. The main difference between this approach and the previous one is that, here, the penalty function is applied to both objective functions simultaneously. These modified objective functions are given by:

$$f_C^m(x) = f_C(x) \cdot \epsilon^{r_{ind}} \tag{5}$$

$$f_E^m(x) = f_E(x) \cdot \epsilon^{r_{ind}} \tag{6}$$

in which

$$r_{ind} = \sum_{i \in \mathcal{A}} \max \left\{ 0, \sum_{j \in \mathcal{T}} (a_{ij} \cdot x_{ij}) - b_i \right\}$$

4. Numerical Results

In reference [Subtil et al. 2010], it has been proposed an integer enhanced NSGA-II for solving the MoGAP. That algorithm was compared with a basic NSGA-II and with an exact method for the mono-objective problem. The comparison, in [Subtil et al. 2010], was guided by visual inspection, based on five independent runs of each algorithm. It was noticed that the algorithm proposed there clearly dominated the basic one, and was able to provide good approximations to the exact optimal cost solution in five of six instances. Besides, it was always possible to choose a solution which provided favorable ratio *% gain in equilibrium* vs. *% loss in cost*. The algorithm of [Subtil et al. 2010] will be referred in the remainder of this paper as $\mathcal{A}_1$.

The current work proposes an enhancement of that previous algorithm, in order to improve its convergence. This new algorithm, denoted by $\mathcal{A}_2$, is compared with the previous version, $\mathcal{A}_1$. The comparison with the basic GA is not performed here, since this algorithm is clearly dominated by $\mathcal{A}_1$. Before presenting the results, it is important to explain how the comparisons were conducted.

Comparison scheme

The algorithm comparison methodology which is considered here is inspired on the evaluation schemes proposed in [Carrano et al. 2008, Carrano et al. 2011]. It can be summarized as follows:

- Each algorithm is executed k times, for a fixed number of function evaluations (stop criterion).

- For each algorithm i:
  - For each run j:
    - Evaluate a merit criterion, $m_{crit}(i, j)$, of the final archive achieved in the run j of the algorithm i.
  - Perform bootstrapping [Efron 1979] with the values $m_{crit}(i, :)$. Each bootstrapping iteration delivers a sample of the mean of the merit function for the algorithm i. The whole set provided by the bootstrapping is an empirical approximation of the probability distribution function (PDF) of such a mean.
- Compare the empirical PDF's of the algorithms using One-Way ANOVA [Lindman 1974], for a significance level $\alpha = 0.05$.

Comparison criteria

Two independent merit criteria have been considered in the comparison:

- Hyper-volume indicator: the hyper-volume indicator [Zitzler 1999] is used to estimate the quality of the final Pareto approximation achieved in each run;
- Decision-making: a solution is automatically chosen for each algorithm run. Such a solution is the one which maximizes:

$$i^* = \arg \min_i \frac{f_E^{*C} - f_E^i}{f_C^i - f_C^{*C}} \cdot \frac{f_C^{*C}}{f_E^{*C}}$$

$$\text{subject to: } \begin{cases} f_C^i \neq f_C^{*C} \\ f_E^i \neq f_E^{*C} \end{cases}$$

  in which $[f_C^i \;\; f_E^i]'$ is the function vector assigned to the solution i and $[f_C^{*C} \;\; f_E^{*C}]'$ is the function vector of the optimal cost solution (known a priori).

The rule for decision-making has been chosen arbitrarily and, therefore, it could be replaced by any other reasonable rule. The main goal of considering these two criteria is to estimate the quality of the global Pareto approximation found and the quality of a particular solution picked from such approximations.

Computational procedures

The algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ have been tested in six instances, which have been taken from the OR-Library [Beasley]:

- $A05 \times 100$: 5 agents and 100 tasks;
- $A05 \times 200$: 5 agents and 200 tasks;
- $A10 \times 100$: 10 agents and 100 tasks;
- $A10 \times 200$: 10 agents and 200 tasks;
- $A20 \times 100$: 20 agents and 100 tasks;
- $A20 \times 200$: 20 agents and 200 tasks.

Both algorithms have been set with the following parameters:

- Number of runs: 100 runs per algorithm;
- Population size: 50 individuals;
- Stop criterion: maximum number of generations;
- Maximum number of generations: 300 generations;
- Crossover probability: 0.90 per pair;
- Mutation probability: 0.03 per integer.

The analysis of the results observed in the six instances has been divided in two sections, one for each merit criterion.

Table 1. Hyper-volume indicator - 95% confidence interval

	95% confidence interval	
instance	$\mathcal{A}_1$	$\mathcal{A}_2$
$A05 \times 100$	$[32333\ ;\ 33711]$	$[32788\ ;\ 34105]$
$A05 \times 200$	$[295322\ ;\ 305575]$	$[307332\ ;\ 311554]$
$A10 \times 100$	$[103116\ ;\ 111013]$	$[106169\ ;\ 112955]$
$A10 \times 200$	$[279897\ ;\ 301474]$	$[308347\ ;\ 322609]$
$A20 \times 100$	$[88885\ ;\ 100839]$	$[95353\ ;\ 106252]$
$A20 \times 200$	$[226007\ ;\ 254017]$	$[263840\ ;\ 290605]$

4.1. Hyper-volume indicator

After running the algorithms 100 times each, the Pareto approximations have been used for finding the values of the hyper-volume associated to each algorithm. These values have been used for finding 95% confidence intervals, which are shown in Table 1.

From Table 1, the bounds of the intervals observed for $\mathcal{A}_2$ are always higher than the ones noticed for $\mathcal{A}_1$. Since higher values of hyper-volume indicate better Pareto approximations, this analysis suggests that the algorithm $\mathcal{A}_2$ has found better results than $\mathcal{A}_1$. This hypothesis is supported by the ANOVA tests, which have indicated that there are significant differences between $\mathcal{A}_1$ and $\mathcal{A}_2$ in all instances[1].

In order to perform an evaluation of the amount of the difference that was found, a visual comparison of the Pareto approximations has been performed. Three Pareto approximations have been chosen: the approximation with lower hyper-volume, the median and the approximation with higher hyper-volume. These comparisons have shown that the Pareto approximations achieved by $\mathcal{A}_2$ always dominate most part of the $\mathcal{A}_1$ solutions. This can be noticed in Figure 4, which illustrates such a comparison for a median run.

4.2. Decision-making

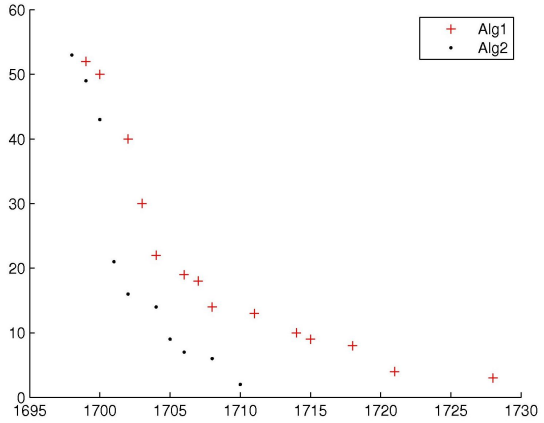
A deterministic rule for choosing a solution from each Pareto approximation has been adopted: the solution which provides better ratio *gain in equilibrium* vs. *loss in cost*. Therefore, a value of maximum ratio is assigned to each algorithm run and these values are used in the statistical comparison scheme discussed earlier in this section. The 95% confidence intervals for these ratios are shown in Table 2.

These intervals indicate that the algorithm $\mathcal{A}_2$ is considerably better than $\mathcal{A}_1$ in this merit criterion, since it is able to provide solutions with better ratios in all instances considered here. Such a conclusion was corroborated by the ANOVA tests, which detected significant differences between the algorithms in the six instances[2].

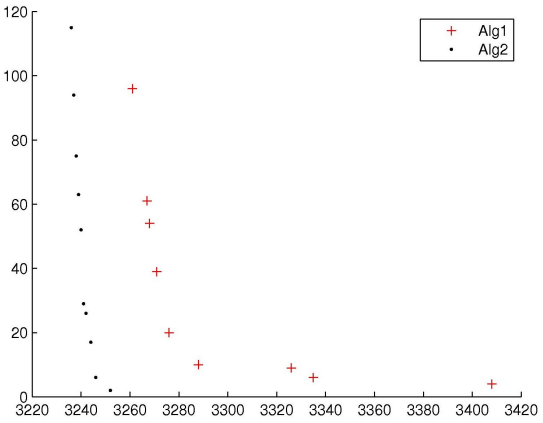
Considering both the Pareto approximation quality and the decision-making step, it can be concluded that the new proposed algorithm is better than the former one, at least for the instances considered.

[1]The higher p-value observed was lower than 1×10^{-16}.

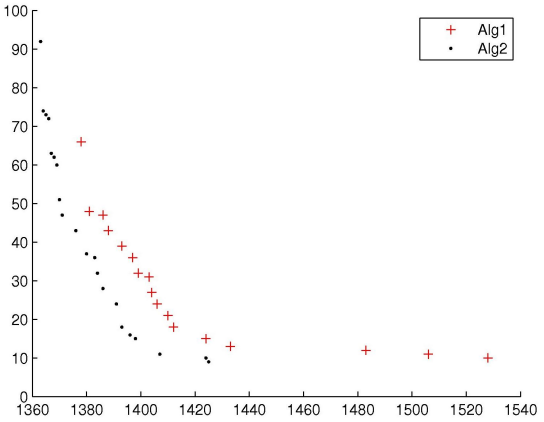
[2]All p-values were lower than 1×10^{-16}.



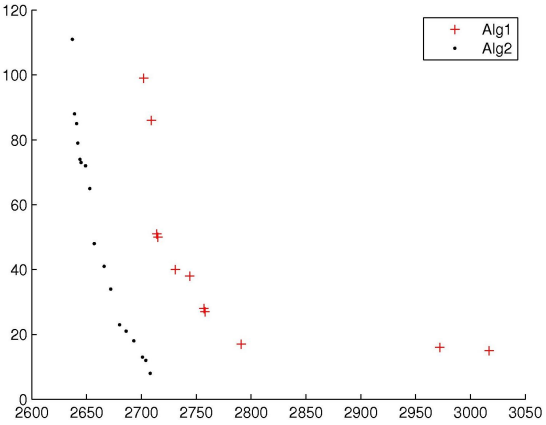
(a) $A05 \times 100$



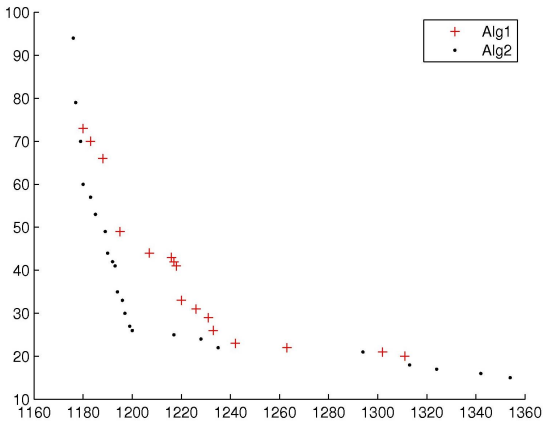
(b) $A05 \times 200$



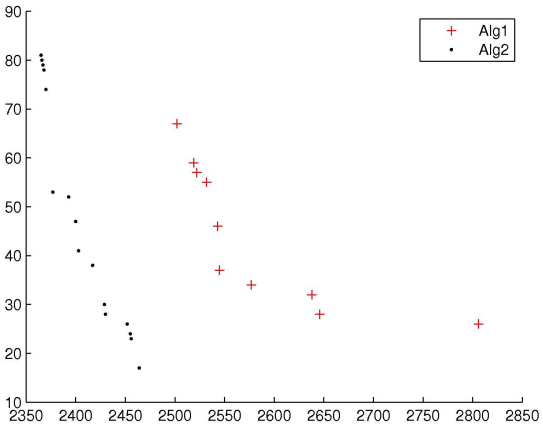
(c) $A10 \times 100$



(d) $A10 \times 200$



(e) $A20 \times 100$



(f) $A20 \times 200$

Figure 4. Comparison of the Pareto approximations (for medians)

Table 2. Decision-making - 95% confidence interval

	95% confidence interval	
instance	$\mathcal{A}_1$	$\mathcal{A}_2$
$A05 \times 100$	$[80.1\ ;\ 341.7]$	$[245.6\ ;\ 400.5]$
$A05 \times 200$	$[43.2\ ;\ 2280.0]$	$[518.4\ ;\ 3135.0]$
$A10 \times 100$	$[23.5\ ;\ 70.9]$	$[50.6\ ;\ 115.7]$
$A10 \times 200$	$[9.5\ ;\ 25.6]$	$[43.3\ ;\ 275.4]$
$A20 \times 100$	$[5.3\ ;\ 21.2]$	$[18.2\ ;\ 61.6]$
$A20 \times 200$	$[4.0\ ;\ 9.9]$	$[18.2\ ;\ 81.1]$

5. Conclusion

An enhanced version of an integer NSGA-II algorithm, specialized for the multiobjective generalized assignment problem (MoGAP), is proposed in this paper. This algorithm introduces two new specific genetic operators for improving the search for efficient solutions which perform with a higher probability operations that enhance the cost function or which enhance the solution feasibility. It is interesting to notice that these operators still perform stochastic operations that may still perform operations that do not immediately enhance the solutions. This has been found to be much better than deterministic local searches.

A statistical comparison of the proposed algorithm and the original version indicates that the new algorithm is more efficient for solving the considered problem. With identical computational cost, the new algorithm was able to find Pareto approximations which dominate the ones achieved by the former algorithm. Besides, the employment of a decision-making procedure has shown that the new approach delivers solutions with more favorable trade-offs, which allow significant gains in the task homogeneity assignment with small losses in the cost.

Acknowledgments

The authors acknowledge the support from the Brazilian agencies FAPEMIG, CAPES and CNPq.

References

Balachandran, V. (1976). An integer generalized transportation model for optimal job assignment in computer networks. *Operations Research*, 24:742–759.

Beasley, J. E. Or-library. http://www.brunel.ac.uk/depts/ma/research/jeb/orlib/gapinfo.html.

Carrano, E. G., Takahashi, R. H. C., and Wanner, E. F. (2008). An enhanced statistical approach for evolutionary algorithm comparison. In *Proceedings of the Genetic and Evolutionary Computation Conference - GECCO'08*, Atlanta, USA.

Carrano, E. G., Wanner, E. F., and Takahashi, R. H. C. (2011). A multi-criteria statistical based comparison methodology for evaluating evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*. to appear.

Chu, P. C. and Beasley, J. E. (1997). A genetic algorithm for the generalised assignment problem. *Computers and Operations Research*, 24:17–23.

Coello, C. A. C. (2000). An updated survey of GA-based multiobjective optimization techniques. *ACM Computing Surveys*, 32(2):109–143.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7:1–26.

Fisher, M. L. and Jaikumar, R. (2006). A generalized assignment heuristic for vehicle routing. *Networks*, 11:109–124.

Fonseca, C. M. and Fleming, P. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation*, 3(1):1–16.

Lenstra, J. K., Shmoys, D. B., and Tardos, E. (1990). Approximation algorithms for scheduling unrelated parallel machines. *Mathematical Programming: Series A and B*, 46:259–271.

Lindman, H. R. (1974). *Analysis of Variance in Complex Experimental Designs*. W. H. Freeman & Co., San Francisco, USA.

Ross, G. T. and Soland, R. M. (1977). Modeling facility location problems as generalized assignment problems. *Management Science*, 24:345–357.

Sahni, S. and Gonzalez, T. (1976). P-complete approximation problems. *Journal of the ACM*, 23:555–565.

Subtil, R. F., Carrano, E. G., Souza, M. J. F., and Takahashi, R. H. C. (2010). Using an enhanced integer NSGA-II for solving the Multiobjective Generalized Assignment Problem. In *Proc. IEEE World Congress on Computational Intelligence*, Barcelona.

Turek, J., Wolf, J. L., and Yu, P. S. (1992). Approximate algorithms scheduling parallelizable tasks. In *Proc. ACM Symposium on Parallel Algorithms and Architectures*, San Diego, USA.

Yagiura, M., Glover, F., and Ibaraki, T. (2006). A path relinking approach with ejection chains for the generalized assignment problem. *European Journal of Operational Research*, 169:548–569.

Zitzler, E. (1999). *Evolutionary algorithms for multiobjective optimization: Methods and applications*. PhD thesis, Computer Engineering and Networks Laboratory - Swiss Federal Institute of Technology Zurich.

SIMPel: Sistema Inteligente Híbrido para Apoio a Manutenção Preditiva em Equipamentos Lubrificados

Fabiano Losilla de Carvalho[1], **Veronica Oliveira de Carvalho**[2]

[1]Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP)
Araraquara – SP – Brasil

[2]Instituto de Geociências e Ciências Exatas – Univ Estadual Paulista (UNESP)
Caixa Postal 178 – Rio Claro – SP – Brasil

logsilla@gmail.com, veronica@unesp.com.br

Abstract. *Aiming to improve productivity ALL, as many other companies, has sought to improve its predictive maintenance processes to ensure equipment availability. Such maintenance is performed on lubricated equipments through oil analysis and other information about the equipment. However, obtaining a consistent diagnosis is laborious due to the amount of rules and conditions involved. Therefore, this article presents the SIMPel, a hybrid intelligent system, based on fuzzy set theory and expert systems, that automates the diagnosis of maintenance of ALL's lubricated equipments. The system is in use, and presented an excellent performance in tests.*

Resumo. *Em busca do aumento da produtividade a ALL, como muitas outras empresas, tem buscado aprimorar seus processos de manutenção preditiva para garantir a disponibilidade dos equipamentos. Essa manutenção em equipamentos lubrificados é realizada por meio da análise de óleo e outras informações sobre o equipamento. Entretanto, a obtenção de um diagnóstico consistente é trabalhosa devido à quantidade de regras e condições envolvidas. Nesse contexto, este artigo apresenta o SIMPel, um sistema inteligente híbrido, baseado na teoria de conjuntos fuzzy e sistemas especialistas, que automatiza o diagnóstico de manutenção dos equipamentos lubrificados da ALL. O sistema encontra-se em uso, tendo apresentado um excelente desempenho no testes realizados.*

1. Introdução

A necessidade das empresas de serem cada vez mais competitivas faz com que elas busquem a cada dia novas técnicas visando o aumento de produtividade e, consequentemente, maiores lucros. Uma maneira de garantir essa produtividade é assegurando a disponibilidade dos equipamentos da empresa, a qual pode ser alcançada com uma boa política de manutenção. Dentre as políticas de manutenção encontra-se a preditiva, a qual verifica a evolução do desgaste dos equipamentos. Esse tipo de manutenção só é realizada quando se constata um desgaste excessivo no equipamento, possibilitando que a ação seja tomada antes que o equipamento efetivamente falhe, reduzindo assim os custos.

A realização de manutenção preditiva em equipamentos lubrificados como locomotivas, empilhadeiras, etc., pode ser realizada a partir da análise de óleo. Essa análise, utilizada em diversos setores das indústrias, tem como objetivo verificar a deterioração

do lubrificante e, juntamente com outras informações, controlar o estado operativo dos equipamentos [Toms and Toms 2010]. Assim, a ação preditiva de manutenção dos equipamentos é condicionada, entre outros fatores, ao diagnóstico de deterioração. Esse diagnóstico é obtido considerando os resultados das análises laboratoriais do óleo, as quais podem ser realizadas por diferentes métodos [Mobley 2002], os quais quantificam os contaminantes (ferro, etc.) encontrados no óleo. Entretanto, os resultados obtidos por esses métodos devem ser interpretados considerando-se os modelos de equipamento, uma vez que os limites de contaminação utilizados para avaliar as condições do óleo e, posteriormente, o estado operativo do equipamento, são dependentes da especificação do modelo e da operação por ele desenvolvida (carregamento de minério de ferro, etc.).

Nesse contexto, foi desenvolvido um sistema inteligente híbrido, denominado SIMPel, para apoiar os especialistas da empresa América Latina Logística (ALL)[1] na tarefa de manutenção preditiva, especificamente os alocados na Oficina de Manutenção de Locomotivas localizada na cidade de Araraquara-SP. O SIMPel, objeto de estudo deste artigo, visa informar a ação preditiva a ser realizada a fim de melhorar a disponibilidade dos equipamentos.

Antes do SIMPel, os especialistas da ALL realizavam manualmente todo o processo de obtenção do diagnóstico preditivo. A obtenção de um diagnóstico consistente se torna trabalhosa devido à quantidade de regras e condições: para se definir as ações preditivas, os especialistas se deparam com uma grande diversidade de equipamentos e com diferentes ambientes aos quais os equipamentos podem ser expostos, fatores que influenciam a decisão a ser tomada. Além disso, o histórico de manutenção dos equipamentos também contribui significativamente para a decisão. Portanto, o esforço envolvido para se obter um único diagnóstico é grande. Desse modo, diante da necessidade de se desenvolver um sistema capaz de lidar com várias regras e condições, a fim de facilitar a obtenção de um diagnóstico preditivo consistente, evitando a sobrecarga da equipe técnica, o SIMPel foi desenvolvido.

Avaliando a problemática e os procedimentos da empresa, observou-se que o SIMPel poderia ser modelado adequadamente como um sistema inteligente híbrido associando-se a teoria de conjuntos fuzzy [Zadeh 1965, Berkan and Trubatch 1997] com a técnica de sistemas especialistas [Giarratano and Riley 2004, Negnevitsky 2005]. O SIMPel, a partir dos resultados das análises laboratoriais, classifica inicialmente o estado do óleo em categorias (Bom, Regular ou Ruim), detectando os contaminantes que contribuíram para a classificação. Em seguida, utiliza a classificação do óleo e os contaminantes identificados como parte do conhecimento necessário para que seja realizado um diagnóstico que permita identificar qual ou quais as ações que devem ser realizadas para evitar a quebra do equipamento.

O SIMPel apresentou um excelente desempenho durante os testes realizados, aproximadamente 97%, diagnosticando inclusive situações que antes não eram notadas pelo especialista. O SIMPel encontra-se em uso e em estudo para ser integrado ao SAP da empresa.

Este artigo encontra-se estruturado da seguinte maneira: na Seção 2 são apresentados os procedimentos adotados antes do SIMPel, os métodos de análise de óleo utilizados

[1] http://www.all-logistica.com/port/index.htm.

pela empresa e, portanto, considerados neste trabalho, assim como outros trabalhos que resolvem esse mesmo tipo de problema; a Seção 3 apresenta o SIMPel e seu funcionamento; a Seção 4 apresenta os resultados obtidos durante os testes; por fim, a Seção 5 apresenta a conclusão e os trabalhos futuros.

2. Background, Conceitos e Trabalhos Relacionados

A princípio, é necessário compreender os procedimentos adotados pela empresa antes do SIMPel para que se entenda a necessidade de se desenvolver um sistema de apoio aos especialistas da ALL e a importância de se utilizar técnicas inteligentes para modelar o problema. Para tanto, essa seção apresenta na Seção 2.1 uma descrição desses procedimentos e a justificativa de escolha da utilização de sistemas fuzzy e especialista com base nessa descrição. Além disso, essa seção apresenta também alguns conceitos necessários ao entendimento do artigo (Seção 2.2) e alguns trabalhos relacionados (Seção 2.3).

2.1. Background

Como mencionado anteriormente, os especialistas da ALL realizavam manualmente todo o processo de obtenção do diagnóstico preditivo. Inicialmente, após a obtenção dos resultados das análises laboratoriais, os especialistas qualificavam linguisticamente cada contaminante contido no óleo em Bom, Regular ou Ruim. Para tanto, utilizavam tabelas adaptadas das constantes no manual do fabricante do equipamento em análise (alteradas de modo a atender as necessidades reais da empresa). Essas tabelas contém os limites de contaminação do óleo que o tornam prejudicial ao equipamento. É importante mencionar que embora os especialistas usassem valores linguísticos (imprecisos em sua natureza), os mesmos estavam associados a limites exatos de contaminação. Em seguida, com base em uma nova tabela por eles construída a partir de suas experiências, os especialistas classificavam o óleo em Bom, Regular ou Ruim, a partir da qualificação atribuída anteriormente aos contaminantes, tornando essa classificação independente do modelo de equipamento utilizado. A partir dessa classificação, os especialistas examinavam a base histórica do equipamento em busca de informações relevantes a fim de emitir um diagnóstico sob a condição operativa do mesmo.

Desse modo, a decisão por se utilizar a teoria de conjuntos fuzzy foi tomada para se realizar a classificação do óleo em função das características do domínio: (i) uma vez que os especialistas qualificam linguisticamente, inicialmente, cada contaminante como Bom, Regular ou Ruim, cada contaminante pode ser modelado como uma variável linguística de entrada, cujos conjuntos fuzzy são definidos em função dos limites de contaminação. Desse modo, se cada modelo contiver seus respectivos conjuntos fuzzy associados a cada variável linguística de entrada é possível qualificar os contaminantes qualquer que seja o modelo utilizado; (ii) assim como os especialistas faziam manualmente, a partir do momento que os contaminantes são qualificados, pode-se utilizar uma base única de regras para classificar o óleo, já que a mesma será composta pela combinação dos termos linguísticos das variáveis de entrada. Assim, a classificação pode ser realizada independente do modelo do equipamento e do ambiente ao qual ele esteja exposto; (iii) indiscutivelmente, a existente necessidade de se modelar o problema por meio de variáveis qualitativas e regras linguísticas, de modo a aproximar o processo de decisão humana da decisão computacional. Já a decisão por se utilizar sistemas especialistas foi tomada para que ações preditivas pudessem ser obtidas, uma vez que essa

técnica gera questionamentos aos usuários para coletar informações necessárias à tomada de decisão, nesse caso, informações relacionadas ao histórico dos equipamentos. Além disso, essa técnica proporciona facilidades na inserção de novos conhecimentos, já que a base de conhecimento não é vinculada à implementação do código do sistema.

2.2. Conceitos

A análise de óleo tem como objetivo verificar a deterioração do óleo para controlar o estado operativo dos equipamentos [Toms and Toms 2010]. A metodologia de controle consiste em medir a taxa de contaminação do óleo e analisar essa contaminação. O que ocorre é que, com o passar do tempo, as peças se desgastam e por receberem a lubrificação do óleo passam a contaminá-lo com os materiais existentes em sua composição [Toms and Toms 2010]. Em uma análise de óleo padrão são aplicados basicamente três métodos. Esses métodos, descritos a seguir, foram os considerados neste trabalho para classificar o óleo em categorias, uma vez que os mesmos são os utilizados atualmente pela empresa.

A *análise físico-química* mede as propriedades do óleo referente a viscosidade e a diluição. A viscosidade determina o valor da resistência que o óleo tem ao escoamento, isto é, o quanto o óleo consegue permanecer entre as partes do equipamento [Toms and Toms 2010]. A viscosidade pode ser medida em VsSu (temperatura ambiente) e em VcSt (temperatura em condições de trabalho). Já a diluição determina o grau de contaminação do óleo através do contato com o combustível utilizado pelo equipamento [Toms and Toms 2010]. A diluição é caracterizada geralmente por vazamentos internos do sistema de combustível ou ainda queima inadequada do combustível. A diluição influencia a viscosidade do óleo deixando-o mais fino, permitindo assim que as peças móveis dos equipamentos entrem em atrito [Mobley 2002]. A *espectrometria por infravermelho* analisa o teor de água e fuligem. Quando o óleo está contaminado com água ocorre a deterioração dos aditivos, além de corrosão das partes metálicas [Mobley 2002]. Já a fuligem determina a eficiência da queima de combustível [Mobley 2002]. A fuligem influencia a viscosidade do óleo deixando-o mais grosso, fazendo com que o equipamento tenha que realizar um esforço maior para obter a mesma condição de trabalho podendo causar, em condições extremas, o travamento do equipamento. A *espectrometria por emissão óptica* realiza a contagem das partículas dos metais presentes no óleo [Toms and Toms 2010]. A medição é contabilizada em ppm: parte por milhão. Os metais considerados pela empresa são: ferro, cromo, chumbo, cobre, alumínio, prata, zinco, estanho e silício. Cada metal está associado com o desgaste de algumas das partes internas dos equipamentos.

2.3. Trabalhos Relacionados

Embora existam diversos softwares comerciais que visem resolver o problema aqui apresentado, esses são em geral de propriedade das empresas que fazem a análise laboratorial do óleo, sendo disponibilizados somente quando se contrata o serviço de análise. Portanto, como forma de garantir propriedade e projetar funcionalidades para suas necessidades optou-se por desenvolver o SIMPel, como no caso descrito em [Kreyn et al. 2007]. [Toms 1996] discute a utilização de sistemas especialistas no contexto da análise de óleo e a necessidade de se modelar o problema em etapas. O autor apresenta um processo dividido em quatro etapas: (i) aquisição de dados, etapa em que os dados são preparados; (ii) indicadores de condição, etapa em que se identificam os sintomas e urgências

de manutenção; (iii) diagnóstico dos problemas (falhas), obtido a partir da avaliação dos sintomas obtidos em (ii); (iv) diagnóstico de manutenção, obtido a partir dos problemas identificados em (iii) juntamente com dados históricos de manutenção. Por outro lado, [Sala et al. 2007, Macián et al. 2006] apresentam um sistema baseado na teoria de conjuntos fuzzy para modelar o problema, uma vez que afirmam que modelagens baseadas em lógicas binárias fazem com que a base de conhecimento cresça de modo exponencial. O sistema por eles proposto é composto por três módulos: (i) pré-processamento dos dados, comparado a etapa (i) do trabalho de [Toms 1996]; (ii) inferência, onde se identificam os problemas encontrados no equipamento a partir de um conjunto de sintomas; (iii) pós-processamento, onde se apresentam os problemas identificados em ordem de relevância. [Sala et al. 2007, Macián et al. 2006] também discutem a necessidade de se acoplar ao módulo de pós-processamento conhecimentos relacionados às ações preditivas a serem realizadas para solucionar os problemas. Além disso, os autores destacam a importância de se considerar outras informações do domínio, como os dados históricos do equipamento, para que o diagnóstico seja o mais confiável possível.

Uma vez que o objetivo do SIMPel é diagnosticar a ação preditiva, diferentemente dos trabalhos acima descritos, os problemas (falhas) não são apresentados ao usuário, embora os mesmos sejam implicitamente analisados com base nos contaminantes considerados fora de padrão (não bons). Por outro lado, assim como nos demais trabalhos, o SIMPel apresenta explicitamente a gravidade dos problemas por meio da classificação do óleo, onde: (i) óleo bom indica que não existem problemas e que o equipamento pode prosseguir trabalhando; (ii) óleo regular indica que existem problemas de gravidade intermediária e que é necessário o agendamento de manutenção; (iii) óleo ruim indica problemas graves e, portanto, que o equipamento deve ter sua operação interrompida e ser encaminhado imediatamente para a manutenção. Além disso, como discutido em [Sala et al. 2007, Macián et al. 2006], o SIMPel considera outras informações para emissão do diagnóstico, entre elas os dados históricos e o ambiente ao qual o equipamento está exposto, permitindo que diferentes taxas de contaminação sejam consideradas para o mesmo equipamento.

3. SIMPel: Objetivo, Arquitetura e Funcionamento

O SIMPel é um sistema inteligente híbrido que tem por objetivo apoiar o processo de obtenção de diagnósticos preditivos para viabilizar o controle do estado operativo dos equipamentos lubrificados da empresa ALL. O SIMPel foi projetado em módulos, conforme arquitetura apresentada na Figura 1. Os módulos Fuzzy e Especialista são os responsáveis por tornar o sistema inteligente, de modo que ações preditivas sejam emitidas corretamente. Nesse processo, o módulo Fuzzy é responsável por classificar o óleo de acordo com o resultado obtido na análise laboratorial, identificando também os contaminantes (ferro, etc.) que contribuíram para a classificação. Com base na classificação obtida e conhecimentos sobre o equipamento a serem fornecidos interativamente ao sistema, o módulo Especialista é responsável por informar a ação preditiva a ser realizada. Já o módulo Básico é responsável pela base de dados e tem a função de gerenciar: (i) as operações de cadastro, alteração, consulta e exclusão dos equipamentos, os quais são descritos pelo modelo, marca e componente; (ii) as funções de pertinência de cada um dos contaminantes para cada modelo existente utilizadas pelo módulo Fuzzy; (iii) os resultados obtidos pelos módulos Fuzzy e Especialista.

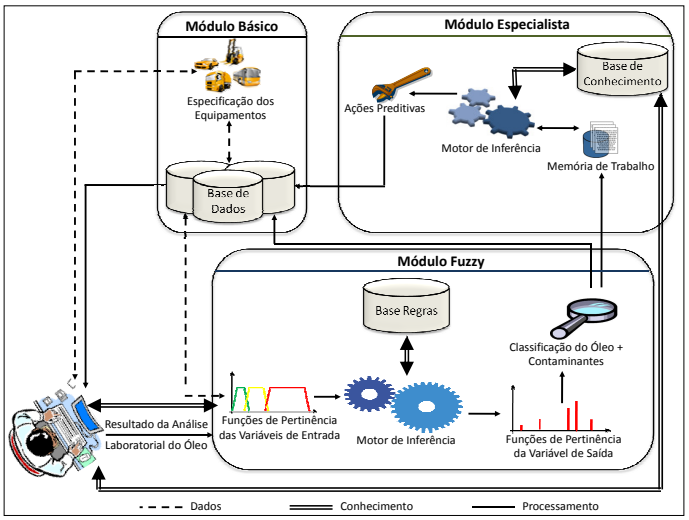


Figure 1. Arquitetura do SIMPel.

O fluxo Dados é ativado sempre que há a necessidade de comunicação com a base de dados. O fluxo Conhecimento fornece o conhecimento do domínio necessário a tomada de decisão. Esse fluxo permite que as funções de pertinência associadas a cada um dos contaminantes, bem como a base de conhecimento, possam ser gerenciadas pelo especialista. Desse modo, novos equipamentos podem ser adicionados ao SIMPel, uma vez que regras relacionadas a novos diagnósticos preditivos podem ser contemplados. Já a base de regras, por ser única para todos os tipos de equipamento, contém todas as associações válidas existentes entre os contaminantes já especificadas não sendo, portanto, acessível. Todo o conhecimento codificado no SIMPel foi obtido por meio de entrevistas junto aos especialistas da empresa. O fluxo Processamento é ativado quando o usuário fornece os valores de cada um dos contaminantes encontrados na análise laboratorial do óleo e solicita que seja realizada a identificação do estado operativo do equipamento. A partir da solicitação, o módulo Fuzzy faz a classificação do óleo e identifica os contaminantes que contribuíram para a classificação. Em seguida, os resultados obtidos nesse módulo são utilizados pelo módulo Especialista, agregados a outros conhecimentos do equipamento, para emissão do diagnóstico.

O SIMPel foi implementado em Java e o banco de dados utilizado foi o MySQL. É importante mencionar que os módulos inteligentes foram projetados de modo a se complementarem, permitindo que o conhecimento do domínio seja facilmente ampliado e ajustado a novas situações. Assim, para melhor compreensão do sistema, são apresentados a seguir os detalhes dos módulos Fuzzy e Especialista.

3.1. Módulo Fuzzy

O módulo Fuzzy é responsável por classificar o óleo a partir do resultado obtido na análise laboratorial, identificando também os contaminantes que contribuíram para a classificação. Para tanto, os contaminantes são inicialmente qualificados na etapa de fuzificação. Cada contaminante compõe uma variável linguística de entrada, as quais possuem os termos linguísticos Bom (B), Regular (A) e Ruim (R). Os contaminantes considerados são: viscosidade (*VcSt* e *VsSu*), diluição (*D*), fuligem (*F*), água (*A*), ferro (*Fe*), cromo (*Cr*), chumbo (*Pb*), cobre (*Cu*), alumínio (*Al*), prata (*Ag*), zinco (*Zn*), estanho (*Sn*)

e silício (*Si*). Como a semântica dos termos linguísticos depende do modelo do equipamento, o módulo Fuzzy só é efetivamente ativado se as funções de pertinência referentes a cada um dos contaminantes para o equipamento em análise estiverem cadastradas na base de dados.

Para permitir que o especialista gerencie os conjuntos fuzzy associados a cada variável linguística de entrada para um determinado modelo de equipamento é disponibilizada a interface apresentada na Figura 2. Esses conjuntos modelam de modo gradativo os limites de contaminação, os quais eram avaliados até então por limites exatos. Para que houvesse uma interface unificada para todos os modelos e considerando os procedimentos utilizados pelos especialistas da empresa, todas as variáveis estão associadas a três termos linguísticos, sendo cada termo representado por uma função de pertinência trapezoidal. Como pode ser observado, o SIMPel foi projetado de modo a gerenciar qualquer equipamento, sendo necessário apenas o cadastramento do mesmo na base de dados assim como de seus respectivos limites de contaminação. Embora a empresa utilize diversos modelos de equipamentos para realizar suas operações de logística, atualmente o SIMPel gerencia apenas dois modelos, os quais representam a maioria dos equipamentos.

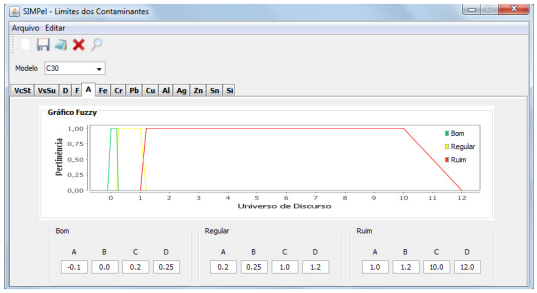


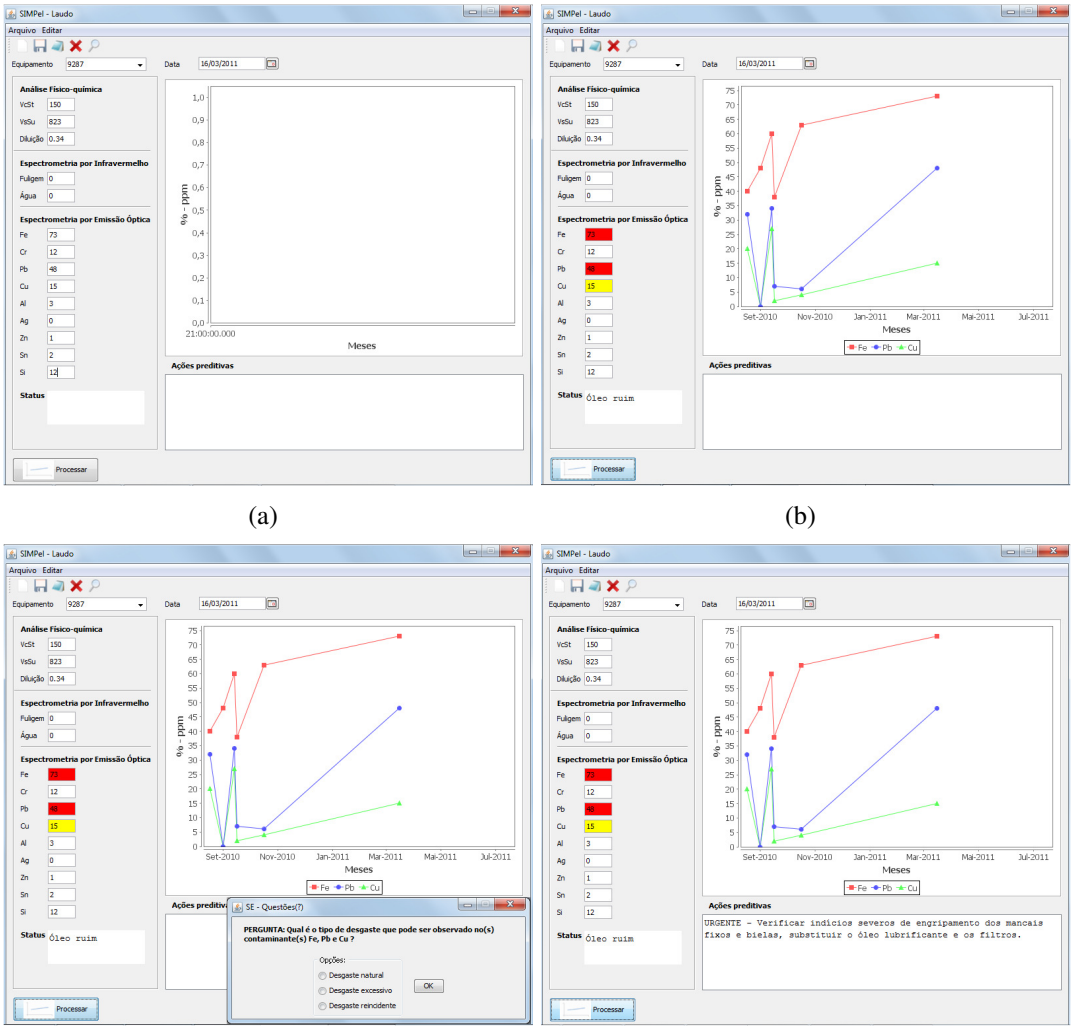
Figure 2. Interface de gerenciamento dos conjuntos fuzzy referentes as variáveis linguísticas de entrada.

O módulo Fuzzy é ativado sempre que o equipamento a ser analisado é selecionado e os valores obtidos no laudo laboratorial do óleo fornecidos (Figura 3(a)). A partir dessas informações o óleo é classificado em Bom, Regular ou Ruim e os contaminantes que contribuíram para a classificação são destacados na cor amarela quando considerados Regular, e vermelha quando considerados Ruim (Figura 3(b)).

A base de regras do módulo foi construída de modo a contemplar apenas as regras consideradas válidas pelos especialistas, regras essas que contêm algumas das possíveis combinações entre os contaminantes. Cada uma das combinações levam a uma das possíveis respostas, ou seja, a uma classificação do óleo com seus respectivos contaminantes. A combinação ocorre por meio do operador E. Uma vez que apenas algumas das combinações são contempladas, caso o SIMPel não consiga classificar o óleo, uma mensagem é emitida informando que não existem regras disponíveis que satisfaçam a condição na qual o óleo se encontra. A regra

```
SE   ((VcSt=Bom) E (VsSu=Bom) E (D=Bom) E (F=Bom) E (A=Bom) E (Fe=Ruim) E (Cr=Bom) E
      (Pb=Bom) E (Cu=Bom) E (Al=Bom) E (Ag=Bom) E (Zn=Bom) E (Sn=Bom) E (Si=Reg))
ENTÃO Resposta="Regular:Fe,Si"
```

representa uma das 220 regras codificadas. Observe que no antecedente as variáveis de entrada são combinadas e que no consequente atribui-se à variável de saída *Resposta* um



(a) (b)

(c) (d)

Figure 3. Interface responsável pela inicialização dos módulos Fuzzy e Especialista em diferentes momentos do processo preditivo.

dos seus termos linguísticos. É importante mencionar que essa base de regras é única para todos os equipamentos, uma vez que os contaminantes são agrupados sempre do mesmo modo, sendo necessário apenas que se considere na etapa de fuzificação os conjuntos fuzzy do modelo do equipamento em análise. A variável linguística de saída, *Resposta*, contém 215 termos linguísticos, cada um representado por um conjunto fuzzy do tipo *singleton*. Cada termo representa uma das possíveis classificações (Bom, Regular ou Ruim), as quais podem conter também os contaminantes que influenciaram a classificação, como na regra acima apresentada.

Em relação ao processamento, o módulo é baseado em regras do tipo Mamdani [Mamdani 1977, Berkan and Trubatch 1997] e utiliza os operadores fuzzy padrão, a saber: max para agregação e min para intersecção e implicação. Nos sistemas Mamdani as regras disparadas geram resultados parciais que são agregados para formar uma única saída final na forma de conjunto fuzzy. Esse conjunto pode ser então defuzificado para a obtenção de uma saída numérica precisa. No caso deste trabalho, a defuzificação não é

realizada, uma vez que o objetivo do módulo é classificar linguisticamente o óleo, assim como identificar os contaminantes que contribuem para a classificação. Desse modo, para que a classificação seja realizada o SIMPel considera como resposta o termo linguístico (conjunto fuzzy do tipo *singleton*) associado a variável de saída *Resposta* que apresentar o maior grau de ativação.

Diante do apresentado, uma vez que o módulo é ativado, o processamento ocorre da seguinte maneira (vide Figura 1): (A) recuperam-se os limites de contaminação (conjuntos fuzzy) do equipamento associados a cada variável linguística de entrada para que a fuzificação seja realizada; (B) após obter-se os graus de pertinência de cada variável de entrada em cada um de seus respectivos conjuntos fuzzy, o processo de inferência é inicializado; (C) após o processo de inferência, realiza-se a agregação dos conjuntos fuzzy associados a variável de saída *Resposta*; (D) ao final, o termo linguístico associado a variável *Resposta* que apresentar o maior grau de ativação determina a classificação.

A implementação desse módulo foi gerada a partir do Xfuzzy[2], ambiente de desenvolvimento de sistemas fuzzy. Assim, inicialmente a modelagem do problema foi realizada nesse ambiente e, após a geração e importação do código Java para o SIMPel, modificações foram realizadas para permitir que diversos equipamentos pudessem ser analisados.

3.2. Módulo Especialista

O módulo Especialista é responsável por fornecer as ações preditivas a serem tomadas para que o equipamento não falhe. Para tanto, o módulo recupera o modelo do equipamento em análise e o resultado obtido no módulo Fuzzy. Esse resultado é combinado com outras informações a serem fornecidas pelo usuário durante o processamento do módulo para que o diagnóstico preditivo seja informado.

Antes que o módulo seja efetivamente inicializado, o SIMPel recupera da base de dados, apenas para os contaminantes identificados no módulo Fuzzy, os resultados numéricos obtidos nas últimas análises laboratoriais realizadas (até dez resultados) e os apresenta graficamente (Figura 3(b)) juntamente com o resultado atual. O gráfico permite que o usuário analise, identifique e, posteriormente, forneça informações sobre o comportamento histórico dos contaminantes envolvidos na classificação atual do óleo.

O processo de aquisição de conhecimento referente a construção da base de conhecimento foi realizado por meio de entrevistas com os especialistas do domínio. Para obtenção das ações preditivas, a base relaciona as informações obtidas no módulo Fuzzy com informações históricas do equipamento. O conhecimento foi representado por meio de regras de produção. A base é composta por 107 regras, sendo 82 conclusivas e 25 geradoras de fatos (informações) (divisão baseada em [Bittencourt 2001]). As regras

```
SE    ((modelo="X") OU (modelo="Y"))
ENTÃO tipoDesgaste={"Desgaste natural","Desgaste excessivo","Desgaste reincidente"}

SE    ((modelo="X") OU (modelo="Y")) E (contaminante="F") E (ClassificacaoOleo="Regular") E
      (tipoDesgaste="Desgaste excessivo") E ((manutencao="Sim") OU (manutencao="Não"))
ENTÃO "Verificar o sistema de combustível com intuito de encontrar uma injeção irregular."
```

representam, respectivamente, uma das 25 regras geradoras e uma das 82 regras conclusivas. As regras geradoras, quando satisfeitas, geram questionamentos ao usuário.

[2] http://www2.imse-cnm.csic.es/Xfuzzy/.

Os possíveis questionamentos a serem feitos ao usuário a partir dessas regras são: (i) Qual é o tipo de desgaste que pode ser observado no(s) contaminante(s) X? [Natural;Excessivo;Reincidente]; (ii) Foi realizada manutenção no equipamento? [Sim;Não]; (iii) O óleo foi substituído? [Sim;Não]; (iv) Em qual equipamento foi realizada a manutenção por contaminação pelo(s) contaminante(s) X? [Anéis;Balancins;etc.]. Os valores entre [] representam as possíveis respostas (exceção a pergunta (iv), na qual estão listadas apenas algumas das possíveis respostas). O X presente nas perguntas (i) e (iv) é preenchido automaticamente pelo SIMPel e se refere aos contaminantes destacados (em amarelo ou vermelho) que influenciaram a classificação atual do óleo. A pergunta (i) está relacionada ao gráfico descrito anteriormente.

O raciocínio implementado no motor de inferência foi o encadeamento progressivo. A fim de facilitar a implementação, esse módulo utiliza o Drools[3], mais especificamente o Drools Expert, um motor de inferência que usa a abordagem baseada em regras para implementar sistemas especialistas. O Drools Expert é totalmente integrável ao Java e já disponibiliza os raciocínios progressivo e regressivo, uma representação de conhecimento baseada em regras, o reteOO (otimização do algoritmo rete [Forgy 1982]) como estratégia de busca, memória de trabalho, estratégias de resolução de conflitos, etc. Complementando a necessidade do módulo, a interface foi construída utilizando-se a API Swing do Java, uma vez que essa não é implementada diretamente pelo Drools.

Diante do apresentado, uma vez que o módulo é ativado, o processamento ocorre da seguinte maneira (vide Figura 1): (A) inicialmente, o modelo do equipamento, a classificação do óleo e os contaminantes que influenciaram a classificação são adicionados à memória de trabalho, ou seja, são eles os fatos iniciais a serem utilizados durante o processo de inferência; (B) a partir dos fatos iniciais, o motor de inferência verifica se alguma das regras conclusivas que compõe a base de conhecimento é satisfeita. Enquanto nenhuma regra conclusiva for satisfeita, ou seja, enquanto não se obtiver um diagnóstico preditivo de manutenção, as regras geradoras serão ativadas, uma por vez, até que se obtenha um diagnóstico. Cada uma dessas regras gerará questionamentos ao usuário (Figura 3(c)). As informações fornecidas pelo usuário, por meio de perguntas e respostas, serão armazenadas na memória de trabalho como novos fatos; (C) uma vez satisfeita uma regra conclusiva, o diagnóstico preditivo é fornecido ao usuário (Figura 3(d)) e a execução termina. Caso todas as regras conclusivas tenham sido verificadas e nenhum diagnóstico tenha sido encontrado, o SIMPel informa que não existe nenhuma regra capaz de provar os fatos informados e solicita ao usuário que uma nova regra seja inclusa na base de conhecimento.

4. Avaliação do SIMPel

Com o objetivo de avaliar o desempenho do SIMPel, 161 casos de teste foram considerados. Os testes basearam-se em diagnósticos preditivos reais gerados pelo especialista da empresa no período de 2004 a 2007 para quatro equipamentos distintos, sendo dois de um determinado modelo e dois de um outro modelo.

O SIMPel obteve 97.52% de acerto. Os 2.48% (4 casos dos 161) considerados inicialmente como erros foram encaminhados ao especialista para que os resultados pudessem ser verificados. Após análise, o especialista concluiu que em dois dos quatro

[3] `http://www.jboss.org/drools`.

casos, se o SIMPel tivesse sido utilizado teria-se evitado que um dos equipamentos viesse a ser danificado ao ponto de necessitar a substituição de peças. Para esses dois casos, o especialista havia concluído que o óleo era bom e, portanto, que o equipamento estava em boas condições de uso. Por outro lado, o SIMPel indicou que o óleo estava regular em função do contaminante *Al* e, considerando o histórico do equipamento, disparou algumas ações preditivas como, por exemplo, a substituição do óleo. A análise realizada pelo SIMPel superou a do especialista em função da qualificação dos contaminantes ser feita de modo gradativo por meio da modelagem Fuzzy.

Nos outros dois casos, embora os diagnósticos preditivos emitidos pelo SIMPel tenham divergido do especialista, os diagnósticos do especialista não geraram, como nos casos anteriores, prejuízos aos equipamentos. Nesses casos, o diagnóstico do SIMPel também foi mais detalhado, considerado o óleo regular em função do contaminante *F*. Entretanto, uma vez que a contaminação evoluiu muito pouco entre os intervalos das análises, o especialista também conseguiu diagnosticar as ações preditivas a serem tomadas antes que o problema se agravasse.

5. Conclusão e Trabalhos Futuros

Este artigo apresentou o SIMPel, um sistema inteligente híbrido projetado para apoiar os especialistas da ALL na tarefa de manutenção preditiva em equipamentos lubrificados. O SIMPel informa a ação preditiva a ser realizada a fim de melhorar a disponibilidade dos equipamentos sem sobrecarregar a equipe técnica.

O SIMPel é composto por dois módulos inteligentes. O módulo Fuzzy é responsável por classificar o óleo e identificar os contaminantes que contribuíram para a classificação. O módulo Especialista é responsável por obter um diagnóstico preditivo de manutenção com base nos resultados obtidos pelo módulo Fuzzy e informações históricas. Os módulos foram projetados de modo a se complementarem, permitindo que o conhecimento do domínio seja facilmente ampliado e ajustado a novas situações. Se o SIMPel tivesse sido implementado apenas com a técnica especialista, a base de regras seria muito maior e as regras seriam muito mais específicas, dificultando a manutenção do conhecimento. Nesse caso, o conhecimento referente aos limites de contaminação associados a cada equipamento teria de ser codificado de modo preciso ou o usuário teria de informar, a priori, a qualificação de cada contaminante. Por outro lado, uma implementação totalmente baseada na teoria de conjuntos fuzzy não seria adequada, uma vez que os dados históricos não necessitam de tratamento linguístico. Além disso, em função da integração, quando novos equipamentos são inseridos no SIMPel, o módulo Fuzzy permanece inalterado, sendo necessário em alguns casos (dependendo do modelo) apenas a inserção de novos conhecimentos no módulo Especialista e a especificação de conjuntos fuzzy relacionados às variáveis de entrada.

O SIMPel apresentou um excelente desempenho durante os testes realizados, diagnosticando inclusive situações que antes não eram notadas pelo especialista. O SIMPel encontra-se em uso e em estudo para ser integrado ao SAP da empresa. Caso essa integração ocorra é provável que parte das informações fornecidas pelo usuário ao módulo Especialista passem a ser automáticas, uma vez que o SAP possui uma base de dados de manutenção integrada.

Como trabalho futuro, pretende-se adicionar ao SIMPel a funcionalidade de

identificação dos padrões de desgaste do óleo. Desse modo, o SIMPel poderá diagnosticar problemas que possam a vir ocorrer antes que a análise de óleo seja realizada. Para tanto, estuda-se o uso de técnicas de aprendizado de máquina.

6. Agradecimento

O segundo autor agradece o apoio financeiro recebido da Fundação para o Desenvolvimento da Unesp (FUNDUNESP).

References

Berkan, R. C. and Trubatch, S. L. (1997). *Fuzzy Systems Design Principles: Building Fuzzy If-Then Rule Bases*. Wiley-IEEE Press.

Bittencourt, G. (2001). *Inteligência Artificial: Ferramentas e Teorias*. UFSC, 2 edition.

Forgy, C. L. (1982). Rete: A fast algorithm for the many pattern/many object pattern match problem. *Artificial Intelligence*, 19(1):17–37.

Giarratano, J. C. and Riley, G. D. (2004). *Expert Systems: Principles and Programming*. Course Technology, 4 edition.

Kreyn, A., Turner, J., and Toms, A. M. (2007). Fuel and water diagnostic enhancements at BNSF railway. In *STLE'07: Society of Tribologists and Lubrication Engineers; Condition Monitoring/Predictive Maintenance*. STLE. 3p.

Macián, V., Tormos, B., Sala, A., and Ramirez, J. (2006). Fuzzy logic-based expert system for diesel engine oil analysis diagnosis. *Insight - Non-Destructive Testing and Condition Monitoring*, 48(8):462–469.

Mamdani, E. H. (1977). Application of fuzzy logic to approximate reasoning using linguistic synthesis. *Transactions on Computers*, 26(12):1182–1191.

Mobley, R. K. (2002). *An Introduction to Predictive Maintenance*. Butterworth-Heinemann, 2 edition.

Negnevitsky, M. (2005). *Artificial Intelligence: A Guide to Intelligent Systems*. Addison Wesley, 2 edition.

Sala, A., Tormos, B., Macián, V., and Royo, E. (2007). Fuzzy diagnosis module based on interval fuzzy logic: Oil analysis application. In *Informatics in Control, Automation and Robotics II*, pages 43–50. Springer Netherlands.

Toms, A. and Toms, L. (2010). Oil analysis and condition monitoring. In Mortier, R. M., Fox, M. F., and Orszulik, S. T., editors, *Chemistry and Technology of Lubricants*, chapter 16, pages 459–495. Springer, 3 edition.

Toms, L. A. (1996). Expert systems: A decade of use for used-oil data interpretation. In *Proc. Technology Showcase: Integrated Monitoring, Diagnostics and Failure Prevention*, pages 65–72.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, 8(3):338–353.

Encoding DL-Atoms in RuleML

Samy Sá[1] **and João Alcântara**[1]

[1]Departamento de Computação, Universidade Federal do Ceará, P.O.Box 12166, Fortaleza, CE, Brazil 60455-760

{samy,jnando}@lia.ufc.br

Abstract. *We consider Description Logic Programs (DLPs) as consisting of a description logic knowledge base L and a generalized extended logic program P, which may contain queries to L through the DL-Atoms. Aiming to provide general syntax for representing rules, RuleML is a markup language and candidate for a standard exchange rule language in the Web. Currently under development from version 0.91 to version 1.0, its syntax still lacks the ability to express DL-Atoms and, consequently, DL-Programs. In order to overcome this gap, we propose an extension for RuleML expressive enough to represent all sorts of DL-Atoms.*

Resumo. *Consideramos Description Logic Programs (DLPs) como estes que consistem em uma base de conhecimento L em Lógica Descritiva e um programa em lógica estendido generalizado P, o qual pode conter consultas a L através de DL-Átomos. Objetivando prover sintaxe geral para a representação de regras, RuleML é uma linguagem candidata a padrão para troca de regras na Web. Atualmente em desenvolvimento da versão 0.91 para 1.0, sua sintaxe ainda não tem a habilidade de expressar DL-Átomos e, consequentemente, DL-Programs. Para superar esse obstáculo, nós propomos uma extensão para RuleML expressiva o suficiente para representar todos os tipos de DL-Átomos.*

1. Introduction

The *Semantic Web* [Berners-Lee and Fischetti 1999] is an extension of the *World Wide Web* technology that aims to establish a pattern for expressing the intended meanings of documents in the web along with their content. The Semantic Web consists of several layers enabling to build ontologies with increasingly expressive power. XML is the first layer and all others are built on top of it, meaning that documents in all superior layers are valid XML documents. The current top layer of the stack being developed comprises OWL and the representation of rules in order to allow reasoning over ontologies [Antoniou and van Harmelen 2008]. Now that OWL reaches a satisfactory level of maturity, the urge to efficiently represent rules in the most variate formats has taken over the Semantic Web development efforts and the W3C (World Wide Web Consortium) has assigned a working group to design a Rule Interchange Format (RIF). As part of this effort, the XML based *Rule Markup Language* (RuleML) [Dema and Hirtle 2006] is a candidate for becoming the standard language for exchange and representation of rules in both syntax and semantics. The main goal of RuleML is to allow the translation from any rule systems to others, in a single general framework. Currently, RuleML is being developed from version 0.91 into version 1.0. Even though there are improvements in the language, it still lacks the ability to express some kinds of rules.

Various notions relating *Description Logics* (DL) [Baader et al. 2003] and *Logic Programming* (LP) [Lloyd 1987] were proposed, including a translation from ontologies in the Description Logic language $\mathcal{ALCIQ}$ to logic programs [Alsaç and Baral 2002] and an algorithm to reduce a $\mathcal{SHIQ}$ knowledge base to a disjunctive datalog program [Hustadt et al. 2007]. Other approaches involve proposing Description Logic Programs as the intersection of horn logic programs and description logics [Grosof et al. 2003] and an extension of answer set programming that supports inverted predicates and infinite domains to integrate description logics and answer set programming [Heymans and Vermeir 2003]. Eiter et al., however, proposed a different concept of *Description Logic Programming* (*DLP*) in [Eiter et al. 2004, Eiter et al. 2008] to represent and reason over ontologies. Their approach enables a great deal of ontology integration on reasoning, as a logic program can access and consider the results of queries over Description Logic ontologies (DL ontologies) represented in expressive languages such as OWL DL or OWL Lite [Eiter et al. 2008]. Also, the results of queries might be based on results from both the logic program and the ontology through special operators. Next, we show a DLP rule taken from [Eiter et al. 2008] and explain its meaning.

Example 1.1 (Sample DLP Rule)

The following rule from a DLP P contains a query to a Description Logic ontology (DL ontology) L as its only condition. The query is described in the DL-Atom $DL[keyword \cup +kw; inArea](P,\ A)$, in which $keyword$ and $inArea$ are terms from the ontology L and kw is a predicate from the program P.

$$r : paperArea(P,\ A) \leftarrow DL[keyword \cup +kw; inArea](P,\ A);$$

The above rule queries L to find to what area A the paper P belongs. In this case, the query returns the results of the role $inArea$ from L as a predicate $inArea(P, A)$ based on the consideration $keyword \cup +kw$. This consideration means that the results of DLP predicate kw augments ($\cup+$) the set of results of the DL ontology role $keyword$, so the result of the query is based on data from both the program P and ontology L.

As it can be taken from example 1.1, programs and DL ontologies can be combined in very complex ways. In fact, DLP rules that include DL-Atoms are very peculiar, as they present queries inside some atoms. This feature is not common and rule exchange languages (RuleML v1.0 included) are not capable of expressing such queries. We point out that this kind of integration is compliant to the the goals of the Semantic Web and, as such, should be considered. In this paper we propose a modular extension capable of comprising DLP rules while respecting XML compatibility and show how to encode the characteristic atoms of these rules. We defend that our extension to RuleML 0.91 should be considered for adoption in the development of RuleML 1.0. This extension makes possible to represent DLP rules and therefore integrate description logic ontologies and logic programs as rules from one another are exchanged in the Semantic Web.

A related work is [Damásio et al. 2006], in which the authors propose a general extension to RuleML 0.9 for representing uncertainty rules from several frameworks such as fuzzy and probabilistic logic programming. Their proposal is aimed at being modular, to make for natural, easy to read translations while also trying to keep changes to the language of

RuleML 0.9 to a minimum. In this paper we adopt the same goals to propose an extension in which it is also possible to represent DL-Atoms.

The rest of the paper is organized as follows. Section 2 briefly introduces Logic Programs, Description Logics and the definitions of programs, rules, atoms and operators used to relate ontologies and programs in *DLP*. Section 3 provides an analysis of available and missing features of RuleML needed to express DL-Atoms and presents our suggested extension. Section 4 shows how to encode DL-Rules in RuleML. We summarize the steps towards proposing such extension, criticize results and comment on future work in section 5.

2. Preliminaries

2.1. Extended Logic Programs

An Extended Logic Programs (ELP) [Gelfond and Lifschitz 1991] comprises a finite sets of inferential rules of the form $r_i : A \leftarrow B_1, \ldots, B_k, notB_{k+1}, \ldots, notB_m, m \geq k \geq 0$, where A, B_i, $m \geq i \geq 0$, are classical literals. We refer to the literal A as the *Head* of the rule and the conjunction of the literals $B_1, \ldots, B_k, notB_{k+1}, \ldots, notB_m$ as the *Body* of the rule and write $Head(r)$ and $Body(r)$ to refer to those. The semantics of such programs are given by its Answer Sets (AS) [Gelfond and Lifschitz 1991].

2.2. Description Logics

Description logics (DL) [Baader et al. 2003] are a family of formal knowledge representation languages based on concepts and roles possibly applied to variables or terms representing particular entities in a domain. A *concept* is a unary predicate that qualifies a set of entities and a *role* is a binary predicate defining a relationship between entities. A Description Logic knowledge base, sometimes referred to as an ontology, consists of two parts traditionally referred to as the *TBox* (Terminological Box) and *ABox* (Assertional Box). The first consists of assertions about concepts and roles in the sense of defining their meaning towards each other (e.g concept inclusion). The second presents specific information about particular entities belonging to concepts or pairs of entities belonging to roles.

2.3. Description Logic Programs

The literature comprises several approaches to relate DL and logic programming such as [Alsaç and Baral 2002, Hustadt et al. 2007, Grosof et al. 2003, Heymans and Vermeir 2003, Eiter et al. 2004, Eiter et al. 2008]. Except by *Description Logic Programs* (DLP) as in Eiter et al. [Eiter et al. 2004, Eiter et al. 2008], the proposed syntax from all others can be easily expressed in RuleML. For instance, in [Heymans and Vermeir 2003] it is proposed the inclusion of a *predicate inversion* operator to disjunctive logic programs syntax, resulting in the extended language *Conceptual Logic Programs* (CLP). This extension allows for direct translation from DL formulas using the *Role Inversion* construct. In CLP we write $P^{-}(X_1, X_2)$ to address the inverse of $P(X_1, X_2)$ (instead of using $P'(X_2, X_1) \leftrightarrow P(X_1, X_2)$), P being a binary predicate, or $P^{-}(X_1)$ as the inverse of $P(X_1)$, $P^{-}(X_1) = P(X_1)$, P being a unary predicate. There are a few ways we can represent such extension to syntax, such as by suggesting a new tag to involve atoms and account for its inversion, a new attribute to the tag <Rel>

to indicate inversion or just by using a simple translation that requires no change to RuleML, such as done in [Grosof et al. 2003]. Since the role inversion construct can be successfully represented in RuleML 0.91, we are left with only the approach due to Eiter et al. in [Eiter et al. 2008] and [Eiter et al. 2004] for consideration. In their work, characteristic atoms are introduced to the body of rules in order to integrate Description Logic ontologies (DL ontologies) and extended logic programs. These atoms require special attention, as they can not be expressed in RuleML 1.0. From this point and on, we mean by DLP the notion introduced in [Eiter et al. 2004, Eiter et al. 2008]. In this section we introduce the elements composing DLPs. Most of the definitions below are taken from [Eiter et al. 2008], but the name of considerations is a suggestion of our own.

Definition 2.1 (DL-Queries) *A* DL-Query *is a question to a DL knowledge base L in one of the forms:*

1. *a concept inclusion axiom F or its negation $\neg F$.*
2. *$C(t)$ or $\neg C(t)$, where C is a concept and t is a term.*
3. *$R(t_1, t_2)$ or $\neg R(t_1, t_2)$, where R is a role and t_1, t_2 are both terms.*

Definition 2.2 (Considerations) *A* Consideration *is a sequence $S_i op_i p_i$, where S_i is either a concept or role from L, $op_i \in \{\cup+, \cup-, \cap-\}$ and p_i is, respectively, a unary or binary predicate from P relative to S_i.*

The operators op_i are able to augment or retract facts about S_i (from L) or its negation through p_i (from P) as (1.) $op_i = \cup+$ increases the set of positive results of S_i by adding instances of p_i while (2.) $op_i = \cup-$ increases the set of results of $\neg S_i$ by adding instances of p_i and (3.) $op_i = \cap-$ restricts possible results of S_i to the ones of p_i.

For instance, suppose a query $Q : inArea$ to L depends on instances of $keyword$ in the TBox of L. In case the consideration $keyword \cup +kw$ is applied to Q, the instances of predicate kw in P are taken as if they were translated to instances of $keyword$ in L before Q is processed and discarded right after, therefore possibly changing the output of Q.

Definition 2.3 (DL-Atoms) *Special atoms allowed in the syntax of DLPs that specify queries to L.* DL-Atoms *have the form*

$$DL[S_1 op_1 p_1, \ldots, S_m op_m p_m; Q](t),\ m \geq 0,$$

where $Q(t)$ is a DL-Query and each sequence $S_i op_i p_i$ is a consideration to the query $Q(t)$.

A query inside DL-Atoms ask about the occurrence or evidence of either a description logic axiom or its negation in L. Whenever a query $Q(t)$ is to be calculated, it takes into account inferences driven from P according to considerations to the query inside the DL-Atom [1].

Definition 2.4 (DL-Rules) *A DL-Rule is an extended logic program rule possibly presenting DL-Atoms in its body. A DL-Rule has the form $r : L_0 \leftarrow L_1, \ldots, L_k, notL_{k+1}, \ldots, notL_m, m \geq k \geq 1$, where L_0 is a classical literal and L_i, $m \geq i \geq 1$, are either classical literals or DL-Atoms. We refer to L_0 as the* head *of r and the conjunction $L_1, \ldots, L_k, notL_{k+1}, \ldots, notL_m$ as the* body *of r.*

[1] For a more detailed explanation, we invite the reader to address to [Eiter et al. 2004, Eiter et al. 2008].

Definition 2.5 (Description Logic Programs) *A* Description Logic Program *(DLP) is a pair KB = (L, P), where L is a Description Logic knowledge base and P is a logic program possibly with occurrence of DL-Rules. DLPs can present queries to L in the body of rules of P and integrate the knowledge base and logic program allowing for rules to work on top of ontologies and ontologies to be built on top of rules.*

The following example is a partial transcript from [Eiter et al. 2004, Eiter et al. 2008].

Example 2.1 (Reviewer Selection) *Suppose we want to assign reviewers to papers, based on certain information about the papers and the available persons, using a description logic knowledge base L_S which contains knowledge about scientific publications. Consider then the DL-Program $KB_S = (L_S, P_S)$, where P_S contains in particular the following DL-Rules:*
(1) $paper(p_1)$*;* $kw(p_1,\ Semantic_Web)$*;*
(2) $paper(p_2)$*;* $kw(p_2,\ Bioinformatics)$*;* $kw(p_2, Answer_Set_Programming)$*;*
(3) $kw(P,\ K_2) \leftarrow kw(P,\ K_1),\ DL[hasMember](S,\ K_1), DL[hasMember](S,\ K_2)$*;*
(4) $paperArea(P,\ A) \leftarrow DL[keyword \cup +kw; inArea](P,\ A)$*;*
(5) $cand(X,\ P) \leftarrow paperArea(P,\ A),\ DL[Referee](X),\ DL[expert](X,\ A)$.

Intuitively, (1) and (2) specify the keyword information of the papers p_1, p_2 which should be assigned to reviewers. Rule (3) augments the set of keywords from P_S for a paper P with similar ones obtained through queries to L_S. The DL-Atoms in (3) query L_S for keywords other than K_1 that are members of the same keyword cluster S. Rule (4) queries L_S to assert a paper P to an area A while considering all results of $keyword$ (from L_S) and kw (from P_S) of P. Finally, rule (5) singles out review candidates based on the information from experts among the reviewers according to L_S. Note that, in view of rules (3) to (5), information flows in both directions between the knowledge encoded in L_S and P_S.

3. Extension of RuleML

RuleML [Tabet et al. 2001] is a normalized modular markup language designed to express and exchange rules in the Web. RuleML resorts to a set of tags and attributes with preconceived nesting[2] admissibility and semantics. About to enter version 1.0 [Dema and Hirtle 2006, Boley et al 2010], RuleML is fully XML compatible and capable of expressing rules in various working frameworks ranging from Datalog (function-free horn logic) [Silberschatz et al. 2001] to HILOG [Chen et al. 1993]. Specialized markup languages are also available for specific rule terminology[3] such as FOL RuleML [Boley et al 2004] for first order logic. Nonetheless, RuleML up to version 0.91 is unable to express the special case of rules including sub-queries to external knowledge bases. In this section we analyze the main issues for representing DL-Atoms and DL-Rules in RuleML 0.91 and propose a minimal and modular extension. It means that (1) changes to RuleML are kept to a minimum while attempting to maximize expressiveness; (2) any valid RuleML document will still be a valid document in our extended version and that future work should also be easily compatible to ours. Our extension also provides a straight-forward translation of DLP-Rules.

[2]For any two particular tags, the tags nesting specifies whether or not one can be used in the scope of the other.

[3]Specialized rule markup languages might present a different nesting or set of tags/attributes than that of RuleML.

3.1. Special needs of DL-Atoms

We now summarize the features of DL-Atoms and which elements are missing for encoding those features in RuleML 0.91.

- DL-Atoms should somehow be identified as a different kind of atom;
- The adequate structure for representing queries is the performative <Query> [Dema and Hirtle 2006] but performatives can not occur nested in RuleML;
- Considerations relate concepts/roles in L to predicates in P and omit variables. Also, there are no performatives or tags suitable to encompass the semantics of a consideration.
- There are no tags for the operators $\cup+$, $\cup-$, $\cap-$.

Some of those issues are solved by employing tags already in RuleML. Relations as concepts, roles and predicates can be comprised under the tag <Rel>. Variables and constants can be expressed respectively by <Var> and <Ind>. Also, in order to associate relations, it is enough to wrap them as the only content inside a tag <Implies>. In such a case, the predicate (condition) comes first and the concept/role (conclusion) follows, just as the first extends the second and not the opposite. In addition, we need to:

- Allow sub-queries in the body of rules.
- Represent the special operators $\cup+$, $\cup-$, $\cap-$ used in considerations to DL-Queries.
- Propose a set of tags and nesting capable of relating concepts/roles to predicates in considerations.

3.2. Changes to syntax

In DLPs there may be atoms able to express taking action in which rules can query other rules. The general construct for expressing action in RuleML are the performatives [Dema and Hirtle 2006]. In plain RuleML there are three performatives: <Assert>, <Query> and <Retract> and they are meant to add information, express queries and eliminate information, respectively. In FOL RuleML 0.9 [Boley et al 2004], however, we have the performative <Consider> meant to add information only locally, i.e., such information is not part of the theory where inference is driven, but extra. Thus <Consider> seems to be the most adequate amongst performatives to express considerations to queries within DL-Atoms.

In previous versions, RuleML had an attribute @kind exclusive to tags <Implies>. This attribute was used to identify the kind of implication by values ”fo” for first order and ”lp” for logic programming, but it was renamed in RuleML 0.91 to @material. Other work such as [Damásio et al. 2006] proposes applying @kind back to elements and connectives to express the language in which such elements should be interpreted. We defend the usefulness of @kind as such, as its use in <Atom> with a new possible value ”dl” is enough for identifying DL-Atoms. Also, note that the nesting of queries inside tags <Atom> identified as DL-Atoms is adequate to express the queries inside DL-Atoms. Considerations should be expressed inside the same element <Atom> as its related query, and we use the performative <Consider> to do so. Relations from both P and L are related inside each consideration through the use of <Implies>. Finally, instead of proposing new tags for the operators $\cup+$, $\cup-$ and $\cap-$, we show how they can

be translated into RuleML as specific sequences of nested tags inside DL-Atoms. We have then extended RuleML by:

- allowing the performatives <Query> and <Consider> to occur inside elements <Atom>;
- adding @kind to elements <Atom> and a new possible value "dl".

We now show the mentioned elements and performatives content models from RuleML 0.91 in the same style as in [Damásio et al. 2006, Dema and Hirtle 2006]. The exact same changes would be applied to the current proposal of RuleML version 1.0. Below, our proposed changes are typed in bold to emphasize which elements are being added as part of our proposed extension.

Query	**Consider**
attribute: @closure	**attribute: @closure**
content: (oid?, (formula)*)	**content: (oid?, (formula)*)**
within Atom …	**within Atom …**
content: (oid?, (formula))	**content: (oid?, (formula))**

Figure 1. Content Model for Performatives

We change the content models for performatives in two ways. First, performative <Query> can now occur inside <Atom> elements. The second change introduces <Consider> in standard RuleML with the same possible content as <Query> and also able to occur inside <Atom> elements. Optional parameters receive a question mark (?). When a parameter can occur zero or multiple times we use a star mark (*).

Atom
attribute: @closure, **@kind**
content: (oid?, **((formula)* | ((Consider)*, Query))**

@kind
[optional] (default:fo | lp | **dl**)

Figure 2. Content Model for Elements

Content models for elements are changed in three ways. Elements <Atom> can now show performatives <Query> and <Consider> and have a new attribute @kind. We also enable the possible value "dl" to @kind. Having introduced these changes, we can now encode *DLP* rules in RuleML as expounded in the sequel.

4. Encoding DLP Rules

In this section we show how to use the extended RuleML syntax proposed in section 3 to encode DLPs. As extended logic programs can be easily expressed in RuleML 1.0, only DL-Atoms are left for encoding. Indeed, each kind of consideration in a DL-Atom requires a specific codification in RuleML. For an easier comprehension of the reader, we opt to show our translation through examples of DLP rules.

4.1. DL-Atoms with no Considerations to the Query

The first kind of DL-Atom to be considered consists of just a query to L, with no considerations. RuleML provides a performative <Query>, but does not allow for its use inside tags <body>. We simply extend the syntax of RuleML so we can have this behavior of tags, and identify the atom as being of kind DL through attribute @kind inside <Atom>. The value "dl" for @kind is a new possible value, added to our extension.

Example 4.1 *Consider the DL-Atom $DL[inArea](P,\ A)$, that specify a query to find in L what papers (P) qualifies in which areas (A). This single DL-Atom can be represented in our extension as*

```
<Atom kind = "dl">
     <Query>
          <Atom>
             <Rel>inArea</Rel>
             <Var>P</Var>
             <Var>A</Var>
          </Atom>
     </Query>
</Atom>
```

In this example, @kind has the value "dl" in <Atom> and the performative <Query> inside it. Both changes are necessary to translate DL-Atoms to RuleML and back. Content of <Query> is still as usual.

4.2. DL-Atoms with Considerations to the Query

In order to represent more complex DL-Rules, we need to deal with considerations and, in particular, to translate the operators $\cup+$, $\cup-$, $\cap-$. In order to introduce considerations inside DL-Atoms, we use the performative <Consider> from FOL RuleML (0.9) [Boley et al 2004]. For simplicity, our examples will only have one consideration at a time and its query content omitted, as they can still be encoded as shown in Subsection 4.1 in a DL-Atom with considerations. For the general case, multiple occurrences of <Consider> take place. We now present a brief account for each kind of consideration found in DLPs.

4.2.1. Considerations to the Query with operator $\cup+$

Whenever the operator $\cup+$ is used in a consideration $S_i op_i p_i$, the instances of p_i in P add to the instances of S_i in L for processing the query. As such, the consideration works in similar way to a normal logic program rule with S_i in the head and p_i in the body. Its result, however, is to be reconsidered each time the query is needed during execution of the DLP. We encode the consideration $S_i \cup +p_i$ using the same structure of general logic programs rules inside an occurrence of the performative <Consider>

Example 4.2 *Let us recall the DL-Atom $DL[keyword \cup+ kw;\ inArea](P,\ A)$ from rule (4) of Example 2.1. The query is the same of Example 4.1, but it has to take into account the extension of keyword (from L_S) by the results of kw (from P_S). By the application of the operator $\cup+$, kw extends the set of positive occurrences of keyword as $kw(p, k)$ allows*

to consider $keyword(p, k)$*. We write the DL-Atom* $DL[keyword \cup + \ kw;\ inArea](P,\ A)$ *in RuleML as:*

```
<Atom kind = "dl">
     <Consider>
          <Implies closure = "universal">
             <Atom>
                 <Rel>kw</Rel>
             </Atom>
             <Atom>
                 <Rel>keyword</Rel>
             </Atom>
          </Implies>
     </Consider>
     <Query> ... </Query>
</Atom>
```

Our considerations are represented through the performative <Consider> with a connective <Implies> as its only content. We call attention to the lack of variables in each occurrence of <Atom> above, as considerations in DL-Atoms do not enumerate the variables for any of the predicates. Both predicates in each consideration are supposed to have the same number and order of arguments.

4.2.2. Considerations to the Query with operator $\cup-$

The operator $\cup-$ in $S_i \cup -p_i$ means that any instance of p_i in P adds to the instances of $\neg S_i$ in L for processing the query. As in Subsection 4.2.1, we use the performative <Consider>, but now the conclusion inside <Implies> is negated.

Example 4.3 *The DL-Atom* $DL[isMale \cup - \ isFemale;\ gender](A,\ S)$ *express a query about the gender of animals who accounts that females are not males. The instances of the predicate* isFemale *in* P *extend the set of negative instances of the concept* isMale *in* L*. It is represented in RuleML as follows:*

```
<Atom kind = "dl">
     <Consider>
          <Implies closure = "universal">
             <Atom>
                 <Rel>isFemale</Rel>
             </Atom>
             <Neg>
                 <Atom>
                    <Rel>isMale</Rel>
                 </Atom>
              </Neg>
          </Implies>
     </Consider>
     <Query> ... </Query>
</Atom>
```

The only difference between two considerations (i) $S_i \cup + p_i$ and (ii) $S_i \cup - p_i$ is that results of p_i add to the set of *positive* instances of S_i in (i) and to the *negative* instances of S_i in (ii). Their representation differs by the negation of the second atom (head/conclusion) through the use of the connective <Neg>.

4.2.3. Considerations to the Query with operator $\cap -$

Considerations of the kind $S_i \cap -p_i$ are such that the failure in proving instances of p_i in P adds to the instances of $\neg S_i$ in L for processing the query, restricting the results of S_i to those of p_i. In this sense, the negation as failure is applied to the condition inside <Implies> and conclusion is negated as in 4.3.

Example 4.4 *Consider the DL-Atom* $DL[Referee \cap - poss_Referee;\ Referee](X)$ *meant to restrict the concept* Referee *(from* L*) up to the values to which* poss_Referee *(from* P*) is true. It can be represented in RuleML as follows:*

```
<Atom kind = "dl">
     <Consider>
          <Implies closure = "universal">
             <Naf>
                 <Atom>
                    <Rel>poss_Referee</Rel>
                 </Atom>
             </Naf>
             <Neg>
                 <Atom>
                     <Rel>Referee</Rel>
                 </Atom>
             </Neg>
          </Implies>
     </Consider>
     <Query> ... </Query>
</Atom>
```

Just as the operator $\cap -$ is defined, we use negation as failure of *poss_Referee* as premise in order to extend the set of negative instances of *Referee*.

The Table 1 summarizes the uses of connectives <Neg> and <Naf> for each operator:

\	<Neg>	<Naf>
$\cup +$	No	No
$\cup -$	Yes	No
$\cap -$	Yes	Yes

Table 1. Usage of <Neg> and <Naf> for each operator in DLP

DL-Rules can be encoded as most extended logic program rules, the only difference being the possible DL-Atoms in the body. We presented how to translate these atoms to RuleML

in a way that both syntax and semantics of DL-Atoms are properly represented. We reinforce that multiple occurrences of the performative <Consider> are needed in case of a DL-Atom with multiple considerations to its query.

5. Conclusion

We have proposed an extension for RuleML capable of expressing DL-Atoms and allowing exchange of DL-Programs rules in the Web. This extension relies on the use of the attribute @kind and occurrence of performatives <Query> and <Consider> inside <Atom> elements and should be considered for inclusion in RuleML 1.0. We also show through examples how this simple extension enables expression of DL-Atoms and, consequently, DL-Programs. This extension does not introduce new terminology to RuleML, for it only suggests new uses and nesting for current tags and an attribute. It provides the necessary expressive power and a straightforward encoding capable of coping with both syntax and semantics of DL-Rules. We defend that allowing performatives to occur inside atoms in the body of rules is very useful, as it also makes possible to easily represent commands from rule based language such as *append(X)* from Prolog through the use of a performative <Assert>. A more profound study of performatives inside atoms is left for future work. Also, we intend to work on the development of RuleML editors to include our suggestions.

6. Acknowledgements

Research partially supported by CAPES (PROCAD).

References

Alsaç, G. and Baral, C. (2002). Reasoning in description logics using declarative logic programming. Technical report, Arizona State University, Arizona.

Antoniou, G. and van Harmelen, F. (2008). *A Semantic Web Primer*. MIT Press, Cambridge, MA, 2. edition.

Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., and Patel-Schneider, P. F., editors (2003). *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press.

Berners-Lee, T. and Fischetti, M. (1999). *Weaving the Web*. Harper.

Boley et al, H. (2004). The first-order logic web language v0.9. `http://ruleml.org/fol/`.

Boley et al, H. (2010). Ruleml v1.0 specification. `http://www.ruleml.org/1.0/`.

Chen, W., Kifer, M., and Warren, D. S. (1993). HILOG: A foundation for higher-order logic programming. *JOURNAL OF LOGIC PROGRAMMING*, 15(3):187–230.

Damásio, C. V., Pan, J. Z., Stoilos, G., and Straccia, U. (2006). An approach to representing uncertainty rules in ruleml. In Eiter, T., Franconi, E., Hodgson, R., and Stephens, S., editors, *RuleML*. IEEE Computer Society.

Dema, T. and Hirtle, D. (2006). Content models for ruleml 0.91. `http://ruleml.org/1.0/xsd/content_models_10.pdf`.

Eiter, T., Ianni, G., Lukasiewicz, T., Schindlauer, R., and Tompits, H. (2008). Combining answer set programming with description logics for the semantic web. *Artif. Intell.*, 172(12-13):1495–1539.

Eiter, T., Lukasiewicz, T., Schindlauer, R., and Tompits, H. (2004). Well-founded semantics for description logic programs in the semantic web. In Antoniou, G. and Boley, H., editors, *RuleML*, volume 3323 of *LNCS*, pages 81–97. Springer.

Gelfond, M. and Lifschitz, V. (1991). Classical negation in logic programs and disjunctive databases. *New Generation Comput.*, 9(3/4):365–386.

Grosof, B., Horrocks, I., Volz, R., and Decker, S. (2003). Description logic programs: Combining logic programs with description logics. In *Proceedings of the World Wide Web Conference (WWW2003)*, Hungary.

Heymans, S. and Vermeir, D. (2003). Integrating description logics and answer set programming. In Bry, F., Henze, N., and Maluszynski, J., editors, *Proc. of International Workshop on Principles and Practice of Semantic Web Reasoning (PPSWR 2003)*, volume 2901 of *LNCS*, pages 146–159. Springer.

Hustadt, U., Motik, B., and Sattler, U. (2007). Reasoning in description logics by a reduction to disjunctive datalog. *J. Autom. Reason.*, 39(3):351–384.

Lloyd, J. (1987). *Foundations of Logic Programming (2nd Extended Edition)*. Springer-Verlag.

Silberschatz, A., Korth, H., and Sudarshan, S. (2001). *Database System Concepts, 4th Edition.* McGraw-Hill.

Tabet, S., Boley, H., and Wagner, G. (2001). Design rationale of RuleML: A markup language for semantic web rules. In Cruz, I. F., Decker, S., Euzenat, J., and McGuinness, D. L., editors, *Proc. Semantic Web Working Symposium*, Stanford University, California.

Determinação da Emoção Demonstrada pelo Interlocutor

Rafael A.M. Gonçalves[1], Diego R. Cueva[1], Marcos R. Pereira-Barretto[1], Fabio G. Cozman[1]

[1]Departamento de Engenharia Mecatrônica e Sistema Mecânicos
Escola Politécnica da Universidade de São Paulo (USP)
Av. Prof. Melo Moraes 2231 – São Paulo - SP – Brazil

marcos.barretto@poli.usp.br

Abstract. *This paper introduces a new model for the determination of conveyed emotion during a human-machine interaction, based on Kalman filtering of instantaneous facial expressions and the emotional trajectory over an emotional surface.*

Resumo. *Este artigo apresenta um modelo para a determinação da emoção demonstrada pelo interlocutor humano em uma interação homem-máquina de longa duração, a partir da expressão facial, basedo na trajetória emocional sobre uma determinada superfície, a partir de sinais instantâneos capturados e submetidos a um filtro de Kalman.*

1. Introdução

Na interação entre pessoas, a comunicação não verbal, como a realizada pelo gestual e pela demonstração de emoções, tem um papel de grande importância. É, portanto, de esperar-se que a comunicação homem-máquina possa também beneficiar-se deste tipo de comunicação. De fato, aplicações modernas incluem comandos gestuais ou por voz, tendo, talvez, iniciado uma revolução na chamada experiência centrada no usuário. O sucesso de tais aplicações é indício concreto de que o ser humano deseja e se beneficia de sistemas capazes de processar uma interação mais natural, o que é base para a Computação Afetiva (*Affective Computing*) e da Robótica Sociável (*Sociable Robots*).

Estudos sustentam que a emoção demonstrada na voz do interlocutor é corretamente reconhecida em aproximadamente 60% dos casos, enquanto que a demonstrada na face é reconhecida em 70% a 98% dos casos [Piccard, 1997], [Ekman, 2003].

No início da década de 1970, Ekman e colaboradores realizaram estudos extensivos sobre expressões faciais, desenvolvendo um método de descrição e identificação de unidades de atuação (*Action Units*, *AUs*) em expressões faciais, publicado sob a sigla FACS (*Facial Action Coding System*) [Ekman, 1978]. A universalidade dos *AUs*, apesar da forte contestação por quase duas décadas, encontrou suporte em estudos interculturais e com populações pré-literatas, levando à sua admissão [Ekman, 2003].

Os trabalhos acima inspiraram diversas abordagens computacionais de classificação a partir de fotografias ou sequências de vídeo como exposto por [Pantic et al., 2000]. Dentre os mais recentes, destacam-se o eMotion, desenvolvido pela Universiteit van Amsterdam [Azcarate, 2005] e o FaceDetect, desenvolvido no Fraunhofer Institute [Fraunhofer, 2011]. Todavia, a identificação de expressões emocionais em uma foto ou uma curta sequência de quadros em um vídeo não permite uma estimação da emoção demonstrada pelo interlocutor durante uma interação; apenas

fornece indicações instantâneas e, como será demonstrado, ruidosas sobre ela. Assim, a detecção de um sorriso, em si, não é conclusiva para que o sistema classifique o usuário como feliz. Também é necessário salientar que as técnicas para avaliação de expressões faciais não contemplam a sobreposição com os efeitos de deformação da face induzidos durante a fala. Logo, aplicações como o eMotion ou o FaceDetect não são treinadas para estes casos e apresentam baixo desempenho quando utilizadas em sequências mais longas, de alguns segundos de interação. Um exemplo ilustrativo do problema pode ser observado pelas saídas do aplicativo eMotion quando aplicado ao vídeo, com quadros ilustrativos na Figura 1.



Figura 1. Imagens do vídeo ilustrativo. As imagens, da esquerda para a direita, foram classificadas pelo eMotion como felicidade (100% de certeza, aos 2,36s), tristeza (70,6%, 0,76s), medo (83,1%, 3,92s) e raiva (76%, 3,04s).

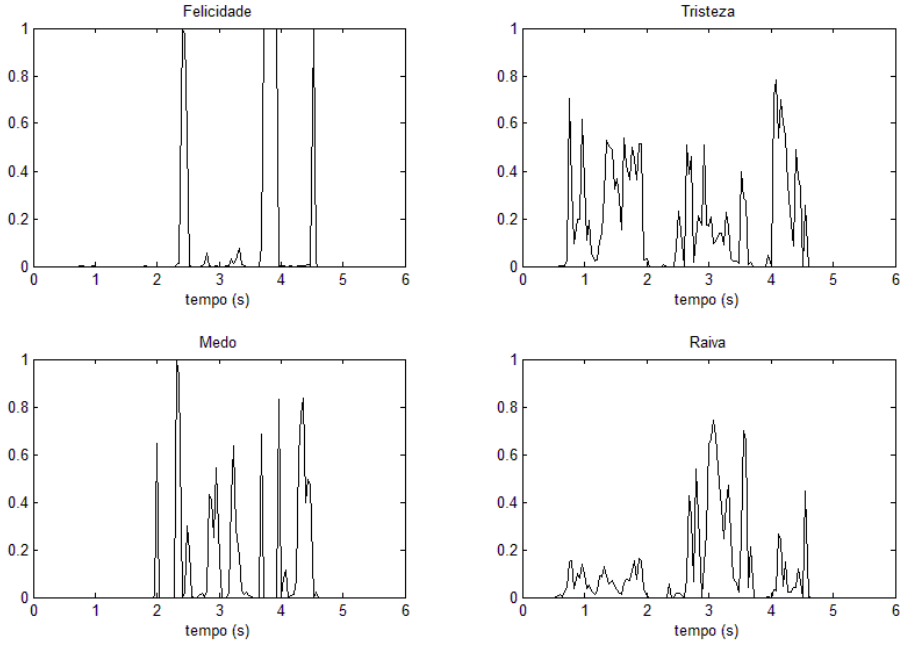


Figura 2. Saídas do eMotion para o vídeo ilustrativo.

Ao analisar as saídas do sensor (Figura 2), um observador humano possivelmente não chegaria a nenhuma conclusão ou até poderia concluir pela predominância da Tristeza. Entretanto, ao assistir ao vídeo, o observador humano facilmente concluiria que a emoção demonstrada é a Raiva.

Neste trabalho é apresentado um modelo para a determinação da emoção demonstrada pelo interlocutor, no qual as indicações instantâneas identificadas pelo aplicativo eMotion são, em certo sentido, integradas no tempo para a determinação da emoção predominante durante uma interação. Na sequência, apresenta-se uma revisão de trabalhos relacionados, para, a seguir, introduzir-se o modelo proposto e os resultados obtidos.

2. Referencial conceitual

As teorias comportamentalistas (*behavioural theories*) dominaram a cena acadêmica da Psicologia entre os anos 30 e 50. De acordo com estas teorias, as emoções eram não diferenciáveis, sendo apenas uma dimensão do comportamento correspondendo ao grau de energia ou atividade. O caráter determinístico e unidimensional da associação evento-emoção é um dos pilares destas teorias: determinado evento causará determinada emoção. As Teorias Cognitivas (*appraisal theories*) assumiram o centro das discussões em meados dos anos 80, embora as ideias iniciais tenham sido lançadas já nos anos 60. De forma simples, determinam que as emoções são elicitadas a partir de avaliações (*appraisals*) de eventos e situações [Roseman, 2001]. Em outras palavras, admitem a existência de processos de elicitação de emoções e que tais processos são comuns a todas as pessoas. Varia a forma com que cada processo se desenvolve individualmente, embora os processos sejam sempre os mesmos. Desta forma, oferecem um modelo que explica (ou descreve) as diferenças individuais, ao mesmo tempo em que determina os aspectos comuns a todas as pessoas.

Este trabalho baseia-se no conceito de *emoção* advindo das Teorias Cognitivas. Segundo Scherer (2001), emoções são "*... an episode of interrelated, synchronized changes in the states of all or most of the five organismic subsystems in response to the evaluation of an external or internal stimulus event as relevant to major concerns of the organism*".

A visão orientada a processos das Teorias Cognitivas levou, de forma natural, à modelagem usando conceitos de dinâmica de sistemas e à ideia de que a elicitação das emoções corresponde ao deslocamento sobre uma superfície, como a da fig.3, discutida originalmente em [Zeeman, 1976] e, depois, por exemplo, [Stewart, 1983], [Scherer, 2000], [van der Maas, 1992], [Sander, 2005].

A Teoria da Catástrofe, que leva à trajetória da Figura 3, foi central para a concepção do modelo apresentado neste artigo, embora não seja diretamente utilizada porque o foco deste artigo é na emoção *demonstrada* pelo interlocutor e não naquela *sentida* pelo receptor dos eventos. A emoção demonstrada permite modelos mais simplificados, como o aqui apresentado.

A utilização das expressões faciais para a inferência da emoção demonstrada fundamenta-se nos estudos de Ekman e seus colaboradores [Ekman, 2003]. As fortes evidências da universalidade das expressões faciais demonstradas permitem que se possa assumir a hipótese de que estas fornecem uma riqueza de indicações sobre as quais um estimador universal pode ser projetado.

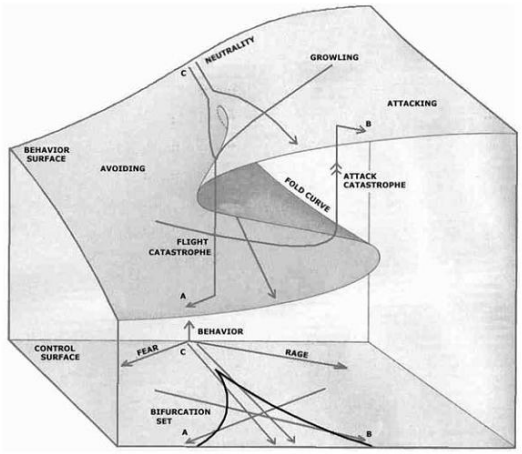


Figura 3. Superfície emocional [Zeeman, 1976].

3. Modelo proposto

O modelo proposto para a estimação da trajetória emocional do interlocutor a partir da análise das expressões faciais baseia-se na modelagem de uma partícula, inicialmente em repouso, sobre uma curva, sujeita a efeitos que alteram sua velocidade instantaneamente e proporcionalmente à probabilidade de detecção de uma expressão facial de conteúdo emocional.

De maneira geral, a curva sobre a qual a partícula se movimenta pode ser estendida a uma superfície N-dimensional contínua, a qual é, portanto, denominada superfície de dinâmica emocional (SDE), pois sua topologia representa os fenômenos envolvidos na evolução da estimação dos estados emocionais e também nas transições entre estes.

As principais vantagens da utilização da SDE no modelo são a possibilidade de abstração das dinâmicas envolvidas e a separação entre o problema de filtragem de sinais de entrada e o de representação dos fenômenos psicológicos das variações dos estados emocionais humanos. Assim, a modelagem de SDE poderia ser entendida como um estudo pertencente ao campo da psicologia onde ao menos duas questões afetam diretamente o desenvolvimento de robôs sociáveis e interfaces inteligentes: a parametrização de uma superfície comum, representativa das interações sociais universalmente aceitas, e a criação de experimentos capazes de reconhecer parâmetros e características das SDE individuais relativos a conceitos como temperamento e personalidade.

Sobre a SDE são posicionados atratores correspondendo a cada emoção a ser percebida.

A partícula, portanto, movimenta-se livremente sobre a SDE segundo efeitos de atração e escoamento, em direção ao estado neutro. Os efeitos de atração compreendem a intensidade com a qual a partícula é atraída para as projeções dos atratores na superfície. O efeito de escoamento é compreendido como a velocidade com a qual a partícula escoa sobre a superfície no sentido de assentá-la em mínimos locais. Assim, a velocidade da particular é dada pela Eq.I.

$$V_p = V_e + \sum_{a=1}^{N} V_a \quad \text{(I)}$$

onde V_p é a velocidade da partícula, V_e a sua velocidade inicial e V_a, a velocidade na direção de cada um dos atratores, computada de forma vetorial.

A intensidade instantânea de atração é dada por um detector instantâneo de emoções, como os de expressões faciais citados anteriormente, ou outros, como os baseados em EEG, condutância da pele, etc. Esta intensidade instantânea pode ser pré-filtrada, para eliminação de ruídos aleatórios, por exemplo.

4. Exemplo

Embora este trabalho apresente o conceito de superfície de dinâmica emocional como uma ferramenta para possibilitar a abstração, o estudo de uma SDE representativa do ser humano foge ao escopo dos trabalhos iniciais relacionados à obtenção de sinais e sua filtragem. Desta maneira, para fins de demonstrar o modelo proposto e realizar experimentos como prova de conceito, optou-se pela utilização de uma superfície simples. Respeitando as condições de continuidade e de existência de ao menos um ponto de mínimo, a superfície escolhida foi um paraboloide na forma:

$$z = f(x, y) = ax^2 + by^2 \quad \text{(II)}$$

$$\gamma(x, y) = (x, y, ax^2 + by^2) \quad \text{(III)}$$

$$a = b = 0{,}6 \quad \text{(IV)}$$

Da mesma forma, o estudo detalhado do posicionamento dos atratores com relação aos aspectos psicológicos foge ao tema central. Assim, adotou-se a configuração mostrada na Tabela 1.

Tabela 1. Posicionamento dos Atratores

Emoção	Atrator
Felicidade	$[60, 60, 0]$
Raiva	$[-60, 60, 0]$
Tristeza	$[-60, -60, 0]$
Medo	$[60, -60, 0]$

Note-se que a localização dos atratores não corresponde ao seu posicionamento no plano Arousal X Valence (Plano AV) [Gunes, 2006]. De fato, a emoção Medo foi posicionada no quarto quadrante ao invés do terceiro, próxima à origem, seu local no Plano AV. Ressalta-se este aspecto para destacar que uma SDE, assim como a mostrada na Figura 3, não precisa ter correspondência direta com o Plano AV. Tal escolha teve

por base o modelo de [Zeeman, 1976] onde medo e raiva são comportamentos ortogonais e mostrou-se viável durante as fases de aprendizado e testes.

Um dos componentes centrais do modelo proposto é a decomposição das velocidades de atração em componentes X e Y, respeitando a topologia da superfície. As informações fornecidas pelos sensores, após pré-filtragem, são interpretadas no modelo como velocidades sobre a trajetória, S, que liga a posição atual da partícula, $\vec{P}$, à posição do atrator relativo à emoção analisada projetado na SDE, $\vec{A}$. Dessa forma, cabe observar que, a rigor, o módulo dos atratores não tem significado especial. Assim, para cada atrator:

$$\overrightarrow{AP} = \vec{A} - \vec{P} = \left[a_{px}, a_{py}, a_{pz}\right] \quad \text{(V)}$$

$$S(x) = \gamma(x, rx) \quad \text{(VI)}$$

$$r = \left|\frac{a_{py}}{a_{px}}\right|, a_{px} \neq 0 \quad \text{(VII)}$$

$$escala = \left|\frac{dS(x)}{dx}\right| = \left|\frac{d\gamma(x, rx)}{dx}\right| = \sqrt{1 + r^2 + [2(a + br^2) * P_x]^2} \quad \text{(VIII)}$$

$$V_{x,a} = \frac{dS}{dt} * \frac{sinal(a_{px})}{escala} * \frac{dS}{dx} \cdot \vec{i} \quad \text{(IX)}$$

$$V_{y,a} = \frac{dS}{dt} * \frac{sinal(a_{py})}{escala} * \frac{dS}{dx} \cdot \vec{j} \quad \text{(X)}$$

Após a decomposição dos efeitos de atração para cada uma das quatro emoções consideradas, o equacionamento para a modelagem do movimento da partícula sobre a SDE é bastante simplificado e foi omitido, por questões de espaço.

Enquanto a SDE pode ser discutida em nível conceitual, a pré-filtragem é fortemente relacionada com o sensor escolhido. Para validação do modelo, escolheu-se o aplicativo eMotion, para a análise quadro a quadro das expressões faciais. Observou-se que este fornecia dados com ruído e também tendência às emoções felicidade e, principalmente, tristeza. Após cautelosa revisão das publicações relativas à área de identificação de expressões, percebeu-se que o problema de eliminação dos ruídos ocasionados pela deformação da face durante a fala não é abordado. Foi possível identificar que a concavidade do arco definido pela boca como principal atributo na detecção de felicidade ou tristeza pelo eMotion. Assim sendo, adotou-se um filtro capaz de analisar as leituras levando em consideração diferentes confiabilidades da informação seria adequado ao tratamento dos dados, o que levou naturalmente ao filtro de Kalman.

A implementação do filtro de Kalman contempla a utilização de um modelo linear subjacente. Posto isso, optou-se por um modelo de primeira ordem da forma apresentada na Equação XI.

$$\frac{Y(s)}{R(s)} = \frac{K_k}{\tau s + 1} \quad \text{(XI)}$$

Nesta forma, o filtro se apresenta como um sistema dinâmico de primeira ordem de modo a garantir que o sinal filtrado também seja contínuo. Mais do que isso, o sistema dinâmico constitui ainda um filtro passa-baixas que pode ser ajustado de forma a atenuar os ruídos. Esta escolha de projeto reduz a filtragem a sistemas do tipo *SISO* (*Single-Input/Single-Output*) e, portanto, não leva em consideração a influência entre os sinais.

Após a implementação dos filtros, valores para as variâncias de entrada (Q) e de saída (R) do filtro de Kalman foram ajustadas mediante testes com os vídeos de amostra. Para o sistema subjacente escolheu-se configurar frequências e ganhos idênticos para os quatro filtros. A Tabela 2 traz os parâmetros utilizados.

Tabela 2. Parâmetros dos Filtros de Kalman

	Q	R	K_k	τ
Felicidade	0,1	0,080	5	1,5
Raiva	0,1	0,100	5	1,5
Tristeza	0,1	0,035	5	1,5
Medo	0,1	0,010	5	1,5

É válido ressaltar que, uma vez que a implementação da rotina de MatLab tem como objetivo execução em tempo real- processando os dados à medida que chegam- os filtros de Kalman devem resolver a equação de Riccati iterativamente durante a execução. Nessa situação, uma estimativa inicial da variância e do ganho de Kalman é necessária para o início do processo. Seguindo uma prática comum, tais estimativas são configuradas como nulas.

5. Experimentos

5.1. Metodologia

O processo experimental computacional iniciou-se com a seleção de vídeos representativos das emoções. Após assistir todos os vídeos do *corpus eNTERFACE'05 Audio-Visual Emotion Database* [Martin, 2005], foram selecionados aqueles para os quais os autores deste trabalho julgaram que o conteúdo emocional transmitido pelas expressões faciais era condizente com a categorização do arquivo na base. Para tais vídeos, realizou-se o ajuste da malha e a análise. Durante o processo foi também observado se houve, em algum quadro, descolamento da malha, ocasionado por perda de algum dos pontos de referência da face. A ocorrência de tal fenômeno desqualificava os resultados, ocasionando exclusão do vídeo da base de testes. A alta taxa de descolamento, não reportada previamente pelos criadores do corpus, foi atribuída

principalmente às deformações faciais ocasionadas pela fala. Este método de seleção resultou na escolha de trinta vídeos.

Os dados quadro a quadro de todos os vídeos foram, em seguida, processados pela rotina desenvolvida em MatLab, a qual realizou dois tipos de tratamento nos sinais: filtro de média com janelamento igual à taxa de quadros do vídeo e aplicação do filtro de Kalman, apresentado anteriormente. Foram calculadas as médias dos dados conforme recebidos e obtidos após filtragem. Em todos os casos houve cuidado especial para análise da máxima variância do filtro de Kalman e também do máximo ganho de inovação (ganho de Kalman), garantindo a convergência do filtro.

Finalmente, utilizando o sinal pós-filtro de Kalman, determinou-se a trajetória sobre a SDE.

5.2. Aplicação ao vídeo ilustrativo

A aplicação dos filtros ao vídeo ilustrativo, cujas saídas em bruto foram mostradas na Figura 2, está demonstrada na Figura 4, onde a linha contínua representa o filtro de Kalman e a linha tracejada, o filtro de média móvel.

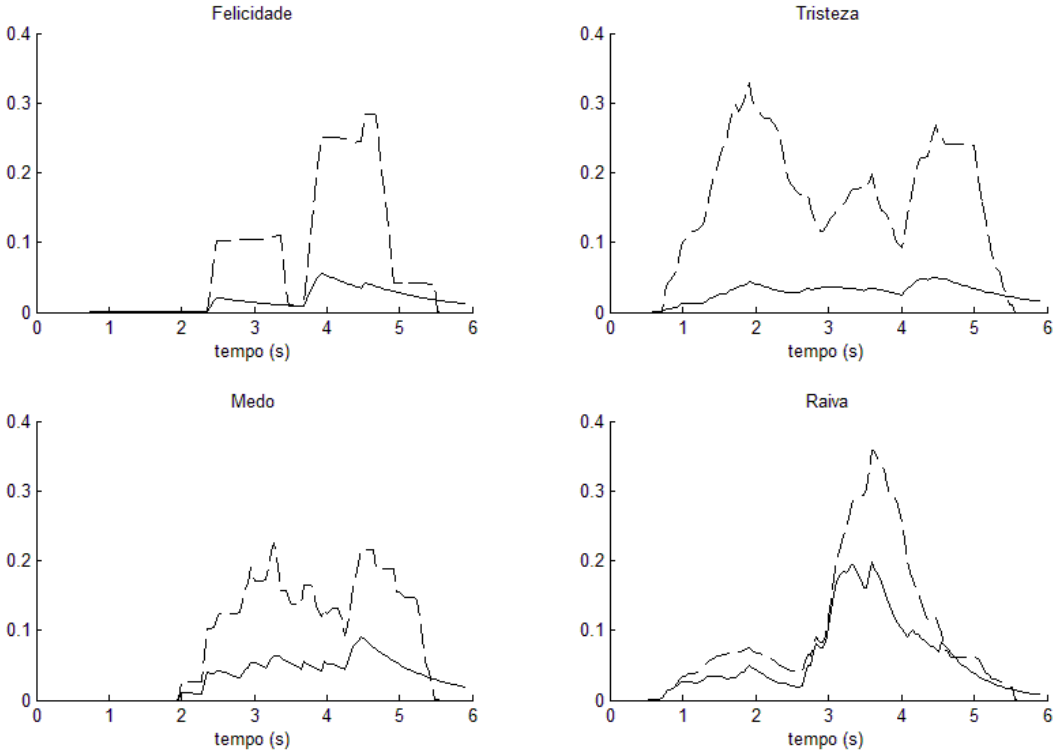


Figura 4. Saída do filtro de Kalman e do filtro de media móvel para o video ilustrativo.

A Tabela 3 apresenta média e desvios padrão normalizados para os dados pré e pós-filtros.

Tabela 3. Análise estatística para o vídeo ilustrativo

	Original		Média Móvel		Kalman	
Emoção	**μ**	**σ**	**μ**	**σ**	**μ**	**σ**
Felicidade	0.175	0.634	0.175	0.237	0.114	0.127
Tristeza	0.377	0.532	0.377	0.254	0.207	0.108
Medo	0.211	0.544	0.211	0.206	0.234	0.203
Raiva	0.236	0.434	0.236	0.257	0.445	0.434

Como se pode perceber, a classificação correta, Raiva, é determinada pelo sistema com filtro de Kalman.

A Figura 5 mostra a trajetória sobre a superfície SDE e ilustra também como o sistema permite estimar a cada instante o estado emocional do interlocutor.

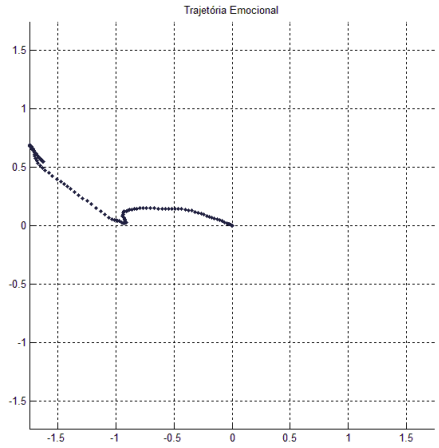


Figura 5. Trajetória sobre a superfície SDE para o vídeo ilustrativo.

A trajetória desenvolve-se no segundo quadrante, onde está posicionado o atrator Raiva. Há momentos mais fortes de tristeza, como se pode perceber.

5.3. Resultados

O vídeo ilustrativo, assim como os demais vídeos utilizados neste trabalho, foram obtidos a partir do *corpus eNTERFACE'05 Audio-Visual Emotion Database* [Martin, 2005], um banco de dados de cenas de curta duração, no qual indivíduos são convidados a expressar uma frase emocional da forma que mais desejarem.

Dos 30 vídeos selecionados, 16 foram utilizados para ajuste dos parâmetros do modelo. O experimento foi repetido para demais 14 vídeos selecionados. Com exceção de um caso (número 14), a trajetória desenvolveu-se corretamente nas amostras, como demonstrado na Tabela 4 onde classificação declarada é aquela registrada no corpus como correspondente ao vídeo e classificação avaliada é aquela obtida com o método descrito.

Tabela 4. Resultados do experimento

#	Arquivo	Classificação		#	Arquivo	Classificação	
		Declarada	Avaliada			Declarada	Avaliada
1	S1sa1	Tristeza	Tristeza	8	S43an4	Raiva	Raiva
2	S38an1	Raiva	Raiva	9	S43fe2	Medo	Medo
3	S38fe3	Medo	Medo	10	S42fe1	Medo	Medo
4	S42sa1	Tristeza	Tristeza	11	S43sa1	Tristeza	Tristeza
5	S43ha1	Felicidade	Felicidade	12	S43sa3	Tristeza	Tristeza
6	S43an2	Raiva	Raiva	13	S43sa4	Tristeza	Tristeza
7	S43an3	Raiva	Raiva	14	S43sa5	Tristeza	Raiva

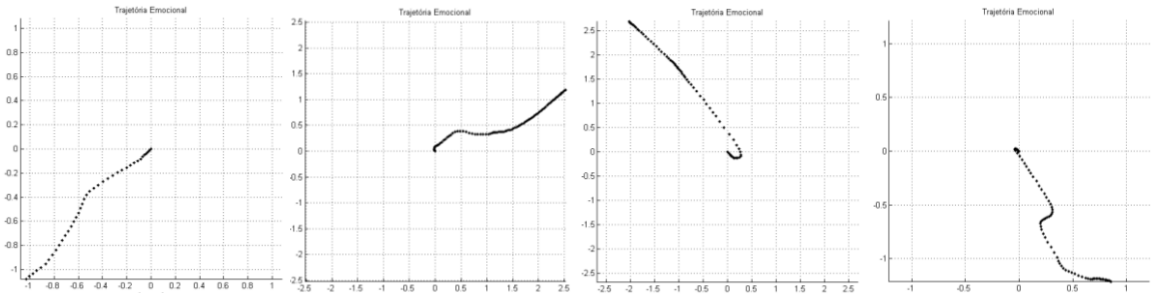


Figura 6. Trajetórias sobre a superfície SDE para vídeos 1, 5, 7 e 10.

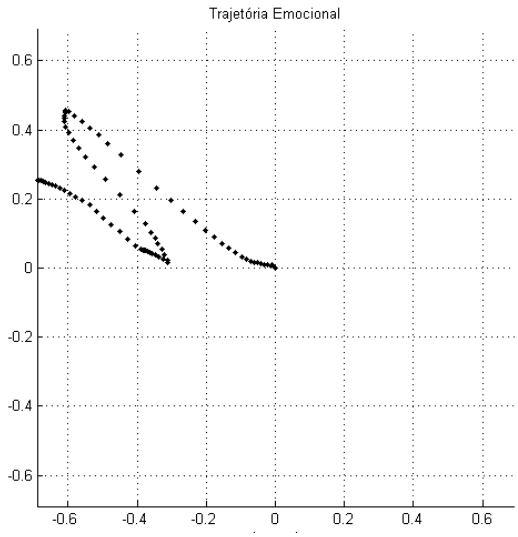


Figura 7. Trajetória sobre a superfície SDE para vídeo 14. Análise reportou oscilações em Raiva para um vídeo de Tristeza.

5.4. Discussão

A comparação entre a análise do sinal como provindo do eMotion e o sinal filtrado, seja pelo filtro de média móvel ou o filtro de Kalman, fornece subsídios para afirmar-se que os processos de filtragem possibilitaram uma classificação mais precisa dos vídeos. O uso de uma trajetória sobre uma superfície permite acompanhar a evolução da emoção no tempo, determinando-se não somente sua característica mais dominante como também permitindo que variações possam ser identificadas.

A experimentação também demonstrou que o "sensor" (no caso, o aplicativo eMotion) é um ponto crítico para a construção de sistemas robustos de determinação da emoção demonstrada pelo interlocutor. Em particular, acredita-se que sensores capazes de estimar mais adequadamente a face neutra do interlocutor e também de tratar as deformações na face induzidas pela fala, possibilitariam grandes ganhos no desempenho.

6. Conclusão

Este trabalho apresentou um método para modelagem e estimação da emoção demonstrada por um interlocutor, utilizando indicações fornecidas pela observação de expressões faciais, capturadas e analisadas preliminarmente por uma aplicação comercial.

O sistema proposto foi implementado em MatLab e testado sobre uma base de dados já existente e disponível para pesquisa, obtendo resultados significativamente melhores à interpretação dos sinais conforme obtidos do sensor. A implementação, mesmo com o Matlab, é executada em tempo real, o que fornece a certeza de que é aplicável em Robótica Sociável e Computação Afetiva.

Em trabalhos futuros, os autores acreditam que seja necessário implementar um sensor capaz de lidar com as principais fontes de ruído: a estimação da posição neutra da face e as deformações durante a fala. Além disso, alternativas de fusão sensorial inteligente e otimização das escolhas de parâmetros dos filtros oferecem possibilidade de melhor desempenho.

Por fim, os autores acreditam que a determinação de superfícies de dinâmica emocional possibilitariam modelar fenômenos mais complexos relacionados à trajetória estimada.

7. Agradecimentos

Os autores agradecem ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), ao Departamento de Engenharia Mecatrônica da Escola Politécnica da Universidade de São Paulo e à FAPESP, através do processo 2008/03995-5, pelo apoio financeiro à pesquisa.

8. Referências

[Azcarate, 2005] Azcarate, A.; Hageloh, F.; Sande, K.; Valenti, R. "Automatic facial emotion recognition". Universiteit van Amsterdam, 2005.

[Ekman, 1978] Ekman, P.; Friesen, W.V. "Facial Action Coding System: a technique for the measurement of facial movement". Consulting Psychologists Press, 1978.

[Ekman, 2003] Ekman, P.; Friesen, W.V. "Unmasking the face". Malor Books, 2003.

[Fraunhofer,2011] Fraunhofer Facedetect. Disponível em: <http://www.iis.fraunhofer.de/en/bf/bv/ks/gpe/demo/>. Acessado em: 10 de abril de 2011.

[Gunes, 2006] Gunes, H; Piccardi, M. "Observer Annotation of Affective Display and Evaluation of Expressivity: Face vs. Face-and-Body". HCSNet Workshop on the Use of Vision in HCI (VisHCI 2006), Canberra, Australia, 2006.

[Martin et al., 2005] Martin, O. Kotsia, I. Macq, B. Pitas, I. The eNTERFACE'05 Audio-Visual Emotion Database. Université Catholique de Louvain; Aristotle University of Thessaloniki, 2005.

[Piccard,1997]Piccard, R.W. "Affective Computing". MIT Press, 1997.

[Pantic, 2000]Pantic, M.; Rothkrantz, L.J.M. "Automatic analysis of facial expressions: state of art". IEEE Trans. On Pattern Analysis and Machine Intelligence, vol.22 no. 12, December, 2000.

[Roseman, 2001] Roseman, I.J.; Smith, C.A. "Appraisal Theory - Overview, Assumptions, Varieties, Controversies". In "Appraisal Processes in Emotion – Theory, Methods, Research" editado por Scherer, K; Schorr, A.; Johnstone, T. Oxford University Press, 2001.

[Sander, 2005] Sander, D.; Grandjean, D.; Scherer, K.R. "A systems approach to appraisal mechanisms in emotion". Neural Networks vol.18 pgs.317-352, 2005.

[Scherer,2000] Scherer, K. "Emotions as episodes of subsystem synchronization driven by nonlinear appraisal processes". In "Dynamic Systems Approaches to Emotional Development", editado por Lewis, M.D.; Granic, I. Cambridge Press, 2000.

[Scherer, 2001] Scherer, K. R. "Appraisal considered as a process of multilevel sequential checking". In "Appraisal Processes in Emotion – Theory, Methods, Research" editado por Scherer, K; Schorr, A.; Johnstone, T. Oxford University Press, 2001.

[Stewart, 1983]. Stewart,I.N.; Peregoy,P.L. "Catastrophe theory modeling in Psychology". Psychological Bulletin vol. 94, no. 2, pags. 336-362, 1983.

[van der Maas,1992] van der Maas, H.L.J.; Molenaar, P.C.M. "Stagewise cognitive development: an application of Catastrophe Theory". Psychological Review vol.99 no.2 pags. 395-417, 1992.

[Zeeman, 1976]. Zeeman, E.C. "Catastrophe theory". Scientific American vol.4 no.254 pags. 65-83, 1976.

Resultados Preliminares na Classificação de Insetos Utilizando Sensores Ópticos

Diego F. Silva[1], Gustavo E. Batista[1 2], Eamonn Keogh[2], and Agenor Mafra-Neto[3]

[1]Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo

[2]Dept. of Computer Science & Engineering – University of California, Riverside

[3]ISCA Technologies, Riverside

diegofsilva@icmc.usp.br, gbatista@icmc.usp.br, eamonn@cs.ucr.edu,

president@iscatech.com

Abstract. *In this work we present a low-cost optical sensor to automatically count and classify disease vector insects in real time. We show that although the counting task is relatively straightforward, the insect classification in species is more elaborated and requires the identification of attributes in data. We evaluate two attributes: the wing-beat frequency and the circadian rhythm; as well as we present additional attributes that can be incorporated to the classifiers in future research. Our results are promising, with data collected with three insect species, we were able to classify them with accuracy higher than 90%.*

Resumo. *Neste trabalho nós apresentamos um sensor óptico de baixo custo para contagem e classificação automática de insetos vetores de doenças em tempo real. Nós mostramos que embora a tarefa de contagem seja relativamente simples, a classificação dos insetos em suas espécies é mais elaborada e requer a extração de atributos dos dados. São avaliados dois atributos: a frequência de batida de asas e o ritmo circadiano; bem como são apresentados outros atributos que podem ser incorporados aos classificadores em pesquisas futuras. Nossos resultados preliminares são promissores, com dados coletados com três espécies de insetos, foi possível classificá-los com acurácia acima de 90%.*

1. Introdução

É inegável a importância dos insetos para a vida humana, seja benéfica ou maleficamente. Por exemplo, insetos são vetores de doenças que matam dois milhões de pessoas por ano, deixando outras 700 milhões adoecidas [WHO 2010], além de causar perdas estimadas em dezenas de bilhões na agricultura e pecuária. Ao mesmo tempo, estima-se que insetos polinizam a maioria das espécies empregadas na agricultura, sendo que um terço de todo alimento consumido no mundo é polinizado somente por abelhas [Dixon 2009].

Dada a importância dos insetos para a vida humana, nas últimas décadas pesquisadores têm desenvolvido métodos de controle de insetos. Por exemplo, foram propostas inúmeras técnicas para controlar infestações de mosquitos responsáveis por transmitir

doenças como a malária e a dengue, tais como a pulverização de inseticidas e larvicidas; a introdução no ambiente de predadores de larvas e insetos adultos; a dispersão de mosquitos machos estéreis; a redução de *habitat* como a remoção de objetos que podem acumular água parada; o uso de armadilhas, entre outras técnicas [Walker 2002]. Para serem empregados, tais métodos de controle requerem o conhecimento da distribuição espaço-temporal dos insetos. Sem tal conhecimento, o emprego dessas técnicas se torna ineficiente.

Atualmente, estudar a distribuição espaço-temporal de insetos é uma tarefa custosa em termos de recursos e tempo de especialistas. Em linhas gerais, contagens de insetos são realizadas por meio de armadilhas, que são recolhidas periodicamente e analisadas por especialistas que identificam e contam manualmente as espécies. Além de ser uma abordagem cara, possui um atraso entre o momento que a armadilha é instalada e analisada, que pode ser de somente uma semana, mas esse tempo pode representar mais de meia vida de um inseto adulto. Dessa maneira, a doença já pode ter contaminado um grande número de pessoas no momento em que os dados estejam disponíveis aos especialistas [Patnaik et al. 2007]. Existe, portanto, uma necessidade por sensores automáticos e precisos, capazes de detectar, classificar e contar em tempo real insetos de diferentes espécies.

Estamos propondo um sensor que utiliza um feixe de luz laser para capturar dados de insetos à distância. Quando um inseto atravessa esse feixe, um fotosensor captura a variação da luz resultante da oclusão parcial do feixe causada pelo inseto. Essa variação da luz possui informações como a frequência de batida de asas do inseto, que podem ser utilizadas para classificar os insetos.

Apesar desse sensor poder ser aplicado a qualquer inseto com asas, estamos interessados na classificação de mosquitos vetores de doenças, em especial dos gêneros *Anopheles* e *Aedes*, vetores da malária e dengue, respectivamente. Estima-se que, por ano, a dengue afeta 50 milhões de pessoas e é considerada uma doença endêmica em mais de 100 países. A malária afeta entre 300 e 500 milhões anualmente, matando aproximadamente um milhão de pessoas.

Este trabalho mostra como os dados obtidos pelo sensor proposto podem ser utilizados para classificar insetos. Concretamente, nós mostramos resultados obtidos com três espécies de insetos, sendo uma espécie de abelhas e duas espécies de mosquitos. Para se obter uma alta acurácia de classificação é necessário identificar e extrair atributos dos dados e utilizar tais atributos para construir os classificadores. Tal argumento é sustentado por uma série de experimentos com algoritmos de classificação que utilizam os dados sem atributos previamente identificados e outros que utilizam atributos como a frequência de batida de asas do inseto e o ritmo circadiano. Ainda, apresentamos alguns atributos adicionais que podem ser extraídos e utilizados em pesquisas futuras.

Este trabalho está organizado da seguinte maneira: na Seção 2 é apresentado o sensor óptico para capturar dados de insetos; na Seção 3 é discutido como os dados de insetos foram coletados e posteriormente utilizados para construir os classificadores; na Seção 4 é realizada uma discussão de quais atributos adicionais podem ser utilizados para classificar os insetos; por fim, na Seção 5 são apresentadas as conclusões deste trabalho.

2. Sensor Classificador de Insetos

Estamos desenvolvendo o hardware e o software de um sensor para capturar informações de insetos à distância. A ideia central desse sensor é utilizar um feixe de luz laser, um refletor total e um fotosensor, como mostrado na Figura 1. Quando um inseto atravessa o feixe de luz, o fotosensor captura a variação resultante da oclusão parcial da luz pelo inseto. Essa variação é registrada na forma de uma série temporal, a qual armazena informações que podem ser utilizadas na classificação. O objetivo é construir um sistema que utilize tais séries temporais como entrada e forneça contagens dos insetos separadas por espécie e sexo, uma vez que somente as fêmeas de mosquitos são vetores de doenças.

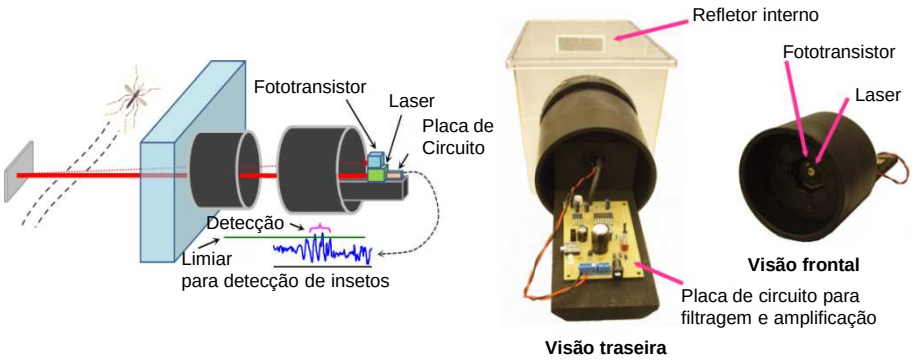


Figura 1. Projeto lógico do sensor em desenvolvimento (esquerda) e uma fotografia do sensor em sua versão atual (direita)

O sensor utiliza um circuito especialmente projetado para filtrar e amplificar os sinais capturados pelo fototransistor, atualmente gravados como arquivos de áudio. Apesar dos dados serem obtidos opticamente, os sinais soam exatamente como o som dos insetos capturados por microfones, com duas vantagens importantes: a primeira é que a luz laser pode trafegar grandes distâncias sem perda significativa de intensidade e, portanto, o sensor pode cobrir uma área de pelo menos algumas dezenas de metros quadrados; a segunda é que o sensor é totalmente *surdo* para qualquer outra interferência que não atravesse a luz laser, como a voz de pessoas, pássaros e aviões, portanto é menos susceptível a falsos positivos.

Nosso objetivo é que haja pelo menos alguns milhares desses sensores espalhados em campo e em países afetados por doenças como a malária e a dengue. Sendo assim, o sensor deve ser de baixíssimo custo, utilizando componentes facilmente encontrados. Idealmente, uma unidade do sensor deve custar menos de R$10,00. Dessa forma, a utilização ampla se torna mais fácil e, por ter baixo valor comercial e de troca, desestimula a prática de roubo.

3. Avaliação Experimental

Iniciamos esta seção explicando a nossa filosofia experimental. Nós projetamos todos os experimentos de forma que eles sejam *facilmente* reproduzíveis. Por exemplo, se o leitor quiser reproduzir qualquer figura deste artigo, ele pode simplesmente executar um programa para essa tarefa. Para isso, foi criado um sítio web[1] com todos os códigos-fonte

[1] `http://www.icmc.usp.br/~diegofsilva/ENIA2011`

de programas e conjuntos de dados, bem como planilhas com resultados experimentais detalhados. Todos os programas estão disponíveis em Matlab, o qual possui um clone totalmente gratuito (Octave) disponível. Além disso, disponibilizamos todos os arquivos de projeto de hardware e software do sensor em um sítio web[2] e estamos distribuindo gratuitamente hardware e software do sensor para pesquisadores em Entomologia interessados em construir uma base de dados com sinais de insetos de diferentes espécies.

A seguir, descrevemos como os dados foram coletados e pré-processados utilizando três espécies de insetos e como os classificadores foram costruídos a partir desses dados. Acreditamos que o grande desafio de construir classificadores precisos neste domínio está na busca por atributos. Para comprovar essa hipótese, iniciamos nossos experimentos utilizando diversos paradigmas de aprendizado para aprender diretamente com os dados coletados como exemplos. Diante dos resultados modestos obtidos com os dados originais, investigamos a extração de atributos, particularmente a frequência de batida de asas e o ritmo circadiano dos mosquitos.

3.1. Coleta de Dados

Os dados foram coletados durante 15 dias de três espécies de insetos: *Bombus impatiens*, espécie de abelhas, benéfica ao ser humano; *Aedes aegypti*, vetor da dengue e da febre amarela; e *Culex quinquefasciatus*, vetor de diversas doenças graves como encefalites (em especial St. Louis e Venezuelana), filariose bancroftiana e febre do Nilo Ocidental.

Os dados foram coletados em laboratório, com condições de temperatura e umidade controladas. As temperaturas durante as coletas variaram de 21^{o}C a 23^{o}C e a umidade entre 50% e 70%. Os arquivos de áudio possuem uma taxa de amostragem de 44100Hz, sendo posteriormente amostrados para 16000Hz, a fim de reduzir os requisitos de armazenamento e processamento dos dados. Essa taxa de amostragem é suficiente para caracterizar os sinais de insetos, uma vez que é capaz de representar até frequências de 8000Hz, sendo que os insetos normalmente apresentam sinais na faixa de 100Hz a 1000Hz.

Os arquivos de áudio consistem em geral de ruído de fundo com "bips" ocasionais, resultado dos breves cruzamentos do inseto com o laser. Na Figura 2, é exibido um exemplo do dado coletado pelo sensor. O inseto em questão é da espécie *Aedes aegypti*. Note que o sinal gerado pelo inseto possui uma amplitude significativamente maior que a amplitude do ruído de fundo. Dessa maneira, é uma tarefa relativamente simples *contar* o número de insetos que cruzam o laser, sendo que *classificar* os sinais nas espécies é uma tarefa bem mais sofisticada.

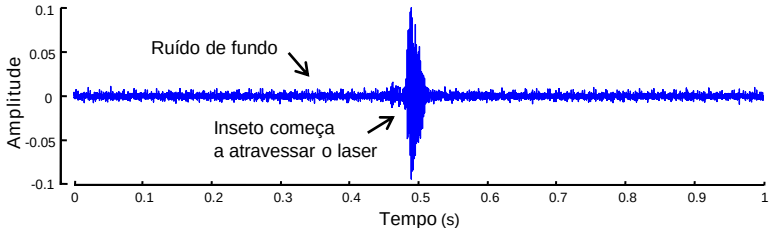


Figura 2. Exemplo do sinal gravado pelo sensor, espécie *Aedes aegypti*

Para separar o ruído de fundo dos sinais gerados pelos insetos, foi implementado

[2] http://www.cs.usr.edu/~eamonn/CE

um detector, descrito no Algoritmo 1. Esse algoritmo utiliza duas sub-rotinas auxiliares: normalizaZ que realiza normalização em índices z, e TRF que calcula a transformada de Fourier. O detector simplesmente utiliza uma janela deslizante sobre os dados, e calcula o espectro do sinal dentro da janela. Como a maioria dos insetos possui frequência de batida de asas na faixa entre 100Hz e 1000Hz, utilizamos a magnitude máxima no espectro do sinal dentro dessa faixa de frequências como um valor de confiança para o detector. Dessa maneira, quanto maior for a magnitude do sinal nessa faixa, maior a confiança de que o sinal não é um ruído de fundo. Todos os sinais com magnitude acima de um limiar especificado pelo usuário são considerados um evento gerado por um inseto. A alta razão sinal-ruído dos dados coletados pelo sensor permite ao usuário especificar valores baixos para o limiar de forma a garantir a identificação de sinais curtos ou de baixa amplitude, sem o risco de falsos positivos. Na Figura 3, ilustramos como o detector funciona.

Algoritmo 1: Detector da passagem do mosquito pelo sensor

Entrada: Um vetor com dados de áudio a
Um limiar de aceitação l
Um tamanho do passo t_p
Uma frequência mínima de interesse min_f = 100 Hz
Uma frequência máxima de interesse max_f = 1000 Hz
início
  a_z = normalizaZ(a)
  para $i = 0$ *até tamanho*(a_z)$-t_p$, *passo* t_p **faça**
    $janela = a_z(i : i + t_p)$
    f = TRF($janela$)
    $d(i : i + ws) = max(abs(f(min_f : max_f)))$
  fim para
  retorna d
fim

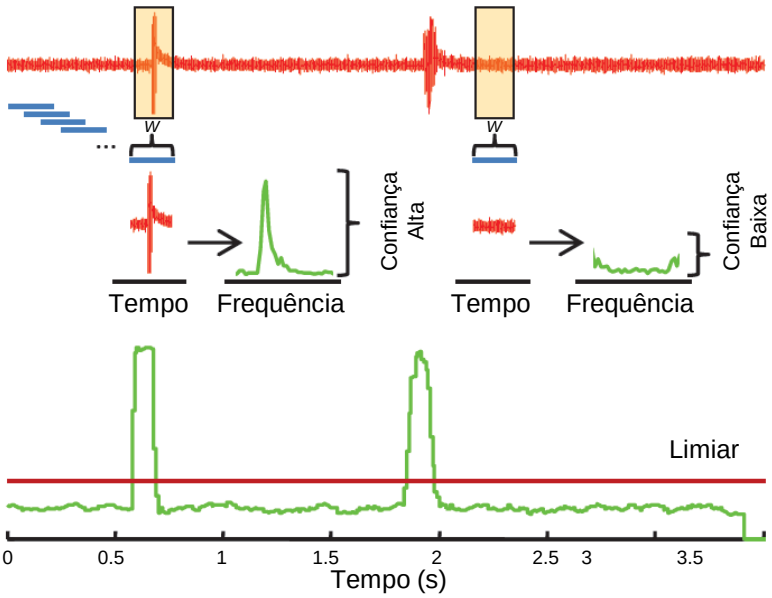


Figura 3. Esquema do detector de batida de asas

O passo final de pré-processamento consiste em extrair fragmentos com eventos

individuais utilizando o detector. Como os eventos gerados pelos insetos possuem durações diferentes, decidimos extrair fragmentos de um segundo. Tais fragmentos são longos o suficiente para armazenar a maioria dos eventos, os quais tipicamente duram um ou dois décimos de segundo. Note que os eventos podem ocorrer com qualquer intervalo de tempo entre eles, de forma que não é possível garantir que cada fragmento de um segundo possui somente um único evento. Entretanto, podemos garantir que cada fragmento possui ao menos um evento. Uma descrição resumida dos dados coletados pelo sensor é mostrada na Tabela 1.

Tabela 1. Descrição resumida dos dados coletados

Total de Amostras	Classe	Exemplos	Distribuição (%)
5982	*Aedes Aegypti*	1231	20,58%
	Bombus impatiens	499	8,34%
	Culex quinquefasciatus	4252	71,08%

3.2. Classificação sem a Identificação de Atributos

Como descrito na Tabela 1, a detecção e pré-processamento dos dados resultou em 5982 séries temporais, sendo que cada uma dessas séries tem duração de um segundo, ou 16000 observações. Uma forma bastante simples de interpretar esses dados é pensá-los como uma tabela atributo-valor com 5982 exemplos e 16000 "atributos". Cada exemplo foi rotulado de acordo com o inseto que gerou os dados, uma vez que os dados foram coletados com cada espécie de inseto separadamente. As limitações dessa representação para este problema são bastante evidentes. Por exemplo, apesar das séries temporais representarem áudios de um segundo, a duração dos sinais é variável, tendo duração típica de um a dois décimos de segundo (vide Figura 2). Por outro lado, tal representação permite a utilização de um grande número de algoritmos de classificação disponíveis e tais algoritmos podem prover resultados que podem ser utilizados como base para comparação para outras técnicas a serem avaliadas.

Neste experimento, utilizamos a ferramenta de domínio público Weka [Witten and Frank 2005]. Para a execução dos experimentos, foram utilizados quatro diferentes algoritmos de classificação: IBk (implementação do método k-vizinhos mais próximos, com os parâmetros k=5 e votação ponderada pelo inverso da distância); classificador Naïve Bayes; J48 (implementação do algoritmo de árvore de decisão C4.5); e SMO (implementação de uma Máquina de Suporte Vetorial). Todos os experimentos foram feitos utilizando o método de *resampling 10-fold cross validation*. Nós reportamos a acurácia média nos 10 conjuntos de teste e os respectivos desvios padrão como as nossas principais medidas de desempenho. Os resultados são exibidos na Tabela 2.

Tabela 2. Descrição resumida dos resultados da classificação utilizando o sinal não processado

Algoritmo	Acurácia	Desvio Padrão
IBk	68,34%	1,58
J48	66,50%	1,86
Naïve Bayes	40,17%	1,72
SMO	67,34%	1,16

Nesses experimentos, o classificador que obteve a melhor acurácia foi o IBk com 68,34%, inferior à porcentagem de exemplos da classe majoritária, a qual é de 71,08%

(vide Tabela 1). Isso significa que esse método possui desempenho inferior ao classificador trivial que classifica todos os exemplos como pertencentes à classe majoritária.

Uma possível crítica a esses experimentos é que os sinais a serem classificados são muito mais curtos do que as séries temporais utilizadas como exemplos. Uma parte significativa das observações é meramente ruído de fundo, uma característica que pode dificultar, por exemplo, a interpretação de distâncias calculadas pelo classificador IBk. Entretanto, é importante ressaltar que o problema de separar sinal e ruído de fundo não é trivial. Isso ocorre, pois a transição entre ruído de fundo e sinal não é abrupta e a duração dos sinais, como observado anteriormente, é variável. Em casos especiais, alguns sinais tem duração próxima a um segundo, como no caso das abelhas que ocasionalmente pairavam em frente ao laser.

Para eliminar a possibilidade de que os resultados modestos apresentados pelos classificadores tenham como origem a curta duração média dos sinais em comparação com a duração total da série temporal, decidimos extrair séries temporais mais curtas, com duração de 3 décimos de segundo. Novamente, os dados foram transformados em uma tabela atributo-valor. Utilizamos o classificador IBk, dado que esse obteve a melhor acurácia nos experimentos anteriores. A acurácia obtida neste novo experimento foi de 70,49%, ou seja, ainda inferior à acurácia do classificador trivial.

Os resultados iniciais evidenciam que o principal desafio está em identificar atributos que permitam construir classificadores precisos. Nas próximas duas seções são mostrados dois exemplos concretos desses atributos.

3.3. Classificação Utilizando a Frequência de Batida de Asas

A frequência de batida de asas é um importante atributo para classificar insetos, o qual está relacionado com o tamanho dos insetos, sendo que insetos maiores tendem a apresentar frequências menores de batida de asas. No caso de mosquitos, esse atributo pode também ajudar a separar mosquitos machos e fêmeas em espécies dimórficas, sendo que as fêmeas tendem a ser maiores do que os machos e portanto apresentam frequências de batida de asas mais baixas.

A Figura 4-*a* mostra o espectro do sinal da Figura 2. Pode-se observar, além do pico na frequência de batida de asas do mosquito (frequência fundamental), as 3 harmônicas em frequências múltiplas inteiras da frequência fundamental.

Neste trabalho, utilizamos o conceito de *cepstro*, proveniente da áera de processamento de sinais, para identificar a frequência de batida de asas. Informalmente, o cepstro é capaz de identificar frequências harmônicas representá-las em um único valor. A Figura 4-*b* apresenta o cepstro do mesmo sinal. O maior pico está localizado na frequência de 0,001563s, aproximadamente 639,79Hz.

Os dados utilizados são os mesmos cuja coleta foi descrita na Seção 3.1. Foi calculado o cepstro para cada fragmento de um segundo e o valor máximo do cepstro foi anotado como a frequência da batida de asas do inseto. Na Figura 5 é mostrado um histograma criado a partir dos resultados obtidos com as três espécies. É possível perceber que as classes se assemelham a uma distribuição Gaussiana com uma longa cauda esquerda. Também pode-se notar que que as frequências da espécie *Bombus Impatiens* (abelhas) são linearmente separáveis das espécies de mosquitos. Analisando somente os mosquitos, *Aedes*

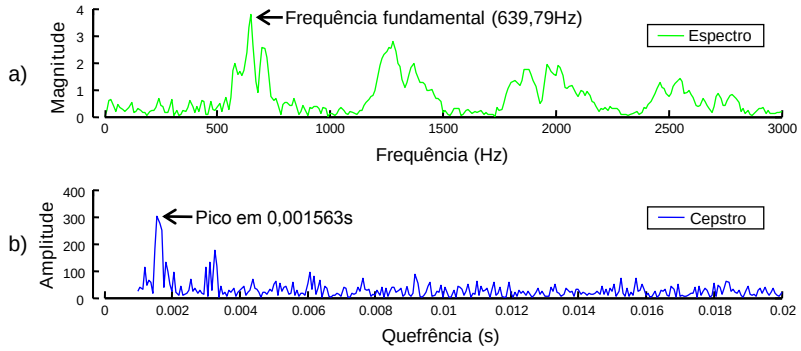


Figura 4. Espectro (a) e cepstro (b) do sinal obtido pelo sensor, que são utilizados para calcular a frequência da batida de asas de um inseto

aegypti possui uma frequência de batida de asas mais alta que o *Culex quinquefasciatus*. Entretanto, existe uma sobreposição nas frequências de batida de asas dessas duas espécies. A Tabela 3 provê algumas estatísticas para as frequências de batida de asas.

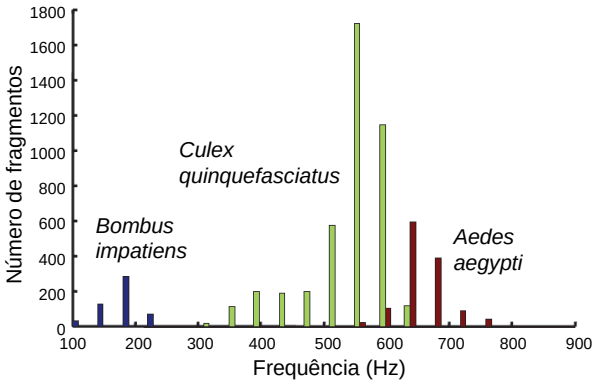


Figura 5. Histograma da frequência da batida de asas das espécies *Bombus impatiens*, *Aedes aegypti* e *Culex quinquefasciatus*

Tabela 3. Características dos dados de batida de asas

Classe	Exemplos	Média (Hz)	Desvio Padrão (Hz)
Aedes aegypti	1231	644,76	34,86
Bombus impatiens	499	173,76	26,42
Culex quinquefasciatus	4252	528,12	62,67

Nós podemos utilizar um classificador Bayesiano combinado com o critério de máximo a posteriori (MAP). A idéia geral do critério MAP é classificar um inseto, com uma dada frequência de batida de asas v, na classe de maior probabilidade a posteriori:

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(c|f = v)$$

no qual, C é o conjunto de todas as classes, f é uma variável aleatória para a frequência de batida de asas e v é o valor medido da frequência de batida de asas para o caso de teste. É possível utilizar a regra de Bayes para calcular a probabilidade a posteriori de cada classe, $P(c|f = v)$:

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} \frac{P(f = v|c)P(c)}{P(f = v)}$$

O termo $P(f = v)$ pode ser descartado, pois ele é uma constante independente de c, resultando em:

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(f = v|c)P(c)$$

Nós vamos assumir que as frequências de batida de asas são normalmente distribuídas para cada espécie. Assim, é possível estimar a quantidade $P(f = v|c)$ dada a equação da curva Gaussiana:

$$P(f = v|c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(v-\mu_c)^2}{2\sigma_c^2}}$$

no qual, μ_c e σ_c^2 são as médias e variâncias das frequências de batida de asas para cada classe c. Esses parâmetros, bem como as probabilidades a priori das classes, $P(c)$, podem ser estimadas a partir dos dados, como mostrado na Tabela 3.

Ao aplicar o classificador Bayesiano MAP aos dados, foi obtida uma acurácia respeitável de 96.04%. Os resultados para cada classe podem são apresentados na Tabela 4.

Tabela 4. Sumário do desempenho do classificador Bayesiano para cada classe

	Predito			Acurária na classe
	Bombus impatiens	*Culex quinquefasciatus*	*Aedes aegypti*	
	Bombus impatiens	499	0	0	100,00%
Real	*Culex quinquefasciatus*	0	4139	113	97,34%
	Aedes Aegypti	0	124	1107	89,92%

Os resultados obtidos com a frequência de batida de asas são bastante promissores. Estamos coletando dados para mais espécies de insetos e ao considerar tais espécies adicionais, novos atributos serão necessários para manter uma alta acurácia. Na próxima seção, exploramos o fato de que insetos possuem períodos distintos de atividade como atributo adicional para o classificador.

3.4. Ritmo Circadiano

Muitas espécies de mosquitos são mais ativas durante determinados períodos do dia. Essa atividade é conhecida como ritmo circadiano, fato conhecido desde a antiguidade e que tem sido estudado por quase cem anos [Roubaud 1918].

Nós utilizamos o sensor para medir o ritmo circadiano de mosquitos, conforme é apresentado na Figura 6 para as espécies *Culex quinquefasciatus* e *Aedes aegypti*. Durante cinco dias, foi medida a atividade dos mosquitos em uma sala com iluminação natural. Para a espécie *Culex quinquefasciatus* foram coletados mais de 200.000 eventos e para a espécie *Aedes aegypti* cerca de 30.000 eventos. Os ritmos circadianos mostram que as duas espécies possuem padrões de atividade bastante distintos. Os mosquitos da espécie *Culex*

quinquefasciatus são predominantemente noturnos com um pico de atividade vespertina; os mosquitos da espécie *Aedes aegypti* são diurnos, também com um pico de atividade verpertina.

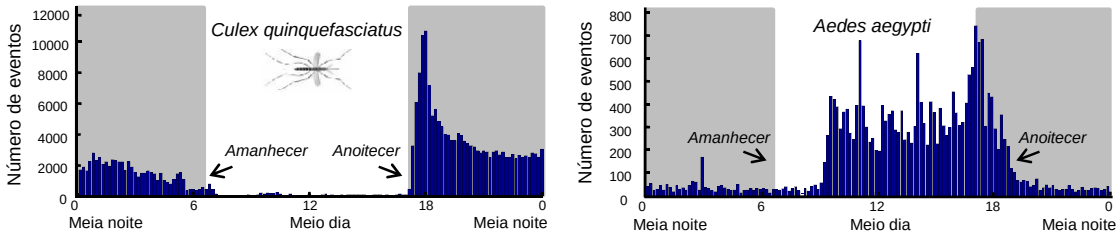


Figura 6. Ritmos cicadianos dos mosquitos das espécies *Culex quiquefasciatus* (*esquerda*) e *Aedes aegypti* (*direita*)

A Figura 7 fornece uma intuição de como é possível adicionar os ritmos circadianos à informação de batida de asas. Suponha um problema de classificar um inseto como *Aedes aegypti*, *Culex quinquefasciatus* ou *Anopheles stephensi* e que foi medida uma frequência de batida de asas a 428Hz. Utilizando somente essa informação, o inseto é igualmente provável de pertencer à classe *Anopheles stephensi* ou *Culex quinquefasciatus*. Entretanto, se o inseto foi observado às 11 horas, podemos verificar que é bem mais provável que seja um *Anopheles stephensi*.

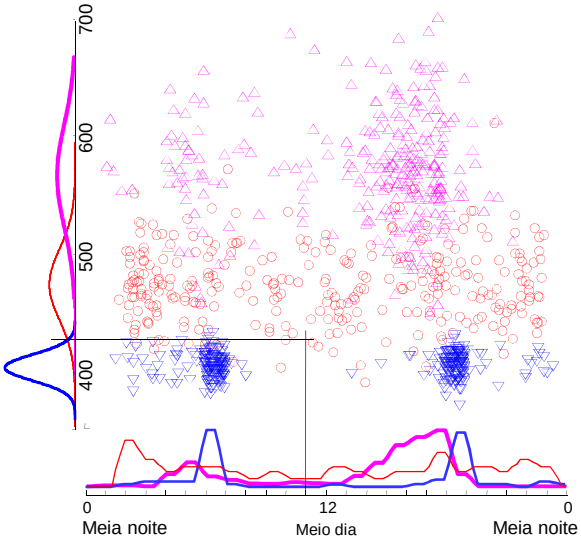


Figura 7. Gráfico de dispersão da frequência de batida de asas vs. horário de observação de 300 exemplos de *Aedes aegypti* △, *Culex quinquefasciatus* ▽ e *Anopheles stephensi* ○. Um inseto observado às 11 horas com frequência de batida de asas de 428Hz é quase certo ser um *Anopheles stephensi*

Atualmente, estamos coletando dados com informação temporal e de frequência de batida de asas para diversas espécies de interesse. Entretanto, podemos fazer um exercício para avaliar como os algoritmos de aprendizado de máquina estendidos com ambos os atributos poderiam beneficiar o aprendizado. Nós utilizamos dados gerados a partir de distribuições reais disponíveis na literatura para frequência de batida de asas [Reed et al. 1942] e ritmos circadianos [Taylor 1998] para as espécies *Aedes aegypti*, *Culex quinquefasciatus* e *Anopheles stephensi* (vide Figura 7).

Os experimentos utilizam 1000 exemplos, dois algoritmos de aprendizado de máquina (Naïve Bayes e k-vizinhos mais próximos) e três combinações de atributos. Na primeira, somente com a frequência de batida de asas foi utilizada no aprendizado, na segunda, apenas o ritmo circadiano e na terceira, ambos os atributos. Em todos os experimentos, foi utilizada a acurácia obtida com *leaving-one-out cross-validation* como principal medida para avaliar os resultados, apresentados na Tabela 5.

Os classificadores aprendidos a partir dos ritmos circadianos apresentam as acurácias mais baixas. Esse é um resultado esperado, uma vez que existe um alto grau de sobreposição nos períodos de atividade das três espécies (vide Figura 7). Os classificadores aprendidos com a frequência de batida de asas obtiveram acurácias consideravelmente mais altas. No experimento final, ambos os classificadores foram beneficiados pelo uso dos dois atributos combinados, com o Naïve Bayes sendo o classificador que obteve a maior melhora de desempenho, 93,18%. Essa é uma acurácia respeitável, considerando que estamos classificando espécies de mosquitos com características físicas e de comportamento similares, as quais até mesmo entomologistas têm dificuldades em distinguir.

Tabela 5. Acurácia para os classificadores k-vizinhos mais próximos e naïve bayes

Atributos	Naïve Bayes	k-NN
Ritmo circadiano	70,69%	68,50%
Frequência de batida de asas	90,73%	91,30%
Atributos combinados	93,18%	91,80%

4. Atributos Complementares

Nesta seção, são discutidos alguns atributos adicionais que podem ser utilizados em classificadores com o intuito de distinguir um número maior de espécies.

4.1. Características Meteorológicas

Temperatura, pressão atmosférica e umidade são as principais variáveis meteorológicas que afetam os insetos, por dois motivos principais. Em primeiro lugar, algumas espécies são mais adaptadas para viver em determinadas condições ambientais. Por exemplo, muitas espécies de mosquitos são originárias de regiões tropicais e subtropicais, onde o clima é quente e úmido. Portanto, em períodos de clima nessas condições, a prevalência de insetos dessas espécies pode aumentar. A segunda razão é que pelo menos a temperatura influencia o metabolismo dos insetos e as propriedades aerodinâmicas do ar. Em particular, esperamos que a frequência de batida de asas aumente proporcionalmente à temperatura. Há pesquisas significativas que sugerem que a temperatura tem efeito linear em uma ampla faixa de valores para a maioria dos insetos [Reed et al. 1942].

4.2. Características Intrínsecas

A frequência não é a única característica que pode ser retirada do sinal coletado pelo sensor. Insetos têm diferenças anatômicas que podem causar pequenas variações na forma das séries temporais coletadas pelo nosso equipamento. Por exemplo, a maioria dos insetos têm dois pares de asas, mas alguns deles têm apenas um par. Ainda, outras espécies têm um segundo par asas muito pequeno, usado principalmente para a estabilização de voo, como é o caso das moscas domésticas. Além disso, existem centenas de recursos de áudio que podem ser extraídos de arquivos de som. Um exemplo é o uso da magnitude das harmônicas de segunda ordem e superior como atributos [Moore and Miller 2002].

4.3. Características Espaço-temporais

Nossos sensores devem ser colocados em algum lugar no espaço e no tempo e esses dados espaço-temporais oferecem atributos potencialmente úteis [Grüebler et al. 2008]. As características espaço-temporais alteram a probabilidade a priori de uma determinada espécie, e essas informações podem ser inseridas no modelo de classificação. As características espaciais incluem altitude, distância a uma fonte de água doce, tipo de solo, velocidade média do vento, densidade populacional humana / insetos, tipo de atividade agrícola local, etc. As características temporais incluem padrões de época do ano e hora do dia (circadiano), que, como discutido anteriormente, serão particularmente interessantes.

5. Conclusões

Neste artigo, apresentamos um novo sensor para contar e classificar insetos à distância utilizando laseres. Mostramos que tal sensor pode facilmente contar insetos e que a tarefa de classificação é factível por meio da extração de atributos a partir dos dados extraídos pelo sensor. Concretamente, demonstramos o uso de dois atributos: a frequência de batida de asas e o ritmo circadiano; e ainda apresentamos outros atributos que podem ser incorporados aos classificadores em pesquisas futuras.

Referências

Dixon, K. W. (2009). Pollination and Restoration. *Science*, 325(5940):571–573.

Grüebler, M. U., Morand, M., and Naef-Daenzer, B. (2008). *Agriculture, Ecosystems and Environment*, 123(1–3):75–80.

Moore, A. and Miller, R. H. (2002). Automated identification of optically-sensed Aphid (Homoptera: Aphidae) wingbeat waveforms. *Annals of the Entomological Society*, 95(1):1–8.

Patnaik, J. L., Juliusson, L., and Vogt, R. L. (2007). Environmental predictors of human west nile virus infections, colorado. *Emerging Infectious Diseases*, 13(11):1788–1790.

Reed, S. C., Williams, C. M., and Chadwick, L. E. (1942). Frequency of wing-beat as a character for separating species races and geographic varieties of drosophila. *Genetics*, 27:349–361.

Roubaud, E. (1918). Rhythmes physiologiques et vol spontan chez l'anopheles maculipennis. *C. R. Hebdomadaires des Seances de l'Academie des Science*, 167:967–969.

Taylor, B. (1998). Biological clocks in mosquitoes. *Website: http://antbase.org/ants/africa/personal/crhtml/covercr.htm*.

Walker, K. (2002). A review of control methods for african malaria vector. Technical Report 108, Bureau for Global Health, Washington, DC, USA.

WHO (2010). The world malaria report. Technical report, World Health Organization.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco, 2nd edition.

Uma Aplicação de PSO na Qualidade da Energia Elétrica

Ricardo de Andrade Lira Rabêlo[1], **Daniel Barbosa**[2],
Ivan Nunes da Silva[1], **Mário Oleskovicz**[1], **Denis Vinicius Coury**[1]

[1] Av. Trabalhador São Carlense, 400, Centro, 13.566-590, São Carlos, SP
Universidade de São Paulo - Escola de Engenharia de São Carlos
Departamento de Engenharia Elétrica

[2]Rua Ponciano de Oliveira, 126, 2º andar, Rio Vermelho, 41.950-275, Salvador, BA
Universidade Salvador
Mestrado em Energia

{ricardor, insilva, olesk, coury}@sc.usp.br, daniel.barbosa@pro.unifacs.br

Abstract. *Electrical Power Systems Quality refers to any problem observed in the voltage, current or frequency. The voltage and current waveforms from an Electrical Power System (EPS) are not considered pure sinusoids due to the presence of, amongst others, the harmonic distortion. This work presents an approach based on the Particle Swarm Optimization (PSO) method for the harmonic component estimation in an EPS. PSO is a technique of search/optimization modeling the social behavior observed in many species of birds, schooling fish and even human social behavior. The technique uses a population of particles to search inside a multidimensional search space. The objective of the PSO is to adjust the speed and position of each particle, seeking for the best solution within the search space. The results demonstrate that the method can precisely identify the harmonic components in the distorted waveforms and it shows considerable advantages if compared to the most common algorithm for this purpose, the Discrete Fourier Transform (DFT).*

Resumo. *Qualidade da energia elétrica refere-se a qualquer problema manifestado na tensão, corrente ou desvio de frequência. As formas de onda das tensões e correntes nos Sistemas Elétricos de Potência (SEPs) não são puramente senoidais, devido à presença de distorções harmônicas. Este trabalho propõe uma abordagem baseada em Otimização por Enxame de Partículas (Particle Swarm Optimization - PSO) para a estimação dos componentes harmônicos em SEPs. PSO é uma técnica de busca/otimização que modela o comportamento social observado em muitas espécies de pássaros, cardumes de peixes, e até mesmo do comportamento social humano. PSO emprega uma população de partículas para realizar buscas dentro de um espaço de busca multidimensional. O objetivo do PSO consiste em ajustar a velocidade e a posição de cada partícula visando alcançar melhores soluções dentro do espaço de busca. Os resultados obtidos mostram que a abordagem proposta pode identificar os componentes harmônicos nas formas de onda distorcidas e apresentam vantagens em relação a algoritmos tradicionalmente empregados, como, por exemplo, a Transformada Discreta de Fourier (TDF).*

1. Introdução

A qualidade da energia elétrica (QEE) refere-se a qualquer distúrbio manifestado na tensão, corrente ou desvio de frequência, que resulta em falha ou operação indevida do equipamento do consumidor [Dugan et al. 2003]. Estes problemas podem ocorrer em várias partes do sistema elétrico de potência (SEP), seja no sistema supridor da concessionária, ou nas instalações dos consumidores finais. Como principais causas dos problemas associados à uma má QEE pode-se citar: saída de unidades geradoras, perda de linha de transmissão, chaveamento de bancos de capacitores e operações de cargas com características não-lineares.

Da prática, tem-se que as formas de onda das tensões e correntes nos sistemas elétricos de potência não são puramente senoidais, em função, principalmente, das distorções harmônicas sobrepostas ao componente fundamental. Recentemente, com a crescente aplicação dos dispositivos não lineares e dos bancos de capacitores em sistemas industriais, as preocupações com o monitoramento e a análise das causas e dos efeitos dessas distorções têm aumentado, fazendo com que os agentes do setor elétrico designem mais recursos para a observação destes fenômenos [Sabin et al. 1999]. Essa conscientização é o resultado do aumento observado dos níveis de distorção harmônica em muitos SEPs e pode ser denotada por meio das novas regulamentações do setor elétrico, como os Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional (PRODIST) [Lamin et al. 2010], que abordam tais distúrbios, entre outros.

As preocupações e precauções no contexto da QEE estão diretamente associadas aos prejuízos financeiros para vários setores da economia, como para o setor industrial, uma vez que estes podem ter seus processos produtivos paralisados ou, mesmo, perder uma linha de produção inteira em decorrência de um evento de QEE [Baggini 2008].

Dentre os diversos distúrbios de QEE existentes, destacam-se as distorções harmônicas, uma vez que estes fenômenos são distorções periódicas de alta frequência presentes nas tensões e nas correntes e podem estar associadas a operação continua de cargas com características não lineares, como os dispositivos que usufruem de certa eletrônica de potência [Macedo et al. 2002]. Estes equipamentos modernos, além de produzirem as harmônicas, são, geralmente, mais sensíveis aos efeitos das variações da qualidade da energia elétrica do que os seus antecessores [Arrilaga et al. 1997].

Embora a avaliação da distorção harmônica seja uma tarefa importante para a correta operação do sistema elétrico, a mensuração destes é difícil e requer a aplicação de métodos rápidos para a medição e a estimação. Tal necessidade reside no fato de que as fontes harmônicas possuem natureza dinâmica e as amplitudes são variantes no tempo [Dugan et al. 2003].

Desta forma, engenheiros e pesquisadores buscam continuamente inovações que melhorem a precisão e a velocidade dos algoritmos aplicados na estimação dos componentes harmônicos. Dentre as diversas técnicas propostas, [Kamwa and Grondin 1992] propuseram um algoritmo que minimiza o erro médio entre os valores estimados e os medidos da função baseado no método dos mínimos quadrados *Least Squared Method*. Os autores em [Chen et al. 2010] reportaram um algoritmo utilizando o filtro de Kalman para estimar o conteúdo harmônico variante no tempo. A TDF (Transformada Discreta de Fourier), técnica baseada na teoria de funções ortogonais, também é comumente uti-

lizada para executar esta função. Todavia, a sua má aplicação pode levar a resultados incorretos, e o seu desempenho pode ser afetado pelo componente de corrente contínua (CC) e pelo ruído presentes no sinal analisado [Girgis et al. 1991]. Para melhorar a estimação em sinais ruidosos, [Lin 2007] apresentou uma técnica imune a este problema baseada em Redes Neurais Artificiais (RNAs). Em 2002, [Macedo et al. 2002] apresentaram a estimação harmônica como sendo um problema de otimização e utilizaram os Algoritmos Genéticos (AGs) para solucioná-lo. Entretanto, a otimização de parâmetros com forte correlação dentro da função objetivo (função objetivo epistática [Fogel 2006]) e a convergência prematura impedem que os AGs alcancem o máximo rendimento em muitas aplicações, reduzindo a eficácia na busca pela solução ótima [Eberhart and Shi 1998, Das and Venayagamoorthy 2006, Gaing 2004].

Assim, para superar algumas das limitações encontradas com o uso de AGs, a técnica de Otimização por Enxame de Partículas (*Particle Swarm Optimization* - PSO) [Eberhart et al. 2001] tem chamado a atenção dos pesquisadores como uma ferramenta robusta para problemas de busca/otimização. Otimização por enxame de partículas constitui um conceito muito simples, e seus paradigmas podem ser implementados em poucas linhas de código computacional [Kennedy and Eberhart 1995]. Além do mais, PSO exige somente operadores matemáticos elementares e é computacionalmente barata em termos de requisito de memória e velocidade [Eberhart and Kennedy 1995]. Diferentemente de outras técnicas heurísticas, PSO tem mecanismos flexíveis e bem balanceados para realçar as habilidades de exploração global e local do espaço de busca [Shi and Eberhart 1998]. Como vantagem adicional em relação aos AGs, um sistema de enxame de partículas tem memória. Mesmo com a possibilidade do uso do elitismo, mudanças na composição genética da população durante a execução de um AG resultam na destruição do conhecimento prévio do problema armazenado nos cromossomos. Na Otimização por Enxame de Partículas, o conhecimento de boas soluções é armazenado pelas partículas do sistema.

Neste contexto, este trabalho propõe uma abordagem baseada em PSO para a estimação dos componentes harmônicos em SEPs, principalmente, no que tange o componente CC e sua constante de decaimento. Os resultados obtidos mostram que o método proposto pode identificar os componentes harmônicos nas formas de onda distorcidas e apresentam vantagens em relação ao algoritmo da TDF.

2. Representação da Modelagem Harmônica

Pela literatura consultada, verifica-se que um sinal periódico e distorcido pode ser representado em função de sua frequência fundamental e de seus componentes harmônicos, sendo estes expressos como a superposição ponderada de senoides complexas conhecida como a série de Fourier. Cada harmônico é múltiplo inteiro da frequência fundamental do sistema e pode ser representado matematicamente pela equação que segue [Madisetti and Williams 1998]:

$$x(t) = x_0 e^{-\lambda t} + \sum_{i=1}^{N} A_{c,i} cos(i\omega_0 t + \theta_{c,i}) + A_{s,i} sen(i\omega_0 t + \theta_{s,i}), \quad (1)$$

na qual x_0 é a componente contínua do sinal; λ é a constante de tempo do sistema, $A_{c,i}$, $A_{s,i}$, $\theta_{c,i}$ e $\theta_{s,i}$ são amplitudes dos cossenos e senos e os ângulos de fase dos i-ésimos harmônicos, respectivamente; ω_0 é a frequência fundamental, em radianos; i é a ordem

do componente harmônico evidenciado e *N* é o número de harmônicas presente no sinal utilizado para representar $x(t)$.

Todavia, embora o sinal seja contínuo no tempo, este deve ser digitalizado para que algoritmos computacionais possam ser empregados. Assim, a representação do sinal amostrado é determinado pela equação 2, sendo $e(t_k)$ o erro associado a cada instante de tempo amostrado (t_k) e m o número total de amostras [Souza 2008].

$$\begin{bmatrix} x(t_1) \\ x(t_2) \\ \vdots \\ x(t_k) \end{bmatrix} = [M] \begin{bmatrix} x_0 \\ A_{c,1} \\ A_{s,1} \\ \vdots \\ A_{c,m} \\ A_{s,m} \end{bmatrix} + \begin{bmatrix} e(t_1) \\ e(t_2) \\ \vdots \\ e(t_k) \end{bmatrix}, \tag{2}$$

sendo a matriz de cossenos e senos (M) dada pela equação 3.

$$[M] = \begin{bmatrix} e^{-\lambda t_1} & cos(\omega_0 t_1 + \theta_{c,1}) & sin(\omega_0 t_1 + \theta_{s,1}) & \cdots & cos(N\omega_0 t_k + \theta_{c,N}) & sin(N\omega_0 t_k + \theta_{s,N}) \\ e^{-\lambda t_2} & cos(\omega_0 t_2 + \theta_{c,1}) & sin(\omega_0 t_2 + \theta_{s,1}) & \cdots & cos(N\omega_0 t_2 + \theta_{c,N}) & sin(N\omega_0 t_2 + \theta_{s,N}) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ e^{-\lambda t_m} & cos(\omega_0 t_m + \theta_{c,1}) & sin(\omega_0 t_m + \theta_{s,1}) & \cdots & cos(N\omega_0 t_m + \theta_{c,N}) & sin(N\omega_0 t_m + \theta_{s,N}) \end{bmatrix} \tag{3}$$

Na representação fornecida pela equação 2, geralmente o número de amostras (m) é superior ao número de parâmetros a ser estimado ($N+1$). Portanto, para resolver o problema, este trabalho aplicou o PSO com o objetivo de minimizar o vetor de erro ($e[]$) e, consequentemente, proporcionar um sinal estimado que seja o mais próximo possível do sinal amostrado. A figura 1 ilustra o procedimento supracitado.

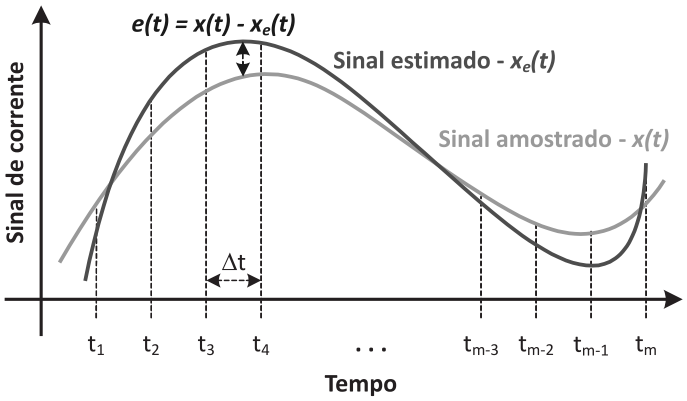


Figura 1. Representação do erro ($e[]$) a ser minimizado.

3. Otimização por Enxame de Partículas

A Otimização por Enxame de Partículas (*Particle Swarm Optimization* - PSO) é uma técnica de busca/otimização que modela o comportamento social observado em muitas espécies de pássaros e cardumes de peixes, e até mesmo do comportamento social humano [Serapião 2009]. O PSO emprega uma população de partículas para realizar buscas

dentro de um espaço de busca multidimensional. Neste espaço de busca, em um determinado intervalo de tempo, cada partícula ocupa uma posição específica e movimenta-se com uma velocidade, a qual é modificada baseada na experiência adquirida pela própria partícula e na experiência das outras partículas do grupo (enxame). A experiência de cada partícula inclui informações relevantes da sua trajetória no espaço de busca, de forma a armazenar a melhor posição já ocupada pela própria partícula. Portanto, a melhor posição já ocupada por cada partícula corresponde à memorização do conhecimento associado às boas soluções obtidas durante o processo iterativo. A posição e a velocidade de cada partícula são ajustadas a cada iteração, de acordo com a melhor posição já ocupada pela própria partícula e com a melhor posição já ocupada por uma partícula dentre todas as partículas do enxame [AlRashidi and El-Hawary 2009]. Desta forma, as partículas tendem a se movimentar no espaço de busca de modo a alcançar regiões cada vez mais promissoras.

Os elementos básicos da técnica PSO são definidos a seguir [Abido 2002]:

- Posição da partícula j ($X_j(t)$);
- População $Pop(t)$;
- Velocidade da partícula j ($V_j(t)$);
- Função de avaliação $f(X(t))$;
- Melhor posição prévia da partícula j até o instante de tempo t ($X^*(t)$);
- Melhor posição prévia do enxame de partículas até o instante de tempo t ($X^{**}(t)$);
- Coeficientes de aceleração (c_1 e c_2);
- Critério de parada para definir quando o processo iterativo de busca/otimização será finalizado;
- Peso de inércia ($w(t)$).

O algoritmo para implementar o PSO segue os passos descritos pela Figura 2.

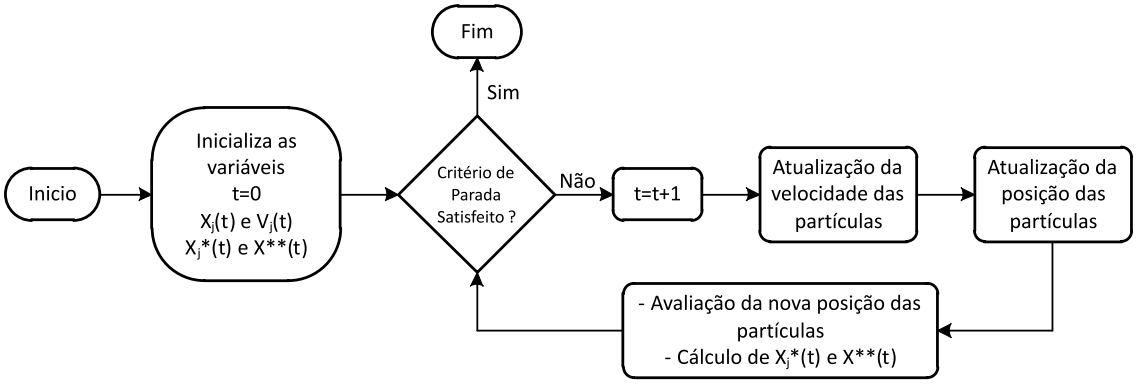


Figura 2. Fluxograma do PSO.

Os passos do algoritmo PSO estão descritos a seguir:

1. Inicialização:
   (a) Inicialize o contador de iterações, $t = 0$. Inicialize aleatoriamente a posição e a velocidade de cada partícula j da população. Atribuem-se os valores iniciais de $X_j^*(t)$ e $X^{**}(t)$;
2. Processo Iterativo:
   (a) Atualize o contador de tempo, $t = t + 1$;

(b) Atualização da velocidade da partícula j na dimensão k de acordo com a equação abaixo:

$$\begin{aligned} v_{j,k}(t) &= w(t) \cdot v_{j,k}(t-1) \\ &+ c_1 \cdot r_1 \cdot (x^*_{j,k}(t-1) - x_{j,k}(t-1)) \\ &+ c_2 \cdot r_2 \cdot (x^{**}_{k}(t-1) - x_{j,k}(t-1)). \end{aligned} \quad (4)$$

Os valores de r_1 e r_2 são números aleatórios uniformemente distribuídos no intervalo [0,1];

(c) Atualização da posição da partícula j na dimensão k de acordo com a seguinte equação:

$$x_{j,k}(t) = v_{j,k}(t) + x_{j,k}(t-1). \quad (5)$$

(d) Cada partícula é avaliada de acordo com a posição atualizada. Se a avaliação da posição atualizada for melhor do que a avaliação da melhor posição anterior, faça $X^*_j(t) = X_j(t)$;

(e) Procure a melhor posição prévia de todo o enxame e faça $X^{**}(t) = X^*_g(t)$ onde g é o índice da partícula cuja avaliação da melhor posição anterior é a melhor de todo o enxame;

(f) Se um dos critérios de parada for satisfeito, então pare, caso contrário volte ao passo 2.

4. Aspectos da Implementação

A abordagem proposta, baseada em otimização por enxame de partículas, foi implementada usando a linguagem de programação C++ [Stroustrup 2000]. O desenvolvimento do software foi executado em um processador Intel Core 2 Duo de 1,83GHz, e memória RAM de 3,00GB, em um sistema operacional *Microsoft Windows* Vista de 32 bits.

4.1. Representação das Partículas

O primeiro passo na utilização do PSO consiste em representar as partículas, isto é, os parâmetros do problema. A representação das partículas é feita especificando cada uma das dimensões do espaço de busca. Neste trabalho, consideram-se como parâmetros a componente contínua do sinal (x_0), a constante de tempo do sistema (λ) e as amplitudes dos senos ($A_{s,i}$) e cossenos ($A_{c,i}$) dos harmônicos. Desta forma, cada partícula representa os parâmetros para a estimação de harmônicas. Ressalta-se que cada um dos parâmetros é armazenado de forma contínua, isto é, por meio de números reais.

4.2. Função de Avaliação

A função de avaliação é responsável por medir a qualidade de cada partícula do enxame como solução potencial do problema. Como o objetivo deste trabalho consiste em proporcionar um sinal estimado que seja o mais próximo possível do sinal amostrado, a função de avaliação deve contemplar a minimização do vetor de erro de estimação ($e[]$) associado à equação 1.

Desta forma, para proporcionar uma melhor estimação dos componentes harmônicos contidos nos sinais, adota-se como objetivo a maximização da função de avaliação

Tabela 1. Parâmetros do PSO proposto.

Parâmetros	
Np	30
t_{max}	15000
c_1	$2,0$
c_2	$2,0$
w_{min}	$0,40$
w_{max}	$0,90$
v^{max}	$0.2 * (x^{max - x^{min}})$

(FA):

$$FA = \frac{1}{\sqrt{\frac{\sum_{i=1}^{p} e_p^2}{p}} + \Delta} \tag{6}$$

na qual Δ é uma constante para evitar um possível *overflow* quando o erro for nulo (Δ = 0,00001) e *p* é o número de amostras do sinal.

4.3. Aspectos Gerais

O algoritmo do PSO tem um grupo de parâmetros que precisa ser estabelecido. Um dos primeiros parâmetros para ajuste é o número de partículas do enxame, ou seja, o tamanho da população. Para esta aplicação, a população (Np) foi definida considerando-se 30 partículas. Verificou-se que 30 partículas garantiam o bom desempenho do algoritmo, bem como um esforço computacional aceitável. O critério de parada adotado consiste em parar a execução do algoritmo caso o número máximo de iterações (t_{max}) seja excedido. O número máximo de gerações (iterações) considerado foi de 15000. Os valores adotados para os coeficientes de aceleração foram 2.0 tanto para o componente cognitivo (c_1) como para o componente social (c_2). Deve ser mencionado que a atribuição dos valores para os coeficientes de aceleração também segue as várias aplicações práticas de PSO. Para o algoritmo implementado, adototou-se 0,4 e 0,9 como limites inferior (w_{min}) e superior (w_{max}), respectivamente para o peso de inércia. Adotou-se o limite máximo da velocidade da partícula (v_k^{max}) correspondente a 20% da amplitude máxima em cada dimensão. A Tabela 1 resume os valores dos parâmetros utilizados na implementação do algoritmo de PSO.

5. Casos de Teste

A validação do método proposto foi realizada com a utilização de formas de onda de tensão e de corrente provenientes de um sistema elétrico de potência, cujos coeficientes da série de *Fourier* são previamente conhecidos, e que podem ser modeladas matematicamente pelas Equações 7 e 8.

$$\begin{aligned} V(t) \;=\; & 0,0550e^{-0.4t} + 0,9829cos(\omega t) + 0,1842sen(\omega t) + \\ & 0,0141cos(2\omega t) + 0,0245sen(2\omega t) + \\ & 0,0077cos(3\omega t) + 0,0197sen(3\omega t) + \\ & 0,0050cos(4\omega t) + 0,0168sen(4\omega t) + \end{aligned}$$

$$0,0039cos(5\omega t) + 0,0154sen(5\omega t) + 0,0033cos(6\omega t) + 0,0161sen(6\omega t) + 0,0033cos(7\omega t) + 0,0230sen(7\omega t) \quad (7)$$

$$\begin{aligned} I(t) = {} & 0,2491e^{-0.4t} + 0,95872cos(\omega t) + 0,2841sen(\omega t) + \\ & 0,0619cos(2\omega t) + 0,1054sen(2\omega t) + \\ & 0,0329cos(3\omega t) + 0,0811sen(3\omega t) + \\ & 0,0206cos(4\omega t) + 0,0643sen(4\omega t) + \\ & 0.0146cos(5\omega t) + 0,0528sen(5\omega t) + \\ & 0,0116cos(6\omega t) + 0,0448sen(6\omega t) + \\ & 0,0052cos(7\omega t) + 0,0401sen(7\omega t) \end{aligned} \quad (8)$$

Os resultados do PSO foram obtidos com a aplicação da função de aptidão em uma janela de dados contendo um ciclo com 64 amostras, e estes são comparados com a tradicional Transformada Discreta de *Fourier* (TDF). É factível salientar que apesar da apresentação de apenas dois casos, diversos testes foram realizados com o intuito de verificar o comportamento e a confiabilidade do algoritmo PSO na estimação de componentes harmônicos.

A Tabela 2 apresenta as estimativas obtidas pela TDF e pelo PSO para a forma de onda da tensão. Todavia, devido a característica heurística e aleatória do método proposto, os testes foram repetidos dez vezes e os resultados encontrados estão sumarizados na tabela mencionada.

Tabela 2. Amplitude dos componentes harmônicos que compõe o sinal de tensão.

		TDF			PSO	
	Referência	Estimação	Erro (%)	Est. Média	$\pm\sigma(10^{-8})$	Erro (%)
V_0	0,0550	0,1102	100,3636	0,0550	0,00063	0,0025
λ_V	0,4000	-	-	0,4008	1,78696	0,2016
$V_{c,1}$	0,9829	0,9832	0,0305	0,9829	0,00036	0,0001
$V_{s,1}$	0,1842	0,1834	0,4343	0,1842	0,00049	0,0032
$V_{c,2}$	0,0141	0,0139	1,4184	0,0141	0,00007	0,0063
$V_{s,2}$	0,0245	0,0245	0,0000	0,0245	0,00016	0,1628
$V_{c,3}$	0,0077	0,0077	0,0000	0,0077	0,00098	0,0076
$V_{s,3}$	0,0197	0,0197	0,0000	0,0197	0,00061	0,0002
$V_{c,4}$	0,0050	0,0050	0,0000	0,0050	0,00133	0,0106
$V_{s,4}$	0,0168	0,0168	0,0000	0,0168	0,00081	0,0002
$V_{c,5}$	0,0039	0,0039	0,0000	0,0039	0,00074	0,0142
$V_{s,5}$	0,0154	0,0154	0,0000	0,0154	0,00023	0,0003
$V_{c,6}$	0,0033	0,0033	0,0000	0,0033	0,00030	0,0200
$V_{s,6}$	0,0161	0,0161	0,0000	0,0161	0,00012	0,0005
$V_{c,7}$	0,0033	0,0034	3,0303	0,0033	0,00024	0,0327
$V_{s,8}$	0,0230	0,0229	0,4348	0,0230	0,00060	0,0011
		Erro Médio:	7,0475		Erro Médio:	0,0290

A Figura 3 apresenta a forma de onda utilizada como referência e as formas de ondas reconstruídas por meio das componentes harmônicas estimadas. Observa-se que

os resultados encontrados pelo PSO produz uma forma de onda que fica sobreposta a referência, elucidando a viabilidade do método proposto. Tal precisão se deve a estimação correta do componente CC, uma vez que a constante de decaimento também é considerada na modelagem.

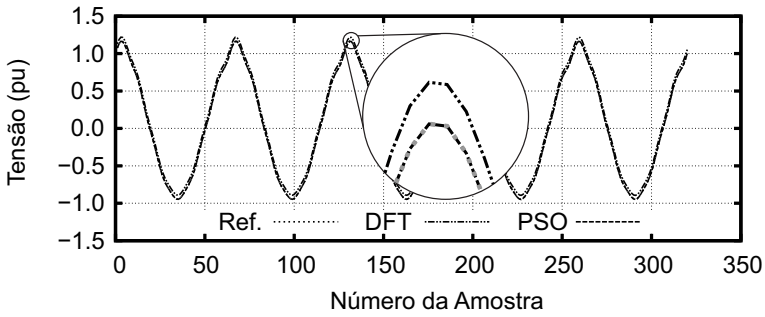


Figura 3. Parâmetros estimados para a tensão.

Os resultados obtidos para a estimação das componentes harmônicas na forma de onda da corrente podem ser verificados na Tabela 3 e na Figura 4. É factível verificar que as formas de onda de referência e do PSO estão sobrepostos, o que confirma o erro inferior a $0,03\%$ encontrado nos testes realizados.

Tabela 3. Amplitude dos componentes harmônicos que compõe o sinal de corrente.

		TDF			PSO	
	Referência	Estimação	Erro (%)	Est. Média	$\pm\sigma(10^{-7})$	Erro (%)
I_0	0,2491	0,4911	97,1497	0,2491	0,00492	0,001
λ_I	0,4000	-	-	0,4003	2,35544	0,065
$I_{c,1}$	0,9587	0,9592	0,0522	0,9587	0,00006	0,002
$I_{s,1}$	0,2841	0,2838	0,1056	0,2841	0,00317	0,002
$I_{c,2}$	0,0619	0,0619	0,0000	0,0619	0,00004	0,003
$I_{s,2}$	0,1054	0,1055	0,0949	0,1054	0,00175	0,001
$I_{c,3}$	0,0329	0,0330	0,3040	0,0329	0,00007	0,005
$I_{s,3}$	0,0811	0,0812	0,1233	0,0811	0,00109	0,001
$I_{c,4}$	0,0206	0,0208	0,9709	0,0206	0,00014	0,008
$I_{s,4}$	0,0643	0,0643	0,0000	0,0643	0,00084	0,001
$I_{c,5}$	0,0146	0,0148	1,3699	0,0146	0,00013	0,012
$I_{s,5}$	0,0528	0,0527	0,1894	0,0528	0,00074	0,002
$I_{c,6}$	0,0116	0,0118	1,7241	0,0116	0,00007	0,016
$I_{s,6}$	0,0448	0,0447	0,2232	0,0448	0,00040	0,004
$I_{c,7}$	0,0052	0,0054	3,8462	0,0052	0,00016	0,342
$I_{s,7}$	0,0401	0,0399	0,4988	0,0401	0,00004	0,000
		Erro Médio (%):	7,1101		Erro Médio (%):	0,0290

A análise dos resultados encontrados permite que seja observada a viabilidade da aplicação do PSO na estimação de componentes harmônicos de potência, uma vez que o desvio padrão nas diversas estimativas é muito pequeno, o que caracteriza uma alta confiabilidade nos resultados obtidos. Este fato pode ser reflexo da consideração do decaimento exponencial na modelagem e a consequente não linearização do componente CC como ocorre pela aplicação da TDF.

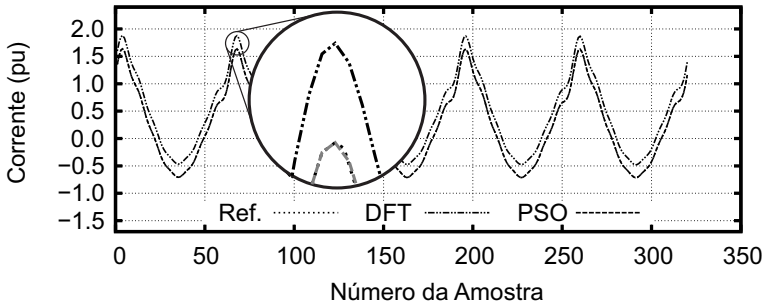


Figura 4. Parâmetros estimados para a tensão.

6. Conclusão

Este trabalho apresentou uma nova técnica baseada em otimização por enxame de partículas para a estimação de componentes harmônicos em sistemas elétricos de potência (SEP). Esse método foi testado com sucesso em formas de onda de corrente e tensão provenientes de simulações computacionais cujos os coeficientes de *Fourier* são conhecidos.

Os resultados apresentados no trabalho foram comparados com o método tradicional de *Fourier* e alguns ganhos foram observados. Entre as vantagens verificadas estão a melhora na obtenção da componente CC e a maior precisão na estimação dos demais componentes. Além disso, o erro médio obtido pela método proposto é praticamente nulo e se manteve constante em todos testes realizados.

A estimação de harmônicos apresentada neste trabalho supôs uma modelagem de um sistema real, o que confere maior precisão nos resultados obtidos. Nesse sentido, é factível observar que a função utilizada para caracterizar as formas de onda estudadas, e aplicada durante o processamento do algoritmo proposto, possui duas variáveis para definir a componente CC, possibilitando um aumento na eficiência do método nesta componente, uma vez que os valores da constante de decaimento (λ) e do valor médio (x_0) são considerados.

Outros testes estão sendo realizados em sistemas mais complexos e os resultados iniciais demonstram que o uso de PSO pode ser uma boa alternativa na estimativa de componentes harmônicos em SEPs. Nestes casos, o objeto de estudo principal é a influência dos parâmetros de configuração do PSO nos resultados encontrados.

Agradecimentos

Os autores gostariam de agradecer aos Laboratórios de Automação Inteligente de Processos e Sistemas (LAIPS) e de Sistemas de Energia Elétrica (LSEE), da Escola de Engenharia de São Carlos (EESC-USP) e a Universidade Salvador pela infra-estrutura proporcionada para a realização desta pesquisa.

Referências

Abido, M. (2002). Optimal design of power-system stabilizers using particle swarm optimization. *IEEE Transactions on Energy Conversion*, 17(3):406–413.

AlRashidi, M. and El-Hawary, M. (2009). A survey of particle swarm optimization applications in electric power systems. *Evolutionary Computation, IEEE Transactions on*, 13(4):913 –918.

Arrilaga, J., Smith, B. C., Watson, N. R., and Wood, A. R. (1997). *Power System Harmonic Analysis*. John Wiley & Sons, New Zealand.

Baggini, A. (2008). *Handbook of Power Quality*. Wiley.

Chen, C. I., Chang, G. W., Hong, R. C., and Li, H. M. (2010). Extended real model of kalman filter for time-varying harmonics estimation. *IEEE Transactions on Power Delivery*, 25(1):17–26.

Das, T. and Venayagamoorthy, G. (2006). Optimal design of power system stabilizers using a small population based PSO. In *IEEE Power Engineering Society General Meeting*, page 7. IEEE.

Dugan, R. C., Santoso, S., McGranaghan, M. F., and Beaty, H. W. (2003). *Electrical Power Systems Quality*. McGraw–Hill, New York, 2 edition.

Eberhart, R. and Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. IEEE.

Eberhart, R. and Shi, Y. (1998). Comparison between genetic algorithms and particle swarm optimization. In *Proceedings of the 7th International Conference on Evolutionary Programming*, pages 611–616. Springer-Verlag.

Eberhart, R., Shi, Y., and Kennedy, J. (2001). *Swarm Intelligence*. Morgan Kaufmann.

Fogel, D. (2006). *Evolutionary Computation: toward a new Philosophy of Machine Intelligence*. Wiley-IEEE Press.

Gaing, Z. (2004). A particle swarm optimization approach for optimum design of PID controller in AVR system. *IEEE Transactions on Energy Conversion*, 19(2):384–391.

Girgis, A., Chang, W., and Makram, E. (1991). A digital recursive measurement scheme for online tracking of power system harmonics. *IEEE Transactions on Power Delivery*, 6(3):1153 –1160.

Kamwa, I. and Grondin, R. (1992). Fast adaptive schemes for tracking voltage phasor and local frequency in power transmission and distribution systems. *IEEE Transactions on Power Delivery*, 7(2):789–795.

Kennedy, J. and Eberhart, R. C. (1995). Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948. Perth, Australia.

Lamin, H., Mattar, C. A. C., and de Sousa, R. E. F. (2010). Procedimentos de distribuição – PRODIST: destaques, inovações e a importância para o setor elétrico brasileiro. In *The International Congress on Electricity Distribution (CIDEL)*, Argentina.

Lin, H. C. (2007). Intelligent neural network-based fast power system harmonic detection. *IEEE Transactions on Industrial Electronics*, 54(1):43–52.

Macedo, R. A., Silva, D. d., Coury, D. V., and Carvalho, A. C. P. L. F. d. (2002). A new technique based on genetic algorithms for tracking of power system harmonics. In *VII Brazilian Symposium on Neural Networks*, pages 7–12.

Madisetti, V. K. and Williams, D. B., editors (1998). *Digital Signal Processing Handbook*. CRC Press. ISBN-13: 978-0849385728.

Sabin, D. D., Brooks, D., and Sundaram, A. (1999). Indices for assessing harmonic distortion from power quality measurements: definitions and benchmark data. *IEEE Transactions on Power Delivery*, 14(2):489 –496.

Serapião, A. B. S. (2009). Fundamentos de otimização por inteligência de enxames: uma visão geral. *Sba: Controle & Automação Sociedade Brasileira de Automatica*, 20:271–304.

Shi, Y. and Eberhart, R. (1998). A modified particle swarm optimizer. In *IEEE World Congress on Computational Intelligence Evolutionary Computation Proceedings*, pages 69–73. IEEE.

Souza, S. A. d. (2008). *Algoritmos genéticos aplicados à proteção e estimação de harmônicos em sistemas elétricos de potência*. PhD thesis, Escola de Engenharia de São Carlos (EESC), São Carlos.

Stroustrup, B. (2000). *A Linguagem de Programação C++*. Bookman.

Uma Abordagem para Seleção de Rotas em Redes de Sensores Sem Fio Utilizando Sistemas Fuzzy Genéticos

Líliam Barroso Leal[1], **Marcus Vinícius de S. Lemos**[1], **Raimir Holanda Filho,**[1]
Fábbio Anderson da Silva Borges[2], **Ricardo de Andrade Lira Rabêlo**[2]

[1] Mestrado em Informática Aplicada - MIA
Universidade de Fortaleza
Fortaleza, Ceará, Brasil

[2]Centro de Ensino Unificado de Teresina - CEUT
Faculdade de Ciências Humanas, Saúde, Exatas e Jurídicas de Teresina
Teresina, Piauí, Brasil

{liliamleal, marvin}@gmail.com, raimir@unifor.br

fabbioanderson@gmail.com, ricardor_usp@yahoo.com.br

Abstract. *Wireless sensor networks (WSNs) are composed of sensor nodes in order to detect and transmit features from the physical environment. Generally, the sensor nodes transmit information to a special node, called sink. The approach proposed by this paper presents the application of Fuzzy Genetic Systems (FGSs) to the selection of routes in WSNs, in order to make the communication between multiple sensor nodes and multiple sink nodes. The Fuzzy Inference System (FIS) of Mamdani is used to estimate the most suitable sink for communication at a given moment, based on some network features such as energy and the number of hops. Genetic Algorithms (GAs) are employed to adjust the design parameters of the FIS. Proposed route selection was applied through of computer simulations, to show the viability of the implemented approach. The results obtained through simulation show a sensors network with longer lifetime, because the adequated choice of the sink used for sending packets through the network in order to find the best routes.*

Resumo. *Redes de Sensores Sem Fio (RSSFs) são compostas por um conjunto de nós sensores com o objetivo de detectar e transmitir alguma característica do meio físico. Estes nós sensores, depois de captar algum evento, devem se comunicar com um nó especial, denominado sink node. A abordagem proposta por este trabalho apresenta a aplicação de Sistemas Fuzzy Genéticos (SFGs) para a seleção de rotas em RSSFs, de modo a realizar a comunicação entre múltiplos nós sensores e múltiplos sink nodes. Um Sistema de Inferência Fuzzy de Mamdani é utilizado para estimar o sink node mais adequado para a comunicação em um determinado instante, baseado em algumas características da rede, como a energia e o número de saltos. Algoritmos Genéticos (AGs) são utilizados para ajustar os parâmetros de projeto do sistema de inferência fuzzy de Mamdani. A abordagem proposta para seleção de rotas foi aplicada, por meio de simulações computacionais, para demonstrar a viabilidade da abordagem implementada. Os resultados obtidos apresentam uma rede de sensores*

com maior tempo de vida, por meio da escolha adequada do sink node utilizado para o envio de pacotes, de forma a encontrar as melhores rotas.

1. Introdução

Nos últimos anos temos presenciado um aumento considerável de pesquisas envolvendo Redes de Sensores Sem Fio (RSSFs), devido a sua aplicabilidade em diversas áreas como, segurança, saúde, agricultura, ambientes inteligentes, automação industrial, dentre outras [Yeh et al. 2009], [Mainwaring et al. 2002], [Xu 2002]. Redes de sensores sem fio possuem características peculiares bem distintas das redes tradicionais, principalmente no que diz respeito a quantidade de nós presentes nas redes, restrições de energia, poder de processamento, memória disponível e largura de banda de comunicação [Hill et al. 2000]. Estas características dificultam a reutilização de vários algoritmos desenvolvidos para sistemas computacionais tradicionais, uma vez que o projeto de qualquer solução para redes de sensores deve considerar de forma relevante restrições como o consumo de energia, memória disponível, capacidade de processamento, dentre outras.

Usualmente em uma RSSF, vários dispositivos sensores são depositados com o objetivo de observar determinados eventos, tais como, pressão atmosférica, temperatura, nível de poluição, etc. Estes nós sensores após captar algum evento devem se comunicar com um nó especial, denominado *sink node*, o qual irá analisar os dados recebidos e, de acordo com o objetivo do projeto da rede, deverá tomar alguma decisão.

Alguns dos paradigmas de comunicação mais comuns nas RSSFs envolvem a comunicação de múltiplos nós sensores depositados em uma área de observação, reportando informações a um *sink* (muitos-para-um). Entretanto, o uso de um único *sink* representa um gargalo em uma rede, especialmente para aplicações de tempo real e fluxo de dados contínuo [Boukerche and Martirosyan 2007]. Nesse sentido, algumas pesquisas têm direcionado esforços para o uso de múltiplos *sinks*, nos quais a comunicação aborda um paradigma alternativo, composto de múltiplos nós sensores e múltiplos *sinks* (muitos-para-muitos).

Este trabalho foca a seleção de rotas para o envio de dados entre os nós sensores e os múltiplos *sink nodes* de uma RSSF. A abordagem combina sistemas de inferência *fuzzy* de Mamdani [Mamdani 1974] e Algoritmos Genéticos (AGs) [Goldberg 1989]. Sistemas de inferência *fuzzy* de Mamdani são utilizados para classificar as rotas de forma a determinar o *sink node* mais adequado. A classificação das rotas, por meio da consideração de algumas características da RSSF, como a energia e o número de saltos, é feita para auxiliar o protocolo de roteamento no processo de seleção do melhor caminho dentre as várias rotas possíveis em um determinado instante de tempo. Algoritmos Genéticos são empregados para obter o ajuste otimizado dos parâmetros de projeto do sistema de inferência *fuzzy* de Mamdani.

No projeto de um sistema de inferência *fuzzy*, constitui-se aspecto relevante a definição da base de conhecimento *fuzzy*, a qual é formada pela base de dados *fuzzy* e pela base de regras *fuzzy* [Cordón et al. 2001, Hoffmann 2001]. A base de dados *fuzzy* é composta principalmente pelas variáveis linguísticas [Zadeh 1973] e seus termos primários (termos linguísticos), os quais são definidos por conjuntos *fuzzy* e suas funções de pertinência [Pedrycz and Gomide 1998]. A base de regras é composta por um conjunto de regras de produção *fuzzy*, que definem a estratégia de tomada de decisão para um determinado do-

mínio de aplicação. Vale ressaltar que alguns estudos têm mostrado que a performance do sistema de inferência *fuzzy* é muito mais sensível à escolha da base de dados *fuzzy* do que à composição da base de regras de produção *fuzzy* [Cordón et al. 2000, Cordón et al. 2001]. Entretanto, a interdependência entre a base de dados *fuzzy* e a base de regras *fuzzy* sugere que um projeto automático simultâneo destes dois componentes consiste em uma metodologia mais apropriada e robusta [Homaifar and McCormick 1995].

O projeto de um sistema de inferência *fuzzy* pode ser visto como um problema de busca/otimização em um espaço de busca de alta dimensionalidade (multidimensional). Cada ponto do espaço de busca representa uma base de conhecimento *fuzzy* particular (base de dados *fuzzy* + base de regras de produção *fuzzy*). Portanto, encontrar o melhor projeto de um sistema de inferência *fuzzy* corresponde a obter um ponto ótimo deste espaço de busca. Entretanto, este espaço de busca é caracterizado como infinitamente grande, não-diferenciável, complexo, ruídoso, multimodal e "enganador" (*deceptive*) [Shi et al. 1999]. Desta maneira, a obtenção de um sistema de inferência *fuzzy* otimizado para uma determinada aplicação de interesse pode ser visto como uma tarefa muito complexa.

Com a aplicação dos AGs, pretende-se obter simultaneamente uma base de dados *fuzzy* e uma base de regras *fuzzy* para maximizar a performance da aplicação do sistema de inferência de Mamdani na classificação de rotas em Redes de Sensores Sem Fio. Portanto, o sistema *fuzzy* obtido deve auxiliar um protocolo de roteamento a escolher o melhor caminho de comunicação entre os nós sensores e os *sink nodes* de forma a maximizar o tempo de vida da rede. Os sistemas inteligentes híbridos obtidos da integração entre modelos de inferência *fuzzy* e Algoritmos Genéticos são denominados Sistemas *Fuzzy* Genéticos (SFGs) [Herrera 2008]. Importante ressaltar que os AGs são algoritmos de otimização global, baseados nos mecanismos de seleção natural e da genética, que têm se mostrado eficientes em uma grande variedade de problemas, pois se sobrepõem a muitas das limitações encontradas nos métodos de busca/otimização tradicionais [Haupt and Haupt 1998].

2. Seleção de Rotas em Redes Multi-Sink

2.1. Aspectos Gerais

Em nossa proposta consideramos que na região observada os nós sensores são posicionados de modo uniforme e os múltiplos *sinks* são dispostos de forma a abranger a cobertura de toda a área. Ressalta-se que os *sink nodes* são dispositivos com características bem superiores aos nós sensores, não possuindo limitações de energia.

Cada *sink node* é responsável por receber mensagens de notificação de eventos ocorridos em qualquer ponto da rede. Para isso, cada nó sensor deve selecionar o *sink node* mais apropriado em um determinado instante, levando-se em consideração características da rede, como por exemplo, energia, número de saltos, colisões e *delay*. Devemos enfatizar que neste trabalho consideramos inicialmente apenas duas características, energia e número de saltos, principais métricas consideradas em RSSFs [García-Hernández et al. 2007].

2.2. Simulador

O simulador Sinalgo [sinalgo 2010] é utilizado nos experimentos para aplicação da abordagem proposta. Sinalgo é um *framework* implementado na linguagem Java. Ao contrário

de outras ferramentas, Sinalgo foca na verificação de algoritmos abstraindo-se das camadas mais baixas (como camada física, camada de enlace, entre outras).

2.3. Protocolo de Roteamento

É importante destacar que é necessária a escolha de um protocolo de roteamento, pois o Sistema *Fuzzy*-Genético funcionará em conjunto com ele. Para a comunicação dos nós com os *sink nodes*, utilizamos o protocolo de roteamento *Directed Diffusion* [Intanagonwiwat et al. 2003], que é projetado para Redes de Sensores Sem Fio, onde o projetista da rede é responsável por definir o tipo de evento que deverá ser observado pelos nós sensores e a área de interesse. Na abordagem proposta, o *Directed Diffusion* seleciona a melhor rota baseado em um índice, o nível *fuzzy* (FL - *Fuzzy Level*) associado a cada rota. O FL é usado para classificar cada rota que um nó possa utilizar. Rotas com maior valor de FL são consideradas preferenciais. O valor do FL é obtido por meio do modelo de inferência *fuzzy* de Mamdani, cujas entradas correspondem aos valores da energia e do número de saltos. Desta forma, o FL auxilia um nó sensor no processo de seleção do melhor caminho para envio de dados dentre as várias rotas possíveis em um determinado instante de tempo, com o objetivo de aumentar o tempo de vida da rede.

3. Aspectos da Implementação do Sistema Fuzzy Genético

O sistema de inferência *fuzzy* implementado tem duas variáveis de entrada, a Energia e o Número de saltos, e uma variável linguística de saída, o nível *fuzzy* (FL) que mensura a qualidade de cada rota. A sintaxe das regras do sistema *fuzzy* está representada pelas seguintes declarações condicionais linguísticas:

- Regra 1: Se (*Energia é* A_1) e (*Número de saltos é* B_1), Então (*FL é* C_1), ou
- ...
- Regra n_r: Se (*Energia é* A_p) e (*Número de saltos é* B_q), Então (*FL é* C_r)

onde: n_r é o número de regras; A_j, B_k e C_l são os termos primários (valores linguísticos) associados às variáveis linguísticas Energia, Número de saltos e FL, respectivamente; p, q e r correspondem à quantidade de termos primários das variáveis linguísticas Energia, Número de saltos e FL, respectivamente.

A abordagem adotada neste trabalho propõe o uso de AGs para determinar, de forma simultânea:

- A base de dados *fuzzy*, através da especificação da quantidade de termos primários para as variáveis linguísticas e do ajuste das funções de pertinência associadas a cada termo primário;
- A base de regras *fuzzy* através da composição do conjunto de regras linguísticas do sistema *fuzzy* de Mamdani.

O primeiro aspecto a ser considerado no uso de AGs é a representação cromossomial do problema, para que os AGs possam solucioná-lo de forma adequada. A representação cromossomial consiste em mapear a informação dos parâmetros do problema em uma maneira viável de ser tratada pelo AG. Ao projetar um sistema *fuzzy* usando um AG, deve-se mapear o sistema *fuzzy* dentro de um cromossomo. Um sistema *fuzzy* é especificado somente quando a base de regras e a base de dados *fuzzy* são especificadas

[Shi et al. 1999]. Neste trabalho os cromossomos contêm informações relativas à quantidade de termos primários, às funções de pertinência dos termos primários das variáveis linguísticas, e à composição linguística da base de regras do sistema de inferência *fuzzy*.

A estrutura dos cromossomos para armazenar os parâmetros do sistema de inferência *fuzzy* foi dividida em três partes. A primeira parte codifica a quantidade de termos primários das variáveis linguísticas. Esta parte do cromossomo contém 9 bits, em que cada conjunto de 3 bits representa a quantidade de termos primários de cada uma das 3 variáveis linguísticas. A segunda parte armazena as informações da base de regras *fuzzy*, isto é, ela codifica os valores lingüísticos do conseqüente das regras de produção *fuzzy*. O número máximo de regras no sistema de inferência *fuzzy* é igual a 25, pois o número máximo de termos primários de cada variável linguística foi definido como sendo igual a 5, e a quantidade de regras é então estabelecida por meio de uma multiplicação entre a quantidade de termos primários das variáveis linguísticas de entrada (Energia e Número de saltos). Portanto, a segunda parte é representada por 75 bits, sendo que cada conjunto de 3 bits representa o termo primário associado ao consequente de uma regra de produção do sistema de inferência. A terceira parte codifica os parâmetros da localização das funções de pertinência de cada variável linguística no universo de discurso. Como as funções de pertinência têm parâmetros contínuos que necessitam de uma grande precisão, adotou-se a representação da terceira parte do cromossomo por meio de números reais [Srikanth and kamala 2008], ao invés da representação binária [Holland 1975] utilizada nas duas primeiras partes. As funções de pertinência utilizadas para as variáveis linguísticas são triangulares (Figura 1).

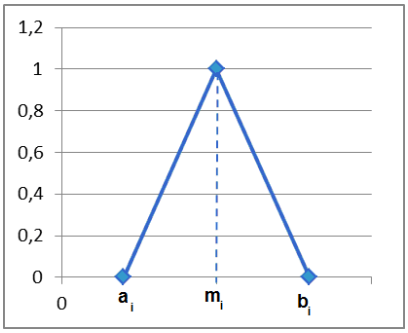


Figura 1. Função de pertinência triangular.

Entretanto, ao invés dos cromossomos armazenarem os valores a_i, m_i e b_i para cada função de pertinência, eles armazenam os coeficientes de ajustes (parâmetros) δ_i e η_i. Cada função de pertinência i pode ser montada a partir das seguintes equações [Park et al. 1994]:

- $a_i = (a_i + \delta_i) - \eta_i$;
- $m_i = (m_i + \delta_i)$;
- $b_i = (b_i + \delta_i) + \eta_i$.

O coeficiente δ_i é responsável por deslocar a função de pertinência para a direita ou esquerda, enquanto o coeficiente η_i pode expandir ou encolher o suporte da função da pertinência em relação ao seu universo de discurso. Como cada variável linguística possui no máximo 5 termos primários, e cada função de pertinência associada a um termo

primário é representada pelos dois parâmetros δ_i e η_i, são necessários no máximo 10 genes para representar as funções de pertinência de uma variável linguística do sistema de inferência *fuzzy* aplicado. Portanto, a terceira parte do cromossomo tem tamanho igual a 30. Como cada cromossomo armazena as informações sobre os parâmetros do sistema de inferência *fuzzy*, tem-se um cromossomo de 114 (9 + 75 + 30) genes.

Após a definição da representação cromossomial, o projeto do AG enfoca a especificação de uma função de avaliação. Como o objetivo do sistema *fuzzy* é de ajudar o protocolo de roteamento a escolher o melhor caminho de comunicação entre os nós sensores e os *sink nodes*, este trabalho empregou como função de avaliação o tempo que leva para o primeiro nó sensor da rede morrer. Portanto, tem-se um problema de maximização, cujo objetivo está em encontrar uma base de conhecimento *fuzzy* para o sistema de inferência que ajude o protocolo de roteamento a selecionar as melhores rotas para maximizar o tempo de vida dos nós sensores e da rede.

Após o cálculo da função de avaliação para todos os indíviduos, o processo de seleção escolhe um subconjunto de indivíduos da população atual, para compor uma população intermediária, de forma a aplicar os operadores genéticos. O método de seleção adotado neste trabalho foi o método do torneio. Vale mencionar que o tamanho do torneio adotado foi igual a 2. Em combinação com o modulo de seleção, foi usada uma estratégia elitista, com a manutenção do melhor indivíduo de uma geração para outra.

Para fazer com que a população passe por uma evolução, deve-se aplicar os operadores genéticos. Os operadores genéticos, cruzamento e mutação, são utilizados para transformar a população, de forma a estender a busca/otimização a um resultado satisfatório. O cruzamento é o operador responsável pela recombinação genética dos pais, para permitir que a próxima geração herde as características genéticas da população atual. Neste trabalho, empregou-se o cruzamento discreto [Herrera et al. 2005]. Este operador engloba os principais operadores de cruzamento para a representação binária, os quais são diretamente aplicáveis para a representação real [Herrera et al. 2003]. Portanto, tem-se um operador de cruzamento compatível com a representação cromossomial adotada. O operador genético da mutação [Hinterding et al. 1995] é necessário para introduzir e manter a diversidade genética da população, por meio da mudança aleatória de genes dentro dos cromossomos, o que fornece meios para incorporação de novas características genéticas dentro da população. Portanto, a mutação assegura a possibilidade de se chegar a qualquer ponto do espaço de busca, além de ajudar a contornar o problema de mínimos locais. Entretanto, a mutação é aplicada de forma menos frequente que o cruzamento, com o intuito de preservar o relacionamento exploração-aproveitamento [Herrera et al. 1998]. Neste trabalho, empregou-se a mutação aleatória [Michalewicz 2011].

O sistema de inferência *fuzzy* de Mamdani obtido por meio da aplicação de AGs é composto por n_r regras disjuntivas de inferência. A inferência de cada regra consiste na avaliação do antecedente, seguida da aplicação do operador de implicação para determinar o conjunto *fuzzy* do consequente. A agregação dos consequentes consiste em agregar, ou combinar, os conjuntos *fuzzy* dos consequentes obtidos pela inferência de cada regra. O procedimento de defuzzificação obtém o valor numérico do FL, ou seja, um índice quantitativo que indica a qualidade da rota. Desta forma, o sistema de inferência *fuzzy* é responsável por estimar a qualidade das rotas baseado em dois critérios: energia e número de saltos. A estimação da qualidade das rotas, por meio do FL, auxilia o protocolo de rote-

Parâmetros		
Sistema Fuzzy		
	Funções de Pertinência	Triangular
	Operador de Implicação	Mínimo de Mamdani
	Operador de Agregação	Máximo
	Método de Defuzzificação	Centro de Área
Algoritmo Genético		
	Tamanho da População	80
	Representação	Mista (Binária e Real)
	Método de Seleção	Torneio (k=2)
	Operador de Crossover	Discreto
	Probabilidade de Crossover	100%
	Operador de Mutação	Aleatório
	Probabilidade de Mutação	5%
	Número Máximo de Gerações	100

Tabela 1. Valores para os principais parâmetros do Sistema *Fuzzy*-Genético implementado.

amento na tomada de decisão para selecionar o melhor caminho dentre os vários possíveis para o envio de dados em uma RSSF. Importante mencionar que rotas com maior valor de FL são consideradas preferenciais.

A Tabela 1 resume os valores dos parâmetros utilizados na implementação do Sistema *Fuzzy*-Genético:

4. Aplicações e Discussões

Os resultados da aplicação da abordagem proposta baseada em Sistemas *Fuzzy*-Genéticos são apresentados baseados em duas métricas. As métricas são:

- Tempo de morte do primeiro nó: expressa o tempo de morte do primeiro nó na rede. Esta métrica permite analisar por quanto tempo todos os nós sensores permanecem vivos;
- Tempo de vida da rede: registra por quanto tempo a rede permanece viva, isto é, até quando a rede poderá manter ativas as comunicações necessárias, mesmo com a morte de alguns nós sensores;

Descrevemos a seguir os cenários simulados:

- Cenário 1: o simulador utiliza o protocolo de roteamento *Directed Diffusion* em sua forma tradicional, sem uso de sistemas de inferência *fuzzy*. Paralelamente, a simulação aplica

balanceamento de carga de forma a distribuir uniformemente a carga de trabalho (os pacotes transmitidos) entre as várias rotas existentes. Adicionalmente, para este cenário, considerou-se um rodízio de rotas de forma que todas as rotas sejam utilizadas. Com isso, pretende-se auxiliar o protocolo *Directed Diffusion* para minimizar o consumo de energia das rotas, uma vez que evita-se o consumo exaustivo de uma determinada rota em detrimento das outras;

- Cenário 2: o simulador aplica sistema de inferência *fuzzy* de Mamdani para estimar a qualidade das rotas baseado na energia e no número de saltos. Contudo, a concepção do sistema de inferência *fuzzy* é feita de maneira empírica, em que a definição da base de conhecimento *fuzzy* é feita baseada em vários testes com configurações diferentes (tentativa e erro);

- Cenário 3: como no cenário 2, o simulador considera a aplicação de um sistema de inferência *fuzzy* de Mamdani. Entretanto, neste cenário, a base de conhecimento do sistema de inferência *fuzzy* é ajustada automaticamente por meio de Algoritmos Genéticos.

4.1. Características da Rede

As principais características da rede são:

1. Topologia: a rede simulada é fixa e composta por apenas dois tipos de nós: *sink nodes* e nós sensores. Os nós sensores possuem características similares, caracterizando uma rede plana, na qual cada nó da rede possui um identificador único e um alcance de rádio fixo;

2. Quantidade de nós: a rede de sensores é composta por 100 nós sensores distribuídos uniformemente. O número de *sink nodes* variou de 1 a 4, de forma a ilustrar a influência do número de *sink nodes* no tempo de vida dos nós sensores e da rede.

4.2. Resultados

Apesar da abordagem proposta ser direcionada a ambientes com múltiplos *sink nodes*, inicialmente apresenta-se o comportamento da RSSF por meio das simulações computacionais com os 3 cenários descritos considerando-se apenas um único *sink node*. Estes experimentos foram realizados com o proposito de verificar a viabilidade e eficiência da aplicação de um Sistema *Fuzzy*-Genético em um ambiente com um único *sink* (mono-sink). O comportamento das RSSF também foi avaliado em cada cenário para verificar a qualidade da classificação das rotas em ambientes multi-*sink*. Adicionalmente, as simulações com ambientes mono-*sink* e multi-*sink* permitem verificar a relevância da quantidade de *sink nodes* no ambiente monitorado e a influência deles no tempo de vida da rede.

Para a primeira métrica de avaliação, tempo de morte do primeiro nó, a Tabela 2 e a Figura 2 ilustram os resultados obtidos para todos os 3 cenários simulados. Pode-se perceber que o tempo de morte do primeiro nó apresenta um comportamento crescente entre os cenários, alcançando o maior valor para o cenário 3. Isso implica que a utilização de um Sistema *Fuzzy*-Genético maximiza o tempo de vida dos nós sensores presentes na RSSF, quando comparado aos outros cenários. Pode-se observar também que maiores tempo de vida são obtidos quando mais *sink nodes* são inseridos na RSSF. Esse fato se deve ao aumento do número de rotas possíveis que cada nó sensor pode utilizar para a transmissão de dados pela rede.

Para a outra métrica de avaliação, tempo de morte da rede, a Tabela 3 e a Figura 3 ilustram os resultados obtidos para todos os 3 cenários simulados. Esta métrica refere-se ao tempo em que

Tempo de Morte do Primeiro Nó [segundos]			
Número de *Sink Nodes*	Cenário 1	Cenário 2	Cenário 3
1	14144	18254	19468
2	18334	20674	26656
3	19114	22388	27926
4	20441	23811	28329

Tabela 2. Tempo de Morte do Primeiro Nó da RSSF.

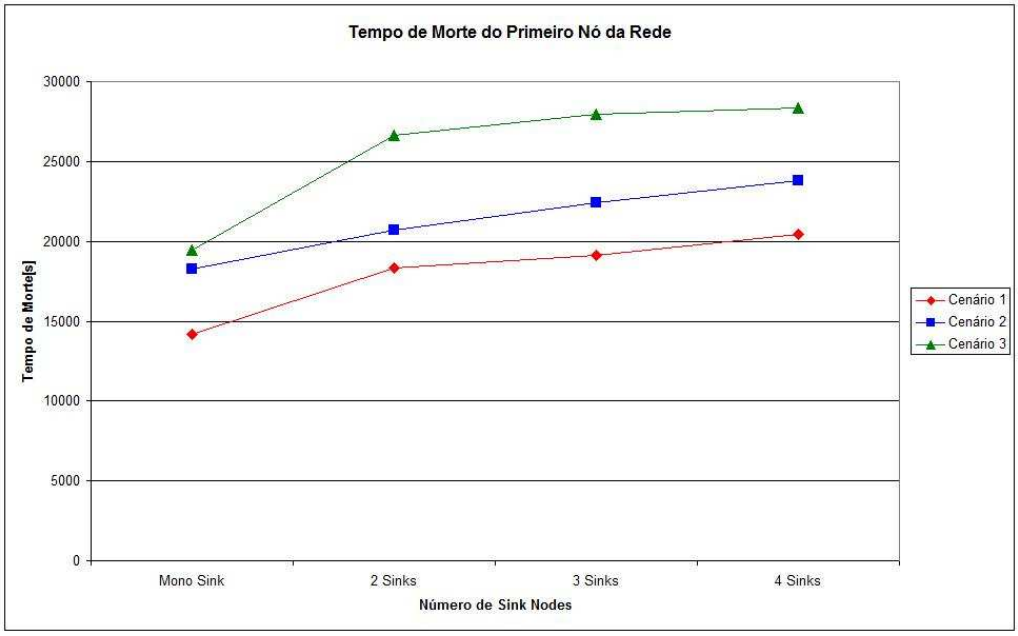


Figura 2. Tempo de Morte do Primeiro Nó da Rede.

a comunicação da rede é interrompida devido a morte de alguns nós sensores que compõem as rotas do transmissor.

Novamente, pode-se observar um melhor comportamento para o cenário 3. Esse fato representa que o uso de um Sistema *Fuzzy*-Genético para auxiliar o protocolo de roteamento foi responsável por maximizar o tempo de vida da rede. Entretanto, deve-se notar que para um número de *sink nodes* igual a 3 e a 4, o cenário 2 teve um comportamento pior quando comparado ao cenário 1. Isso é resultado de um ajuste "pobre" na base de conhecimento *fuzzy* do sistema de inferência de Mamdani. Portanto, esse resultado valida o uso de um AG para otimizar os parâmetros do Sistema *Fuzzy*. Da mesma forma que para a primeira métrica, o tempo de morte da rede cresce com o número de *sink nodes*.

5. Conclusões

Este trabalho propõe uma abordagem baseada em Sistemas *Fuzzy*-Genéticos aplicada à Rede de Sensores Sem Fio com múltiplos *sink nodes*. O Sistema *Fuzzy*-Genético trabalha em conjunto com o protocolo de roteamento *Directed Diffusion* de forma a auxiliar um nó sensor no processo de seleção do melhor caminho dentre as várias rotas possíveis em um determinado instante. Baseado na energia e no número de saltos, um sistema de inferência *fuzzy* de Mamdani é usado para estimar a qualidade das rotas. Com isso, o protocolo de roteamento *Directed Diffusion* pode determinar o

Tempo de Vida da Rede [segundos]			
Número de *Sink Nodes*	Cenário 1	Cenário 2	Cenário 3
1	21518	23314	31586
2	31136	32816	43284
3	37918	34992	45991
4	38291	35200	46871

Tabela 3. Tempo de Vida da Rede.

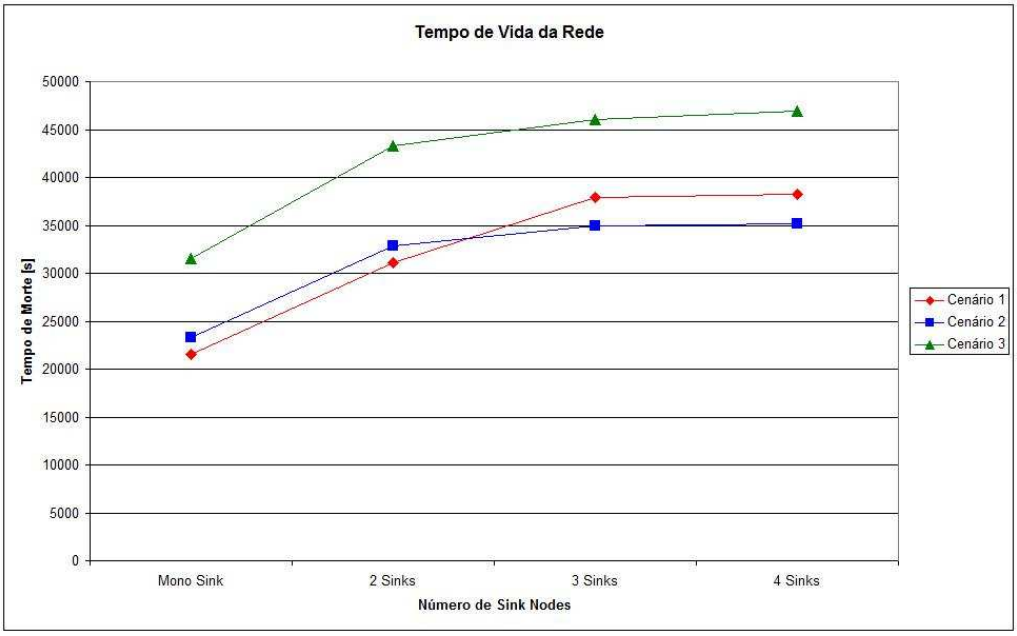


Figura 3. Tempo de Vida da Rede.

sink node mais adequado para o envio de dados com o objetivo de aumentar o tempo de vida da rede.

O projeto de um sistema de inferência *fuzzy* pode ser caracterizado como um problema de busca/otimização em um espaço de busca de alta dimensionalidade (multidimensional), muito grande, não diferenciável, complexo, ruidoso, multimodal e "enganoso" (*deceptive*). Estas características motivaram os autores a aplicar AGs para o projeto (ajuste) do sistema de inferência *fuzzy* de Mamdani, visto que AGs se sobrepõem a muitas das limitações encontradas nos métodos de busca/otimização tradicionais.

A estimação da qualidade das várias rotas da RSSF através de Sistema *Fuzzy*-Genético foi simulada computacionalmente para demonstrar a viabilidade da abordagem implementada. O simulador Sinalgo foi utilizado para realizar as simulações computacionais para todos os 3 cenários considerando as mesmas características da rede. Os resultados obtidos pelo Cenário 3 (uso de Sistema *Fuzzy*-Genético) apresentam uma rede de sensores com maior tempo de vida. Assim por meio do uso de AGs foi possível derivar uma base de conhecimento *fuzzy* (base de dados *fuzzy* + base de conhecimento *fuzzy*) que auxilia o protocolo de roteamento a definir as melhores rotas possíveis para o envio de dados baseado apenas na energia e no número de saltos. Com a seleção das melhores rotas, a rede terá um maior tempo de vida e poderá manter ativas as comunicações necessárias por muito mais tempo.

Como trabalhos futuros, pretendemos analisar o comportamento da seleção de rotas em ambientes mais complexos, bem como incorporar novas variáveis linguísticas para a inferência *fuzzy*. Adicionalmente, pretendemos aplicar otimização por enxame de partículas (PSO - *Particle Swarm Optimization*) em substituição ao Algoritmo Genético para o ajuste da base de conhecimento *fuzzy*.

Referências

Boukerche, A. and Martirosyan, A. (2007). An energy efficient and low latency multiple events' propagation protocol for wireless sensor networks with multiple sinks. In *PE-WASUN '07: Proceedings of the 4th ACM workshop on Performance evaluation of wireless ad hoc, sensor,and ubiquitous networks*, pages 82–86, New York, NY, USA. ACM.

Cordón, O., Herrera, F., and Villar, P. (2000). Analysis and Guidelines to Obtain a Good Uniform Fuzzy Partition Granularity for Fuzzy Rule-Based Systems Using Simulated Annealing*. *International Journal of Approximate Reasoning*, 25(3):187–215.

Cordón, O., Herrera, F., and Villar, P. (2001). Generating the Knowledge Base of a Fuzzy Rule-Based System by the Genetic Learning of the Data Base. *IEEE Transactions on Fuzzy Systems*, 9(4):667–674.

García-Hernández, C. F., Ibargüengoytia-González, P. H., García-Hernández, J., and Pérez-Díaz, J. A. (2007). Wireless sensor networks and applications : a survey. *Journal of Computer Science*, 7(3):264–273.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.

Haupt, R. L. and Haupt, S. E. (1998). *Practical Genetic Algorithms*. Wiley New York.

Herrera, F. (2008). Genetic Fuzzy Systems: Taxonomy, Current Research Trends and Prospects. *Evolutionary Intelligence*, 1(1):27–46.

Herrera, F., Lozano, M., and Snchez, A. M. (2003). A taxonomy for the crossover operator for real-coded genetic algorithms: An experimental study. *International Journal of Intelligent Systems*, 18(3):309–338.

Herrera, F., Lozano, M., and Snchez, A. M. (2005). Hybrid crossover operators for real-coded genetic algorithms: an experimental study. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 9(4):280 – 298.

Herrera, F., Lozano, M., and Verdegay, J. (1998). Tackling Real-Coded Genetic Algorithms: Operators and Tools for Behavioural Analysis. *Artificial Intelligence Review*, 12(4):265–319.

Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., and Pister, K. (2000). System architecture directions for networked sensors. *SIGPLAN Not.*, 35(11):93–104.

Hinterding, R., Gielewski, H., and Peachey, T. (1995). The Nature of Mutation in Genetic Algorithms. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 65–72. Citeseer.

Hoffmann, F. (2001). Evolutionary Algorithms for Fuzzy Control System Design. *Proceedings of the IEEE*, 89(9):1318–1333.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. Ann Arbor: The University of Michigan Press.

Homaifar, A. and McCormick, E. (1995). Simultaneous Design of Membership Functions and Rule Sets for Fuzzy Controllers using Genetic Algorithms. *IEEE Transactions on Fuzzy Systems*, 3(2):129–139.

Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., and Silva, F. (2003). Directed diffusion for wireless sensor networking. volume 11, pages 2–16, Piscataway, NJ, USA. IEEE Press.

Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., and Anderson, J. (2002). Wireless sensor networks for habitat monitoring. In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97, New York, NY, USA. ACM.

Mamdani, E. H. (1974). Application of Fuzzy Algorithms for Control of Simple Dynamic Plant. *Proceedings of IEE Control and Science*, 121(12):1585–1588.

Michalewicz, Z. (2011). *Genetic Algorithms + Data Structures = Evolution Programs*. Springer.

Park, D., Kandel, A., and Langholz, G. (1994). Genetic-based New Fuzzy Reasoning Models with Application to Fuzzy Control. *IEEE Transactions on Systems, Man and Cybernetics*, 24(1):39–47.

Pedrycz, W. and Gomide, F. (1998). *An Introduction to Fuzzy Sets*. MIT Press Cambridge, MA.

Shi, Y., Eberhart, R., and Chen, Y. (1999). Implementation of Evolutionary Fuzzy Systems. *IEEE Transactions on Fuzzy Systems*, 7(2):109–119.

sinalgo (2010). http://disco.ethz.ch/projects/sinalgo/index.html.

Srikanth, T. and kamala, V. (2008). A real coded genetic algorithm for optimization of cutting parameters in turning. *IJCSNS International Journal of Computer Science and Network Security*, 8(6):189 – 193.

Xu, N. (2002). A survey of sensor network applications. *IEEE Communications Magazine*, 40.

Yeh, L., Wang, Y., and Tseng, Y. (2009). ipower: an energy conservation system for intelligent buildings by wireless sensor networks. *International Journal of Sensor Networks*, 5:1–10.

Zadeh, L. (1973). Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. *IEEE Transactions on Systems, Man, and Cybernetics*, 3(1):28–44.

Desvio de Obstáculos Utilizando um Método Estéreo Semi-global

Caio César Teodoro Mendes[1], **Denis Fernando Wolf**[1]

[1] Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo (USP) – São Carlos, SP – Brasil

{caiom,denis}@icmc.usp.br

Abstract. *This paper presents an obstacle avoidance system using a stereo camera as its primary sensor and a modified version of the VFH algorithm. In order to get a high accuracy disparity map and yet meet the real-time needs of the navigation system this work relies on a semi-global stereo method. We show the usability and advantages of the method by realizing experiments using our platform (an electric vehicle). The results show a navigation where the vehicle maintain a safe distance from the obstacles while moving.*

Resumo. *Este artigo apresenta um sistema de desvio de obstáculos utilizando uma câmera estéreo como sensor primário e uma versão modificada do algoritmo VFH. Para garantir a precisão do mapa de disparidades e atender as restrições de desempenho este trabalho faz uso de um método estéreo semi-global. Nós mostramos a usabilidade e vantagens do método utilizado através de experimentos utilizando nossa plataforma (um veículo elétrico). Como resultado apresentamos uma navegação onde o veículo mantém uma distância segura dos obstáculos enquanto se movimenta.*

1. Introdução

A robótica móvel é uma área relativamente nova de pesquisa que lida com o controle de veículos autônomos ou semi-autônomos. Sendo a mobilidade o que distingue a robótica móvel da robótica industrial convencional, seu principal foco de estudo é a navegação.

Grande parte da pesquisa realizada nesta área leva em consideração apenas ambientes estruturados e internos onde o problema da navegação pode ser simplificado de várias formas. A navegação em ambientes externos consiste em um problema mais complexo por contar com poucos ou nenhum padrão estrutural. A irregularidade do terreno e a dinâmica do ambiente são alguns dos fatores que dificultam a navegação de um robô, além de tornar a leitura dos sensores imprecisa. Uma aplicação direta dessa tecnologia é o desenvolvimento de um veículo autônomo inteligente. Este tipo de trabalho contribui com uma nova tecnologia que poderá auxiliar na redução do número de acidentes no trânsito, na diminuição de engarrafamentos nas grandes cidades e na melhoria da mobilidade de deficientes físicos e idosos.

Um exemplo dos recursos humanos e tecnológicos empregados na área é a competição *DARPA Grand Challenge* realizado pela *Defense Advanced Research Projects Agency* (DARPA). Em 2005 o prêmio oferecido pelo projeto foi de 2 milhões de dólares para o veículo que conseguisse atravessar 212 km de terreno acidentado [DARPA 2005].

Em 2007 foi realizado pela DARPA o *Urban Challenge* com o mesmo prêmio, onde o desafio foi o de navegar 96 km em área urbana autonomamente [DARPA 2007].

Lasers, sonares e câmeras são alguns dos sensores mais comumente utilizados nessa tarefa. Os lasers são os principais sensores na maioria dos robôs porque eles provêm informações de profundidade confiáveis e de alta qualidade. Câmeras também são amplamente utilizadas, mas em quase todos os sistemas de navegação práticos o laser é o sensor principal [Thrun et al. 2006]. Motivos para isso são a grande quantidade de ruído presente nas imagens e sua dependência em relação a questões tal como iluminação. Apesar disso, com o preço de um laser comum é possível comprar diversas câmeras, uma câmera estéreo pode prover informações de profundidade para duas dimensões além das imagens associadas, enquanto que o laser comum nesse tipo de aplicação fornece apenas informações de profundidade em um eixo. O principal desafio na utilização de uma câmera estéreo esta em como estimar a profundidade da cena com precisão.

Este trabalho utiliza uma câmera estéreo como sensor principal. Uma câmera estéreo assemelha-se ao olho humano, possuindo duas lentes deslocadas horizontalmente. A grande utilidade de múltiplas lentes em posições diferentes é a possibilidade de relacionar as imagens geradas e assim obter informações sobre a profundidade da cena. Tais informações irão guiar a navegação do veículo servindo de base para detecção de obstáculos.

Existem diversos métodos para se extrair informações de profundidade a partir de uma câmera estéreo (métodos estéreos) e seus resultados variam quanto a qualidade e custo computacional envolvido. Uma avaliação envolvendo vários desses métodos é apresentada em [Scharstein and Szeliski 2002]. A grande maioria dos sistemas de navegação com câmera estéreo utilizam métodos estéreos de correspondência locais providos pela própria empresa que comercializa a câmera [Murarka and Kuipers 2009] [Erkan et al. 2007] [Sermanet et al. 2008], sem uma avaliação de outros métodos e trabalhando com código fechado.

Por conta das limitações de métodos locais este trabalho tem como objetivo utilizar e avaliar um método estéreo semi-global de código aberto para fins de navegação e utiliza-lo juntamente com uma versão modificada do algoritmo de desvio de obstáculos *vector field histogram* (VFH) [Borenstein and Koren 1991].

2. Trabalhos relacionados

Vários trabalhos utilizam uma câmera estéreo para detecção de obstáculos e caminhos navegáveis, sua utilização está intimamente ligada com o método estéreo utilizado, podendo este, computar um mapa de disparidades esparso ou denso e com maior ou menor precisão. De fato, por conta de seu alto custo computacional, apenas por volta do ano 1997 métodos estéreos densos começaram a ser utilizados em navegação robótica e ainda assim limitados a uma velocidade de 2 Hz e a uma resolução de 128 x 120 [Murray and Jennings 1997]. Hoje a grande maioria dos trabalhos envolvendo câmera estéreo utiliza o mesmo método estéreo utilizado em [Murray and Jennings 1997]: o método chamado *block-matching*. Algumas das razões para isso são sua simplicidade e relativo baixo custo computacional.

Grande parte das técnicas de detecção de obstáculos se baseiam na procura de descontinuidades e elevações no mapa de disparidades [Rankin et al. 2005], além disso é

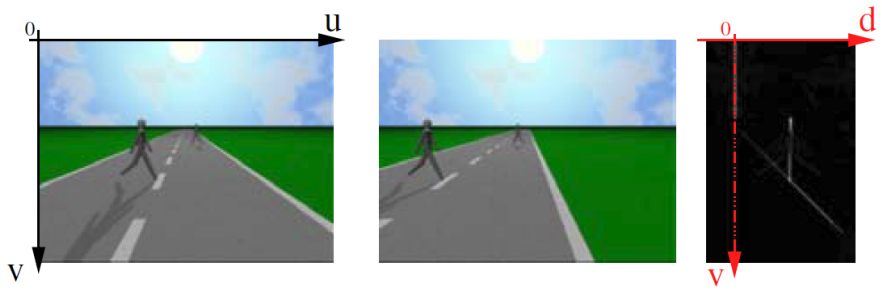


Figura 1. Par de imagens estéreo e o respectivo mapa de disparidades em ”V”[Broggi et al. 2005].

comum a utilização das intensidades naturalmente associadas às informações de profundidade. Alguns trabalhos tem como a primeira tarefa a estimação do plano em que o veículo esta situado dentro do mapa de disparidades, para isso a técnica de gerar um mapa de disparidade em ”V”é amplamente utilizada [Labayrade et al. 2002] [Broggi et al. 2005] [Caraffi et al. 2007]. A criação deste mapa é feita transformando o mapa de disparidade original de forma com que as colunas representem os valores de disparidade em ordem crescente, as linhas sejam mantidas na mesma ordem e a intensidade pode ser ajustada para representar a quantidade de disparidades iguais. Visualizando a Figura 1 é notável a facilidade com que o plano principal pode ser estimado após esta transformação e com a utilização de heurísticas também é possível estimar a localização de obstáculos, como foi realizado no trabalho [Lima and G.A.S. 2010].

Outros trabalhos [Konolige et al. 2008] [Happold and Ollis 2006] utilizaram o paradigma *RANdom SAmple Consensus* (RANSAC) [Fischler and Bolles 1981] para estimar os parâmetros do plano geométrico em que o veículo se situa, com isso é possível calcular a distância de qualquer ponto do mapa de disparidades em relação ao plano e então os pontos que não fizerem parte do plano (dado um limiar) são considerados obstáculos.

[Murarka et al. 2008] utilizou um método estéreo global baseado em segmentação por cores e enquadramento de planos para gerar o mapa de disparidades. Em conjunto com um método baseado em movimento para detecção de quedas, o trabalho construiu um mapa bidimensional discreto onde cada célula poderia possuir os seguintes valores: chão, acima do chão, abaixo do chão, borda da queda e indefinido. Apesar de ter obtido uma boa precisão em se tratando de câmera estéreo, apenas o cálculo do mapa de disparidades demorava 4.5 segundos, limitando assim suas aplicações. O trabalho também não realizou navegação, utilizando a plataforma experimental apenas para obtenção dos dados.

3. Método

O método utilizado consiste em: uma câmera estéreo que fornece um par de imagens (informações do ambiente), estas são processadas pelo método estéreo semi-global para se obter o mapa de disparidades. O paradigma RANSAC é então aplicado ao mapa de disparidades para obtenção do plano onde se situa a plataforma veicular, com isso é possível gerar o mapa de navegabilidade e utiliza-lo como entrada para o algoritmo de desvio de obstáculos. O algoritmo então fornece os comandos de esterçamento e aceleração do veículo. Cada uma destas etapas será explicada em detalhes nas seções seguintes, a Figura 2 ilustra o método utilizado.

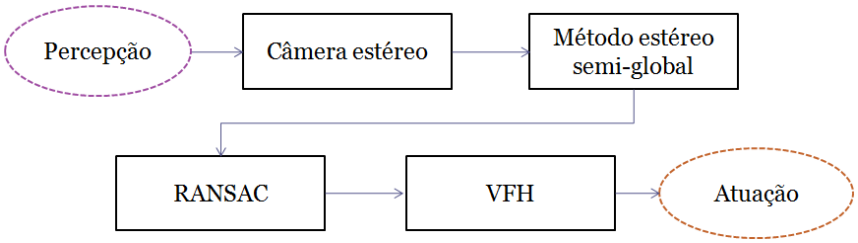


Figura 2. Diagrama de blocos do método utilizado.

3.1. Visão Estéreo e Métodos Estéreo

Uma câmera estéreo possui duas lentes horizontalmente deslocadas capturando, deste modo, duas imagens similares. Este par de imagens possui um pequeno deslocamento entre posições relativas de partes locais dependendo da distância que estes componentes locais estão da câmera. Ao estimar a diferença da posição de um ponto entre uma imagem e outra é possível aferir sua profundidade relativa, esse método assemelha-se a forma com que a visão humana nós proporciona noção de profundidade. A Figura 3 mostra a geometria de um sistema estéreo padrão.

Em um sistema ideal, sabendo-se a distância focal, o deslocamento base, x_r e x_t é possível, por triangulação, saber a distância relativa do ponto P à base da câmera. A equação 1 mostra como o cálculo é feito, sendo Z a distância, B o deslocamento base, f a distância focal e d a disparidade.

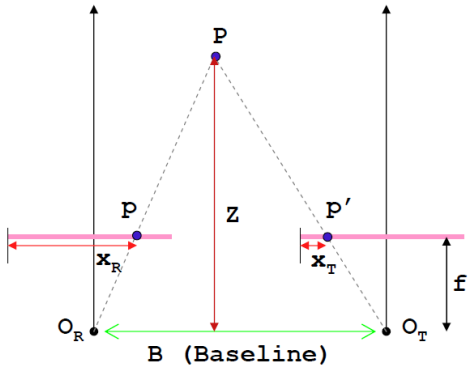


Figura 3. Sistema padrão (canônico) de uma câmera binocular com distância focal f e lentes deslocadas por uma distância B [Mattoccia 2009].

$$Z = \frac{B \cdot f}{x_r - x_t} = \frac{B \cdot f}{d} \tag{1}$$

Para estimar a disparidade, ou seja, a distância no eixo x dos pontos correspondentes p e p' é necessário utilizar um algoritmo de correspondência ou método estéreo, seu objetivo consiste em encontrar pontos correspondentes entre o par de imagens estéreo. Tal procura tem um alto custo computacional e por isso deve-se minimizar ao máximo sua área de busca, em um sistema canônico perfeito tal busca poderia ser limitada horizontalmente, entretanto, na realidade isso não acontece. Câmeras possuem distorções

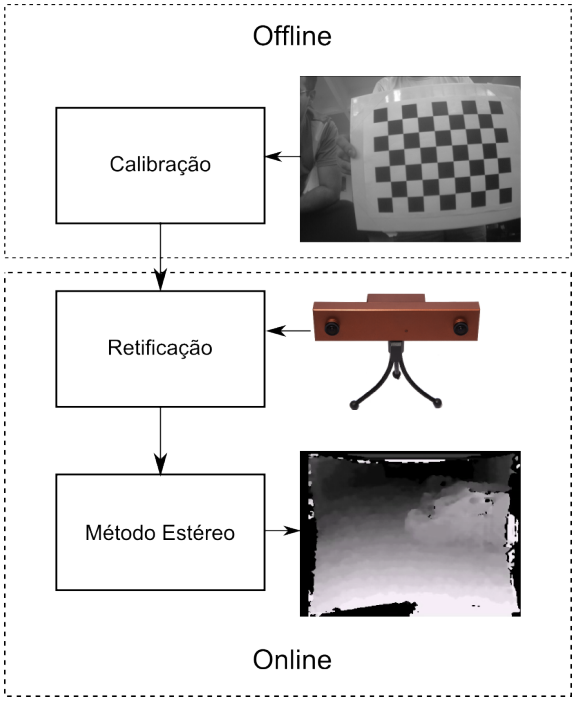


Figura 4. Fluxo resumido presente na utilização de uma câmera estéreo.

e não são perfeitamente alinhadas, por esse motivo temos que realizar uma calibração prévia das câmeras e retificar suas imagens a fim de corrigir as imperfeições e limitar horizontalmente a busca por pontos correspondentes.

A Figura 4 mostra o fluxo resumido referente a utilização de uma câmera estéreo. A calibração estima os parâmetros que possibilitam a remoção da distorção presente na imagem e sua retificação.

Com o auxilo da biblioteca de código aberto *Open Source Computer Vision* (OpenCV) [1] foi realizada a calibração da câmera. A partir das imagens retificadas deu-se início aos testes dos métodos estéreos disponíveis na biblioteca, sendo eles: (I) método local *block matching* [Konolige 1997]; (II) método semi-global *semi-global block matching*; (III) método global *graph cuts* [Kolmogorov and Zabih 2001].

Como esperado, utilizando o processador *Intel Core i7 720QM* [2] e uma resolução de 400 por 300 píxeis, o método global *graph cuts* demorou em torno de um minuto para o cálculo do mapa de disparidades tornando-se assim inviável para o uso na navegação almejada. O método *block matching* foi testado e seu resultado sofreu das conhecidas limitações encontradas em métodos locais [Hirschmuller 2005]. A Figura 5 mostra uma comparação entre os diferentes tamanhos de bloco utilizando o método. Para diminuir a ambiguidade e consequentemente minimizar os ruídos é necessário aumentar o tamanho do bloco, mas, no caso do tamanho 25 (25 por 25 píxeis) os efeitos colaterais (circulado na Figura 5) do aumento se tornam visíveis, o método é incapaz de distinguir os dois cones.

[1] http://opencv.willowgarage.com/wiki/
[2] http://ark.intel.com/Product.aspx?id=49024

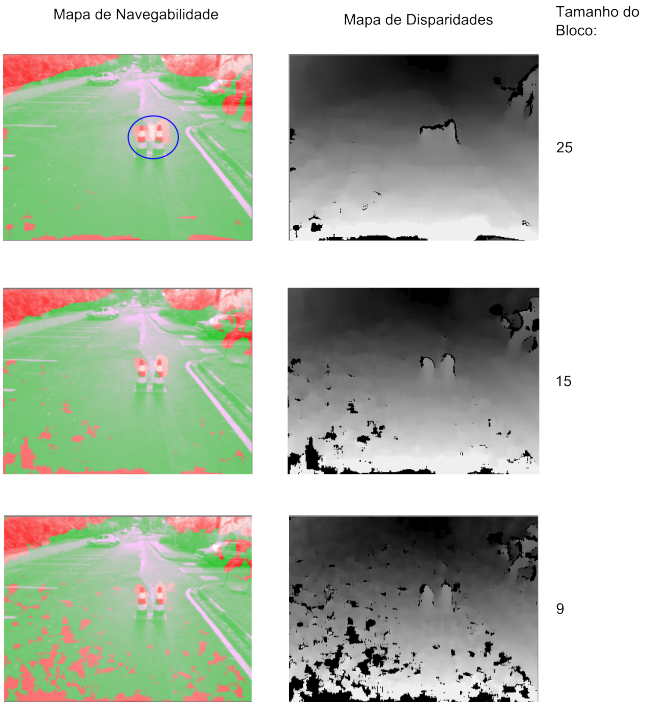


Figura 5. Comparação entre diferentes tamanhos de bloco utilizando o método local *block-matching*.

Com base nos testes citados optamos pelo método *semi-global block matching*, foram feitos testes exaustivos para seleção de parâmetros que melhor se adequem ao ambiente onde seria feita a navegação e a posição e orientação da câmera relativa à plataforma veicular. Os parâmetros mais influentes no mapa de disparidades foram o número de disparidades e o tamanho do bloco utilizado. O número de disparidades define o espaço de busca por combinações, na prática quanto maior esse número mais próximo a câmera consegue detectar um obstáculo, porém cresce o custo computacional. O tamanho do bloco influencia a densidade do mapa gerado ou seja, um bloco grande possibilita um mapa mais denso, mas pode ocultar detalhes da cena.

Um descrição detalhada do método estéreo semi-global utilizado pode ser vista em [Hirschmuller 2005] e [Hirschmuller 2006].

3.2. Detecção e Desvio de Obstáculos

Possuindo o mapa de disparidades gerado, o paradigma RANSAC, por causa de sua robustez à ruídos, foi utilizado para estimar os parâmetros do plano (via urbana) onde o veículo se situa. Utilizando heurísticas, os píxeis da imagem são classificados em navegáveis (verde) ou não navegáveis (vermelho).

A classificação é realizada com base na distância dos pontos ao plano, ou seja, pontos acima do plano (cones, pessoas) e abaixo do plano (buracos) são classificados como não navegáveis. Um limiar definido através de testes exaustivos é utilizado na classificação. A Figura 6 apresenta uma imagem classificada pelo método.

Este trabalho utiliza uma versão modificada do algoritmo VFH para realizar o desvio de obstáculos. O algoritmo utilizado no presente trabalho se vale das seguintes



Figura 6. Mapa de navegabilidade onde vermelho representa não navegável e verde navegável.

diferenças: (I) a construção do mapa cartesiano bidimensional é desconsiderada; (II) o histograma gerado não possui representatividade angular, ou seja, não é um histograma polar; (III) o número de setores do histograma é igual a resolução horizontal da imagem; (IV) a direção resultante do algoritmo é escalonada de acordo com o angulo de visão da câmera; (V) não é realizado o cálculo da velocidade, esta é mantida constante.

As modificações I, II, III e IV são justificadas pela simplicidade, natureza bidimensional das informações de profundidade providas pela câmera estéreo e a precisão alcançada através do método estéreo semi-global. A modificação V foi empregada apenas pela simplicidade.

A geração do histograma é feita de acordo com a equação 2. Onde H_k é a magnitude da célula k do histograma H, $IC_{(u,k)}$ é o valor presente na linha u e coluna k da matriz IC referente a classificação binária realizada, sendo navegável igual a zero e não navegável igual a um, a e b são constantes positivas utilizadas para escalar a influência da distância do veículo em relação ao obstáculo e n a resolução vertical do mapa de disparidades (iniciando pelo topo da imagem).

$$H_k = \sum_{u=0}^{n} IC_{(u,k)}(a - bu) \tag{2}$$

Utilizando o histograma gerado, o esterçamento do veículo é definido de acordo com a equação 3. Onde θ é o angulo de esterçamento, k_n representa a borda próxima do vale, k_f a borda distante do vale e c é a constante de escalonamento referente ao angulo de visão da câmera. As definições aqui utilizadas podem ser encontradas em [Borenstein and Koren 1991].

$$\theta = \frac{k_n + k_f}{2} c \tag{3}$$

3.3. Arquitetura do Sistema

A arquitetura relacionada à paralelização do sistema desenvolvido pode ser vista na Figura 7. As *threads*: (I) *stereo thread* é responsável pelo cálculo do mapa de disparidades

utilizando o método semi-global; (II) *main thread* calcula, com base no plano estimado, o mapa de navegabilidade e o histograma além de realizar a comunicação com sensores e atuadores; (III) *plane thread* faz o cálculo do plano onde se situa o veículo utilizando o paradigma RANSAC.

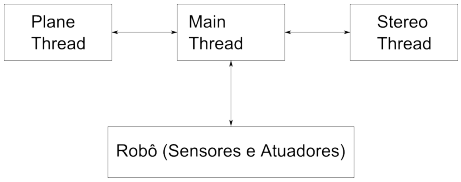


Figura 7. Arquitetura do programa relacionada a sua paralelização.

As *threads main thread* e *stereo thread* são sincronizadas a cada ciclo do programa, assim garantindo a utilização do mapa de disparidades mais recente. A *plane thread* é sincronizada apenas quando esta terminar de calcular o plano. Apesar de gerar robustez em relação à inclinação do terreno o cálculo contínuo do plano principal pode se tornar problemático quando o cenário contiver uma grande densidade de obstáculos.

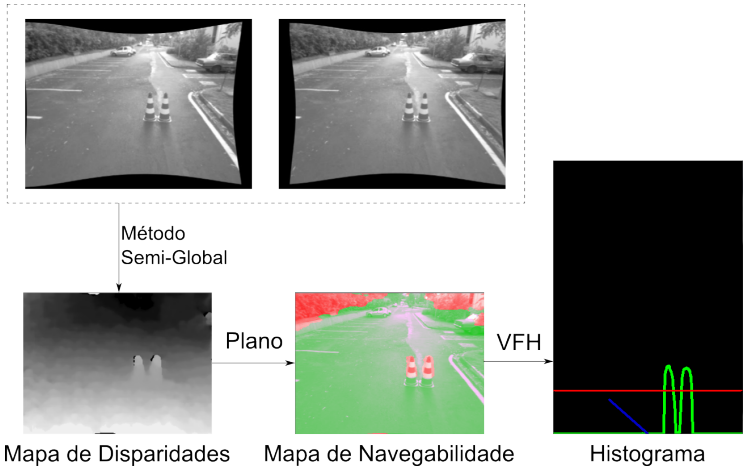


Figura 8. Fluxo principal do programa.

A Figura 8 mostra do fluxo geral do método onde o par de imagens estéreo capturadas pela câmera são retificadas para o cálculo do mapa de disparidade pelo método estéreo semi-global. Dado o mapa de disparidades o paradigma RANSAC estima o plano onde o veículo se situa e com base no plano o mapa de navegabilidade é criado. O histograma é então gerado a partir do mapa de navegabilidade, onde a linha vermelha representa o limiar escolhido, o segmento azul a direção para a qual o veículo deve se movimentar e em verde a densidade de obstáculos.

4. Resultados

Realizou-se um experimento onde o carro elétrico do projeto CArro Robótico Inteligente para Navegação Autônoma (CARINA) disponível no Laboratório de Robótica Móvel (LRM - USP) foi apto a desviar de obstáculos (pessoas e cones de trânsito). Utilizamos a câmera estéreo STOC-15CM-M-MINI da *Videre Design* [3] como sensor principal,

[3]http://www.videredesign.com



Figura 9. Carro elétrico desviando de cones de trânsito durante o experimento realizado.

o dispositivo da marca *Roboteq* [4], modelo AX-2850 para esterçamento e a plataforma aberta *Arduino* para o controle da aceleração. Em termos de *software* o experimento fez uso do *software The Player Project* [5] para se comunicar com o atuador da direção.

O ambiente escolhido para o experimento foi uma via urbana presente na USP, campus São Carlos, onde foram posicionados cones de trânsito a fim de testar o método desenvolvido. A Figura 9 ilustra o desvio de obstáculos realizado, como resultado o carro apresentou o comportamento esperado realizando adequadamente o desvio de obstáculos proposto. O mapa de navegabilidade se mostrou preciso o suficiente para realizar a navegação, a pequena porcentagem de ruídos é justificada pelo uso do método semi-global. A Figura 10 ilustra o mapa de navegabilidade gerado e o resultado do método de desvio de obstáculos.

Todo o sistema teve um desempenho compatível com sua aplicação em tempo real respeitando uma limitação de velocidade. O esterçamento, e portando, a *main thread*, foi atualizado a uma taxa de 5 Hz utilizando uma resolução de 400 por 300 para o par de imagens estéreo enquanto que o cálculo do plano foi realizado a uma de taxa de 1 Hz. Os testes foram feitos em um notebook da marca *Alienware* [6] modelo M15x e processador *Intel Core i7 720QM*. O número de disparidades do método estéreo foi de 16 com bloco

[4]http://www.roboteq.com/
[5]http://playerstage.sourceforge.net/
[6]http://www.alienware.com/

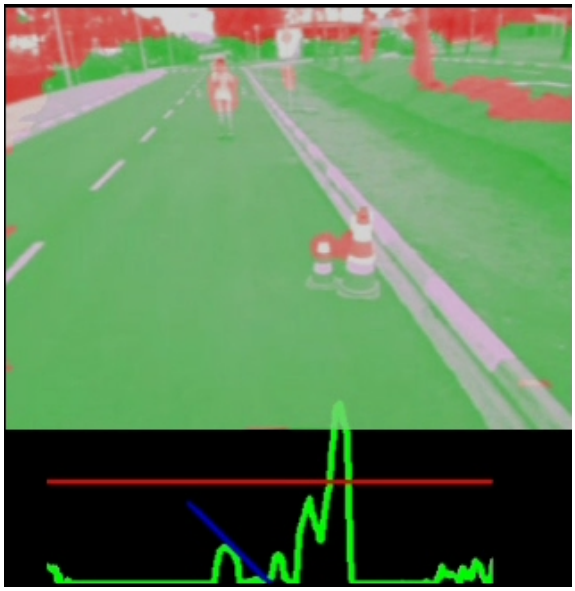


Figura 10. Mapa de navegabilidade gerado e ilustração do método de desvios de obstáculos utilizado.

do tamanho 7, o cálculo dos parâmetros do plano por meio do paradigma RANSAC foi limitado à 1000 iterações.

Os vídeos referentes ao experimento podem ser vistos em [http://www.youtube.com/watch?v=J8uWtgZWrUg] e [http://www.youtube.com/watch?v=TqncAL_ose8].

5. Conclusão e Trabalhos Futuros

Este artigo apresentou um sistema de desvio de obstáculos em tempo real utilizando uma câmera estéreo e um método estéreo semi-global. Demonstrando assim, que tal método pode ser utilizado na navegação de robôs móveis autônomos em ambientes externos, os quais exigem um alto grau de precisão por parte do sistema de percepção empregado. Tal precisão é classicamente obtida com o uso de sensores de alto custo como, por exemplo, lasers.

Utilizando o método estéreo *block-matching*, apresentou-se uma breve comparação dos efeitos relacionados ao tamanho do bloco escolhido em métodos locais. O resultado mostra como as já conhecidas limitações de métodos locais.

Como trabalhos futuros almejamos a detecção robusta de obstáculos frente à curvatura de vias urbanas (plano navegável) com o uso da distância relativa entre pontos e inclinação. Também é prevista a detecção de calçadas através de uma máquina de estados finitos.

Agradecimentos

Os autores agradecem a FAPESP pela bolsa concedida (processo número: 2010/04798-9) e a toda equipe do Laboratório de Robótica Móvel da USP - São Carlos pelo apoio e suporte.

Referências

Borenstein, J. and Koren, Y. (1991). The vector field histogram-fast obstacle avoidance for mobile robots. In *Robotics and Automation, IEEE Transactions on*, volume 7, pages 278 –288.

Broggi, A., Caraffi, C., Fedriga, R., and Grisleri, P. (2005). Obstacle detection with stereo vision for off-road vehicle navigation. In *Computer Vision and Pattern Recognition - Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*, page 65.

Caraffi, C., Cattani, S., and Grisleri, P. (2007). Off-road path and obstacle detection using decision networks and stereo vision. In *IEEE Transactions on Intelligent Transportation Systems*, volume 8, pages 607 –618.

DARPA (2005). DARPA Grand Challenge 2005. `http://replay.waybackmachine.org/20090327053503/http://www.darpa.mil/grandchallenge05/`. Acessado em 6 de abril de 2011.

DARPA (2007). DARPA Urban Challenge. `http://replay.waybackmachine.org/20090825041140/http://www.darpa.mil/grandchallenge/index.asp`. Acessado em 6 de abril de 2011.

Erkan, A., Hadsell, R., Sermanet, P., Ben, J., Muller, U., and LeCun, Y. (2007). Adaptive long range vision in unstructured terrain. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2421 –2426.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In *Commun. ACM*, volume 24, pages 381–395, New York, NY, USA. ACM.

Happold, M. and Ollis, M. (2006). Autonomous learning of terrain classification within imagery for robot navigation. In *Systems, Man and Cybernetics, 2006. SMC '06. IEEE International Conference on*, volume 1, pages 260 –266.

Hirschmuller, H. (2005). Accurate and efficient stereo processing by semi-global matching and mutual information. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 807 – 814 vol. 2.

Hirschmuller, H. (2006). Stereo vision in structured environments by consistent semi-global matching. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pages 2386–2393, Washington, DC, USA. IEEE Computer Society.

Kolmogorov, V. and Zabih, R. (2001). Computing visual correspondence with occlusions via graph cuts. In *In International Conference on Computer Vision*, pages 508–515.

Konolige, K. (1997). Small vision systems: Hardware and implementation. 333 Ravenswood Avenue, Menlo Park, CA 94025.

Konolige, K., Agrawal, M., Bolles, R., Cowan, C., Fischler, M., and Gerkey, B. (2008). Outdoor mapping and navigation using stereo vision. In Khatib, O., Kumar, V., and Rus, D., editors, *Experimental Robotics*, volume 39 of *Springer Tracts in Advanced Robotics*, pages 179 –190. Springer Berlin / Heidelberg.

Labayrade, R., Aubert, D., and Tarel, J.-P. (2002). Real time obstacle detection in stereovision on non flat road geometry through "v-disparity"representation. In *Intelligent Vehicle Symposium, 2002. IEEE*, volume 2, pages 646 – 651 vol.2.

Lima, D. and G.A.S., P. (2010). Um sistema de visão estéreo para navegação de um carro autônomo em ambientes com obstáculos. In *XVII Congresso Brasileiro de Automática*, pages 224 –231.

Mattoccia, S. (2009). Stereo Vision: Algorithms and Applications. `http://www.vision.deis.unibo.it/smatt/Seminars/StereoVision.pdf`. Acessado em 6 de abril de 2011.

Murarka, A. and Kuipers, B. (2009). A stereo vision based mapping algorithm for detecting inclines, drop-offs, and obstacles for safe local navigation. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 1646 –1653.

Murarka, A., Sridharan, M., and Kuipers, B. (2008). Detecting obstacles and drop-offs using stereo and motion cues for safe local motion. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 702 –708.

Murray, D. and Jennings, C. (1997). Stereo vision based mapping and navigation for mobile robots. In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, volume 2, pages 1694 –1699 vol.2.

Rankin, A., Huertas, A., and Matthies, L. (2005). Evaluation of stereo vision obstacle detection algorithms for off-road autonomous navigation. In *In AUVSI Symp. on Unmanned Systems*.

Scharstein, D. and Szeliski, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In *Int. J. Comput. Vision*, volume 47, pages 7–42, Hingham, MA, USA. Kluwer Academic Publishers.

Sermanet, P., Hadsell, R., Scoffier, M., Muller, U., and LeCun, Y. (2008). Mapping and planning under uncertainty in mobile robots with long-range perception. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 2525 –2530.

Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., Lau, K., Oakley, C., Palatucci, M., Pratt, V., Stang, P., Strohband, S., Dupont, C., Jendrossek, L.-E., Koelen, C., Markey, C., Rummel, C., van Niekerk, J., Jensen, E., Alessandrini, P., Bradski, G., Davies, B., Ettinger, S., Kaehler, A., Nefian, A., and Mahoney, P. (2006). Stanley: The robot that won the darpa grand challenge. In Periodicals, W., editor, *Journal of Field Robotics*, volume 23, pages 661–692.

TRACTS: Um método para a classificação de trajetórias de objetos móveis usando séries temporais

Irineu Jr. Pinheiro dos Santos[1], Luis Otavio Alvares[1,2]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

{ijpsantos,alvares}@inf.ufrgs.br

[2]Depto. de Informática e Estatística – Universidade Federal de Santa Catarina (UFSC)
Caixa Postal 476 – 88.040-900 – Florianópolis – SC – Brasil

alvares@inf.ufsc.br

Abstract. *Trajectory classification is a relatively new research issue, but some methods have already been proposed. Most of them are developed for a specific application. A few are more general and can be applied to several domains. This paper presents a new trajectory classification method that transforms trajectories into time-series, in order to obtain more discriminative features for classification. The results of the processed experiments have been compared with the TraClass algorithm and have presented better results.*

Resumo. *A classificação de trajetórias é um assunto relativamente novo de pesquisa, mas alguns métodos já foram propostos. A maioria destes métodos foi desenvolvido para uma aplicação específica. Poucos propuseram um método mais geral, aplicável a vários domínios ou conjuntos de dados. Esse trabalho apresenta um novo método de classificação que transforma as trajetórias em séries temporais de forma a obter características mais discriminativas para a classificação. Os resultados dos experimentos realizados foram comparados com os obtidos pelo método TraClass e se mostraram, em geral, superiores.*

1. Introdução

Nos anos recentes, observa-se o crescimento do uso de sistemas baseados em localização, tais como GPS, triangulação de rádio e de celulares, sistemas wireless e RFID. Esses sistemas possibilitaram e facilitaram o rastreamento de objetos móveis com uma resolução de apenas alguns metros no espaço, ou com maior precisão ainda, em alguns casos.

O aumento do uso de tais sistemas está levando à maior disponibilidade de dados espaço-temporais de pessoas, veículos e animais, por exemplo. Essa coleção de dados espaço-temporais representa um ambiente extremamente rico para análise, pois dentro dessa grande massa de dados estão diversas informações a respeito de comportamentos individuais e coletivos dos objetos móveis envolvidos.

Uma das formas de se extrair informações interessantes a partir de um conjunto de dados espaço-temporais é através da análise de trajetórias, que podem ser construídas a partir dos dados coletados. Uma trajetória é uma sequência de pontos (id, x, y, t), em que x e y indicam a posição do objeto móvel identificado por id no tempo t. A

mineração e a descoberta de conhecimento a partir dessas trajetórias têm emergido como um tópico frequente em pesquisas acadêmicas e em aplicações práticas.

A análise de diversas trajetórias de objetos móveis permite, por exemplo, realizar a identificação de lugares importantes, ou POIs (*Points of Interest*), para esse determinado objeto móvel. Isso possibilita a armazenagem de históricos de sua localização em cada um desses POIs [Hariharan and Toyama 2004], descobrir a sequência de locais que geralmente são visitados pelo objeto móvel [Zhou et al. 2007], realizar a predição de destinos possíveis [Ashbrook and Starner 2003] e até mesmo permitir que um usuário seja avisado em tempo real que está pegando um ônibus errado [Liao et al. 2006]. Focando-se nas propriedades geométricas das trajetórias, existe a possibilidade de detecção de trajetórias similares, estabelecendo correlação entre objetos móveis [Tiakas et al. 2009], padrões de congruência [Gudmundsson et al. 2007] e predições de comportamento, como, por exemplo, no tempo de viagem de um veículo [Tiesyte and Jensen 2008]. Indo além da análise das propriedades geométricas das trajetórias e considerando-se também os objetos geográficos estáticos em suas proximidades, tais como hotéis, pontos turísticos de uma cidade e pedágios, é possível atribuir semântica a essas trajetórias, estabelecendo, por exemplo, rotas turísticas ou trechos de congestionamento no fluxo do trânsito [Palma et al. 2008].

Uma forma de análise dos dados espaço-temporais, que ainda foi pouco explorada na literatura atual, é a classificação de trajetórias. Essa abordagem utiliza o conceito de classificação tradicional da mineração de dados [Tan et al. 2006], onde é aprendida uma função alvo f que mapeia um conjunto de atributos x para um dos rótulos de classe predefinidos y. Na classificação de trajetórias, a função f, ou modelo de classificação, mapeia cada trajetória para um determinado rótulo de classe. Esse rótulo de classe y determina o objetivo do processo de classificação, que pode ser desde a identificação de diferentes tipos de objetos móveis [Lee et al. 2008], até a identificação de meios de transporte (carro, bicicleta, etc.) utilizado pelo objeto móvel [Zheng et al. 2008].

Entre os métodos existentes para classificação de trajetórias, a técnica utilizada para a identificação das características para a classificação pode variar bastante. Pode ser utilizando, por exemplo, técnicas de análise de similaridade [Panagiotakis et al. 2009] ou de análise de regiões do espaço [Lee et al. 2008].

Este trabalho apresenta um novo método para classificação de trajetórias, denominado *TRACTS* (***Tra****jectory* ***C****lassification using* ***T****ime* ***S****eries*), baseado na transformação de cada trajetória em um conjunto de séries temporais para que destas sejam extraídas as características que serão submetidas ao processo de geração do modelo de classificação. Este método é, no nosso conhecimento, o primeiro método que utiliza séries temporais para a classificação de trajetórias.

O restante do trabalho está organizado da seguinte forma: na Seção 2 são apresentados os trabalhos relacionados e uma breve base conceitual; na Seção 3 é apresentado o *TRACTS*, o novo método para classificação de trajetórias; a Seção 4 descreve alguns experimentos realizados utilizando-se o método proposto e os resultados obtidos; por fim, a Seção 5 conclui o artigo e apresenta trabalhos futuros.

2. Trabalhos relacionados

Para a classificação de dados convencionais, como no exemplo clássico da planta Iris, o conjunto de dados contém atributos descritivos da planta como o comprimento e largura da pétala e da sépala. Cada registro corresponde a um exemplo, com os dados de uma planta específica: os valores de suas características e a sua classe. Já para realizar a classificação de trajetórias, os dados fornecidos se resumem a uma sequência (id, x, y, t), onde cada identificador da trajetória id está associado a uma classe, o que dificulta bastante o processo.

Entre a base de dados de trajetórias e a construção do modelo de classificação final, algum tipo de processamento dos dados da trajetória deve ser realizado. O trabalho proposto por [Panagiotakis et al. 2009] introduz um método de classificação de trajetórias que une técnicas de análise de similaridade de trechos das trajetórias com uma função de votação definida pela distância entre esses trechos, possibilitando a construção de um modelo de classificação a partir dessa função de votação.

A segmentação e extração de características de trajetórias também tem sido uma estratégia bastante utilizada nos métodos existentes para classificação de trajetórias. Os segmentos, e suas características, podem definir, por exemplo, comportamentos específicos e, quando ordenados em sequência, podem estabelecer a classificação de uma trajetória. Essa abordagem permitiu, no trabalho de [Lee and Hoff 2007], a identificação de atividades humanas a partir de trajetórias, adquiridas por vídeo, de diferentes modalidades esportivas.

Da mesma forma, no trabalho desenvolvido por [Zheng et al. 2008], foi possível inferir o meio de transporte utilizado a partir das trajetórias de diversos objetos móveis. Uma trajetória podia ter sido realizada com mais de um meio de transporte e, neste caso, o meio de transporte utilizado em cada sub-trajetória era definido. O método usa conhecimento da aplicação, por exemplo, para a passagem de um meio de transporte para outro a velocidade deveria ficar por alguns instantes próxima de zero e usa como principais características a serem consideradas para a classificação a velocidade média e a velocidade máxima na subtrajetória. Com isso diferenciava entre as 4 classes consideradas: carro, ônibus, bicicleta e a pé.

No trabalho de [García et al. 2006] são utilizados filtros de Kalman para identificar modos de voo (subtrajetórias em trajetórias de aviões como curvas, acelerações, aproximação para pouso, etc.).

Jiawei Han e seu grupo propõem o framework *TraClass* para a classificação de trajetórias, com dois métodos para a extração de características em [Lee et al. 2008]. O primeiro método, denominado *Region Based Clustering* (RB) descobre regiões com trajetórias essencialmente de uma classe, independente do tipo de movimento realizado. O segundo método, denominado *Trajectory Based Clustering* (TB) descobre subtrajetórias que descrevem padrões de movimento comum para cada classe. Nenhuma característica geométrica, global ou pontual, da trajetória é extraída para análise. Somente propriedades de movimentação (de subtrajetórias) e de regiões das trajetórias são utilizadas para a sua classificação.

Como pode ser observado, a maioria dos trabalhos descreve métodos específicos para a aplicação sendo considerada. O trabalho de [Lee et al. 2008] é que apresenta um

método mais geral. O método *TRACTS* também é geral. Ele utiliza algumas técnicas que serão sucintamente descritas a seguir, como base conceitual para o restante do trabalho.

2.1 Séries Temporais

Séries temporais consistem de sequência de valores ou eventos obtidos sobre repetidas medidas de tempo [Han and Kamber 2006]. Os valores são geralmente medidos em intervalos regulares de tempo (a cada hora, diariamente, semanalmente, etc.). Aplicações envolvendo séries temporais têm sido muito utilizadas em áreas como bolsa de valores, análise orçamentária, projeções de produtividade, controle de qualidade, observações de fenômenos naturais (temperatura, ventos, terremotos, etc.) e tratamentos médicos.

Devido ao grande tamanho e a alta dimensionalidade de muitas séries temporais, a redução de dados frequentemente serve como um primeiro passo na análise de séries temporais. A redução de dados leva não somente a um espaço de armazenamento muito menor, mas também a um processamento dos dados muito mais rápido.

Entre as estratégias para redução de dados, é possível encontrar a **redução dimensional**, que emprega técnicas de processamento de sinal para obter versões reduzidas do dado original. Essa estratégia de redução se mostra extremamente importante, pois, por exemplo, para computar a relação entre duas curvas de séries temporais, a redução da série temporal do comprimento (ou dimensão) n para k pode levar a redução de $O(n)$ para $O(k)$ em termos de complexidade computacional. Se $k \ll n$, a complexidade da computação será largamente reduzida.

A figura 1 mostra algumas técnicas de redução dimensional em séries temporais.

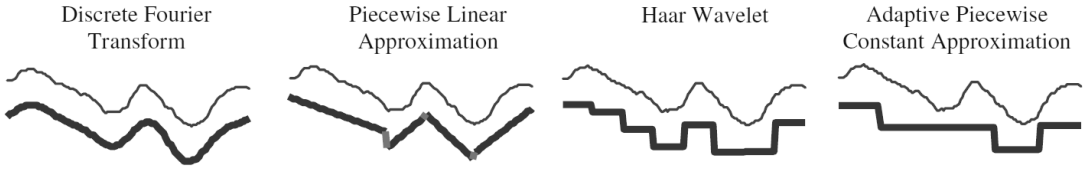


Figura 11: Redução dimensional de séries temporais

2.2 O método SAX

O método ***S**ymbolic **A**ggregate Approx**i**mation* (SAX), proposto por [Lin et al. 2007], permite que uma série temporal de tamanho *n* seja reduzida para uma sequência de caracteres de tamanho *w*, onde $w \leq n$. O alfabeto utilizado para a sequência de caracteres é de tamanho *a*, onde $a \geq 2$. No método SAX, para cada um dos valores obtidos nas séries temporais do conjunto de dados, é realizada uma transformação que permite a sua representação simbólica.

A faixa de valores correspondentes a cada símbolo é determinada pelo valor dos elementos das diversas séries temporais consideradas e pela quantidade de faixas que são desejadas na representação. Nessa abordagem, todos os valores $(v_1, v_2, v_3, \ldots, v_n)$ de todas as séries temporais consideradas são primeiramente ordenados em ordem crescente de valor. Esse conjunto de valores é então dividido igualmente em f faixas $(f_1, f_2, f_3, \ldots, f_n)$, sendo possível obter um número de $\frac{n}{f}$ elementos em cada faixa. Dessa forma, é possível estabelecer que $a = f$. A aplicação do método SAX em duas séries temporais, C_1 e C_2, é ilustrado na figura 2.

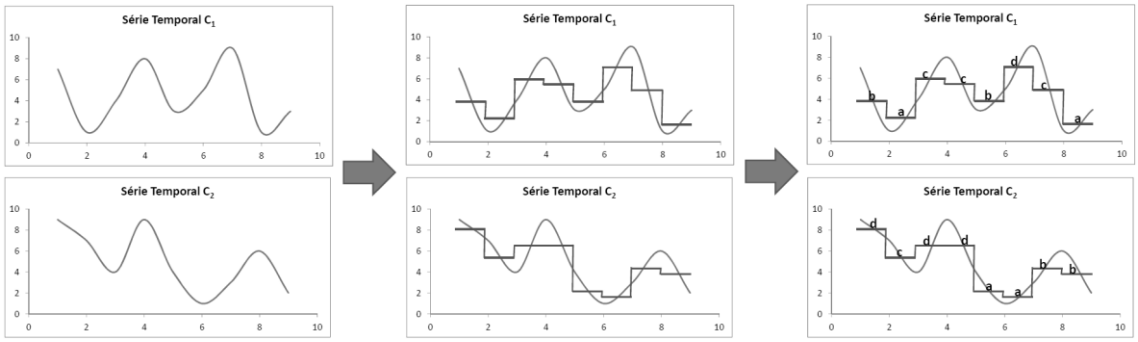


Figura 22: Aplicação do método SAX em duas séries temporais

2.3 O método Time-Series Bitmaps

Entre os métodos atualmente disponíveis para mineração de séries temporais que possuem capacidade de tratamento de sequências de caracteres, encontra-se o método que transforma séries temporais em mapas de bits, chamado *Time-Series Bitmaps* (TSB). Esse método foi proposto por [Kumar et al. 2005]. A ideia central do método é transformar uma sequência de caracteres, como, por exemplo, uma representação discretizada de uma série temporal, em uma matriz de mapa de bits, onde cada célula da matriz realizará a contagem das ocorrências de uma *substring* na *string* principal. Nesse método, a *substring* deve ter tamanho maior ou igual a um, definindo dessa forma a profundidade de análise desejada.

3. *TRACTS*: o método proposto para classificação de trajetórias

O método para a classificação de trajetórias *TRACTS* contém três etapas principais. A Figura 3 apresenta uma visão geral do método, que será detalhado nas seções seguintes.

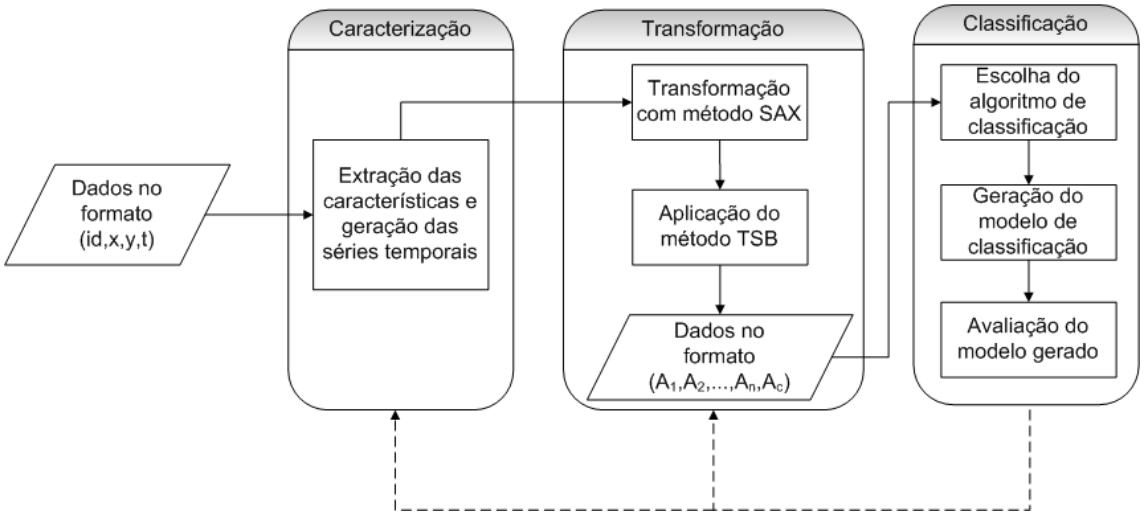


Figura 33: Visão geral do método *TRACTS*

Na figura não está representada a etapa inicial, presente em qualquer método de mineração de trajetórias, que é a parte de limpeza e eventualmente reconstrução das trajetórias, de acordo com a necessidade das operações que serão realizadas nas etapas seguintes. Um exemplo de caso onde ocorre esse tipo de situação é na segmentação de trajetórias em sub-trajetórias menores.

3.1 Caracterização

Na etapa de caracterização dos dados é realizada a identificação e extração das características importantes a partir do conjunto de dados. Essa é uma etapa fundamental para o sucesso do método *TRACTS*, pois ela fornece a semântica necessária para a análise no processo de classificação.

Uma tarefa não trivial nessa etapa é a identificação dos atributos que são relevantes para o domínio considerado. Entretanto, como estamos trabalhando com trajetórias de objetos em movimento, certas características serão sempre universais para todas as trajetórias, tais como velocidade, aceleração e direção vetorial absolutas, assim como a variação de cada uma dessas características em determinado intervalo. Essas são as características utilizadas por *default*. Cada trajetória é então representada como um conjunto de séries temporais, uma para cada característica considerada. Esse é um diferencial do método *TRACTS*, pois é o primeiro a considerar não apenas valores globais da trajetória como média e desvio padrão, mas o valor das características em cada ponto da trajetória.

3.2 Transformação

Nesta etapa, os dados contínuos das séries temporais geradas na etapa anterior são processados de tal forma que um conjunto de séries temporais seja resumido a um único registro com as características relevantes dessas séries, em formato adequado para os algoritmos tradicionais de classificação.

A primeira técnica utilizada é o método de agregação simbólica SAX, introduzido na subseção 2.2. Na aplicação deste método, o conjunto das séries temporais de cada característica é considerado, de tal forma que cada letra do alfabeto utilizado tenha o mesmo número de ocorrências no conjunto das séries temporais daquela característica.

Com as séries temporais transformadas em sequências de caracteres, é necessário utilizar um método de tratamento de *strings* para possibilitar o processo de classificação dessas trajetórias. Para isso é utilizado o método TSB, explicado na subseção 2.3, que realiza a transformação das sequências de caracteres produzidas pelo método SAX em matrizes. É construída uma matriz TSB para cada característica de cada trajetória. Conforme explicado anteriormente, elas mapearão o número de *substrings* da *string* principal formada, em cada uma das células da matriz. Entretanto, como as trajetórias dos domínios podem ter quantidades diferentes de pontos, utiliza-se o percentual de cada uma delas. A Figura 4 apresenta um exemplo de geração da matriz TSB com profundidade um (considera como *substring* somente um símbolo por vez) para uma série temporal com quatro faixas (*a, b, c, d*). Quando a profundidade é maior, é obtida uma análise mais detalhada dos eventos sequenciais da série temporal, porém a complexidade computacional cresce exponencialmente para maiores níveis de profundidade.

Cada atributo de classificação para cada característica considerada é mapeado para uma das células da matriz TSB. Por exemplo, se na Figura 4 estivéssemos representando os valores de velocidade de uma trajetória, teríamos quatro atributos (com seus respectivos valores entre parênteses): *velocidade_a* (8,82%), *velocidade_b* (20,59%), *velocidade_c* (29,41%) e *velocidade_d* (41,18%).

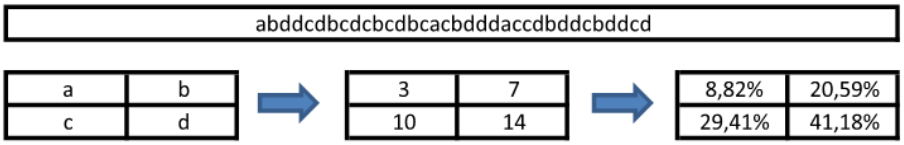


Figura 4: Geração de uma matriz TSB

Portanto, após a aplicação do método TSB, cada célula da matriz, juntamente com seu valor, vira um atributo, possibilitando a construção do conjunto de dados no formato $(A_1, A_2, A_3, \ldots, A_n, A_c)$, onde $A_1, A_2, A_3, \ldots, A_n$ correspondem aos atributos de característica e o atributo A_c ao atributo classe, que servirá como entrada para a próxima etapa, onde será realizada a classificação propriamente dita.

3.3 Classificação

Finalmente, o conjunto de atributos obtidos na etapa anterior é submetido a um algoritmo de classificação tradicional, tal como C4.5 ou Naive Bayes. O desempenho do algoritmo de classificação deve ser medido, permitindo estabelecer se o modelo de classificação gerado foi satisfatório ou não.

Assim como o processo tradicional de descoberta de conhecimento estabelece [Fayyad et al. 1996], caso seja necessário, no método *TRACTS* também é possível retornar a alguma das etapas anteriores, possibilitando o ajuste de parâmetros para geração de um novo modelo de classificação.

4. Experimentos Realizados

O método *TRACTS* foi testado e validado utilizando os mesmos dados do trabalho realizado pelo grupo de Jiawei Han [Lee et al. 2008], que introduziu o método *TraClass*, também para classificação de trajetórias. Isso permitiu uma base comparativa para os resultados obtidos.

Três bases de dados foram utilizadas nos experimentos. A primeira é o conjunto de trajetórias de três tipos de animais, a segunda é uma base de dados com trajetórias de navegação de dois barcos, e por fim, uma base de dados contendo trajetórias de rastreamento de furacões. Para cada base de dados, foi realizado um experimento em separado.

Para os três experimentos foi utilizado o mesmo conjunto de características: velocidade (v_n), aceleração (a_n), direção (d_n) e variação da direção (vd_n), calculadas da seguinte forma (onde ΔD e Δt são a distância e o tempo entre dois pontos consecutivos da trajetória, respectivamente, $\theta(p_{n-1}, p_n)$ retorna o ângulo entre a reta formada pelos pontos p_{n-1} e p_n e o norte geográfico e $diff$ retorna a diferença entre dois ângulos):

$$v_n = \frac{\Delta D^{p_n}_{p_{n-1}}}{\Delta t^{p_n}_{p_{n-1}}} \qquad a_n = \frac{v_{n+1} - v_n}{t_{n+1} - t_n} \qquad d_n = \theta(p_{n-1}, p_n) \qquad vd_n = diff(d_n, d_{n+1})$$

Com isso obtiveram-se quatro séries temporais para cada trajetória.

Na etapa de transformação, cada um dos valores das características obtidos na etapa anterior foi mapeado para um símbolo, através do método SAX (como as trajetórias eram relativamente pequenas, com uma média de 433 pontos em 2033 trajetórias, não houve necessidade de redução de dimensionalidade) e, logo após,

inserido em uma matriz de mapa de bits de série temporal. Para realizar a agregação simbólica com o método SAX, todos os n valores, para cada uma das quatro características obtidas de cada domínio, são considerados juntamente. Para cada caraterística, é formada uma lista ordenada do menor para o maior valor, que é dividida em um número equiprovável de elementos, de acordo com o número de faixas estabelecido. Conforme explicado na seção 3, essas faixas irão formar os símbolos SAX que serão mapeados para cada uma das trajetórias do domínio.

As séries temporais geradas pelo método SAX foram então submetidas ao método TSB para a geração das matrizes e, consequentemente, dos atributos utilizados para a classificação.

Com o arquivo de classificação contendo os atributos e seus respectivos valores, foi utilizada a ferramenta *Weka* [Frank et al. 2005], que possibilitou a utilização de diversos algoritmos de classificação, bem como a análise comparativa da acurácia de cada um dos modelos gerados. Em todas as execuções foi utilizada validação cruzada com 10 subconjuntos.

4.1 Experimento com a base de dados de animais

A base de dados com a trajetória de animais está disponível em [Pacific Northwest Research Station 2005]. Ela contém a trajetória de três diferentes tipos de animais: alce, gado e veado, que determinam a classe das trajetórias. Todos foram rastreados a partir de coleiras RFID.

Nesse conjunto de dados, houve necessidade de eliminação de ruídos, já que foi observado que a velocidade média entre alguns pontos era maior que a velocidade possível para o respectivo animal que originava essa trajetória. Também foi possível observar certa irregularidade no intervalo de obtenção das coordenadas geográficas, o que eventualmente ocasionava grande tempo decorrido entre a aquisição de dois pontos consecutivos. Dessa forma uma segmentação das trajetórias em sub-trajetórias menores foi realizada.

A tabela 1 mostra os resultados comparativos, para a acurácia de classificação, entre o método TRACTS e o método *TraClass*.

Tabela 11: Resultado comparativo para as trajetórias dos animais

Método	Acurácia	Algoritmo de Classificação Utilizado
TRACTS.7.2	95,26%	*MultiLayerPerceptron*
TRACTS.3.4	93,28%	*SVM (SMO)*
TRACTS.5.3	93,28%	*SVM (SMO)*
TRACTS.3.5	93,28%	*SVM (SMO)*
TRACTS.3.3	91,70%	*Adaboost+NaiveBayes*
TraClass RB-	83,30%	*SVM*
TraClass TB-	50,00%	*SVM*

Cada uma das linhas da tabela onde o método *TRACTS* é citado representa um parâmetro de entrada para o método, onde o primeiro algarismo representa o tamanho do alfabeto SAX e o algarismo seguinte representa a profundidade de análise no método TSB. Foram executados experimentos para alfabetos SAX de tamanhos 3, 4, 5 e 7, com profundidade da matriz TSB variando de 1 a 5 e diversos algoritmos disponíveis na ferramenta Weka. Nas tabelas de resultados comparativos mostrados nesse trabalho, são exibidos os cinco melhores resultados obtidos com o método *TRACTS*. *TraClass RB-TB*

corresponde ao método *TraClass* aplicado com as duas opções de geração de características, *region-based* e *trajectory–based*. *TraClass TB-only* corresponde a execução do método *TraClass* apenas considerando a característica baseada na trajetória. Não implementamos o método *TraClass*. Nas tabelas com os resultados dos experimentos, no que se refere ao método *TraClass,* reproduzimos os valores constantes no artigo que descreve o método [Lee et al. 2008]. Como pode ser observado na Tabela 1, o método *TRACTS* obteve resultados sensivelmente melhores que o *TraClass* em várias configurações de parâmetros.

4.2 Experimento com a base de dados de barcos

Para o segundo experimento foi utilizada a base disponível em [Monterey Bay Aquarium Research Institute 2001], contendo a trajetória de dois barcos. Como as trajetórias dos barcos eram apenas duas trajetórias extremamente longas, para que a análise fosse possível, cada uma das trajetórias foi dividida em 200 sub-trajetórias menores, com distribuição equivalente de pontos. A classe correspondia ao nome do barco.

Da mesma forma que na base de dados dos animais, as mesmas características foram extraídas, as strings SAX foram transformadas em matrizes de bitmaps e a ferramenta *Weka* foi utilizada para a etapa de classificação. O resultado desse experimento é apresentado na Tabela 2. Como pode ser observado, o resultado foi melhor que o do método *TraClass* em várias configurações de parâmetros.

Tabela 22: Resultado comparativo para as trajetórias de barcos

Método	Acurácia	Algoritmo de Classificação
TRACTS.5.2	100,00%	*Adaboost+J48*
TRACTS.7.2	100,00%	*Adaboost+SimpleCart*
TRACTS.3.3	99,75%	*NaiveBayes*
TRACTS.3.4	99,75%	*BayesNet*
TRACTS.3.5	99,75%	*BayesNet*
TraClass RB-TB	98,20%	*SVM*
TraClass TB-Only	84,40%	*SVM*

4.3 Experimento com a base de dados de furacões

As trajetórias de furacões, disponível em [Unisys 2009], rastreou um conjunto de furacões que se originaram próximos ao golfo do México. Nessa base de dados, cada ponto continha a informação de força do furacão, que ia de F0 (mais fraco) até F5 (mais forte). Para que fosse possível classificar a trajetória como um todo, a trajetória inteira de cada furacão recebeu como atributo classe a força máxima que ele chegou. Nesse experimento, foram considerados somente os furacões de classe F2 e F3, como nos experimentos relatados no artigo que descreve o método *TraClass*, para possibilitar a comparação.

Como nos experimentos anteriores, foram executadas as etapas de caracterização, transformação e classificação, obtendo os resultados exibidos na Tabela 3.

Como pode ser observado pelos resultados obtidos, a acurácia foi bem menor do que nos experimentos anteriores, porque a trajetória dos furacões não está fortemente correlacionada com a sua intensidade. O método que obteve os melhores resultados foi o *TraClass RB-TB*, pois analisa a região de ocorrência das trajetórias (*Region Based*), já

que furacões mais fortes tendem a atingir locais mais afastados do que furacões mais fracos. No método *TRACTS*, os melhores resultados do método foram a partir de análises mais profundas do alfabeto SAX (com três ou quatro *substrings* sendo avaliadas por vez).

Tabela 33: Resultado comparativo para as trajetórias de furacões

Método	Acurácia	Algoritmo de Classificação
TraClass RB-TB	73,10%	*SVM*
TRACTS.4.3	69,45%	*Adaboost+J48*
TRACTS.7.3	69,45%	*SVM (SMO)*
TRACTS.3.4	68,41%	*Adaboost+RandomForest*
TRACTS.5.3	66,32%	*BayesNet*
TraClass TB-Only	65,40%	*SVM*
TRACTS.3.2	64,23%	*BayesNet*

4.5 Resultado geral

Considerando os melhores resultados de cada uma das execuções dos experimentos, é possível compilar um resultado geral para os experimentos, através da média de acurácia dos resultados, conforme pode ser visualizado na Tabela 4.

Tabela 44: Resultado comparativo da aplicação do método *TRACTS* com o método *TraClass*

Método	Acurácia			
	Animais	Barcos	Furacões	Média
TRACTS	95,26%	100,00%	69,45%	**88,24%**
TraClass RB-TB	83,30%	98,20%	73,10%	**84,87%**
TraClass TB-Only	50,00%	84,40%	65,40%	**66,60%**

Mesmo com a dificuldade de classificação no domínio de dados dos furacões, o método *TRACTS* apresentou resultado superior quando é considerada a média da acurácia dos experimentos realizados.

5. Conclusão e trabalhos futuros

Nesse trabalho foi apresentado o *TRACTS*, um novo método para classificação de trajetórias, utilizando séries temporais e técnicas de mineração de dados para esse tipo de representação, possibilitando a criação de modelos de classificação com boa acurácia.

O método *TRACTS* demonstrou-se eficaz na tarefa de criar um modelo de classificação com boa acurácia, no geral superior a um método proposto para o mesmo fim.

Fica possível observar que o método, apesar de possuir uma grande independência de domínio, possui certa dificuldade para realizar a classificação de domínios caóticos, tal como o domínio dos furacões. Em parte, isso se deve as características extraídas de cada uma das trajetórias desse domínio, já que furacões não se movem mais rápida ou vagarosamente de acordo com algum critério conhecido. Outra possível causa da dificuldade de classificação nesse domínio foi estabelecer o atributo classe como a força máxima atingida pelo furacão, já que a força máxima pode ter sido atingida somente em um breve período de tempo, enquanto o método considera a trajetória inteira. Seria possível segmentar a trajetória dos furacões em subtrajetórias para cada classe, o que melhoraria de forma significativa a acurácia, mas como o

método usado na comparação não tratou o conjunto de dados dessa forma, essa formatação não foi considerada.

Como trabalho futuro será realizando um estudo mais aprofundado, considerando outros domínios de aplicação, sobre a influência do tamanho do alfabeto utilizado no método SAX e do tamanho da *substring* avaliada no método TSB na acurácia de classificação. Também será investigado o uso de outros métodos de tratamento de *strings*, além do TSB.

6. Referências

Ashbrook, D., & Starner, T. (2003). "Using GPS to Learn Significant Locations and Predict Movement Across Multiple Users". *Personal and Ubiquitous Computing Journal*, 275 - 286.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). "From Data Mining to Knowledge Discovery in Databases". *AI Magazine*, 37-54.

Frank, E., Hall, M., Holmes, G., Kirkby, R., & Pfahringer, B. (2005). "WEKA - A Machine Learning Workbench for Data Mining". In: *Collection of The Data Mining and Knowledge Discovery Handbook* (pp. 1305-1314). Springer.

García, J., Concha, O. P., Molina, J. M., & Miguel, G. (2006). "Trajectory classification based on machine-learning techniques over tracking data". *Proceedings of 9th International Conference on Information Fusion*, (pp. 1-8). Florence, Italia.

Gudmundsson, J., Kreveld, M. v., & Speckmann, B. (2007). "Efficient Detection of Patterns in 2D Trajectories". *GeoInformatica*, 195-215.

Han, J., & Kamber, M. (2006). "Mining Stream, Time-Series and Sequence Data". In: *Data Mining: Concepts and Techniques*. (pp. 467-489). San Francisco, CA, EUA: Morgan Kaufmann.

Hariharan, R., & Toyama, K. (2004). "Project Lachesis: Parsing and Modeling Location Histories". *Proceedings of the 3rd International Conference on Geographic Information Science* (pp. 106–124). Adelphi, EUA: Springer.

Kumar, N., Lolla, V. N., Keogh, E., Lonardi, S., Ratanamahatana, C. A., & Wei, L. (2005). "Time-series Bitmaps: A Practical Visualization Tool for working with Large Time Series Databases". *Proceedings of 5th SIAM International Conference on Data Mining - SDM'05, Proceedings* (pp. 531-535). Newport Beach, CA, EUA: SIAM.

Lee, J. Y., & Hoff, W. (2007). "Activity Identification Utilizing Data Mining Techniques". *Motion and Video Computing, IEEE Workshop on* (p. 12). Austin, Texas, EUA: IEEE Computer Society.

Lee, J.-G., Han, J., Gonzalez, H., & Li, X. (2008). "TraClass: trajectory classification using hierarchical region-based and trajectory-based clustering". *VLDB Endowment* (pp. 1081-1094). Auckland, Nova Zelândia: VLDB Endowment.

Liao, L., Patterson, D., Fox, D., & Kautz, H. (2006). "Building Personal Maps from GPS Data". *Annals of the New York Academy of Sciences*, 249 - 265.

Lin, J., Keogh, E., Wei, L., & Lonardi, S. (1 de Outubro de 2007). "Experiencing SAX: a novel symbolic representation of time series". *Data Mining and Knowledge Discovery*, pp. 107-144.

Monterey Bay Aquarium Research Institute. (01 de Setembro de 2001). Acesso em 16 de Julho de 2010, disponível em MUSE Project: http://www.mbari.org/MUSE/platforms/ships.htm

Pacific Northwest Research Station. (2005). *US Forest Service*. Acesso em 16 de Julho de 2010, disponível em The Starkey Project: http://www.fs.fed.us/pnw/starkey/data/tables/

Palma, A. T., Bogorny, V., Kuijpers, B., & Alvares, L. O. (2008). "A Clustering-based Approach for Discovering Interesting Places in Trajectories". *Proceedings of 23rd Annual Symposium on Applied Computing*, (pp. 863-868). Fortaleza, Ceara, Brasil.

Panagiotakis, C., Pelekis, N., & Kopanakis, I. (2009). "Trajectory Voting and Classification based on Spatiotemporal Similarity in Moving Object Databases". *Proceedings of the 8th International Symposium on Intelligent Data Analysis: Advances in Intelligent Data Analysis VIII* (pp. 131-142). Lyon, France: Springer.

Tan, P.-N., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining.* Boston: Addison-Wesley.

Tiakas, E., Papadopoulos, A. N., Nanopoulos, A., Manolopoulos, Y. P., Stojanovic, D., & Kajan, S. D. (2009). "Searching for similar trajectories in spatial networks". *Journal of Systems and Software*, 772-788.

Tiesyte, D., & Jensen, C. S. (2008). "Similarity-based prediction of travel times for vehicles traveling on known routes". *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems* (pp. 105-114). Irvine, CA, EUA: ACM.

Unisys. (2009). *Atlantic Tropical Storm Tracking by Year*. Acesso em 16 de Julho de 2010, disponível em Unisys Weather: http://weather.unisys.com/hurricane/atlantic/

Zheng, Y., Liu, L., Wang, L., & Xie, X. (2008). "Learning Transportation Mode from Raw GPS Data for Geographic Applications on the Web". *Proceeding of the 17th international conference on World Wide Web* (pp. 247-256). Pequim, China: ACM.

Zhou, C., Bhatnagar, N., Shekhar, S., & Terveen, L. (2007). "Mining Personally Important Places from GPS Tracks". *IEEE 23rd International Conference on Data Engineering Workshop* (pp. 517-526). Istambul, Turquia: IEEE Computer Society.

Usando Redes de Petri e Resolvedores ISCAS para Tratar Planejamento como Satisfatibilidade

Razer Montaño[1], **Marcos Castilho**[1], **Fabiano Silva**[1], **Luis Künzle**[1]

[1]Departamento de Informática – Universidade Federal do Paraná (UFPR)
Centro Politécnico – Jardim das Américas
Caixa Postal 19.081 – 81.531-980 – Curitiba – PR – Brazil

{razer, fabiano, marcos, kunzle}@inf.ufpr.br

Abstract. *This work presents an approach to solve classical planning problems in AI as a non-clausal satisfiability problem. An SAT instance in ISCAS format is generated by translating planning problem based in a Petri Net, in which submarking-reachability problem must be solved. So, the SAT instance is solved by well-known non-clausal SAT solvers.*

Resumo. *Este trabalho apresenta uma abordagem para resolver o problema de planejamento clássico em IA como um problema de satisfatibilidade não clausal. Uma instância SAT em formato ISCAS é gerada pela conversão do problema de planejamento tendo como base uma rede de Petri na qual o problema de alcançabilidade de sub-marcação deve ser resolvido. A instância ISCAS é então resolvida usando-se resolvedores SAT não-clausais conhecidos.*

1. Introdução

Planejamento é uma subárea do Raciocínio sobre Ações. Um problema de planejamento clássico é dado por um estado inicial, um estado objetivo e um conjunto de possíveis ações a serem tomadas. Uma solução para esse problema é uma sequência de ações que, ao serem executadas, transformam o estado inicial no estado objetivo. Os programas que resolvem esses problemas e encontram esse conjunto de ações são os planejadores.

Em [Silva et al. 2000] foi mostrado o relacionamento entre planejamento e alcançabilidade em Redes de Petri (RdP). O algoritmo PETRIPLAN possibilita que um grafo de planos possa ser transformado em uma RdP ordinária e limitada [Murata 1989]. A vantagem no uso de RdP é que a própria dinâmica da rede, sem o uso de estruturas auxiliares, representa pré-condições e efeitos da execução de uma ação, bem como conflitos (também conhecidos como *mutex*). Entretanto, alcançabilidade em RdP é um processo exponencial [Esparza 1996], para redes que possuem conflitos. Já para redes acíclicas e livres de conflitos, este processo é polinomial.

Nesta abordagem são usadas as RdP pelas vantagens estruturais desta representação. Os conflitos são representados na própria estrutura, a dinâmica da rede já representa efeitos e pré-condições de ações, há a possibilidade de se tratar problemas temporais, entre outros.

Existem muitas abordagens para o tratamento do problema de planejamento. O SATPLAN [Kautz and Selman 1992] trata planejamento como um problema de satisfatibilidade. O GRAPHPLAN [Blum and Furst 1995] usa um grafo construído a partir de

uma descrição STRIPS [Fikes and Nilsson 1971] para diminuir a representação do espaço de busca do problema, conhecido como *Grafo de Planos*.

O HSP [Bonet and Geffner 2001] é um planejador baseado em funções heurísticas que guiam um processo de subida de encosta. Usando-se esta mesma estratégia, tem-se o FF [Hoffmann and Nebel 2001] e o LPG [Gerevini et al. 2003].

Em 1985, Drummond [Drummond 1985] sugere pela primeira vez o uso de RdP no tratamento de problemas de planejamento. Em 2000 surge o PETRIPLAN [Silva et al. 2000], que trata planejamento como alcançabilidade. Alcançabilidade em RdP também é tratada através da técnica de desdobramento[1] [Hickmott et al. 2007].

O objetivo deste trabalho é mostrar uma nova técnica para se resolver o problema do planejamento representado como uma RdP usando SAT. Um problema em PDDL [Mcdermott et al. 1998] é lido e representado como uma RdP acíclica e esta rede é expandida e transformada em uma instância SAT. Em [Montaño et al. 2007], esta instância era uma fórmula proposicional na NNF (*Negation Normal Form*), a qual era transformada para o formado FNNF (*Factored Negation Normal Form*), através de um tableaux KE. Esta fórmula resultante possui propriedades que permitem se efetuar a verificação SAT em tempo polinomial.

O principal ponto que gera problemas de desempenho nesta técnica é a transformação de NNF para FNNF. Neste trabalho, para evitar esta transformação, a instância SAT é gerada em formato ISCAS e resolvida diretamente pelo NFL-SAT [Jain and Clarke 2009], um resolvedor SAT escrito para verificação de circuitos eletrônicos de processadores.

Planejamento e alcançabilidade em RdP são problemas equivalentes [Silva et al. 2000]. RdP possui uma dinâmica que é aproveitada para enriquecer a representação do problema, evitando estruturas auxiliares para representar e propagar conflitos. Esta é a principal diferença nesta abordagem com o BLACKBOX [Kautz and Selman 1999] e SATPLAN [Kautz and Selman 1992], que precisam de passos de propagação de conflitos. Além disso, RdP fornecem um mesmo formalimo para o tratamento de planejamento clássico e planejamento com recursos ou informações temporais.

A abordagem não-clausal para se resolver SAT é ideal neste trabalho, pois a RdP resultante do problema de planejamento está naturalmente na NNF. Uma transformação para CNF (formato comum de entrada dos resolvedores SAT) acrescentaria mais um passo computacionalmente caro de transformação e poderia destruir a estrutura original da fórmula, informação que pode ser utilizada para otimizar o processo.

Na seção 2 descreve-se brevemente a teoria de redes de Petri. Na seção 3 trata-se dos métodos para SAT não-clausal, enquanto que na seção 4 estes resultados são aplicados ao problema de alcançabilidade em RdP, cuja solução nos dá, pela equivalência dos problemas, a solução do problema de planejamento original.

[1] Do inglês *unfolding*

2. Redes de Petri

Rede de Petri (RdP) é uma ferramenta de modelagem com uma representação gráfica e com uma sólida base matemática de análise de propriedades estruturais e comportamentais dos modelos nela construídos. Seu campo de aplicação são os sistemas dinâmicos a eventos discretos, ou seja, que possuem características de paralelismo, concorrência, não determinismo e assincronia [Murata 1989].

Graficamente, uma rede de Petri é representada por um grafo direcionado bipartido constituído por lugares e transições representados, respectivamente, por círculos e retângulos. Arcos, representados por setas, associam lugares a transições e transições a lugares (pré e pós-condições da ocorrência de eventos), enquanto que marcas (caracterizadas por um ponto preto) representam o estado atual da rede.

Definição 1 (Redes de Petri) *Uma rede de Petri* $\mathtt{RdP}$ *é uma tupla* $\langle \mathtt{P}, \mathtt{T}, \mathtt{F}, \mathtt{M_0} \rangle$ *onde* $\mathtt{P}$ *é um conjunto finito não-vazio de lugares,* $\mathtt{T}$ *é um conjunto finito não-vazio de transições,* $\mathtt{F}$ *é uma dupla* $\langle \mathtt{Pre} : (\mathtt{P} \times \mathtt{T}) \rightarrow \mathbb{N}, \mathtt{Pos} : (\mathtt{P} \times \mathtt{T}) \rightarrow \mathbb{N} \rangle$ *que representa a incidência dos arcos que ligam lugares às transições e transições à lugares, respectivamente, e* $\mathtt{M_0} : \mathtt{P} \rightarrow \mathbb{N}$ *é a marcação inicial da rede.*

O comportamento dinâmico da rede é dado pelo disparo das transições, mas para isso é necessário que a transição esteja habilitada. A condição de habilitação de uma transição $\mathtt{t}$ é dada por $\forall \mathtt{p} \in \mathtt{P}, \mathtt{M(p)} \geq \mathtt{Pre(p,t)}$, onde $\mathtt{M(p)}$ define o número de marcas em um lugar $\mathtt{p} \in \mathtt{P}$. Com a transição habilitada, então o disparo pode ocorrer. Assim, os efeitos do disparo da transição $\mathtt{t}$ é o consumo de $\mathtt{Pre(p,t)}$ marcas de cada lugar pertencente ao seu pré-conjunto e colocar $\mathtt{Pos(p,t)}$ marcas em cada lugar do seu pós-conjunto, levando a rede a uma nova marcação, ou seja, $\mathtt{M'} = \mathtt{M} - \mathtt{Pre(p,t)} + \mathtt{Pos(p,t)}$.

Quando uma sequência de transições $\mathtt{s} = \{\mathtt{t_1}, \mathtt{t_2}, ..., \mathtt{t_n}\}$ é disparada levando a rede de uma marcação $\mathtt{M_0}$ a uma nova marcação $\mathtt{M'}$, então é dito que $\mathtt{M'}$ é alcançável a partir de $\mathtt{M_0}$. O conjunto de marcações alcançáveis ($\mathtt{RM}$) são todos os estados que uma dada rede de Petri pode alcançar a partir do disparo de alguma sequência de transições.

Definição 2 (Problema de Alcançabilidade) *Um problema de alcançabilidade consiste em verificar se existe uma sequência de disparos de transições tal que a partir da marcação inicial* $\mathtt{M_0}$ *se possa alcançar uma dada marcação* $\mathtt{M}$*.*

Cada lugar da rede também pode ter uma capacidade máxima $\mathtt{k}$ a ele associada. Isto significa que o número de marcas do lugar não poderá exceder sua capacidade e que isto deve ser respeitado para todas as marcações acessíveis da rede.

Definição 3 (Rede de Petri k-limitada) *Uma rede de Petri* $\mathtt{RdP} = (\mathtt{P}, \mathtt{T}, \mathtt{F}, \mathtt{M_0})$ *é dita k-limitada se, e somente se, para todas as marcações alcançáveis de* $\mathtt{RM}$*,* $\forall \mathtt{p} \in \mathtt{P}, \mathtt{M(p)} \leq \mathtt{k(p)}$*, onde* $\mathtt{k} \in \mathbb{N}$ *é o número máximo de marcações de um lugar. As redes 1-limitadas também são chamadas de redes seguras.*

O PETRIPLAN [Silva et al. 2000] é o algoritmo de planejamento que recebe como entrada um problema de planejamento em PDDL e gera uma RdP Lugar/Transição, acíclica, onde cada transição pode disparar uma única vez. A geração é similar à geração do Grafo de Planos [Blum and Furst 1995], exceto pelo fato de que a própria dinâmica da rede já representa os conflitos e suas propagações. Resolver alcançabilidade nesta rede é o mesmo que resolver o problema de planejamento subjacente.

O estado inicial é representado por marcações efetuadas na rede. O estado objetivo são lugares onde as marcas devem estar após o disparo de transições. Os conflitos são lugares com uma marca, que pode ser consumida por, neste caso, no máximo duas transições. Portanto, o objetivo é encontrar uma sequência de disparos na qual a marcação resultante contenha o estado objetivo.

O principal problema de uma rede como esta são os conflitos. Se nesta rede não houvessem conflitos, um simples caminhamento pelos lugares e transições resolveria alcançabilidade. Portanto, o objetivo principal nesta rede é a eliminação dos conflitos. Quando uma sequência consistente de disparos é encontrada, é convertida para a representação de um plano (sequência de ações), resolvendo assim o problema de planejamento.

3. SAT Não-Clausal

SAT é o problema de encontrar uma valoração para uma determinada fórmula lógica que a torne $Verdadeira$. A grande maioria das aplicações usa fórmulas em CNF, em virtude de otimizações que podem ser efetuadas no tratamento desse tipo de fórmula. Entretanto, muitos problemas, quando analisados como satisfatibilidade, geram fórmulas que não estão em CNF, sendo necessária uma conversão prévia para que se trate o problema como SAT tradicional.

Converter uma fórmula qualquer para CNF se faz de duas formas: pela aplicação sucessiva de operações distributivas ou pela transformação de Tseitin [Tseitin 1968].

A aplicação de operações de distribuição tende a aumentar o tamanho da fórmula de forma exponencial, tornando inviável sua aplicação.

A transformação de Tseitin baseia-se em converter uma igualdade, entre a nova variável e um trecho da fórmula, em uma dupla implicação que, de forma direta, é transformada para seu equivalente na CNF. A transformação é feita em tempo linear em relação ao tamanho da fórmula. O problema deste método é que a estrutura original da fórmula, que poderia ser aproveitada pelo resolvedor, é perdida.

Assim, o processo de conversão para CNF e aplicação de um resolvedor SAT para CNF ou eleva drasticamente o tempo de execução e consumo de memória do planejador, ou perde informações estruturais da fórmula original, tornando interessante o estudo de resolvedores SAT para fórmulas não-clausais [Thiffault et al. 2004].

Uma abordagem para resolver esse problema foi proposta por Montaño [Montaño et al. 2007], na qual a RdP era transformada para uma fórmula proposicional, baseada nos conflitos da rede. Esta fórmula estava na NNF e era convertida para d-DNNF [Darwiche and Marquis 2002], usando-se um tableau KE. Qualquer modelo desta fórmula era um modelo para o problema de planejamento subjacente.

O problema desta abordagem é que, apesar da busca por um modelo de uma fórmula na d-DNNF ser polinomial, transformar uma fórmula da NNF para d-DNNF é P-SPACE [R. Hähnle and N. Murray and E. Rosenthal 2005]. Assim sendo, o melhor caminho a ser tomado é evitar esta transformação, usando-se resolvedores SAT diretamente em fórmulas na NNF.

O formato ISCAS é comumente usado para representar circuitos de processado-

res para validação. As fórmulas lógicas usadas normalmente contém muitas variáveis e muitos operadores, e qualquer conversão para d-DNNF ou CNF seria inviável, em termos de recursos computacionais, e poderia desfazer a estrutura inicial da fórmula [Thiffault et al. 2004].

O NFLSAT [Jain and Clarke 2009] é um resolvedor SAT para fórmulas não-clausais, usando uma variação do algoritmo DPLL. Este resolvedor transforma a fórmula para uma estrutura em árvore, com as negações somente nas folhas (NNF). Esta representação interna favorece a aplicação de modernas técnicas como: observação de literais, retrocesso não-cronológico e aprendizado de cláusulas.

4. Alcançabilidade em Redes de Petri via SAT para Formas Não-Clausais

O algoritmo PETRIPLAN [Silva et al. 2000] gera uma rede de Petri Lugar/Transição acíclica, muito similar ao grafo de planos. Nesta rede, cada transição pode disparar no máximo uma vez e todos os lugares possuem no máximo dois arcos de saída. A situação inicial é dada por marcas em lugares que representam os estados iniciais e nos conflitos. A situação objetivo são lugares onde estas marcas deverão estar após a execução das ações. A execução de ações é dada pelo disparo de transições e escolha de conflitos.

A principal diferença entre o grafo de planos e a rede de Petri está no fato dos conflitos serem representados como meta-informações no grafo, enquanto que na rede de Petri eles são parte da rede. Ainda, o grafo de planos é uma estrutura estática e os métodos de busca devem percorrer a estrutura até encontrar um caminho livre de conflitos. Na rede de Petri existe uma dinâmica associada, fato que permite reduzir sensivelmente o número de conflitos na rede em relação ao grafo de planos.

A rede obtida pelo PETRIPLAN representa o comportamento dinâmico da aplicação de ações, partindo do estado inicial em busca de um estado final. Resolver planejamento usando esta RdP é encontrar todas as transições a serem disparadas, na ordem correta, para que o estado objetivo seja encontrado a partir do estado inicial. Esta sequência de ações é chamada *plano*.

Na busca pelo estado final encontram-se os conflitos. Em uma rede sem conflitos, o estado final é obtido, se for possível obtê-lo, de forma direta pelo caminhamento nesta rede. A presença de conflitos indica que existem transições que não podem ser disparadas em conjunto para a obtenção do mesmo plano, portanto são mutamente exclusivas.

Isto posto, se os conflitos da RdP forem eliminados de forma consistente, um simples caminhamento da rede é necessário para a obtenção de um plano, se este existir. Portanto, o objetivo aqui é eliminar esses conflitos.

O procedimento proposto pode ser visualizado através do Algoritmo 1.

4.1. Método original

A abordagem original deste trabalho [Montaño et al. 2007] baseia-se no fato de que o problema de satisfatibilidade em uma fórmula na *Deterministic Decomposable Negation Normal Form* (d-DNNF) é polinomial [Darwiche and Marquis 2002]. Também são polinomiais a contagem e enumeração de modelos. Assim sendo, foi proposto um pro-

Algoritmo 1 Planejador

```
Carrega o PDDL
repeat
  Expandir a rede até que os objetivos sejam encontrados consistentemente
  Gerar a fórmula no formato ISCAS
  Aplicar resolvedor NFLSAT
  if NFLSAT encontrou valoração para a fórmula then
    Constrói o plano a partir da valoração
  end if
until encontrar plano
```

cesso de conversão da fórmula lógica obtida da RdP para d-DNNF, baseado em Tableau KE.

Fórmulas na d-DNNF (decomponíveis e determinísticas) são um caso particular de fórmulas na *Negation Normal Form* (NNF). Uma sentença está na NNF se é construída a partir de literais, usando-se somente disjunções e conjunções.

Definição 4 (Decomponibilidade) *Dada uma fórmula na NNF, para cada conjunção $\alpha_1 \wedge \ldots \wedge \alpha_n$, se uma variável não aparece em mais de um termo α_i, isto é, os átomos encontrados em um termo desta conjunção não são os mesmos encontrados em outros termos, então ela é decomponível.*

Uma fórmula decomponível permite que certos tipos de problemas computacionais sejam decompostos em problemas menores, dado que suas subsentenças não compartilham átomos. Sempre que um nodo and for encontrado, cada um de seus ramos terá, seguramente, informações diferentes, podendo ser computado de forma independente.

Definição 5 (Determinismo) *Dado uma fórmula na NNF, se para toda disjunção $\vee_i \alpha_i$, todo par de termos α_i e α_j, onde $i \neq j$, $\alpha_i \wedge \alpha_j$ é uma contradição, isto é, são mutuamente exclusivos, então a fórmula é determinística.*

Decomponibilidade é a propriedade que torna o teste de satisfatibilidade computacionalmente tratável. É a propriedade de determinismo que torna a contagem de modelos e sua enumeração tratável.

Para se transformar uma teoria proposicional em d-DNNF deve-se aplicar, recursivamente, uma operação chamada *condicionamento*[2] [Palacios et al. 2005]. Esta operação é derivada da identidade conhecida como expansão de Shannon dada por:

$$\mathcal{F} = (\mathcal{F}[V/p] \wedge p) \vee (\mathcal{F}[F/\neg p] \wedge \neg p)$$

onde $\mathcal{F}[x/p]$ indica a troca de todas as ocorrências de p em $\mathcal{F}$ por x.

Uma fórmula na *Factored Negation Normal Form* (FNNF) também está na d-DNNF, mas regras de identidade, simplificações, são aplicadas à fórmula para geração de fórmulas menores e equivalentes.

[2]Do inglês *conditioning*

O tableau KE [D'Agostino and Mondadori 1994] é um procedimento de prova baseado em tableau, mas que inclui a regra de *cut*, que classicamente não é considerada. Conforme mostrado em [R. Hähnle and N. Murray and E. Rosenthal 2005] o procedimento de Tableau KE, pode ser usado para a geração da FNNF.

No procedimento de transformação para FNNF somente são usadas a regra *cut*, regras α e regras de transformação de tableau. As regras β são substituídas por um mecanismo de propagação de restrições, conhecido como *simplificação* [Massacci 1998]. Este mecanismo tem o mesmo propósito que a subjugação para a resolução e que a regra de cláusulas unitárias para o procedimento de Davis-Putnam.

O tableau resultante tem literais em todos os nodos ou a árvore inteira será reduzida a $Falso$ ou $Verdadeiro$. Esse tableau é chamado *Tableau FNNF saturado*.

Aqui, não há necessidade de se obter um tableau saturado pois qualquer ramo saturado desse tableau já é um modelo para a fórmula que está sendo processada. Este processo de parada caracteriza um o procedimento como um SAT não-clausal, sendo que do ramo saturado do Tableau KE se consegue extrair um modelo para a fórmula em NNF original. Esta valoração é transformada em cortes na RdP original, sendo que esses cortes incidem sobre os conflitos, já que a fórmula é formada por informações provenientes destes. Esses cortes resolvem todos os conflitos da rede e, a partir deste ponto, pode-se aplicar um procedimentos de busca para encontrar o plano de forma eficiente, pois não será efetuado nenhum *backtracking*.

Apesar de SAT ser polinomial em uma fórmula na FNNF, a conversão de NNF para FNNF é P-SPACE [R. Hähnle and N. Murray and E. Rosenthal 2005], o que degrada todo o processo. Isto posto, deve-se obter um processo que não exiga esta transformação e no qual se possa fazer SAT em uma fórmula não-clausal de maneira eficiente.

Analisando-se o formato não-clausal ISCAS, percebeu-se que a RdP poderia ser facilmente convertida para esta representação. Ademais, as fórmulas geradas seriam menores, pois o uso e reaproveitamento de operadores neste formato pode ser usado para representar lugares ou transições que são visitados várias vezes, uma vez só.

Assim, o processo apresentado aqui, propõem que a RdP seja transformada diretamente para o formato ISCAS e que neste resultado um resolvedor seja usado. Diferentemente da proposta original, o passo de transformação para FNNF pode ser removido, já que existem resolvedores ISCAS eficientes (como o NFLSAT).

4.2. Geração de fórmulas ISCAS a partir de uma Rede de Petri

A partir da RdP original, partindo de cada lugar pertencente ao estado objetivo, varre-se a rede até os lugares iniciais construindo uma fórmula lógica onde seus átomos representam unicamente os conflitos. A retirada dos conflitos da rede se dá a partir de um resolvedor SAT para fórmulas não-clausais, que dará valores aos literais dessa fórmula, neste caso o NFLSAT. Na estrutura da RdP, esta valoração indica a eliminação dos conflitos através da imposição de uma escolha.

Nas RdP geradas pelo PETRIPLAN usado nas implementações, os conflitos são sempre binários (um lugar possuindo no máximo dois arcos de saída), portanto pode-se usar uma variável proposicional para representar cada conflito, e estas são as entradas (INPUT) da fórmula. Por exemplo, na Figura 1 $\neg p$ indica o disparo da transição t_1 e p

indica o disparo da transição t_2.

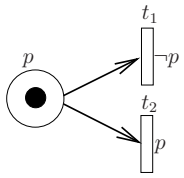


Figura 1. Conflito com a proposição p associada

Analisando-se a estrutura da RdP, percebe-se que os lugares pertencentes ao estado final podem ser descritos como uma fórmula baseada nas variáveis proposicionais associadas aos conflitos. Caso cada um dos lugares que chegam a t tenham fórmulas associadas ($\Delta_1, ..., \Delta_n$), como mostra a Figura 2, a fórmula associada a t é $AND(\Delta_1, ..., \Delta_n)$. Caso cada uma das transições que chegam a p tenham fórmulas associadas ($\Delta_1, ..., \Delta_n$), como mostra a Figura 3, a fórmula associada a p é $OR(\Delta_1, ..., \Delta_n)$.

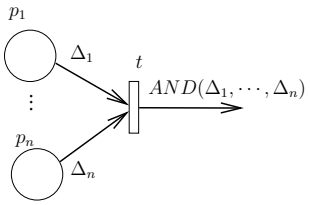


Figura 2. Conjunção na RdP com fórmula associada

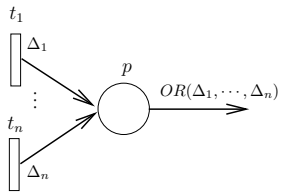


Figura 3. Disjunção na RdP com fórmula associada

Seja M_o a marcação final e M_i a marcação inicial da RdP. Seja $\theta(V)$ a quantidade de elementos de V, onde V é um vetor, diferentes de zero e $\mu(t)$, definido sobre uma transição t, indicando se t é uma transição de um conflito, isto é, $\mu(t)$ é V se $\exists p,\ Pre(p,t) > 0\ \wedge\ \theta(Pre(p,.)) > 1$ e F caso contrário. Assume-se que existe uma ordenação qualquer entre as transições, que pode ser a ordem de inserção na fórmula.

A fórmula $\mathcal{F}$ que representa os conflitos de uma RdP é dada por:

$$\mathcal{F} \quad = \bigwedge_i L(p_i),\ p_i \in M_o$$

$$L(p) = \begin{cases} \bigvee_i T(t_i);\ \forall t_i,\ Pos(p,\ t_i)\ >\ 0 & se\ \theta(Pos(p,\ .))\ \geq\ 1 \\ V & se\ \theta(Pos(p,\ .))\ =\ 0\ e\ p\ \in\ M_i \\ F & se\ \theta(Pos(p,\ .))\ =\ 0\ e\ p\ \notin\ M_i \end{cases}$$

$$T(t) = \begin{cases} \bigwedge_i \alpha(p_i,\ t);\ \forall p_i,\ Pre(p_i,\ t)\ >\ 0 & se\ \theta(Pre(.,\ t))\ \geq\ 1 \\ F & se\ \theta(Pre(.,\ t))\ =\ 0 \end{cases}$$

$$\alpha(p,\ t) = \begin{cases} L(p) & se\ \neg\mu(t) \\ \beta(p,t)\ \wedge L(p) & se\ \mu(t) \end{cases}$$

$$\beta(p,\ t) = \begin{cases} p & se\ \exists t', Pre(p,\ t')\ >\ 0\ e\ t\ >\ t' \\ \neg p & se\ \exists t',\ Pre(p,\ t')\ >\ 0\ e\ t\ <\ t' \end{cases}$$

Baseado na estrutura de geração da fórmula, os únicos conectivos presentes são conjunções e disjunções. Ademais, as negações possuem escopo limitado às variáveis ligadas aos conflitos, portanto a fórmula gerada está na NNF.

Como esta fórmula possui somente literais atrelados a conflitos, e é gerada a partir da interpretação dos lugares e transições, qualquer modelo (valoração que a satisfaça) pode ser interpretado como uma configuração consistente dos conflitos da rede original, eliminando-os, permitindo assim disparos em transições para a obtenção do estado final. Isto indica que existe um plano para o problema de planejamento subjacente. No caso da fórmula ser insatisfatível, conclui-se que não há configuração consistente para os conflitos, de forma que se chegue ao estado final, indicando que não existe plano para o problema de planejamento associado.

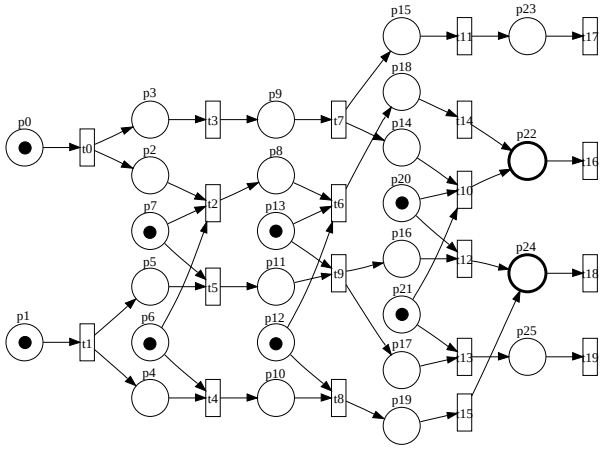


Figura 4. Problema completo com marcações

Como exemplo, a Figura 4 mostra uma RdP com uma marcação inicial (círculos pretos nos lugares) e com uma marcação final (lugares com contornos mais fortes). Nesta rede, deve-se extrair uma fórmula lógica para $p22 \wedge p24$, através de um caminhamento reverso partindo dos objetivos até o estado inicial. A fórmula obtida é:

```
INPUT(p6)
INPUT(p7)
INPUT(p12)
INPUT(p13)
INPUT(p20)
INPUT(p21)
OUTPUT(out)
not_p6 = NOT(p6)
not_p7 = NOT(p7)
not_p12 = NOT(p12)
not_p13 = NOT(p13)
not_p20 = NOT(p20)
not_p21 = NOT(p21)
aux1 = AND(not_p7, not_p6,
           not_p12, not_p13)
aux2 = AND(not_p20, not_p21)
p22 = OR(aux1, aux2)
aux3 = AND(p6, p12)
aux4 = AND(p7, p13, p20)
p24 = OR(aux3, aux4)
out = AND(p22, p24)
```

Aplicando-se o NFLSAT, pode-se obter um modelo, como por exemplo: $\{p6, p7, p12, \neg p20, \neg p21\}$. Como este modelo representa escolhas de conflito, a RdP pode ser cortada, eliminando-se o que não faz parte da escolha. Obtém-se então a rede da Figura 5, na qual um caminhamento simples retorna o plano do problema de planejamento subjacente.

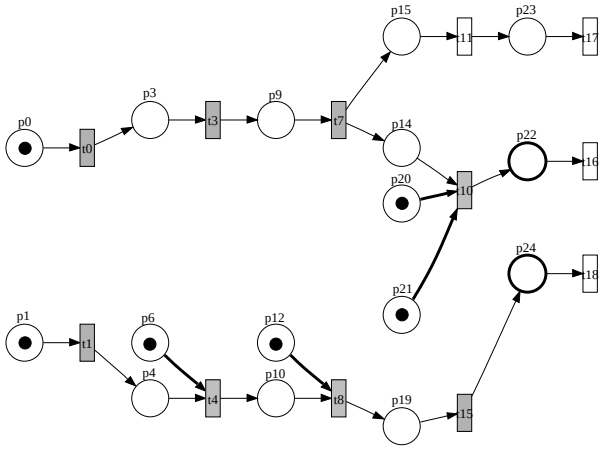


Figura 5. Rede de Petri cortada

5. Experimentos

A implementação efetuada baseia-se no algoritmo PETRIPLAN [Silva et al. 2000] para leitura e representação de um problema de planejamento em PDDL. Com a RdP carregada, o mesmo processo de geração de fórmulas mostrado em Montaño [Montaño et al. 2007] é efetuado, mas aqui o formato gerado é ISCAS, conforme mostrado na seção anterior.

Com a fórmula obtida, o resolvedor NFLSAT é chamado para resolver os conflitos. Se uma solução for obtida, ela é transformada em cortes na rede para que um processo de caminhamento, logo após, determine o plano. Caso uma solução não seja obtida, então uma nova expansão na rede é feita e o processo todo se inicia.

A Tabela 1 mostra os tempos obtidos para a resolução de alguns problemas usando-se a abordagem original de Montaño, SATPLAN e a nova técnica (RDP_ISCAS). Conforme pode-se observar, a abordagem original na qual a fórmula é convertida para d-DNNF não é viável, visto que quase todos os problemas geram estouro no uso de recursos da máquina.

Tabela 1. Comparação de Tempos SATPLAN e RDP_ISCAS (em segundos). MEM indica que o procedimento esgotou recursos de memória antes de terminar. * indica que o problema experimentado não era tipado.

Problema	Expansões	Original	SATPLAN	RDP_ISCAS
Sussman	3	0.04	0	0.01
Blocks-04	3	MEM	0.01	0.088
Blocks-05	6	MEM	0.02	0.317
Blocks-06	6	MEM	0.03	0.97
Blocks-10-2	15	MEM	0.37	78.149
Gripper 2g 2b	3	0.05	0.01	0.01
Gripper 2g 4b	7	MEM	0.04	0.031
Gripper 2g 6b	11	MEM	0.43	5.57
Gripper 2g 8b	15	MEM	19.09	130.95
Logistics 1	9	MEM	0.04*	34.709

Os resultados são preliminares mas promissores, visto que o SATPLAN usa uma

das mais eficientes técnicas para planejamento[3]. Apesar de, na média, os tempos serem maiores do que o SATPLAN, percebe-se uma aproximação de resultados, indicando que a implementação de técnicas de otimização podem levar a resultados melhores.

Para melhorar o resolvedor SAT pode-se implementar novas técnicas e considerar informações estruturais da fórmulas ISCAS, visto que há um padrão de geração da fórmula. Para melhorar a codificação do problema, deve-se usar um outro substrato teórico que, ou que diminua o número de operadores da fórmula, ou produza fórmulas com uma estrutura mais alinhada com as técnicas do resolvedor SAT.

6. Considerações

Ao contrário do BLACKBOX, que usa um grafo de planos como representação interna do problema de planejamento, aqui é usado uma rede de Petri. Como vantagens tem-se: a dispensa no uso de estruturas auxiliares para representar conflitos e a estrutura da rede servindo como mecanismo de propagação de ações, pré-condições e efeitos.

Percebe-se com este trabalho que a técnica de representação de problemas de planejamento como redes de Petri e seu posterior processamento com resolvedores SAT para NNF está indo ao encontro dos resultados obtidos pelo SATPLAN, com a vantagem de se ter um modelo que naturalmente representa o problema.

Os resultados são preliminares, mas dois trabalhos estão sendo desenvolvidos em paralelo, que terão implicação direta nesses resultados: melhoramento no resolvedor NFLSAT e melhoramento no processo de geração das fórmulas ISCAS, usando o novo algoritmo PETRIPLAN, a Rede de Planos [Silva 2005].

Referências

Blum, A. and Furst, M. (1995). Fast planning through planning graph analysis. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI 95)*, pages 1636–1642.

Bonet, B. and Geffner, H. (2001). Heuristic search planner 2.0. *AI Magazine*, 22(3):77–80.

D'Agostino, M. and Mondadori, M. (1994). The taming of the cut: Classical refutations with analytic cut. *Journal of Logic and Computation*, 4(3):285–319.

Darwiche, A. and Marquis, P. (2002). A Knowledge Compilation Map. *Journal of Artificial Intelligence Research*, 17:229–264.

Drummond, M. E. (1985). Refining and extending the procedural net. In *Proceedings of the 9th international joint conference on Artificial intelligence - Volume 2*, pages 1010–1011, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Esparza, J. (1996). Decidability and complexity of petri net problems - an introduction. In *Petri Nets*, pages 374–428.

Fikes, R. and Nilsson, N. (1971). Strips: A new approach to the application of theorem proving to problem solving. Technical Report 43r, AI Center, SRI International, 333 Ravenswood Ave, Menlo Park, CA 94025. SRI Project 8259.

[3]SATPLAN venceu a competição de planejadores na quarta edição do IPC (*4th International Planning Competition*) e empatou em primeiro lugar em sua quinta edição

Gerevini, A., Saetti, A., and Serina, I. (2003). Planning through stochastic local search and temporal action graphs in lpg. *J. Artif. Int. Res.*, 20:239–290.

Hickmott, S., Rintanen, J., Thiébaux, S., and White, L. (2007). Planning via petri net unfolding. In *Proceedings of the 20th international joint conference on Artifical intelligence*, pages 1904–1911, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Hoffmann, J. and Nebel, B. (2001). The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, 14:253–302.

Jain, H. and Clarke, E. M. (2009). Efficient SAT solving for Non-Clausal formulas using DPLL, graphs, and watched cuts. In *46th Design Automation Conference (DAC)*.

Kautz, H. and Selman, B. (1999). Unifying SAT-based and graph-based planning. In Minker, J., editor, *Workshop on Logic-Based Artificial Intelligence, Washington, DC, June 14–16, 1999*, College Park, Maryland. Computer Science Department, University of Maryland.

Kautz, H. A. and Selman, B. (1992). Planning as satisfiability. In *Proceedings of the Tenth European Conference on Artificial Intelligence (ECAI'92)*, pages 359–363.

Massacci, F. (1998). Simplification: A general constraint propagation technique for propositional and modal tableaux. *Lecture Notes in Computer Science*, 1397:217–231.

Mcdermott, D., Ghallab, M., Howe, A., Knoblock, C., Ram, A., Veloso, M., Weld, D., and Wilkins, D. (1998). PDDL - the planning domain definition language. Technical report, CVC TR-98-003/DCS TR-1165, Yale Center for Computational Vision and Control.

Montaño, R., Silva, F., Castilho, M., and Kunzle, L. (2007). Planejamento como satisfatibilidade: uma abordagem não-clausal. In *Proceedings of the VI Encontro Nacional de Inteligência Artificial (ENIA)*.

Murata, T. (1989). Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580.

Palacios, H., Bonet, B., Darwiche, A., and Geffner, H. (2005). Pruning conformant plans by counting models on compiled d-DNNF representations. In *ICAPS*, pages 141–150.

R. Hähnle and N. Murray and E. Rosenthal (2005). Normal forms for knowledge compilation. *Lecture Notes in Computer Science*, 3488:304–313.

Silva, F. (2005). *Rede de Planos: Uma Proposta para a Solução de Problemas de Planejamento em Inteligência Artificial usando Redes de Petri*. PhD thesis, CEFET, Curitiba PR, Brasil.

Silva, F., Castilho, M., and Künzle, L. (2000). Petriplan: a new algorithm for plan generation (preliminary report). In *Lecture Notes in Artificial Intelligence*, volume 1952, pages 86–95. International Joint Conference IBERAMIA'2000 - SBIA'2000.

Thiffault, C., Bacchus, F., and Walsh, T. (2004). Solving non-clausal formulas with dpll search. In *In SAT*, pages 663–678. Springer.

Tseitin, G. (1968). On the complexity of derivation in propositional calculus. *Studies in constructive mathematics and mathematical logic*, 2(115-125):10–13.

ORION – Um Framework para Refinamento de Ontologias Através de Técnicas de Revisão de Teorias

Felipe Leão, Kate Revoredo, Fernanda Baião

NP2Tec – Núcleo de Pesquisa e Prática em Tecnologia
Departamento de Informática Aplicada
Universidade Federal do Estado do Rio de Janeiro (UNIRIO)
Av. Pasteur, 458 – Rio de Janeiro – RJ – Brazil

{felipe.leao, katerevoredo, fernanda.baiao}@uniriotec.br

Abstract. *Ontology have been frequently used for knowledge representation. As it evolves, however, keeping its consistency becomes a costly and complex task, if no support is provided. This work proposes ORION – Ontology Refinement through theory revisION, which applies existing Theory Revision techniques to support Ontology evolution. Preliminary tests were successful in obtaining a more accurate refined Ontology.*

Resumo. *A utilização de Ontologias para representação do conhecimento sobre um domínio requer um trabalho contínuo de manutenção, uma vez que a observação de novas instâncias pode trazer inconsistências às regras previamente definidas. No entanto, a modificação manual da Ontologia pode ser muito custosa e complexa. Este trabalho propõe o ORION, um framework para refinamento automático de Ontologias através do uso de técnicas de revisão de teorias. Testes preliminares mostraram a viabilidade da proposta e a obtenção de uma Ontologia refinada mais precisa.*

1.Introdução

Ontologias propiciam descrições concisas e desambíguas sobre conceitos e relacionamentos em um domínio de interesse. A utilidade de uma ontologia depende da precisão com a qual a estrutura do domínio, descrita pela ontologia, representa as instâncias deste domínio. Quando novas instâncias válidas são recebidas, as regras previamente especificadas na ontologia podem deixar de ser verdade. Quando isto acontece, assumindo-se que as novas instâncias sejam observações inequívocas da realidade, estas regras devem ser revisadas. Em cenários reais, onde o número de instâncias e regras podem se tornar muito grande (como na Web Semântica), detectar e corrigir manualmente estas inconsistências se torna impraticável. É interessante então ter um mecanismo para automaticamente refinar uma ontologia.

Por outro lado, a área de Programação em Lógica Indutiva (ILP) [Muggleton, 1992] utiliza técnicas de revisão de teoria [Wrobel, 1996] para aprimorar uma base de conhecimento de forma automática, visando reduzir os erros, a dificuldade e o tempo despendido na aquisição de conhecimento. Um exemplo de sistema de revisão de teorias é o FORTE (*First Order Revision of Theories from Examples*) [Richards e Mooney, 1995], que revisa bases de conhecimento descritas em cláusulas de primeira ordem Horn e livre de funções (i.e., programas em PROLOG livres de funções).

Motivados então pela possibilidade de um refinamento automático de ontologias, em [Leão et al., 2010] nós propomos utilizar o sistema FORTE para refinar

os axiomas de uma ontologia exemplo, descritos em cláusulas de primeira ordem Horn. Entretanto, uma ontologia pode estar representada em outras linguagens, como OWL, RDF, OBO, lógicas de descrição, etc. Dessa forma, neste artigo, nós propomos um framework para refinamento de ontologias. Neste framework uma ontologia (descrita através de linguagens como OWL, RDF, OBO etc.) é mapeada para uma teoria a ser revisada pelo FORTE. A teoria revisada é então convertida novamente para uma ontologia refinada representada na sua linguagem original.

O trabalho está organizado da seguinte forma. A seção 2 apresenta as definições de ontologia, técnicas de revisão de teoria e cita a aplicabilidade de técnicas de revisão de teoria no refinamento de ontologias. Na seção 3 é proposto o framework ORION. A seção 4 aborda trabalhos relacionados. Finalmente, a seção 5 conclui o trabalho.

2. Conhecimento Preliminar

2.1. Ontologia

Uma ontologia é uma especificação formal e explícita de uma conceitualização compartilhada [Gruber, 1995]. Ela define um vocabulário comum entre aqueles que compartilham informações de um mesmo domínio. Formalmente, uma ontologia é definida através dos conjuntos finitos $\mathcal{C}$, $\mathfrak{R}$ e $\mathfrak{I}$, onde $\mathcal{C}$ é um conjunto de conceitos, $\mathfrak{R}$ é um conjunto de relacionamentos binários, que relacionam dois conceitos de $\mathcal{C}$, sendo $\mathcal{C} \cap \mathfrak{R} = \emptyset$. $\mathfrak{I}$ é um conjunto de expressões em lógica de primeira ordem, descrevendo dependências entre elementos de $\mathfrak{R}$.

Em um domínio que represente uma família podemos dizer, por exemplo, que Professor e Pessoa são dois conceitos, relacionados entre si com a semântica de especialização ("É UM") evidenciando, dentre outras coisas, que uma instância do conceito Professor é também uma instância do conceito Pessoa. A expressão (regra) **pessoa(x) ← professor(x)** descreve a semântica do relacionamento "É UM" entre as instâncias dos conceitos Pessoa e Professor, pertencendo ao conjunto $\mathfrak{I}$.

Uma ontologia pode ser representada como um grafo direcionado G(V,E), onde V é um conjunto finito de vértices, e E um conjunto finito de arestas . Cada vértice em V é rotulado por um conceito de $\mathcal{C}$ e cada aresta em E é rotulada por um relacionamento em $\mathfrak{R}$. Alternativamente, uma ontologia pode ser representada através de modelos baseados em RDF [Klyne e Carroll, 2004] como na Figura 1, onde duas instâncias de Pessoa são descritas.

```
1  <?xml version="1.0"?>
2  <rdf:RDF
3  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
4  xmlns:fam="http://meu.site.com.br/RDF/universidade-ns#">
5
6      <rdf:Description rdf:about="http://meu.site.com.br/FelipeLeao">
7          <fam:primeironome>Felipe</fam:primeironome>
8          <fam:ultimonome>Leão</fam:ultimonome>
9      </rdf:Description>
10
11     <rdf:Description rdf:about="http://meu.site.com.br/KateRevoredo">
12         <fam:primeironome>Kate</fam:primeironome>
13         <fam:ultimonome>Revoredo</fam:ultimonome>
14     </rdf:Description>
15
16 </rdf:RDF>
```

Figura 1 - Instâncias de Pessoa representadas em sintaxe RDF

Entretanto, parte da ontologia, neste caso o conjunto $\Im$, não pode ser representada em RDF, dado que esta linguagem não provê mecanismos para especificar ou descrever a semântica dos relacionamentos entre os conceitos. Para superar estas deficiências, o RDF foi estendido para outros modelos como o RDF Schema[1] e OWL[2]. A Figura 2 ilustra a definição dos conceitos Pessoa, Professor e Aluno e os relaciona através da combinação dos modelos RDF Schema e OWL.

```
1  <?xml version="1.0"?>
2
3  <rdf:RDF
4  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
5  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
6  xmlns:owl="http://www.w3.org/2002/07/owl#"
7  xml:base="http://meu.site.com.br/universidade#">
8
9      <rdfs:Class rdf:ID="Pessoa" />
10
11     <rdfs:Class rdf:ID="Professor">
12         <rdfs:subClassOf rdf:resource="#Pessoa"/>
13         <owl:disjointWith rdf:resource="#Aluno"/>
14     </rdfs:Class>
15
16     <rdfs:Class rdf:ID="Aluno">
17         <rdfs:subClassOf rdf:resource="#Pessoa"/>
18     </rdfs:Class>
19
20 </rdf:RDF>
```

Figura 2 - Definição e relacionamento entre conceitos expressos pelas sintaxes RDF Schema e OWL

2.2. Revisão de Teoria

A aquisição de conhecimento em ILP é uma tarefa que consome tempo e suscetível a erros. Sistemas de revisão de teoria visam aperfeiçoar, de forma automática, bases de conhecimento, através de métodos de aprendizado de máquina [Wrobel, 1996].

A Figura 3 apresenta um esquema genérico para revisão de teoria. O sistema de revisão de teoria recebe uma teoria inicial e um conjunto consistente de exemplos. A teoria inicial é composta por um componente invariante, denominada *conhecimento preliminar* (CP) e uma componente que pode ser modificada (A). O conjunto de exemplos é dividido em exemplos positivos (E^+) e exemplos negativos (E^-). Os exemplos positivos são afirmações corretas, enquanto exemplos negativos são afirmações incorretas sobre o domínio.



Figura 3 - Esquema para Revisão de Teoria

O Sistema de Revisão de Teorias deverá gerar uma teoria final "minimamente revista" capaz de provar todos os exemplos positivos em E^+ e refutar todos os exemplos negativos em E^-. Desta forma, a teoria final torna-se consistente com a base de dados. Um exemplo de sistema de revisão de teorias é o FORTE [Richards e Mooney, 1995].

[1] http://www.w3.org/TR/rdf-schema/
[2] http://www.w3.org/TR/owl2-overview/

A qualidade das revisões produzidas depende crucialmente de quais cláusulas da teoria são escolhidas para serem modificadas e qual a modificação aplicada, definindo então um conjunto de **pontos de revisão**. Após especificar onde a teoria deve ser modificada, é preciso determinar como modificá-la, definindo então **operadores de revisão**. Exemplos de operadores de revisão são: exclusão de cláusula, adição de antecedente, exclusão de antecedente e adição de cláusula. Uma função de avaliação é utilizada para escolher a melhor proposta de revisão a ser implementada. Um possível algoritmo de revisão que utiliza uma abordagem gulosa de subida de encosta (*greedy hill-climbing*) é exibido na Figura 4.

Algorithm 1 Algoritmo de revisão de teorias de primeira-ordem

```
repita
   gerar pontos de revisão;
   para cada ponto de revisão
      gerar possíveis revisões;
      avaliar possíveis revisões de acordo com alguma métrica;
      atualizar melhor revisão possível;
   implementar melhor revisão;
até que nenhuma revisão melhore a teoria
```

Figura 4 - Algoritmo do Sistema de Revisão de Teorias

2.3. Refinamento de Ontologias Através de Revisão de Teoria

O Refinamento de ontologias através de técnicas de revisão de teoria foi proposto inicialmente em [Leão et. al., 2010], utilizando a ontologia da Família. O conjunto de axiomas da ontologia foi mapedo para cláusulas Horn de primeira ordem, definindo o conjunto CP e o conjunto A das entradas do sistema FORTE. Além disso, as instâncias da ontologia foram utilizadas como exemplos positivos e na geração de exemplos negativos. Após a revisão das cláusulas constantes em A pelo FORTE, estas são convertidas para a linguagem utilizada para descrever os axiomas. É importante observar que, devido ao seu poder de expressão, a lógica de primeira ordem (neste caso clausula de Horn) é capaz de expressar todo o conhecimento originalmente expresso em OWL e que as alterações propostas pelo FORTE, também são possíveis de serem representadas em OWL

A proposta se provou de grande utilidade ao gerar axiomas revisados, capazes de representar o domínio de estudo com maior acurácia. Entretanto, a variedade de linguagens de representação de ontologias ocasiona o aumento de complexidade no uso de sistemas de revisão de teoria para o refinamento de ontologias, dado que para cada linguagem um determinado procedimento de mapeamento deverá ser executado. Além disso, é necessário um conhecimento prévio de como executar o sistema de revisão.

3. ORION – Um Framework para Refinamento de Ontologias

Objetivando facilitar a utilização de sistemas de revisão de teoria no refinamento de ontologias, propõe-se criar o framework ORION (Ontology Refinement through RevisION techniques), capaz de receber uma ontologia e devolvê-la refinada, abstraindo as conversões necessárias e o conhecimento sobre o funcionamento do sistema de revisão de teoria. O framework converte automaticamente os dados de uma ontologia para o formato aceito como entrada pelo FORTE e o executa. Uma vez revisados os

axiomas, o conversor atua novamente, realizando o caminho inverso, convertendo a saída do sistema de revisão para uma nova ontologia. A ontologia resultante será um modelo aprimorado da ontologia inicial, possuindo um maior poder de expressividade. O esquema de funcionamento do framework ORION pode ser visto na Figura 5.

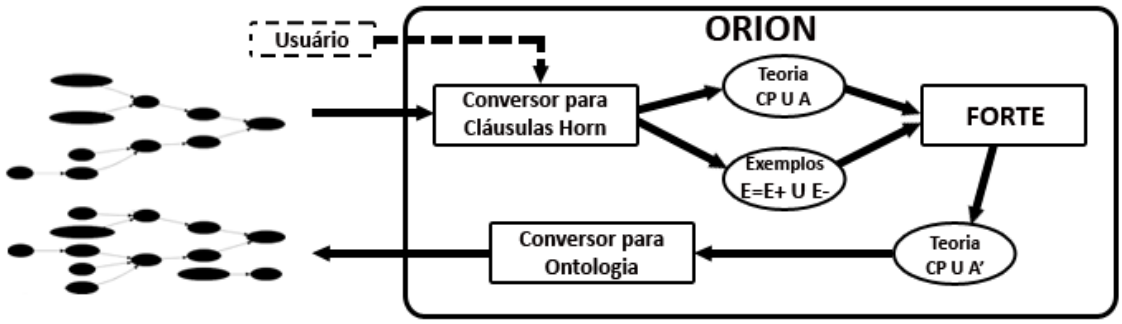


Figura 5 - Esquema de funcionamento do framework ORION

Dependendo do nível de complexidade do domínio retratado pela ontologia, o mapeamento terá de ocorrer de forma semi-automática, fazendo uso da interação com o usuário para estabelecer, por exemplo, axiomas base, que não devem ser revisados decidindo assim o que fará parte dos conjuntos CP e A.

Neste trabalho será considerada apenas a OWL como sintaxe para construção de ontologias, tanto por ser um padrão recomendado pelo W3C [Klyne e Carroll, 2004] e uma das linguagens mais utilizadas pela comunidade acadêmica quanto pela sua importância para a viabilidade da Web Semântica.

3.1. Mapeamento de uma Ontologia para o Modelo de Teoria do FORTE

Esta seção descreve como alguns elementos previstos na sintaxe da linguagem OWL são mapeados para elementos utilizados no modelo de teoria do sistema FORTE.

3.1.1. Propriedades de Objetos (Relacionamentos entre classes)

Object Properties são a representação dos relacionamentos entre conceitos em uma Ontologia. Para cada *Object Property*, é gerada uma cláusula para representar o relacionamento. No metamodelo da OWL, uma *Object Property* é descrita pelos atributos *About*, *Domain* e *Range*, que são mapeados, respectivamente, como o nome da cláusula, o seu primeiro termo e o seu segundo termo. Adicionalmente, os tipos do domínio (*domain*) e do escopo (*range*) do relacionamento são adicionados como antecedentes da cláusula, para restringir as interpretações possíveis das instâncias que podem estar relacionadas no conjunto de fatos. Um exemplo de conversão do *Object Property* **cursa** pode ser visto na Figura 6. Uma vez gerados, os relacionamentos são incluídos no conjunto CP ou no conjunto A, dependendo da indicação do especialista do domínio se tal cláusula deve estar sujeita (A) ou não (CP) à revisão.

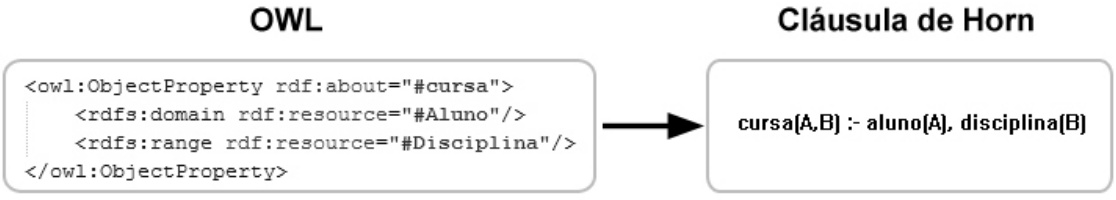


Figura 6 - Mapeamento de *Object Property* para uma Cláusula de Horn

3.1.2. Propriedades de Tipos de Dados (Atributos da classe)

Uma propriedade de tipo de dado (*Datatype Property*) representa, na interpretação mais comum da sintaxe OWL, os atributos de uma classe da Ontologia. O mapeamento de um *Datatype Property* ocorre de forma semelhante à conversão dos *Object Properties*, sendo utilizados principalmente os atributos *About* e *Domain*. A Figura 7 ilustra a conversão do *Datatype* **genero** da ontologia para o atributo que deverá ser inserido entre os *object_attributes* do FORTE.

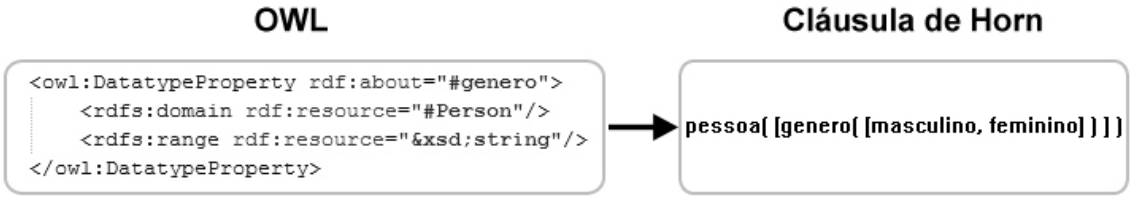


Figura 7 - Mapeamento de um *Datatype Property* para uma Cláusula de Horn

Vale ressaltar que em ontologias de domínio reais pode haver um número muito grande de atributos (*Datatype Properties*) para cada classe da Ontologia, o que acarretaria em um aumento significativo do tamanho da teoria a ser revista e consequentemente um aumento do custo de revisão, podendo tornar o processo inviável na prática e diminuir a acurácia da teoria revisada. No entanto, a relevância de cada atributo para o domínio em questão e para o processo de revisão é bastante variável, existindo atributos essenciais para a revisão (por exemplo, o atributo **gênero** de uma Pessoa quando se fala de um domínio médico, onde seria incorreto dizer que uma pessoa do gênero masculino pode estar grávida) e atributos que poderiam ser descartados do processo de revisão (por exemplo, o atributo email de uma Pessoa). Este problema é análogo ao problema de seleção de atributos na área de mineração de dados [Han e Kamber, 2006], para o qual existem várias técnicas e algoritmos propostos na literatura. Existe um consenso, no entanto, de que a decisão pelos atributos relevantes a serem considerados deve ser do especialista do domínio. Portanto, ao mapear as propriedades de tipos de dados, o ORION prevê interação com o usuário para definir quais delas, que de outra forma seriam ignoradas, devem ser incluídas.

Uma restrição do sistema FORTE é não permitir atributos contínuos, portanto é necessário informar os valores que o *object_attribute* poderá assumir. No caso exibido na Figura 7 foi necessário informar ao conversor que os valores possíveis para **genero** eram **masculino** e **feminino**.

Alguns casos extras podem ser previstos, como a possibilidade de um *Datatype Property* se referir a mais de um conceito em seu atributo *Domain*. Neste caso seria necessário gerar a referencia do atributo para cada um dos conceitos no FORTE.

3.1.3. Class

A tag *class* da OWL é utilizada na definição de um conceito da ontologia, podendo agregar em si múltiplos axiomas, que juntos formam a sua descrição formal. Para o sistema de revisão o conceito é uma regra, definida através de relacionamentos.

Toda *class* deve possuir pelo menos um axioma *subClassOf*, relacionando-a a um outro conceito da ontologia, indicando quais propriedades deverá herdar. Caso o axioma seja omitido, considera-se que o objeto é subclasse do objeto raiz *Thing*. O axioma *subClassOf* pode ainda ser utilizado em conjunto com a tag *Restriction* para definir algumas restrições para a semântica da classe. Em conjunto, todas as

características e propriedades de uma classe formarão a cláusula de Horn que definirá o conceito para o sistema de revisão. Na Figura 8 é mostrado o conceito **Professor,** composto por três axiomas *subClassOf*, sendo dois deles restrições sobre relacionamentos obrigatórios ao conceito, um com valor específico e um fazendo referência a qualquer instância de um determinado conceito.

```
<owl:Class rdf:about="#Professor">
    <rdfs:subClassOf rdf:resource="#Pessoa"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#temQualificacao"/>
            <owl:hasValue rdf:resource="#DSc"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#temTopicoPesquisa"/>
            <owl:someValuesFrom rdf:resource="#TopicoPesquisa"/>
        </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

Figura 8 - Definição OWL para o conceito Professor

O conversor deverá mapear a classe gerando a clausula Horn **professor(A)**, cujo corpo é mostrado pela Figura 9.

```
professor(A):- pessoa(A),temQualificacao(A, DSc),temTopicoPesquisa(A, T),TopicoPesquisa(T)
```

Figura 9 - Cláusula de Horn gerada pelo conversor definindo o conceito Professor

A especificação OWL comporta ainda outros tipos de axiomas, como o *disjointWith,* que pode ser visto na Figura 10 e indica a disjunção entre os conceitos **Aluno** e **Professor** (para o domínio exemplo nenhum **Aluno** pode ser **Professor**).

```
<owl:Class rdf:about="#Aluno">
    <rdfs:subClassOf rdf:resource="#Pessoa"/>
    <owl:disjointWith rdf:resource="#Professor"/>
</owl:Class>
```

Figura 10 - Definição OWL para o conceito Aluno

A conversão do axioma *disjointWih* não é direta como a do axioma *subClassOf*, já que isto requer a representação de antecedentes negativos a uma cláusula, e o sistema de revisão FORTE não é capaz de revisar cláusulas com antecedentes negativos. Neste caso, optou-se por definir um novo conceito para o conjunto disjunto de instâncias, e restringir que este novo conceito seja incluído como antecedente negativo em cláusulas pertencentes ao conjunto CP, ou seja, não sujeitas à revisão. O novo conceito então poderá ser utilizado como antecedente tanto em A quanto em CP. Foi convencionado utilizar a prefixo **não** para indicar este tipo de cláusula. No exemplo, definiu-se o conceito **naoProfessor**, e então utilizou-se este como antecedente no corpo da cláusula que define o conceito **Aluno**. A Figura 11 mostra, respectivamente, a definição das cláusulas de Horn **naoProfessor(A)** e **aluno(A)**.

```
naoProfessor(A) :- not profesor(A).
-------------------------------------
aluno(A) :- pessoa(A), naoProfessor(A).
```

Figura 11 - Cláusula de Horn definindo os conceitos naoProfessor e Aluno

3.1.4. Instâncias

As instâncias da ontologia servirão para formar o conjunto E a ser aplicado ao sistema de revisão. Baseado neste conjunto, o sistema de revisão é capaz de verificar inconsistências na teoria e propor modificações, a fim de provar todos os exemplos positivos e negar todos os negativos O mapeamento das instâncias OWL em exemplos do FORTE é explicado a seguir.

3.1.4.1. Exemplos Positivos (E^+)

As instâncias da ontologia, por natureza, constituem o subconjunto de exemplos positivos, dado que se referem aos indivíduos existentes na ontologia. Uma única instância da ontologia é capaz de gerar mais de um exemplo para o conjunto E^+.

O primeiro exemplo a ser obtido é o conceito instanciado. Na Figura 12 pode-se ver que uma instância pode se referir ao conceito instanciado pela própria *tag*. Neste caso seria gerado o exemplo positivo **topicoPesquisa(inteligenciaArtificial)**.

```
<TopicoPesquisa rdf:about="#InteligenciaArtificial">
    <rdf:type rdf:resource="&owl;Thing"/>
</TopicoPesquisa>
```

Figura 12 - Representação OWL para uma instância de TopicoPesquisa

A Figura 13 ilustra uma instância do conceito Professor, de onde é possível extrair mais exemplos positivos com base nos relacionamentos descritos pela instância.

```
<Professor rdf:about="#FernandaBaiao">
    <rdf:type rdf:resource="#Professor"/>
    <firstName rdf:datatype="&xsd;string">Fernanda</firstName>
    <lastName rdf:datatype="&xsd;string">Baião</lastName>
    <genero rdf:datatype="&xsd;string">feminino</genero>
    <matricula rdf:datatype="&xsd;int">210503082</matricula>
    <homepage xml:lang="en">http://uniriotec.br/~fernanda.baiao</homepage>
    <temTopicoPesquisa rdf:resource="#InteligenciaArtificial"/>
    <temTopicoPesquisa rdf:resource="#Ontologia"/>
    <pertenceCentro rdf:resource="#CCET"/>
    <temQualificacao rdf:resource="#DSc"/>
    <orientadorDe rdf:resource="#FelipeLeao"/>
</owl:Professor>
```

Figura 13 - Instância utilizada para a extração de exemplos positivos

Desta instância podemos retirar exemplos como:

```
professor(fernandaBaiao);
temTopicoPesquisa(fernandaBaiao, inteligenciaArtificial);
pertenceCentro(fernandaBaiao, ccet);
temQualificacao(fernandaBaiao, dsc);
orientadorDe(fernandaBaiao, felipeLeao).
```

É possível ainda extrair outros exemplos com base nos *Datatype Properties* incorporados pela instância, como por exemplo, **genero(fernandaBaiao, feminino)**.

3.1.4.2.Exemplos Negativos (E^-)

Para ser capaz de revisar uma teoria, o sistema de revisão precisa receber um subconjunto de exemplos negativos, que seriam as afirmações incorretas do estado atual do domínio. No entanto, como a ontologia representa somente as afirmações corretas sobre o domínio representado, não é possível extrair os exemplos negativos diretamente da ontologia.

Os Exemplos negativos deverão ser gerados através da negação de afirmações presentes na ontologia. Uma combinação de métodos pode ser utilizada. Por exemplo, os axiomas de classe *disjointWith*, apresentados na subseção 3.1.3. Segundo este axioma, os conceitos **Aluno** e **Professor** são disjuntos (Figura 10), portanto , o exemplo positivo **professor(fernandaBaiao)**, pode ser utilizado para gerar o exemplo negativo **aluno(fernandaBaiao)**, da mesma forma que um exemplo positivo de aluno pode ser utilizado na geração de um exemplo negativo de professor. Outro método para construção de exemplos negativos seria considerar o axioma de classe *subclassOf*. Sendo a classe **Monitor** (Figura 14) uma subclasse de **Aluno** (Figura 10 da subseção 3.1.3), um exemplo positivo **aluno(felipeLeao)** pode ser utilizado na criação do exemplo negativo **monitor(felipeLeao)**, já que uma instância definida como Aluno não deve ser entendida como uma instância da classe Monitor.

```
<owl:Class rdf:about="#Monitor">
    <rdfs:subClassOf rdf:resource="#Aluno"/>
</owl:Class>
```

Figura 14 - Definição OWL para o conceito Monitor

3.2. Mapeamento do Modelo Revisado para uma Ontologia

Com base na teoria revisada pelo sistema de revisão o framework ORION montará uma ontologia refinada em OWL capaz de representar com maior precisão o domínio estudado. A ontologia refinada comportará o mesmo conjunto de instâncias da ontologia original, dado que o sistema de revisão não permitirá que exemplos positivos deixem de ser provados pela teoria revisada.

As cláusulas de Horn que representam relacionamentos possuem em sua cabeça a indicação do relacionamento entre duas variáveis, enquanto as cláusulas que definem conceitos possuem em sua cabeça somente uma variável. O framework mapeará relacionamentos para *Object Properties* e conceitos para *Classes*. A Figura 15 exibe duas cláusulas de Horn que serão convertidas, respectivamente, para um objeto *Class* e um *Object Property*.

```
centro(A) :- temCurso(A,C),curso(C).
-------------------------------------
temCurso(X,Y) :- centro(X), curso(Y).
```

Figura 15 – Cláusulas de Horn representando um conceito e um relacionamento

Cláusulas de Horn que representam conceitos terão em seu corpo relacionamentos e referências a outros conceitos. Relacionamentos deverão ser traduzidos para restrições (através da combinação entre o axioma *subClassOf* e a tag

Restriction), e referências a outros conceitos poderão representar a relação de subclasse de um outro conceito ou auxiliar na definição de uma restrição (como a variável C auxilia a definir qualquer instância do conceito Curso). Caso um relacionamento se refira a uma instância específica, a restrição deverá se referir a este valor, caso faça referência a uma variável, será utilizado o atributo RDF *someValuesFrom.* O conceito expresso pela cláusula **centro(a)** na Figura 15 foi mapeado para um objeto *Class* da OWL na Figura 16.

```
<owl:Class rdf:about="#Centro">
    <rdfs:subClassOf rdf:resource="&owl;Thing"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#temCurso"/>
            <owl:someValuesFrom rdf:resource="#Curso"/>
        </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>
```

Figura 16 - Definição do conceito Centro na linguagem OWL

Ao mapear um relacionamento para um *ObjectProperty*, o conceito expresso pela primeira variável será o *Domain*, enquanto o segundo será o *Range*. O resultado do mapeamento do relacionamento **temCurso(X,Y)**, mostrado previamente pela Figura 15, pode ser visto na Figura 17.

```
<owl:ObjectProperty rdf:about="#temCurso">
    <rdfs:domain rdf:resource="#Centro"/>
    <rdfs:range rdf:resource="#Curso"/>
</owl:ObjectProperty>
```

Figura 17 - Definição do relacionamento temCurso na linguagem OWL

Na seção 3.1.3 falou-se sobre a geração de cláusulas com o prefixo **nao**, necessárias para indicar negação no corpo de cláusulas que deverão ser revisadas pelo sistema de revisão de teorias. Estas cláusulas não serão mapeadas diretamente, mas servirão para indicar disjunções (através do axioma *disjointWith*) em objetos *Class*.

3.3. Discussões e Resultados Obtidos

O framework ORION se encontra em estágio de desenvolvimento, com a implementação do mapeamento conforme especificado neste trabalho. Testes preliminares foram executados com pequenos trechos em OWL, resultando nas cláusulas de Horn ilustradas anteriormente. A viabilidade de aplicação do sistema FORTE para revisão de uma ontologia no domínio da família também já foi verificada em Leão et al. (2010), onde foram obtidos resultados positivos, com a teoria sendo revisada com sucesso aumentando sua acurácia de representação.

Atualmente estão sendo realizados testes com o framework Jena[3], que permite manipular e realizar inferências a partir de um modelo ontológico. O Jena está sendo utilizado para recuperar os dados da ontologia, que serão convertidos pelo ORION para o formato aceito pelo FORTE.

[3] http://jena.sourceforge.net/

4. Trabalhos Relacionados

Alguns trabalhos na literatura abordam o tema de revisão de ontologias. Kang e Lau (2007) propuseram a aplicação de técnicas de revisão de crenças (*belief revision*) para revisão de ontologias. No entanto, a técnica de revisão de crenças é monotônica, ou seja, as modificações na base de conhecimento são restritas à remoção de assertivas da base de conhecimento. Por conta disto, Sun et al. (2006) propuseram um novo operador de revisão para lidar com a não monotonicidade do problema de revisão de crenças, e ilustrou a utilização deste operador em um exemplo.

As vantagens do framework ORION advêm do fato de que a restrição da monotonicidade não se aplica à técnica de revisão de teorias, a qual contempla operadores de generalização e de especialização do conjunto de regras e já apresenta resultados satisfatórios para a revisão de teorias descritas em cláusulas de primeira ordem Horn e livre de funções.

5. Conclusão

Ontologias são artefatos de grande utilidade na descrição de domínios para diversos fins. Tipicamente ontologias são construídas por especialistas, considerando seu conhecimento preliminar sobre o domínio. Novas instâncias do domínio podem ser constantemente incorporadas a ontologias, o que pode resultar em um modelo inconsistente com algumas instâncias, especialmente quando estas instâncias representam situações previamente desconhecidas pelo especialista. Quando a ontologia se encontra em estado inconsistente é preferível refiná-las automaticamente ao invés de reaprendê-la do zero, ou tentar modificá-la manualmente. Motivados por estes fatos, neste trabalho propusemos um framework capaz de receber uma ontologia e refiná-la, através do uso de técnicas de revisão de teorias. Foi descrito como os elementos da ontologia devem ser convertidos para uma teoria a ser processada pelo sistema de revisão e como a teoria revista pode ser mapeada para um modelo ontológico que represente com maior acurácia o domínio alvo.

Trabalhos futuros incluem a extensão do framework para trabalhar com múltiplas linguagens de definição de ontologias além da OWL, como lógica de descrição (*Description Logic*), lógica de primeira ordem (*First-Order Logic*) e OBO (*Open Biomedical Ontologies*). Por outra perspectiva, é possível que o domínio esteja associado à incerteza, possuindo axiomas probabilísticos, formalmente representados utilizando ICL [Poole, 1993], BLP [Kersting e De Raedt, 2001]ou Probabilistic Prolog [De Raedt et al., 2007], desta forma propomos também trabalhar com o sistema de revisão de teorias probabilísticas PFORTE [Paes et al., 2006] para refinar ontologias probabilísticas. Finalmente, pretende-se propor técnicas adicionais para geração de exemplos negativos.

Agradecimentos

O primeiro autor gostaria de agradecer à FAPERJ pelo apoio recebido

Referências Bibliográficas

Buntine, W. (1991). “Theory refinement on Bayesian networks”. In Proceedings 17th Conference Uncertainty in Artificial Intelligence, pages 52–60, San Mateo, CA.

De Raedt, L., Kimmig, A., and Toivonen, H. (2007). "Problog: A probabilistic prolog and its application in link discovery". In Proc. IJCAI-2007, pp 2462–2467.

Garcez, A. e Zaverucha, G. (1999). "The connectionist inductive learning and logic programming system". Applied Intelligence, 11:59–77.

Gruber, T. (1995). "Toward principles for the design of ontologies used for knowledge sharing". International Journal of Human-Computer Studies, 43:907-928.

Han, J.; Kamber, M. (2006). Data Mining: Concepts and Techniques, Morgan Kaufmann Pubs

Kang, S. and Lau, S. (2007) Ontology Revision: An Application of Belief Revision Approach, in Sharman, R. et al (eds.) Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems, Springer, pp. 297-318.

Kersting, K. and De Raedt, L. (2001). "Towards combining inductive logic programming with Bayesian networks". In Proceedings of the 12th Int. Conference on Inductive Logic Programming, LNAI 2157 Springer Verlag, Strasbourg, 118–131.

Klyne, G. e Carroll, J. (2004). "Resource description framework (rdf): W3C recommendation". available at: www.w3.org/TR/2004/REC-rdf-concepts-20040210/.

Leão, F., Revoredo, K., Baião, F. (2010) "Ontology Refinement through Theory Revision Techniques", In III Workshop on Computational Intelligence, Brazil.

Mitchell, T. (1997). "Machine Learning'. McGraw-Hill, New York.

Muggleton, S. (1992). "Inductive logic programming". Academic Press, New York.

Paes, A., Revoredo, K., Zaverucha, G., Costa, V. S. (2006). "Pforte: Revising probabilistic fol theories". Proc 18th Brazilian AI Symposium (SBIA-06), LNAI 4140, pp 441–450. Springer.

Poole, D. (1993). "Probabilistic Horn abduction and Bayesian networks". Artificial Intelligence, 64(1):81–129.

Ramachandran, S. e Mooney, R. (1998). "Theory refinement of bayesian networks with hidden variables". Proc 15th Intl Conference on Machine Learning (ICML), pp 454–462.

Richards, B. L. and Mooney, R. J. (1995). "Automated refinement of first-order Horn-clause domain theories. Machine Learning", 19:95–131.

Towell, G. e Shavlik, J. (1994). Knowledgebased artificial neural networks. Artificial Intelligence, 70(1–2):119–165.

Wogulis, J. e Pazzani, M. (1993). "A methodology for evaluationg theory revision systems: results with Audrey II". In Proceedings of the 13th International Join Conference on Artificial Intelligence, pages 1128–1134, Chambery, France.

Wrobel, S. (1996). "First-order theory refinement". In Raedt, L. D., editor, Advances in Inductive Logic Programming, pages 14–33. IOS Press.

Sun, Y., Sui, Y., Zhiping, L. (2006). "One Axiomatic System for the Ontology Revision". Intelligent Information Processing III: 91-100

Fusão de Observações Afetivas em Cenários Realistas

Diego R. Cueva[1], Rafael A. M. Gonçalves[1], Fábio Cozman[1], Marcos R. Pereira-Barretto[1]

[1]Departamento de Engenharia Mecatrônica e Sistema Mecânicos
Escola Politécnica da Universidade de São Paulo (EPUSP)
Av. Prof. Melo Moraes 2231 – São Paulo – SP - Brasil

marcos.barretto@poli.usp.br

Abstract. *This paper demonstrates multimodal fusion of emotion sensory data in realistic scenarios of relatively long human-machine interactions. Fusion, combining voice and facial expressions, has been enhanced with semantic information retrieved from Internet social networks, resulting in more accurate determination of the conveyed emotion.*

Resumo. *Este artigo demonstra uma técnica para a determinação da emoção demonstrada por um interlocutor durante uma interação homem-máquina relativamente longa, utilizando a fusão multimodal de sinais provenientes das emoções na fala e na expressão facial, combinados à análise semântica do conteúdo da fala, realizada com base em informações recuperadas a partir de redes sociais na Internet.*

1. Introdução

Na contínua evolução da computação, o desenvolvimento de experiências de usuário amigáveis tem apresentado lenta inovação desde o surgimento das primeiras interfaces gráficas na década de 70. Ainda que o aprimoramento organizacional, gráfico e de entrada desses sistemas tenha sido de grande relevância, a conservação do paradigma é empecilho para que uma interação mais rica e inteligente entre ser humano e máquina possa prevalecer. O que de fato acaba por existir na atual realidade é a exigência de que o operador aprenda e adapte-se à maneira de operar de cada máquina, sempre de forma restritiva, delimitada pelos casos de uso fixados pelo desenvolvedor.

Emular a capacidade humana de contextualização de uma conversa e de reação flexível ao significado semântico seria a forma mais natural e produtiva de lidar com o problema de interação: se o computador fosse capaz de se utilizar dos recursos dos diversos sinais linguísticos e não linguísticos presentes no cotidiano humano (tais como expressões faciais, o tom da voz e o contexto afetivo da interação), seria possível ter-se máquinas capazes de compreender de forma mais adequada as necessidades e dificuldades do usuário, tornando mais próxima a possibilidade de uma interação genuinamente focada no indivíduo.

Motivados por essas questões, diversos trabalhos buscam elaborar algoritmos computacionais para detecção de emoções, sejam elas extraídas de elementos faciais ou

vocais. Em geral, tais trabalhos suportam-se sobre a conceituada e amplamente utilizada classificação facial elaborada por [Ekman e Friesen, 1977], a qual distingue seis blocos de expressões emocionais básicos: felicidade, tristeza, desagrado, medo, raiva e surpresa. Na última década, com o vasto crescimento da base de dados opinativos da Internet (*blogs*, fóruns, redes sociais), alguns trabalhos têm igualmente tentado observar emoções a partir de elementos semânticos básicos da conversa, comparando-os com o senso comum disponível na rede.

Contudo, ainda que o campo em questão desenvolva-se rapidamente, as técnicas existentes para o tratamento das emoções apresentam-se atualmente limitadas. A dificuldade de adquirir robustez torna esses sistemas pouco confiáveis e frequentemente irrealizáveis em aplicações de alta complexidade. Além disso, a abordagem unimodal (isto é, de observação uma única fonte de emoção) ignora a relação intrínseca entre essas diversas entradas afetivas, cuja relevância é estudada em detalhes por [Scherer e Ellgring, 2007].

Tendo tal problemática como motivação, este artigo investiga a fusão dessas diversas componentes emocionais – ou "sensores" – para determinação da emoção demonstrada pelo interlocutor, em uma solução multimodal. Com esse objetivo, serão propostos dois classificadores para avaliação dos pesos e relações entre as diferentes entradas fornecidas, os quais serão comparados às soluções unimodais. O artigo está organizado de forma a apresentar os trabalhos relevantes nesse contexto na seção 2, descrever as ferramentas utilizadas na seção 3 e elaborar e apresentar experimentos nas seções 4 e 5.

2. Trabalhos Relacionados

O entendimento e modelagem formal do comportamento humano é assunto extensamente discutido pela psicologia e neurociência. Do determinismo comportamentalista às teorias cognitivas mais recentes, observa-se o escalonamento na complexidade e quantidade de variáveis necessárias à compreensão do indivíduo e de suas decisões. Entre os inúmeros trabalhos que discutem o tópico, a linha de estudo das Teorias Cognitivas (*Appraisal Theories*), discutida por [Roseman, 2001] e [Schorr, 2001], fornece um paradigma satisfatório para o escopo de diversos trabalhos da área da computação, ao passo que oferece um modelo que explica as diferenças comportamentais de cada indivíduo, ao mesmo tempo em que determina aspectos comuns a todos. Para essa linha de pensamento, os processos de elicitação de emoções são comuns a todas as pessoas, mas o desenvolvimento desses processos varia individualmente, respeitando a experiência de vida de cada um. A quantificação da dinâmica dessas elicitações é também assunto frequentemente discutido, tal como em [Sander, 2005].

No processo de identificação de emoções por reconhecimento facial, [Bartlett *et al*., 1999] implementaram alguns dos primeiros algoritmos bem sucedidos para automação computacional do processo, comparando diferentes técnicas de obtenção de dados. Na década seguinte, com o aumento da capacidade computacional, as abordagens baseadas em modelos tridimensionais da face e fluxo ótico em tempo real tornaram-se mais comuns, mostrando avanços em relação às anteriores.

Na análise de emoções na voz, pode-se citar os avanços recentes em [Eyben,2009] e [Rachuri, 2010]. Frequentemente, trabalhos discutem a classificação através da valência do tom do discurso (positiva, neutra ou negativa), enquanto outros tentam buscar comportamentos mais bem definidos; [Voigt e André, 2005] fazem um comparativo dessas alternativas sobre um mesmo classificador, discutindo sua eficiência.

Na utilização da Internet como banco de dados para compreensão emocional de palavras, algumas pesquisas recentes, tal como a de [Ptaszynski *et al*., 2009], começam a apresentar resultados animadores.

A análise multimodal, por sua vez, apresenta quantidade ainda pequena e recente de pesquisas e publicações. Além de [Scherer e Ellgring, 2007], [Campanella e Belin, 2007] realizam uma discussão atual dos estudos cognitivos, suportando a correlação entre voz e expressões faciais na demonstração de emoções. Do ponto de vista computacional, [Castellano *et al.*, 2007] apresentam a fusão de dados de voz, face e expressões corporais com mais de dez por cento de aprimoramento em comparação com a abordagem unimodal. [Chetty e Wagner, 2008], por sua vez, investigam o nível em que a fusão deve ocorrer, tomando face e voz como entradas.

3. Ferramentas Utilizadas

Para análise de expressões faciais, o *software* comercial eMotion [eMotion01], desenvolvido na Universidade de Amsterdam, Holanda, foi incorporado ao estudo. Sua escolha está relacionada ao bom comportamento do algoritmo de ajuste de malhas tridimensionais e a capacidade de avaliar os estados emocionais tradados neste artigo.

No processamento de voz, utilizou-se o pacote Emo-Voice [Vogt *et al*., 2008], desenvolvido pelo Instituto de Ciências da Computação da Universidade de Ausburgo. O Emo-Voice é disponibilizado em licença aberta e permite o treinamento personalizado de classificadores para detecção de emoções no discurso. O classificador SVM (*Support Vector Machine*) selecionado para voz foi treinado a partir de amostras do *corpus* (20 para cada emoção), de forma que o algoritmo adaptasse as características de gravação e da língua presentes no banco de dados.

Para a análise do significado emocional do discurso, foi desenvolvida uma ferramenta dedicada, batizada emoCrawler, a qual terá sua validade discutida nos experimentos apresentados neste artigo. A ferramenta utiliza os verbos, adjetivos e substantivos do discurso em buscas em diferentes bases de dados, observando a reação emocional apresentada nos textos.

3.1. O emoCrawler

Por ter sido desenvolvido no contexto deste trabalho, discute-se neste item o emoCrawler. O emoCrawler é um aplicativo modular em desenvolvimento, possuindo como objetivo a avaliação de contextos emocionais sem exigência de tratamentos semânticos complexos. Ignorando as questões abertas de processamento sintático em conversação, o programa realiza buscas em bases de dados *online* de redes sociais, processando grandes volumes de resultados e analisando frequências relativas de expressões de caráter emocional. Dessa forma, o programa conta com um dicionário

inspirado pelas propostas de Goleman (apud [Martines-Miranda,2005]) e [Laros, 2005], o qual mapeia expressões diversas às emoções utilizadas neste artigo.

As palavras consideradas significativas expressadas pelo usuário são extraídas do discurso e buscadas isoladamente na rede social. As mensagens relacionadas a essa consulta são então indexadas e verificadas quanto à existência de expressões de emoção contidas no dicionário. O algoritmo atua através de regras, buscando eliminar falsos positivos, por exemplo, considerando a presença de elementos de negação no tratamento das buscas. O fluxo de dados no aplicativo está ilustrado na Figura 1.

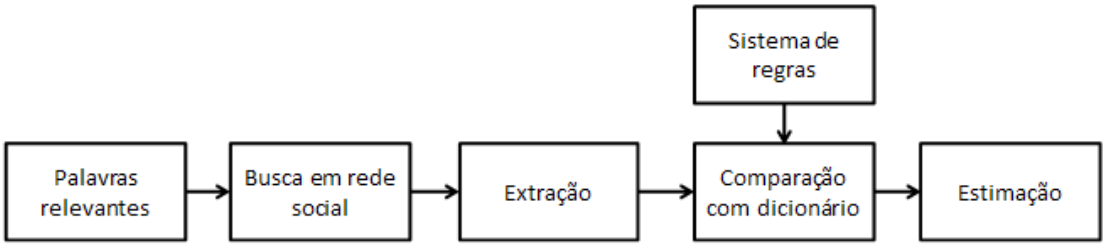


Figura 1. Fluxo de dados no emoCrawler

Dado o contexto modular do emoCrawler, diversas bases de dados podem ser examinadas. Para o presente trabalho, considerou-se uma amostra de consultas retirada da rede social Twitter [twitter01]. A escolha do Twitter está relacionada à abordagem concisa e altamente opinativa dos textos elaborados pelos seus usuários, de forma que mensagens frequentemente demonstram elementos de emoção, ao mesmo tempo em que não contêm construções complexas, dado a limitação de 140 caracteres. Dá-se preferência às consultas em Inglês, o que evita questões ligadas à flexão de verbos, substantivos e adjetivos em Português.

O sistema deve considerar também a questão temporal em sua análise. A busca de emoções relacionadas a um elemento pode apresentar diferentes resultados em diferentes momentos de pesquisa; a opinião popular instantânea pode divergir do passado. Todavia, dado o contexto predominantemente atemporal das frases utilizadas neste experimento, o modo de operação aplicado a este artigo busca apenas o senso comum presente relacionado a cada elemento de busca.

4. Metodologia experimental

Tendo como objetivo a análise multimodal, envolvendo expressões faciais, voz e contexto da fala, foi utilizado o *corpus eNTERFACE'05 Audio-Visual Emotion Database* [Martin *et al*., 2005], um banco de dados de cenas no qual indivíduos são convidados a expressar uma frase emocional da forma que mais desejarem (Figura 2).

Para a execução dos trabalhos, selecionaram-se três subconjuntos de amostras que pudessem ser prontamente qualificadas por observadores humanos, de forma que cenas com ambiguidades emocionais fossem descartadas. Um dos grupos foi utilizado no treinamento do Emo-Voice, enquanto outro foi utilizado para treinamento dos classificadores e, finalmente, o terceiro para validação dos resultados. Ainda que tenha

sido realizada uma seleção prévia do *corpus*, consideraram-se vídeos em condições não ideais para os diversos sistemas de classificação: ruído em áudio, iluminação não uniforme e movimentos abruptos da cabeça são alguns dos fatores que permaneceram intencionalmente no conjunto.



Figura 2. Exemplos de expressões do corpus

Como classificadores para a fusão foram utilizadas redes neurais, escolha comum para situações em que as fontes possuem ruídos diversos e nas quais o grau de confiabilidade relativa entre elas apresenta-se desconhecido. Além disso, tais redes acabam por avaliar também a significância entre os resultados de uma mesma fonte, o que é essencial no caso do emoCrawler.

A escolha das entradas analisadas levou em conta um subconjunto das emoções de Ekman, deixando de fora "surpresa", a qual é frequentemente compreendida como uma expressão sem valência, não abrangendo de fato um estado emocional (a surpresa pode estar atrelada a qualquer estado). Dessa forma, a rede neural possui entradas provindas das três diferentes fontes, uma por emoção de cada fonte. O classificador contém uma saída para cada emoção, de forma a realizar o caminho inverso de classificação nominal, ou seja, conversão para "nomes de emoções". Ainda que alguns trabalhos descrevam transições continuas para algumas emoções, neste contexto seria inviável mapear a saída para uma escala gradual, posto que as emoções trabalhadas não se apresentam ordenáveis em intensidade ou valência. A Figura 3 ilustra o processo.

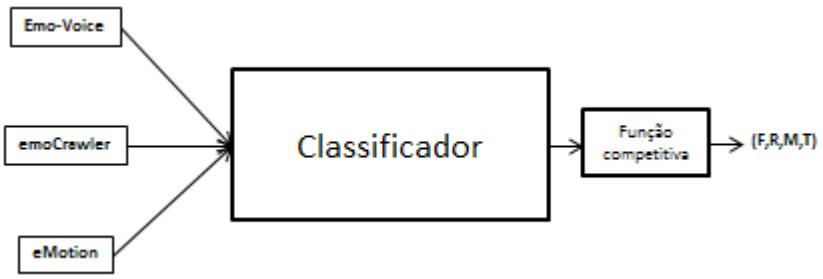


Figura 3. Fusão dos sensores

Para realização dos experimentos, duas abordagens de construção da topologia e algoritmo de treinamento foram consideradas: uma rede neural do tipo *Feed-Forward Back-Propagation* (FFBP) e uma rede neural probabilística (PNN). A rede FFBP é solução comum em problemas de classificação; possui uma camada oculta de nós, de forma a absorver não linearidades existentes e atua com diferentes funções de transferência nas camadas. A PNN, por sua vez, apresenta-se como solução alternativa relevante, dado seu tempo de treinamento ordens de grandeza mais rápido que a FFBP.

Entretanto, seu ajuste é por vezes de difícil realização em conjuntos de dados esparsos – como é o caso.

5. Experimento e Resultados

A primeira etapa de investigação consistiu na análise dos resultados provindos diretamente dos sensores, ou seja, da análise unimodal. A escolha preliminar de algumas amostras para estudo do ferramental sinalizou certos problemas na detecção. A Tabela 1 mostra os resultados de tais testes para um conjunto heterogêneo de amostras, indicando percentuais de acerto das ferramentas.

Tabela 1. Taxa percentual de acertos em análise unimodal - conjunto heterogêneo de amostras

Emoção	**Face**	**Voz**
Felicidade	12,5%	12,5%
Raiva	88,9%	11,1%
Desagrado	0%	0%
Medo	50,0%	50,0%
Tristeza	50,0%	100,0%

A observação da dificuldade dos sensores em captar as expressões de desagrado no conjunto preliminar sugeriu a eliminação delas do banco de entradas da rede neural. Além disso, o processo inicial de classificação desse tipo de expressão por humanos apresentou grande dificuldade de detecção, dado o perfil amador dos atores. Isso tornou difícil a obtenção de um grupo razoável de amostras de desagrado para treinamento e validação das redes. A exclusão evitou, portanto, problemas de contaminação na classificação das outras quatro emoções.

Em um segundo momento, os dados das quatro emoções restantes foram separados em conjuntos de treinamento e testes para as redes neurais, com números iguais de amostras para cada emoção. O treinamento da rede FFBP consistiu da aplicação do algoritmo *Robust Backpropagation* ao conjunto de treinamento. A definição da quantidade de neurônios na camada oculta foi realizada através do estudo dos índices de convergência da rede.

O ajuste da rede probabilística, por sua vez, depende fortemente do parâmetro de espalhamento, um escalar positivo relacionado à distância entre os vetores de treinamento. A metodologia para a escolha do parâmetro consistiu em iniciá-lo com um valor alto (generalista) e realizar passos de redução até garantir total aderência dos dados de treinamento em simulação. Obteve-se através dessa operação um fator de espalhamento de *0,17*.

A Tabela 2 apresenta, após o treinamento supervisionado da rede, os resultados do conjunto final de testes, comparando-se a avaliação isolada das emoções na face e na voz com os dados provindos da fusão multimodal (não há motivo em realizar a

comparação do emoCrawler separadamente, posto sua eficiência estar inerentemente ligada à rede).

Tabela 2. Comparativo das medições individuais com a fusão multimodal: taxas percentuais de acerto para cada método

	Voz	**Face**	**Fusão FFBP (face/voz/semântica)**	**Fusão PNN (face/voz/semântica)**
Felicidade	20%	0%	60%	60%
Raiva	100%	0%	100%	100%
Medo	40%	20%	80%	60%
Tristeza	100%	60%	60%	60%
Acerto Médio	**65%**	**20%**	**75%**	**70%**

Para observação da relevância do emoCrawler sobre a fusão, a rede FFPB foi reconstruída e novamente treinada, agora com oito nós na camada de entrada. A reformulação também acarretou a eliminação de um nó na camada oculta. A Tabela 3 apresenta os resultados desse processo.

Tabela 3 – Avaliação da eficiência do emoCrawler sobre o grupo de teste para a FFBP: taxas percentuais de acerto em cada caso

	emoCrawler desabilitado	**emoCrawler habilitado**
Felicidade	20%	60%
Raiva	60%	100%
Medo	20%	80%
Tristeza	100%	60%
Acerto Médio	**50%**	**75%**

Nota-se o melhor comportamento da fusão quando da incorporação dos elementos de compreensão semântica, particularmente em emoções nas quais os sistemas de face e voz tiveram desempenho ruim. Apesar do resultado coerente, observa-se a diminuição do acerto individual no caso da tristeza, consequência provável da existência de grandes ruídos na base de treinamento do emoCrawler, os quais geraram confusão no tratamento dos dados.

6. Conclusões e Trabalhos Futuros

Dados oriundos dos sistemas de medição utilizados são frequentemente sujeitos a ruídos sistemáticos. Observou-se essa característica, sobretudo, nos dados de face e semântica,

onde frequentemente houve tendência em classificar como mais intenso certo tipo de emoção, independentemente da entrada. As redes neurais foram bem sucedidas em observar essas tendências e realizar correções nos pesos das variáveis.

Verificaram-se indícios de melhoria no desempenho da fusão multimodal com a inclusão do emoCrawler . O grande volume de dados parece ter suprido a ausência de análise sintática profunda. Contudo, a melhoria dos resultados através do uso do emoCrawler deve ser ainda mais extensamente investigada. A quantidade relativamente pequena de expressões linguísticas presentes no *corpus* pode ter afetado os resultados de forma a facilitar a identificação no grupo de testes.

Apesar dos resultados positivos, o trabalho sofreu consideravelmente as implicações de sua premissa de tratar dados fora de condições ideais de operação dos sistemas de detecção. Particularmente, observa-se no caso do *software* eMotion a influência da movimentação da cabeça e dos lábios durante o registro das emoções.

Como trabalhos futuros, planeja-se a utilização de outros classificadores e a análise comparativa em relação aos resultados descritos neste trabalho. Também se planeja a utilização da fusão multimodal em situações ainda mais longas de interação homem-máquina, buscando-se uma maior aproximação com a realidade.

Agradecimentos

Os autores agradecem ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), ao Departamento de Engenharia Mecatrônica da Escola Politécnica da Universidade de São Paulo e à FAPESP (por suporte através do processo 2008/03995-5) pela colaboração e financiamento das pesquisas.

Referências

[Bartlett et al., 1999] Bartlett MS, Hager JC, Ekman P, Sejnowski TJ. “Measuring facial expressions by computer image analysis”. Department of Cognitive Science, University of California, San Diego, USA, 1999.

[Campanella e Belin, 2007] Campanella, S., Belin, P. “Integrating face and voice in person perception”. Trends in Cognitive Sciences, 11, 535−543. 2007.

[Castellano *et al.*, 2007] Castellano, G., Kessous, L. Caridakis, G. “Multimodal emotion recognition from expressive faces, body gestures and speech”. In Fiorella de Rosis, Roddy Cowie (Ed.), Proc. of the Doctoral Consortium of 2nd International Conference on Affective Computing and Intelligent Interaction, Lisbon, September 2007.

[Chetty e Wagner,] Chetty, G. Wagner, M. “A Multilevel Fusion Approach for Audiovisual Emotion Recognition”. International Conference on Auditory-Visual Speech Processing 2008.

[eMotion01] Visual Recognition. Disponível em: <http://www.visual-recognition.nl> Acesso em: 23 de março de 2011.

[Ekman e Friesen, 1977] Ekman, P., Friesen, W. "Facial Action Coding System", Consulting Psychologist Press, 1977

[Laros, 2005] Laros, F.J.M.; Steenkamp, J.E.M. "Emotions in consumer behavior: a hierarchical approach". Journal of Business Research vol.58 pgs.1437-1445, 2005.

[Martin et al., 2005] Martin, O. Kotsia, I. Macq, B. Pitas, I. "The eNTERFACE'05 Audio-Visual Emotion Database". Université Catholique de Louvain; Aristotle University of Thessaloniki, 2005.

[Martinez-Miranda, 2005] Martinez-Miranda, J.; Aldea, A. "Emotions in human and artificial intelligence". Computers in Human Behavior vol.21 pgs.323-341, 2005.

[Pantic, 2000]Pantic, M.; Rothkrantz, L.J.M. "Automatic analysis of facial expressions: state of art". IEEE Trans. On Pattern Analysis and Machine Intelligence, vol.22 no. 12, December, 2000.

[Ptaszynski *et al.*, 2009] Ptaszynski, M., Dybala, P., Shi, W., Rzepka, R., Araki, K. "Towards Context Aware Emotional Intelligence in Machines: Computing Contextual Appropriateness of Affective States". IJCAI'09 Proceedings of the 21st international jont conference on Artifical intelligence. 2009.

[Rachuri et alli, 2010]Rachuri, K.K.; Musolesi, M.; Mascolo, C.; Rentfrow, P.; Longworth, C.; Aucinas, A. "EmotionSense: a mobile phone based adaptive platform for experimental social psychology research". UbiComp '10, Sep 26-Sep 29, 2010, Copenhagen, Denmark.

[Roseman, 2001] Roseman, I.J.; Smith, C.A. "Appraisal Theory – Overview, Assumptions, Varieties, Controversies". In "Appraisal Processes in Emotion – Theory, Methods, Research" editado por Scherer, K; Schorr, ª; Johnstone, T. Oxford University Press, USA, 2001.

[Sander, 2005] Sander, D.; Grandjean, D.; Scherer, K. R. "A systems approach to appraisal mechanisms in emotion" Neural Networks vol. 18 pgs. 317-352, 2005.

[Scherer e Ellgring, 2007] Scherer, K. Ellgring, H. "Multimodal Expression of Emotion: Affect Programs or Componential Appraisal Patterns?". American Psychological Association. 2007.

[Schorr, 2001] Schorr, A. "Appraisal – The Evolution of an Idea". In "Appraisal Processes in Emotion – Theory, Methods, Research" editado por Scherer, K; Schorr, ; Johnstone, T. Oxford University Press, UK, 2001.

[twitter01] Twitter – The best way to discover what's new in your world. Disponível em: <http://www.twitter.com>. Acesso em: 30 de março de 2011.

[Vogt *et al.*, 2008] T. Vogt, E. André and N. Bee, "EmoVoice - A framework for online recognition of emotions from voice," in Proceedings of Workshop on Perception and Interactive Technologies for Speech-Based Systems, 2008.

[Voigt e André, 2005] T. Vogt and E. André, "Comparing Feature Sets for Acted and Spontaneous Speech in View of Automatic Emotion Recognition," in IEEE International Conference on Multimedia & Expo (ICME 2005), 2005.

Método AGCRN-CR para Busca Automática de Redes Neurais Artificiais

Anderson Paulo da Silva[1], Teresa B. Ludermir[1], Leandro M. Almeida[1]

[1]Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Caixa Postal 50740-560 – Recife– PE – Brasil

{aps3,tbl, lma}@cin.ufpe.br

Abstract. *This paper describes an automatic search method, CGANN-DC (Cellular Genetic Algorithm + Neural Networks - Direct enCod), for Artificial Neural Networks (ANNs) using Cellular Genetic Algorithms (CGAs). The main difference between CGAs and Genetic Algorithms (GAs) is the use of a cellular automaton (CA) capable of providing the location for individuals, reducing the possibility of local minima in search space. The CGANN-DC performs evolutionary searches for initial weights, transfer functions, architectures and learning rules. Experimental results have shown that the CGANN-DC can find compact networks with a satisfactory generalization power.*

Resumo. *Este artigo descreve um método de busca automática, AGCRN-CR (Algoritmo Genético Celular + Rede Neural - Codificação Real), por Redes Neurais Artificiais (RNAs) utilizando Algoritmos Genéticos Celulares (AGCs). A principal diferença entre AGCs e Algoritmo Genéticos (AGs) é o uso de um autômato celular (AC) para dar localização aos indivíduos, reduzindo as chances destes caírem em mínimos locais no espaço de busca. O AGCRN-CR executa buscas evolucionárias por pesos iniciais, funções de transferência, arquiteturas e regras de aprendizado. Resultados experimentais mostram que o AGCRN-CR pode encontrar redes compactas com satisfatório poder de generalização.*

1. Introdução

A busca por redes neurais artificiais (RNAs) que se adéquem a determinados problemas é vista como uma questão importante na aplicação de RNAs, uma vez que o poder de generalização de uma rede neural pode ser melhorado de acordo com a devida escolha dos seus parâmetros. Tal busca possui uma série de dificuldades, visto que a construção de uma RNA com configuração que se aproxime do ótimo envolve questões como o exponencial número de parâmetros que precisam ser ajustados, a necessidade do conhecimento *a priori* do domínio do problema e do funcionamento da RNA para definição destes problemas, além da necessidade de um especialista quando este conhecimento não existe [Almeida and Ludermir, 2007]. Essa busca por uma configuração de RNA adequada, de forma geral, é feita manualmente ou através de métodos de busca que consigam encontrar configurações ótimas ou se aproximem do ótimo. Nesse contexto, alguns pesquisadores propõem métodos de busca evolucionária que visam "evoluir" as soluções do problema em questão a partir de uma solução inicial (ou conjunto de soluções iniciais) [Abraham, 2004. Pedrajas, et al. 2005].

Neste trabalho, apresentamos um método de busca que utiliza RNAs e um tipo diferente de AG, conhecido como Algoritmo Genético Celular (AGC) com codificação

real. Este método é denominado AGCRN-CR (Algoritmo Genético Celular + Rede Neural - Codificação Real), e baseia-se no método proposto por [Almeida and Ludermir, 2007] denominado NNGA-DCOD (aNN + GA – Direct enCODe). O NNGA-DCOD faz busca por pesos iniciais, arquiteturas, funções de ativação e regras de aprendizagem para rede neural através de um AG comum e visa encontrar redes compactas com bom poder de generalização. A principal diferença do AGCRN-CR em relação ao NNGA-DCOD é o tipo de algoritmo genético utilizado, onde um Autômato Celular (AC) trabalha em conjunto com um AG, dando localidade aos elementos, formando o AGC. Este trabalho está dividido da seguinte forma: A seção 2 apresenta os conceitos básicos de algoritmos genéticos e sua utilização com RNAs; Seção 3 os conceitos de algoritmos evolucionários celulares; Seção 4 descreve o método AGCRN-CR; Seção 5 descreve os resultados experimentais; Seção 6 descreve as conclusões e apresenta os trabalhos futuros.

2. Algoritmos Genéticos e Redes Neurais Evolucionárias (RNAEs)

Os AGs são técnicas evolucionárias que visam achar soluções ótimas ou quase-ótimas em problemas de otimização e busca. Através da união de técnicas evolucionárias (entre elas AGs) e RNAs na busca dos próprios componentes das RNAs, temos as redes neurais artificiais evolucionárias (RNAEs), que foram definidas por [Yao, 1999]. As RNAEs conseguem ter um melhor aproveitamento (*exploitation*) e exploração (*exploration*) de grande número de aspectos e componentes necessários (como pesos, arquiteturas e funções de transição) para a construção de RNAs com bom poder de generalização.

Mesmo trazendo, em geral, bons resultados, um dos problemas mais comuns na utilização de AGs é a convergência da busca dentro de um ponto de mínimo local no espaço de pesquisa, uma vez que a evolução dos indivíduos sem o correto controle pode gerar a produção de super indivíduos que possuem sempre a mesma carga genética (ou uma carga genética muito parecida) e que irão predominar em relação a todos os outros, reduzindo as chances de encontrar a solução ideal, ou até mesmo soluções diferentes, pois a diversidade genética é essencial para que o AG encontre as soluções ideais [Almeida and Ludermir, 2007].

Ainda assim, diversos pesquisadores têm empregado os algoritmos genéticos com sucesso em suas pesquisas [Yao, 1999. Almeida, 2007]. Dentro das pesquisas, foi possível perceber que apesar de possuírem resultados relevantes, as principais desvantagens dos AGs são as perturbações de boas sub-soluções pelas operações de cruzamento e mutação, além da perda da diversidade genética causada pelo operador de seleção, que constantemente diminui a variedade dos cromossomos [Rajabalipour, et al. 2009].

3. Algoritmos Evolucionários Celulares

Uma das formas encontradas para amenizar as perturbações das sub-soluções causadas pelos operadores dos AGs foi proposta por [Cao and Wu, 1998], onde os pesquisadores introduziram autômatos celulares (ACs) para realizar a localização e vizinhança na estrutura da população dos algoritmos genéticos. Um AC é um modelo discreto estudado na teoria da computabilidade, matemática e biologia teórica. Consiste de uma grelha finita e regular de células, cada uma podendo estar em um número finito de estados, que variam de acordo com regras determinísticas. A grelha pode ser em

qualquer número finito de dimensões. O tempo também é discreto e o estado de cada célula no tempo *t* é uma função do estado no tempo *t-1* de um número finito de células na sua vizinhança. Essa vizinhança corresponde a uma determinada seleção de células próximas (podendo eventualmente incluir a própria célula). Todas as células evoluem segundo a mesma regra de atualização, baseada nos valores de suas células vizinhas. Cada vez que as regras são aplicadas à grelha completa, uma nova geração é produzida. Os autômatos celulares foram introduzidos por Von Neumann e Ulam [Wolfram, 2002] como modelos para estudar processos de crescimento e auto-reprodução. Qualquer sistema com muitos elementos idênticos que interagem local e deterministicamente podem ser modelados usando autômatos celulares.

Assim, a saída encontrada em [Cao and Wu, 1998] foi controlar a seleção com base no grid do AC para evitar a perda rápida da diversidade durante a pesquisa genética, baseando-se na distribuição paralela das populações, onde a idéia de isolar os indivíduos possibilita grande diferenciação genética [Wright, 1943]. Em alguns casos [Alba and Troya, 2002], estes algoritmos utilizando populações descentralizadas conseguem prover uma melhor amostragem do espaço de busca e também melhorar o tempo de execução. Da combinação entre AGs e ACs surgiu o algoritmo genético celular (AGC), que, por sua vez, pertence a um subconjunto dos algoritmos evolucionários chamados de algoritmos evolucionários celulares (AEC).

Nos AECs o conceito de vizinhança é muito utilizado, de modo que um indivíduo interage apenas com seus vizinhos mais próximos. Através desta restrição de interação, os AECs conseguem melhorar o comportamento numérico do algoritmo pela estruturação da população, conseguindo, em geral, melhores taxas e tempos de execução [Alba and Dorronsoro, 2008].

O modelo celular simula a evolução natural do ponto de vista do indivíduo. A principal idéia deste modelo é dar à população uma estrutura especial definida como um grafo conectado, no qual cada vértice é um indivíduo que se comunica com os seus vizinhos mais próximos. Ao visualizar a população de AGs desta forma, podemos dizer que em um AG comum, sua população estaria como um grafo completamente conectado, onde os vértices são os indivíduos e as arestas as suas relações, enquanto num AG celular as interações acontecem apenas com os vizinhos mais próximos como um grafo de estrutura, difundindo lentamente as soluções e mantendo a diversidade mesmo entre os vizinhos próximos. A figura 1 mostra estes grafos [Alba and Dorronsoro, 2008].

Os AGCs são uma subclasse dos algoritmos evolucionários no qual a população é distribuída num grid bidimensional toroidal [Simoncini, et al. 2007]. Além de possuírem todos os conceitos de um AG comum, os AGCs incrementam ao seu processo evolutivo um autômato celular.

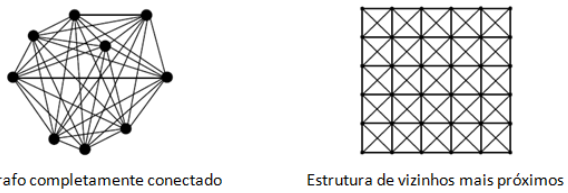


Figura 1. Grafo completamente conectado e grafo de estrutura

Cada célula do AC possui um indivíduo do AG e os operadores genéticos (seleção, reprodução e mutação) são aplicados na vizinhança de um determinado cromossomo escolhido no grid. Como dito anteriormente, a evolução natural aqui se faz voltada para o ponto de vista do indivíduo, ou seja, só há cruzamento entre aqueles indivíduos mais próximos, simulando a influência entre os indivíduos numa sociedade. As interações da população dependem de sua topologia de vizinhança. A figura 2 mostra os dois tipos de vizinhança mais encontrados na literatura, a vizinhança de Neumann e a vizinhança de Moore.

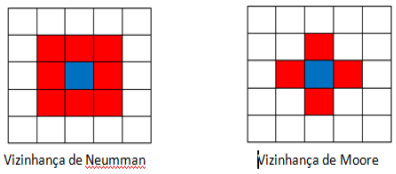


Figura 2. Tipos de Vizinhanças

4. O método AGCRN-CR

O desenvolvimento do AGCRN-CR utiliza RNAs Evolucionárias (RNAEs). A busca evolucionária é subdividida em camadas, onde cada camada possui uma determinada informação da RNA a ser "evoluída" por um AGC específico que trabalha com a informação pertinente à camada em questão. O uso do AGC visa escapar dos mínimos locais e encontrar redes com melhores configurações em relação a um AG comum (assim como o NNGA-DCOD). Essa pesquisa é feita da seguinte forma:

- Na camada mais baixa é feita a busca evolucionária por pesos iniciais da RNA;
- Na camada intermediária ocorre a busca evolucionária por arquiteturas e funções de ativação. Nesta busca também são incluídos o número de camadas escondidas e o número de neurônios por camada;
- Na camada superior a busca evolucionária por regras de aprendizagem (parâmetros dos algoritmos de treinamento) é feita.

A figura 3 mostra uma possível configuração de busca.

Como é possível perceber de acordo com a figura 3, a velocidade de evolução é maior na camada inferior (busca por pesos iniciais), pois neste ambiente o número de arquiteturas e regras de aprendizado é previamente delimitado. Esse formato foi adotado por [Abraham, 2004] onde são feitas buscas mais extensas por arquiteturas e pesos iniciais, tendo as regras de aprendizagem parcialmente definidas.

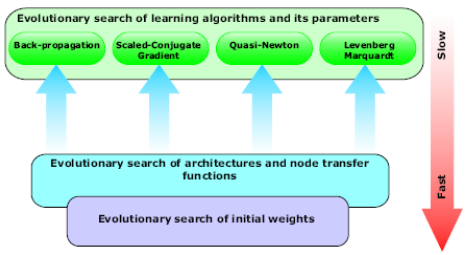


Figura 3. Busca evolucionária em camadas

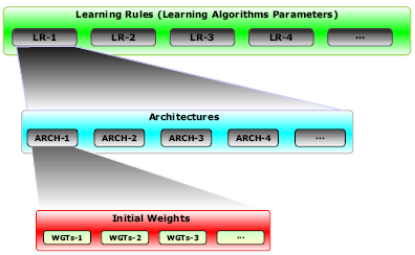


Figura 4. Composição dos dados na busca evolutiva

O AGCRN-CR realiza a busca por RNAs totalmente conectadas com aprendizado supervisionado para problemas de classificação que têm uma arquitetura simples (com poucas camadas ocultas e até 12 nodos por camada), convergência mais rápida com poucas épocas de treinamento e satisfatória performance de generalização. Por exemplo, no trabalho de Ajit Abraham [Abraham, 2004] são usadas até 500 épocas de treinamento para cada rede, enquanto o método AGCRN-CR utiliza até 5 épocas. A composição dos dados na busca evolutiva em camadas do AGCRN-CR é mostrada na figura 4, onde cada camada trabalha com uma população diferente de indivíduos que irá codificar um atributo em questão das RNAs. Na camada inferior há a população de pesos iniciais (PPI), na camada intermediária há a população de arquiteturas e funções de ativação (PAF) e na camada superior há a população de regras de aprendizado (PRA). Como já citado anteriormente, cada uma dessas populações trabalha com os valores reais específicos aos seus domínios de problema, ou seja, a PPI possui populações de cromossomos com os dados reais dos pesos das redes neurais, a PAF populações com os dados reais de configuração de arquiteturas e funções de ativação aplicadas e a PRA possui populações com os dados reais dos parâmetros dos algoritmos de aprendizagem aplicados.

Uma vez que os dados reais são utilizados, o processo de codificação e decodificação dos cromossomos é dispensado, mas é preciso adaptar os operadores genéticos para funcionarem com tal tipo de codificação. Estes operadores foram desenvolvidos em [Almeida and Ludermir, 2007] e foram adaptados para trabalhar com os algoritmos genéticos celulares do AGCRN-CR.

A seleção de n indivíduos é realizada usando a estratégia de torneio, com uma taxa de pressão aleatória $p = 50\%$ e uma taxa de elitismo $e = 10\%$. Esta informação foi encontrada empiricamente após sucessivas execuções. O operador de seleção por torneio é usado tanto para seleção de sobreviventes como para seleção de pais para o cruzamento

Cada PPI está associada a uma arquitetura fixa pré-definida. Dessa forma todos os seus indivíduos possuem as mesmas dimensões e os operadores genéticos trabalham em um conjunto homogêneo de indivíduos de acordo com sua arquitetura. O cruzamento dessa população ocorre como descrito no algoritmo 1. Os filhos serão formados pelo cruzamento dos pais, que serão divididos ao meio para formar novos indivíduos. Esta idéia visa manter as correlações entre os pesos. Portanto, sejam a e b duas matrizes, as metades a_d, a_e, b_d e b_e são obtidas e irão definir a nova matriz filha, por exemplo, w_{filho}=$[a_d; b_e]$ ou $w_{filho} = [a_e; b_d]$

Uma vez que os indivíduos ainda não possuem uma aptidão, a seleção destes para o processo de mutação é feita aleatoriamente. A taxa de mutação empregada é de $m = 40\%$, quarenta por cento dos filhos passarão pelo processo de mutação e estes sofrerão mutação em 40% de sua composição. Este valor obtido no NNGA-DCOD foi encontrado após várias execuções e foi o valor que mais contribuiu para obtenção de bons resultados. Como forma de comparação igualitária o AGCRN-CR utiliza a mesma taxa. Esta mutação é feita através de matrizes esparsas com 40% de seus valores entre $fx = [-0.5, 0.5]$ e os demais valores definidos como zero. Estas matrizes esparsas serão adicionadas aos filhos gerando a mutação.

Algoritmo 1: Operador de cruzamento para indivíduos da PPI

Entrada: $\boldsymbol{ppi_n}$, w // pais, número de matrizes na rede
Saída: $\boldsymbol{vetFilhos}$ // filhos

inicio
  $numFilhos \leftarrow 1$
  enquanto $numFilhos \leq (n/2)$ **faça**
    para $matriz \leftarrow 1: w$ **faça**
      $\boldsymbol{metadeA} \leftarrow fatia(\boldsymbol{ppi_n}(numFilhos).(matriz))$
      $\boldsymbol{metadeB} \leftarrow fatia(\boldsymbol{ppi_n}(numFilhos+1).(matriz))$
      $filho(matriz) \leftarrow [metadeA; metadeB]$
    $\boldsymbol{vetFilhos} \leftarrow filho$
    $numFilhos = numFilhos + 2$
fim

O fitness dos indivíduos da PPI é avaliado de acordo com o erro médio quadrático normalizado (normalized mean square error – NMSE) do conjunto de treinamento da rede.

Para indivíduos da PAF a aptidão I_{fit} é composta por quatro partes de informação, I_{acc} acurácia de classificação, I_{nmse} erro de treinamento, I_{comp} complexidade da rede e I_f peso da função de transferência utilizada.

$$I_{fit} = \propto * I_{acc} + \beta * I_{nmse} + \gamma * I_{comp} + \delta * I_f \quad (1)$$

$$I_{acc} = 100 * \left(1 - \frac{correct}{total}\right) \quad (2)$$

$$I_{nmse} = \frac{100}{N\,P} * \sum_{j=1}^{P} \sum_{i=1}^{N} (d_i - o_i)^2 \quad (3)$$

$$I_{comp} = \frac{c}{c_{total}} \quad (4)$$

$$I_f = \sum_{h=1}^{n} f_{a_h} \quad (5)$$

Na equação 1, I_{acc} é a porcentagem de erro de classificação; I_{nmse} é o NMSE de treinamento gerado pela rede; I_{comp} é a medida da complexidade, considerando o número de nodos usados c e o número máximo de nodos possíveis c_{total}; I_f calcula o peso das funções de transferências usadas. Para isso, cada função de transferência tem um peso associado empiricamente: Linear de 0.2, Tangente de 0.4 e Logaritmica de 0.7, priorizando funções de transferência simples, uma vez que o objetivo da metodologia é encontrar redes simples com alta performance; N e P o número total de saídas e o número total de padrões de treinamento respectivamente; d e o são as saídas desejadas e as saídas obtidas pela rede respectivamente. Como o AGCRN-CR busca redes com até 3 camadas escondidas e o valor de c_{total} é baseado no número de nodos de cada camada (entrada, saída e escondidos): $c_{total} = ent + esc + esc + esc + saida$. Neste trabalho o critério de classificação adotado foi o *winner-takes-all*, no qual o nodo de saída com maior valor vai determinar o padrão de classificação. O cálculo do I_{acc} ocorre através do critério *winner-takes-all.*

As constantes α, β, γ e δ possuem valores entre [0,1] e controlam a influência dos fatores sobre o processo global de aptidão. Por exemplo, para favorecer a precisão da classificação quanto ao erro de treinamento e complexidade, estas constantes são

definidas da seguinte forma $\alpha = 1$, $\beta = 0.90$, $\gamma = 0.90$ e $\delta = 0.20$. Assim, quando indivíduos aparentemente semelhantes forem encontrados, aquele que tiver o menor erro de treinamento, complexidade estrutural ou complexidade de função de transferência prevalecerá. Estes valores foram encontrados após sucessivas execuções. Como já citado anteriormente, o critério de aptidão leva em conta o NMSE da PPI em questão. Caso, durante a evolução da PAF, uma PPI acabe não possuindo tamanhos de vetores necessários à PAF evoluída, uma nova PPI será gerada (e posteriormente evoluída).

A busca por arquiteturas é feita para encontrar configurações com até três camadas escondidas e até 12 nodos por camada escondida. A seleção dos pais é feita pelo algoritmo 2, que começa definindo o número de camadas escondidas que os filhos terão, que será a média arredondada para cima ou para baixo dependendo de certa probabilidade. As dimensões de cada camada escondida e as funções de transferência serão escolhidas aleatoriamente, mas dentro dos limites da união das dimensões e funções de ativação dos pais. Estas etapas do algoritmo 2 se repetem até que o número de cruzamentos desejado seja atingido.

Algoritmo 2: Operador de cruzamento para indivíduos da PAF

Entrada: $\boldsymbol{paf}_n, n, k$ // pais, total de indivíduos, índice
Saída: $\boldsymbol{vetFilhos}$ // filhos
1 **inicio**
2 **enquanto** $k \leq (n/2)$ **faça**
3 $totalCamadas \leftarrow \boldsymbol{paf}(k).numCamEsc + \boldsymbol{paf}(k+1).numCamEsc$
4 **se** $rand(1) > probs$ **faça**
5 $filho.numCamEsc \leftarrow ceil(totalCamadas/2)$
6 **senão**
7 $filho.numCamEsc \leftarrow floor(totalCamadas/2)$
8 $dimensoes \leftarrow [\boldsymbol{paf}(numFilhos).dim; \boldsymbol{paf}(numFilhos+1).dim]$
9 $funcoes \leftarrow [\boldsymbol{paf}(numFilhos).func; \boldsymbol{paf}(numFilhos+1).func]$
10 **para** $elementos \leftarrow 1 : dimensoes_{tamanho}$ **faça**
11 $[filho.dim, filho.func] \leftarrow sorteia(dimensoes, funcoes)$
12 $\boldsymbol{vetFilhos} \leftarrow filho$
13 $numFilhos \leftarrow numFilhos + 2$
10 **fim**

O operador de cruzamento da PAF pode produzir indivíduos similares aos seus pais, ocasionado pela simplicidade do operador. Devido a isso, após o cruzamento é necessário a aplicação do operador de mutação para manter a diversidade da população. O Algoritmo 3 descreve a operação de mutação da PAF. Ele começa com a definição de filhos que sofrerão mutação. Em seguida é gerado um número aleatório no intervalo [0,1]. Este valor será utilizado como uma probabilidade que indicará se uma arquitetura irá sofrer um incremento ou decremento na quantidade de camadas ou no número de neurônios escondidos por camada. Na etapa de mutação, são avaliados todos os casos possíveis de quantidade de camadas (1, 2 ou 3). De acordo com uma determinada probabilidade o indivíduo pode ter sua quantidade de camadas aumentada ou sofrer uma redução de uma ou duas camadas. Caso o indivíduo tenha uma ou duas camadas ocultas, há maiores chances de este passar a ter 3 camadas (visando reduzir o número de nodos por camadas). A probabilidade de redução de camadas só é maior para indivíduos com 3

camadas ocultas (tendo maiores chances de ser 2 camadas). De forma geral, esta etapa prioriza gerar indivíduos com mais camadas e menos nodos por camada. Caso a probabilidade não atinja um valor esperado para adicionar ou remover camadas dentro das arquiteturas, ocorrerá uma mudança nas dimensões das camadas, ou seja, a arquitetura será mantida, mas a quantidade de nodos em cada camada será modificada através de valores gerados entre $arqval = [-2, 5]$ que serão adicionados as dimensões atuais da arquitetura (tendo no mínimo 1 e no máximo 12 nodos por camada). As PAFs são selecionadas de acordo com sua melhor aptidão que, como explicado anteriormente, leva em consideração a aptidão das PPIs (NMSE).

A pesquisa evolucionária por regras de aprendizagem acontece com a busca dos parâmetros dos seguintes algoritmos: Back-Propagation(BP), Levenberg-Marquardt (LM), quasi-Newton (QNA) e Gradiente Conjugado Escalonado (SCG). De acordo com o algoritmo de aprendizagem adotado, uma PRA terá parâmetros deste algoritmo para funcionar. Cada PRA tem uma PAF e sua aptidão é medida através da melhor aptidão dentre todas de sua PAF, ou seja, a PRA é selecionada de acordo com a configuração de parâmetros do algoritmo em questão que proporcione a melhor aptidão para a PAF. A seleção do indivíduo a partir da PRA também usa a estratégia de torneio. As taxas de mutação e elitismo são as mesmas apresentadas anteriormente. Como os indivíduos da PRA têm informações relacionadas a parâmetros do algoritmo de aprendizagem em questão, o seu cruzamento ocorre com a geração de novos valores dentro de um intervalo com base nos valores dos pais. De acordo com uma probabilidade os filhos realizam a mutação adicionando ou subtraindo valores as suas respectivas taxas. O valor da mutação nestas taxas é de 40% para mais ou para menos, dependendo também de uma probabilidade. O funcionamento geral do AGCRN-CR é apresentado no algoritmo 4 e a tabela 1 apresenta a lista de parâmetros utilizados.

O AGCRN-CR inicia sua busca gerando todos os indivíduos e calculando suas aptidões. Em seguida, os operadores genéticos são utilizados nas camadas em busca de novas soluções. Tais operadores tentam manter a diversidade genética utilizando a seleção por torneio, a mutação e o autômato celular, enquanto fazem a evolução das soluções. Como já citado, a manutenção da diversidade é um ponto considerado importante no uso de AGs, visto que tal diversidade evita que a busca caia em pontos de mínimos locais no espaço de pesquisa. A manutenção da diversidade com o uso do autômato celular é feita através da localidade, uma vez que os indivíduos só conseguem cruzar localmente, as chances de escapar dos pontos de mínimo local aumentam, diminuindo a possibilidade de convergência genética.

5. Resultados

Os experimentos foram realizados visando comparar os resultados do método AGCRN-CR em relação aos mesmos resultados com o NNGA-DCOD. Para isso foram utilizadas bases conhecidas do repositório UCI [Newman, et al. 1998]. As bases utilizadas foram Cancer com 9 atributos (atb), 699 exemplos (exp) e 2 classes (cla); Vidros com 9 atb, 214 exp e 6 cla; Cavalos com 58 atb, 364 exp e 3 cla; e Diabetes com 8 atb, 768 exp e 2 cla. A cada iteração os dados foram aleatoriamente divididos em metades. Uma das metades foi utilizada como entrada para os algoritmos com 70% de seus dados para treinamento e 30% para o conjunto de validação. A outra metade da base foi utilizada como conjunto de teste. Para determinar se os resultados obtidos possuem realmente significância estatística, foi utilizado um teste F com o auxílio do software BioStat 5.0.

Algoritmo 3: Operador de mutação para indivíduos da PAF

```
   Entrada: vetFilhos, n, m // vetor com os filhos, número de filhos, taxa de mutação
   Saída: filhosMut // Filhos mutados
1  inicio
2      quantidadeFilhos ← ((m/100) * n) faça
3      enquanto quantidadeFilhos > 0 faça
4          probabilidade ← rand(1)
5          filho ← sortear(vetFilhos)
6          se filho.numCamEsc = 1 faça
7              se probabilidade ≥ 0.6 faça
8                  filho ← adicionaCamadas(3)
9              senão se probabilidade ≥ 0.5 então
10                 filho ← adicionaCamadas(2)
11             senão
12                 filho ← filho + sorteia(arqval)
13
14         senão se filho.numCamEsc = 2 então
15             se probabilidade ≥ 0.6 então
16                 filho ← adicionaCamadas(3)
17             senão se probabilidade ≥ 0.5 então
18                 filho ← reduzCamadaPara(1)
19             senão
20                 filho ← filho + sorteia(arqval)
21
22         senão
23             se probabilidade ≥ 0.6 então
24                 filho ← reduzCamadasPara(2)
25             senão se probabilidade ≥ 0.5 então
26                 filho ← reduzCamadasPara(1)
27             senão
28                 filho ← filho + sorteia(arqval)

29         vetFilhos ← vetFilhos − filho
30         filhosMut ← filho
31         quantidadeFilhos ← quantidadeFilhos − 1
32     filhosMut ← FilhosMut + vetFilhos
10 fim
```

O teste F é um teste de hipóteses que é aproximadamente distribuído com cerca de cinco a dez graus de liberdade e rejeita a hipótese nula de que os dois algoritmos têm a mesma taxa de erro com uma significância de 0,05 se F > 4,74. Esta metodologia foi utilizada devido ao fato do método usual (teste T de Student) gerar um aumento de erros do tipo 1, onde os resultados são erroneamente considerados diferentes com mais freqüência do que o esperado, dado o nível de confiança utilizado.

O método NNGA-DCOD em [Almeida and Ludermir, 2007] teve melhores resultados que a busca manual e obteve RNAs mais compactas, rápidas e com bom poder de generalização. Neste estudo nós comparamos as soluções obtidas com base na informação de erro de dez conjuntos de testes e treinamentos obtidos pelos métodos AGCRN-CR e NNGA-DCOD na Tabela 2.

Algoritmo 4: Funcionamento geral do AGCRN-CR

Entrada:$parametros\ da\ Tabela\ 1$
Saída: $Um\ conjunto\ de\ redes\ neurais\ ótimas\ ou\ quase-ótimas$// novos descendentes

1 **inicio**
2 Gere aleatoriamente as populações de: Pesos iniciais (PPI), Arquiteturas e funções (PAF) e Regras de aprendizagem (PRA)
2 **para cada** $algoritmo\ de\ aprendizagem$ **faça**
3 $Calcule\ a\ aptidão\ para\ todos\ os\ indivíduos$
4 **para cada** $geração \in BERA$ **faça**
5 **para cada** $PAF \in PRA$ **e** $geração \in BEAFA$ **faça**
6 **para cada** $PPI \in PAF$ **e** $geração \in BEP$ **faça**
7 Crie um autômato celular At de acordo com o tamanho da população P;
8 Insira em At a população de acordo com o seu fitness;
9 Crie um autômato auxiliar At_{aux} e faça-o igual a At
10 Inicie o cruzamento através do AGC;
11 Seja Q um número de filhos previamente determinado
12 **para** $numFilhos \leftarrow 1: Q$ **faça**
- Selecione aleatoriamente um indivíduo de At e selecione seus vizinhos por torneio;
- Comece o processo de cruzamento entre os vencedores do torneio para produzir descendentes
- Dependendo de uma probabilidade m aplique a mutação nos descendentes;
- Avalie os descendentes
- Se algum filho gerado for melhor que o pai, substitua o pai pelo filho na mesma posição no grid do autômato At_{aux}

13 Faça $At = At_{aux}$
14 Selecione entre todos os indivíduos, inclusive os novos, sobreviventes para próxima rodada
15 Repita os passos da BEP
16 Repita os passos da BEP
13 Selecionar redes quase-ótimas obtidas com cada algoritmo de aprendizagem
10 **fim**

A média das arquiteturas (arquit.) é baseada na quantidade de neurônios escondidos. O valor do teste F é dado na direita da tabela 2 para todos os casos estudados. De acordo com a tabela 2, ambos os métodos foram capazes de encontrar redes compactas com boa performance, mas o método AGCRN-CR teve resultados estatisticamente melhores de MSE que o método NNGA-DCOD em pelo menos metade dos testes para todas as base de dados. Nas bases Cancer e Cavalos o AGCRN-CR obteve médias estatisticamente melhores em três dos quatro algoritmos avaliados. Nas bases Diabetes e Vidros o AGCRN-CR obteve melhores médias em dois dos quatro algoritmos avaliados. Segundo [Alba and Dorronsoro, 2008], os AGCs conseguem explorar o espaço de busca com maior eficácia, uma vez que a difusão lenta das soluções através da população fornece uma maior diversificação (*exploration*), enquanto a intensificação (*exploitation*) ocorre dentro da vizinhança por meio de operações genéticas.

6. Conclusões

Os algoritmos genéticos celulares possuem melhor equilíbrio entre a diversificação (*exploration*) e a intensificação (*exploitation*) quando comparados a um AG comum.

	Parâmetros para	***Valores***
AGs	*Modo de codificação*	Real
	Taxa de Elitismo	10%
	Taxa de Mutação	40%
	Tipo de seleção	Torneio
	Número de indivíduos por população (PRA, PAF e PPI)	4, 9, 9
	Quantidade de geraçãoes (BERA, BEAFA, BEP)	18, 5, 3
	Cálculo de aptidão	Baseado no NMSE do conj. de treinamento
RNAs	*Tipo da Rede*	MLP feedforward
	Algoritmos de aprendizagem empregados	BP, LM, QNA, SCG
	funções de transferência	FL (linear), FLS (logaritmica Sigmoid) e FTH (tangente hiperbólica)
	Número de camada escondidas	até 3
	Número de unidades escondidas	até 12
	Número de épocas de treinamento	até 5
	Intervalo para os pesos iniciais	[-0.05, 0.05]
	Função de transferência da última camada	FL
Algoritmos de aprendzagem	*BP*	
	Taxa de aprendizado	[0.05, 0.25]
	Momento	[0.05, 0.25]
	LM	
	Taxa de aprendizado	[0.001, 0.02]
	QNA	
	Tamanho do passo	[1.0e-06, 100]
	Limite do tamanho do passo	[0.1, 0.6]
	Fator de escala para determinar a performance	[0.001, 0.003]
	Fator de escala para determinar o tamanho do passo	[0.001, 0.02]
	SCG	
	Fator de controle para cálculo de aproximação da informação de segunda-ordem	[0, 0.0001]
	Fator regulador da falta de definição da matriz hessiana	[0,1.0e-06]

Tabela 1 – Lista de parâmetros e valores

Problemas / Algoritmos		***AGCRN-CR***			***NNGA-DCOD***			***Teste F***
		Arquit.	***Erros***		***Arquit.***	***Erros***		
			Treino	***Teste***		***Treino***	***Teste***	
Cancer	*BP*	14	24.2726	24.3001	9	23.9684	23.9810	≤ 4.74
	LM	12.5	1.6778	2.0790	11.6	2.3036	2.5402	> 4.74
	SCG	11.9	4.4640	4.0842	12.5	5.2004	5.1257	> 4.74
	QNA	11.8	2.9263	2.6734	9.1	3.6066	3.3305	> 4.74
Diabetes	*BP*	7.1	23.9847	24.1965	12.2	24.3103	24.4491	> 4.74
	LM	16.6	13.6740	17.2658	12.3	14.1265	16.9845	≤ 4.74
	SCG	16.3	17.1709	20.1215	15.7	18.6435	21.0650	> 4.74
	QNA	12	18.7571	21.0253	16.5	17.6658	20.0718	≤ 4.74
Cavalos	*BP*	7.5	23.0633	22.6557	9.1	24.1395	23.9741	> 4.74
	LM	17.8	8.8731	15.8538	16.8	9.8776	15.1914	≤ 4.74
	SCG	12.6	15.1021	15.7953	17.9	16.1976	16.4628	> 4.74
	QNA	4.8	16.6637	16.1699	16.1	18.2964	17.2823	> 4.74
Vidros	*BP*	14	14.1346	13.5554	9.7	14.2789	13.8430	> 4.74
	LM	17.8	10.2865	11.0052	16.8	9.5976	11.7114	> 4.74
	SCG	17.9	12.1718	12.0891	18.8	12.3062	11.7663	≤ 4.74
	QNA	14.8	12.4402	12.6701	14.6	12.2023	11.2883	≤ 4.74

Tabela 2 – Resultados Experimentais

Devido a este equilíbrio, os AGCs conseguem escapar mais facilmente dos pontos de mínimo local no espaço de busca, conseguindo atingir, em média, melhores resultados

em relação a um AG comum, exatamente como o adotado pelo método NNGA-DCOD. Os resultados experimentais mostraram que, apesar do NNGA-DCOD ter obtido algumas redes um pouco mais compactas que o AGCRN-CR, na maioria dos casos o método celular obteve médias estatisticamente menores de MSE. Em trabalhos futuros há a pretensão de aplicar o método celular através de uma implementação paralela, visando acelerar a busca genética, uma vez que a pesquisa obteve resultados semelhantes em termos de tempo.

7. Agradecimentos

Os autores gostariam de agradecer a FACEPE, CNPq e CAPES pelo apoio financeiro.

8. Referências

A. Abraham, (2004) "Meta learning evolutionary artificial neural networks". In Neurocomputing, no. 56, pp. 1-38.

D. J. Newman, S. Hettich, C.L. Blake, and C.J. Merz, (1998) "UCI repository of machine learning databases" [Online]. Available: http://archive.ics.uci.edu/ml/

D. Simoncini, P. Collard, S. Verel, M. Clergue, (2007) "On the influence of selection operators on performances in cellular genetic algorithm." In CEC – 2007, *pp.4706-4713*.

E. Alba, B. Dorronsoro, (2008) "Cellular Genetic Algorithms", In *Operations Research Computer Science Interfaces Series*, Vol. 42. ISBN – 978-0-387-77609-5

E. Alba, and J. M. Troya, (2002) "Improving flexibility and efficiency by adding parallelism to genetic algorithms." In *Statistics and Computing*, *pages 91-114*

H. Rajabalipour, H. Haron, M. I. Jambak, (2009) "The Improved Genetic Algorithm for Assignment Problems", In ICSPS, pp. 187 - 191.

L. M. Almeida, (2007) "Uma metodologia de busca por redes neurais artificiais quase-ótimas." Dissertação de mestrado – CIn - Centro de Informática - UFPE.

L. M. Almeida and T. B. Ludermir (2007) "Automatically searching nearoptimal artificial neural networks." In ESANN. pp. 549-554.

N. Garcia-Pedrajas, C. Hervas-Martinez, and D. Ortiz-Boyer, (2005) "Cooperative coevolution of artificial neural network ensembles for pattern classification." In IEEE Trans. Evolut. Computation, Vol. 9, no. 3, pp. 271-302.

R. Linden, (2006), "Algoritmos genéticos – Uma importante ferramenta da inteligência computacional", Rio de Janeiro. Brasport, 1ª edição

S. Wright, (1943) "Isolation by distance." In Genetics, 28, pp. 114 – 138

S. Wolfram, (2002) "A New Kind of Science", Wolfram Media, ISBN 1-57955-008-8

X. Yao, (1999) "Evolving artificial neural networks." In Proceedings of the IEEE, 87(9): pp. 1423-1447.

Y. J. Cao, H. Q. Wu, (1998) "A Cellular Automata Based Genetic Algorithm and its Application in Machine Design Optimization." In *ICC, pages 1593 – 1598.*

Um Modelo Híbrido MLP-SVM para Reconhecimento de Caracteres Manuscritos Cursivos

Washington W. Azevedo[1], Cleber Zanchettin[1]

[1]Centro de Informática – Universidade Federal de Pernambuco (UFPE)
50.732-970 – Cidade Universitária – Recife – PE – Brasil

{wwas, cz}@cin.ufpe.br

Abstract. *This paper presents a hybrid MLP-SVM method for cursive characters recognition. Specialized Support Vector Machines (SVMs) are introduced to significantly improve the performance of Multilayer Perceptron (MLP) in the local areas around the surfaces of separation between each pair of characters in the space of input patterns. This hybrid architecture is based on the observation that when using MLPs in the task of handwritten characters recognition, the correct class is almost always one of the two maximum outputs of the MLP. The second observation is that most of the errors consist of pairs of classes in which the characters have similarities (e.g. (U, V), (m, n), (O, Q), among others). Specialized local SVMs are introduced to detect the correct class among these two classification hypotheses. The hybrid MLP-SVM recognizer showed improvement, significant, in performance in terms of recognition rate compared with an MLP for a task of character recognition.*

Resumo. *Este artigo apresenta um método híbrido MLP-SVM para o reconhecimento de caracteres cursivos. Support Vector Machines (SVMs) especializados são introduzidos para melhorar significativamente o desempenho da Multilayer Perceptron (MLP) nas áreas locais em torno das superfícies de separação entre cada par de caracteres, no espaço de padrões de entrada. Essa arquitetura híbrida é baseada na observação de que ao utilizar MLPs na tarefa de reconhecimento de caracteres manuscritos, a classe correta é quase sempre uma das duas saídas máximas da MLP. A segunda observação é que a maioria dos erros consiste em pares de classes nos quais os caracteres possuem similaridades (ex. (U, V), (m, n), (O, Q), entre outros). SVMs locais especializados são introduzidos para detectar a classe correta entre estas duas hipóteses de classificação. O reconhecedor híbrido MLP-SVM apresentou melhoria, significativa, no desempenho em termos de taxa de reconhecimento em comparação com uma MLP para uma tarefa de reconhecimento de caracteres.*

1. Introdução

A escrita manual é uma das formas mais naturais de comunicação entre as pessoas, constatando-se a geração de uma grande quantidade de dados manuscritos em papel. Muitas vezes é necessário processar os dados contidos nestes documentos de forma automática, sendo extremamente desejável, que computadores tenham capacidade de

'ler' e interpretar documentos manuscritos. O reconhecimento de caracteres manuscritos tem sido uma preocupação da comunidade científica [Plamondon et al. 2000]. As aplicações para um sistema que faça tal tipo de reconhecimento são muitas, podendo citar leitoras automáticas de cheques bancários, máquinas automáticas de processamento de códigos postais, máquinas automáticas para processar qualquer tipo de formulário preenchido manualmente, entre outros [Aires 2005]. O fato dos caracteres em um documento estarem na forma manuscrita torna a tarefa de reconhecimento automática complexa. Isto ocorre devido às variações de estilos existentes na escrita manuscrita, ou seja, o estilo pessoal de cada escritor, onde para cada escritor a forma de um caractere pode ser feita de diferentes maneiras e até o mesmo escritor pode representar o caractere de maneiras diferentes [Aires 2005]. Outro problema no reconhecimento de caracteres manuscritos é a similaridades entre caracteres distintos, por exemplo, U e V, Q e O, entre outros. Neste artigo, demonstramos a vantagem de utilizar Support Vector Machines (SVMs) [Vapnik 1999] para melhorar o desempenho de um sistema ICR (Intelligent Character Recognition) baseado em redes neurais MLP [Rumelhard et al. 1990]. Na seção 2, é feita uma breve descrição sobre reconhecimento de caracteres e apontamos os principais problemas da tarefa de reconhecimento de manuscritos. A seção 3, descreve a base de dados C-Cube. Na seção 4, a extração de características e alguns resultados são brevemente introduzidos. Na Seção 5, explicamos a arquitetura híbrida MLP-SVM. Na seção 6 os experimentos e os resultados obtidos são analisados. Uma discussão sobre análise de erros é feita na Seção 7. A conclusão é dada na seção final.

2. Reconhecimento de Caracteres

O reconhecimento de caracteres é algo que vem sendo estudado amplamente pela comunidade científica desde a invenção do computador [Veloso 1998] e consiste em, a partir de características extraídas de um conjunto de caracteres, separá-los em 10 classes, no caso dos algarismos, ou 26 classes, no caso das letras do alfabeto ocidental.

2.1. Principais Problemas da Tarefa de Reconhecimento de Caracteres

Existem alguns problemas que dificultam a realização do reconhecimento de caracteres. Pode ocorrer, por exemplo, de uma imagem digitalizada possuir baixa qualidade devido a algum inconveniente durante o processo de digitalização do documento, sendo necessário realizar um pré-processamento para a eliminação de ruídos na imagem. Outro problema que dificulta esta etapa é a existência de caracteres distorcidos, principalmente se tratando de documentos manuscritos, devido às características de caligrafia do escritor, que pode dificultar inclusive o reconhecimento dos caracteres por uma pessoa.

Pode-se destacar também como um possível empecilho o fato de um mesmo documento possuir vários tipos de caracteres diferentes como, por exemplo, letras maiúsculas e minúsculas, números, caracteres especiais, letras gregas utilizadas em fórmulas matemáticas, entre outros, o que implicaria no desenvolvimento de um sistema de reconhecimento de caracteres mais generalizado e complexo. Além disto, outra dificuldade a ser considerada é a semelhança entre alguns caracteres, como "I" e "J", "Q" e "O", "U" e "V", entre outros, que pode dificultar a classificação dos caracteres reconhecidos.

O trabalho [Rodrigues et al. 2002] descreve um conjunto de testes através do uso de um modelo de rede neural MLP, para a avaliação de uma técnica de extração de características proposta em [Rodrigues et al. 2001] que utiliza como base de dados um conjunto de letras manuscritas. Em [Rodrigues et al. 2001] foi feita uma avaliação desta técnica, porém a base de dados era composta somente de dígitos. Esta técnica foi desenvolvida principalmente para tratar o problema de reconhecimento de caracteres manuscritos e é baseada na projeção do contorno da imagem sobre os lados de um polígono regular construído em torno de cada caractere.

Em [Vamvakas et al. 2010], é apresentado uma metodologia para o reconhecimento de caracteres manuscritos. A metodologia proposta baseia-se em uma nova técnica de extração de características com base na subdivisão recursiva da imagem do caractere para que o resultado de sub-imagens em cada iteração tenha um número equilibrado (aproximadamente igual) de pixels de primeiro plano (foreground), na medida em que isso é possível. Nos experimentos foram utilizadas duas bases de dados de caracteres manuscritos (CEDAR e CIL), bem como duas bases de dados de dígitos manuscritos (MNIST e CEDAR). A etapa de classificação foi realizada utilizando Support Vector Machine (SVM) com Radial Basis Function (RBF). Os resultados de reconhecimento para a base de dados CEDAR foi de 94.73% e para a base de dados MNIST (99.03%).

No artigo [Cruz et al. 2010] é apresentada uma nova abordagem para o reconhecimento de caracteres cursivos usando múltiplos algoritmos de extração de características e um classifier ensemble. Várias técnicas de extração de características, usando diferentes abordagens, são extraídas e avaliadas. Duas técnicas, Modified Edge Maps e Multi Zoning, são propostas. Com base nos resultados, uma combinação dos conjuntos de características é proposta, a fim de alcançar um alto desempenho de reconhecimento. Esta combinação é motivada pela observação de que os conjuntos de características são, independentes e complementares. O ensemble é realizado através da combinação das saídas geradas pelo classificador em cada conjunto de características separado. A base de dados utilizada para os experimentos foi a C-Cube e uma rede MLP de três camadas, treinada com o Resilient Backpropagation.

Bellili apresentou em seu trabalho [Bellili et al. 2001] um método híbrido MLP-SVM para o reconhecimento de dígitos manuscritos. Essa arquitetura híbrida é baseada na idéia de que a classe correta para um dígito quase sistematicamente pertence às duas saídas máximas da MLP e que alguns pares de classes de dígitos constituem a maioria das substituições da MP (erros). SVMs locais especializados são introduzidos para detectar a classe correta entre estas duas hipóteses de classificação. O reconhecedor híbrido MLP-SVM alcançou uma taxa de reconhecimento de 98.01%, para uma tarefa real de reconhecimento de códigos postais.

Já no trabalho [Camastra 2007] é apresentado um reconhecedor de caracteres cursivos que realiza a classificação de caracteres usando SVM e neural gas. O neural gas é usado para verificar se a versão maiúscula e minúscula de uma determinada letra pode ser unida em uma única classe ou não. Feito isso para cada letra, o reconhecimento de caracteres é feita usando SVMs. SVMs apresentam melhores resultados, em termos de taxas de reconhecimento, comparado com outros classificadores neurais, tais como aprendizagem de quantização vetorial e MLPs. A taxa de reconhecimento das redes

SVMs são as mais elevadas apresentadas na literatura para o reconhecimento de caracteres cursivos.

Nos trabalhos [Rodrigues et al. 2002] e [Vamvakas et al. 2010], foi observado uma tendência de erro entre letras que possuem similaridades (ex. (B, D), (H, N) e (O, Q)). Em [Cruz et al. 2010] foi detectado uma elevada taxa de erro em caracteres que possuem duas formas completamente diferentes de escrita (ex. (a, A) e (f, F) ver figura 1).



Figura 1. Formas diferentes para a versão maiúscula das letras A e F.

Sistemas Inteligentes Hibridos (SIH) tem sido amplamente estudado nos últimos anos como uma alternativa para aumentar a eficiência e precisão de classificadores [Goonatilake et al. 1995]. A principal motivação para a utilização de SIHs é que uma única técnica, por causa de suas limitações e/ou deficiências, pode não ser capaz, por si só, resolver um determinado problema. Para isso a combinação de várias técnicas pode levar a uma solução mais robusta e eficiente. Neste trabalho nós mostramos que através da combinação de classificadores simples (por exemplo, MLP-SVM) para a tarefa de reconhecimento de caracteres, podemos alcançar um desempenho melhor em termos de taxa de reconhecimento, do que técnicas e métodos complexos de difícil treinamento apresentados na literatura, além de tratar o problema de semelhança entre letras (por exemplo, (A, N), (J, S) (M, N) (O, Q), entre outros) apontados em [Rodrigues et al. 2002] e [Vamvakas et al. 2010].

3. Base de Dados C-CUBE

A base de dados C-Cube é uma base pública disponível para download no site Cursive Character Challenge (http://ccc.idiap.ch). A base de dados consiste de 57.293 imagens, incluindo letras maiúsculas e minúsculas, manualmente extraída do CEDAR e United States Post Service (USPS) databases. Como reportado por Camastra [Camastra et al. 2006], esta base de dados apresenta três vantagens:

1. A base de dados já está dividida em conjuntos de treinamento e teste, assim os resultados de diferentes pesquisadores podem ser comparados rigorosamente;
2. A base de dados não contém apenas imagens, mas também seus vetores de características extraídos utilizando o algoritmo proposto por Camastra [Camastra 2007];
3. Os resultados obtidos usando os métodos do estado da arte ainda deixam espaço significativo para melhorias significativas.

A base de dados é dividida em 38.160 (22.274 letras minúsculas e 15.886 letras maiúsculas) imagens para treinamento e 19.133 (11.161 letras minúsculas e 7.972 letras maiúsculas) imagens para teste. Todas as imagens binárias e com tamanho variável. Para cada imagem, informações são fornecidas como a distância a base e a linha superior, a distância do extremo superior da linha base e a distância do extremo inferior

da linha base. O número de exemplos para cada classe é variável e foram selecionados de acordo com sua frequência em documentos extraídos do conjunto de dados do CEDAR e USPS. Nas figuras 2 e 3 são apresentadas à distribuição das letras nas versões minúsculas e maiúsculas, respectivamente. Pode-se observar que há uma grande diferença no número de padrões entre as letras.

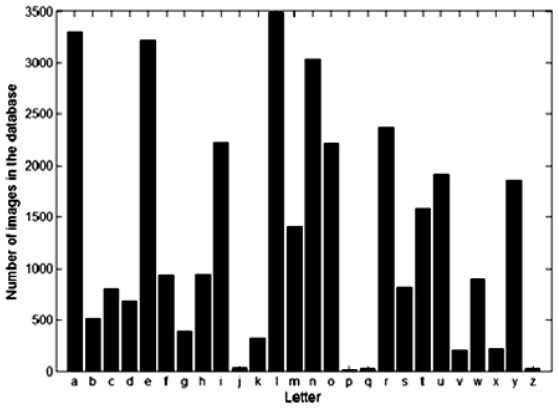


Figura 2. Distribuição de letras minúsculas na Base de Dados C-Cube.

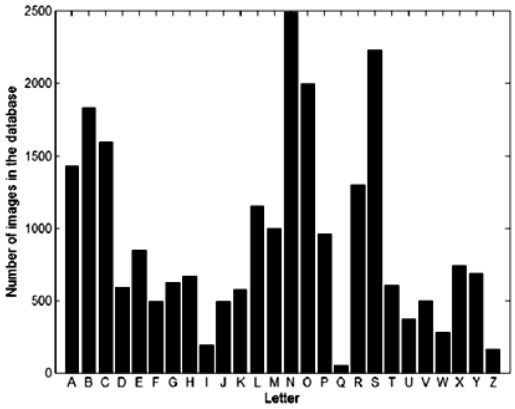


Figura 3. Distribuição de letras maiúsculas na Base de Dados C-Cube.

Thornton [Thornton et al. 2009] observou, através de engenharia reversa, que os arquivos de imagem (test.chr e training.chr) não correspondem aos vetores de características (test.vec e training.vec) disponível no site da C-Cube. Os vetores de características são de uma diferente parte da base de dados (ou seja, algumas imagens estão no conjunto de treinamento em uma parte e no conjunto de teste em outra). Por esta razão Thornton [Thornton et al. 2009] denominou os arquivos dos vetores de características da base de dados (training.vec and test.vec) como Split A e os arquivos de imagens da base de dados (test.chr and training.chr) como Split B. Foi também provado que a Split B consiste em uma divisão mais difícil da base de dados, com resultados em torno de 3 pontos percentuais menor quando comparado com a Split A.

Neste trabalho apenas o Split B foi utilizado para os experimentos, porque os arquivos de imagem do Split A não estão disponíveis para download.

4. Extração de Características

A fase de extração de características pode ser definida como a extração das informações mais importantes e discriminantes para realizar a classificação para uma determinada tarefa [Devijver et al. 1982]. Existem várias técnicas de extração de características e autores sugerem que sua escolha pode ser considerada o fator mais importante para alcançar taxas de alta precisão [Trier et al. 1995].

Em [Cruz et al. 2010] dois experimentos diferentes foram feitos (todos os experimentos foram realizados utilizando uma MLP de três camadas treinada com o algoritmo Resilient Backpropagation [Riedmiller et al. 1993]): Primeiro, dividindo o conjunto de dados com letras maiúsculas e minúsculas e outro experimento com ambos os casos.

Os resultados de classificação para os casos separados e juntos são apresentados nas tabelas 1 e 2, respectivamente. Os resultados são ordenados pelas taxas de reconhecimento. A proposta de modificação do algoritmo Edge Maps apresentou o

melhor resultado global. A maioria dos conjuntos de características apresentou melhor precisão para as letras maiúsculas, com exceção do método proposto por Camastra que apresentou melhor desempenho para letras minúsculas. Este conjunto de características também apresentou a melhor precisão (84,37%) para as letras minúsculas. Pode ser visto que os métodos baseados em gradientes e o modified edge maps apresentaram os melhores resultado. Esses métodos têm em comum o uso da informação direcional. Camastra 34D feature set também usa recursos de características direcionais.

Tabela 1. Taxa de reconhecimento pelo conjunto de características para letras maiúsculas e minúsculas separadas [Cruz et al. 2010].

Método	No. Nodos	Maiúsculo (%)	Minúsculo (%)
Edge	490	**86.52**	81.13
Binary Grad.	490	86.35	79.89
MAT Grad.	300	85.77	79.22
Median Grad.	360	85.10	79.48
CAMASTRA 34D	400	79.63	**84.37**
Zoning	450	84.46	78.07
Structural	320	81.94	77.70
Concavities	530	73.35	81.89
Projections	500	71.73	79.90

Tabela 2. Taxa de reconhecimento pelo conjunto de características para ambos os casos [Cruz et al. 2010].

Método	No. Nodos	Taxa de Reconecimento (%)
Edge	490	**82.49**
Binary Grad.	490	81.46
MAT Grad.	300	80.83
Median Grad.	360	79.96
CAMASTRA 34D	400	79.97
Zoning	450	78.60
Structural	320	77.07
Concavities	530	74.90
Projections	500	73.85

5. Uma Arquitetura Híbrida Combinando MLP-SVM

Um dos problemas que dificultam a tarefa de reconhecimento de caracteres manuscritos, é a similaridade entre caracteres distintos, (por exemplo, U e V, Q e O, entre outros) causando confusões no momento de classificar tais padrões. A figura 4 apresenta algumas similaridades entre caracteres. Esta observação motiva a busca por um método adequado que pode detectar a classificação correta entre padrões de maior confusão tendo um nível de confiança máxima na decisão de classificação. Uma vez que a decisão da MLP é feita, o problema é escolher a classe certa entre duas hipóteses de classificação. Desta forma, a escolha resulta em um problema binário, considerando que estudos realizados por [Bellili et al. 2001] identificaram que em problemas de reconhecimento de caracteres a resposta certa, na maioria dos casos, está entre as duas primeiras saídas da MLP.

Um dos métodos mais eficazes para resolver um problema de classificação binária, com a máxima confiança na decisão, é a introdução de SVMs. A eficácia desses métodos é devido à sua capacidade em construir um hiperplano de separação ótimo entre os exemplos de duas classes diferentes. A margem de separação entre estes exemplos é máxima.

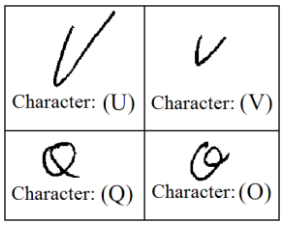


Figura 4. Similaridade entre caracteres distintos.

Esta combinação de métodos resulta na especialização de SVMs nas áreas locais ao redor da superfície de separação entre cada par de caracteres que constituem a maioria dos erros da MLP (confusões). Assim, serão introduzidos SVMs apenas para os pares de classes que constituem as maiores confusões da MLP. O modelo híbrido MLP-SVM é mostrado na figura 5.

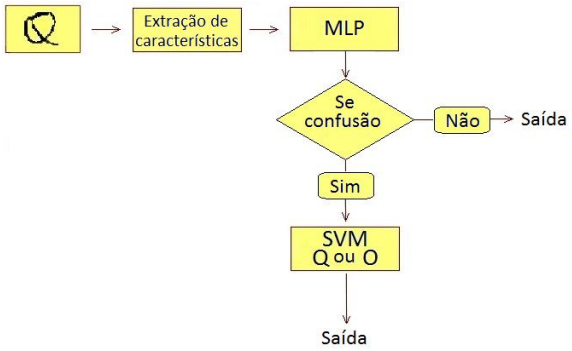


Figura 5. Modelo híbrido ML-SVM: Um padrão é apresentado a MLP. Se a saída classificada pela rede MLP faz parte das classes de confusão, esta saída é apresentada ao SVM especialista para detectar a classe correta para a classificação destas duas hipóteses.

Em [Bellili et al. 2001] a arquitetura MLP+SVM foi usada para classificar dígitos (0...9). Neste trabalho avaliamos essa arquitetura para uma tarefa de classificação de maior complexidade, que foi a classificação de caracteres cursivos com 26 padrões para cada caso (maiúsculo/minúsculo) e 39 padrões para o caso maiúsculo+minúsculo e foram obtidos resultados satisfatórios. A preocupação principal de [Bellili et al. 2001] foi em observar as duas saídas da rede com maior ativação, para analisar se a resposta estava correta. No trabalho proposto, nos preocupamos em avaliar a matriz de confusão e obter quais os pares de caracteres com maior índice de confusão e treinarmos SVMs especialistas nestas classes (conforme Figura 6).

6. Experimentos e Resultados

6.1. Processo de treinamento MLP

Todos os experimentos foram realizados utilizando uma MLP com três camadas (uma camada de entrada, uma camada escondida e uma camada de saída) e treinada com o algoritmo Resilient Backpropagation (RProp) [Riedmiller et al. 1993]. O algoritmo RProp foi escolhido porque apresentou convergência mais rápida e melhores resultados para este problema, quando comparado ao Backpropagation convencional. Os valores dos pesos foram ajustados baseando-se no erro do resultado obtido com o resultado esperado da rede. A medida de erro utilizada foi o SSE (soma dos erros quadráticos).

Com isso espera-se que a cada iteração o erro decresça. O treinamento é finalizado quando o erro do conjunto de validação crescer por cinco iterações seguidas ou até a milésima iteração, sendo estes os critérios de parada utilizados.

A arquitetura da rede MLP usada para todos os casos (maiúsculo, minúsculo e maiúsculo+minúsculo) foi a seguinte:

- Uma camada de entrada contendo 34 neurônios, referente ao número de características extraídas de cada caractere usando o algoritmo proposto por Camastra [Camastra 2007];
- Uma camada escondida contendo a seguinte quantidade de neurônios para cada caso: 530 nodos para o caso maiúsculo, 450 nodos para o caso minúsculo e 400 nodos para o caso maiúsculo+minúsculo. A quantidade de nodos na camada escondida para cada caso, foi baseada em testes preliminares para verificar a melhor configuração para a resolução do problema;
- Uma camada de saída contendo 26 neurônios para os casos maiúsculo e minúsculo ("A" a "Z" e "a" a "z" respectivamente) e 39 neurônios para o caso maiúsculo + minúsculo ((A, B, C+c, D, E, F+f, G, H, I, J+j, K+k, L, M+m, N+n, O+o, P, Q, R, S, T, U+u, V+v, W+w, X+x, Y+y, Z+z) + (a, b, d, e, g, h, i, l, p, q, r, s, t)).

Dois diferentes experimentos foram realizados: Primeiro dividindo os casos maiúsculos e minúsculos, e outro experimento com ambos os casos. Para este último, como algumas letras apresentam a mesma forma em ambas as versões maiúsculas e minúsculas, elas foram agrupadas em uma única classe (c, x, o, w, y, z, m, k, u, j, n, f, v). Camastra [Camastra 2007] utilizou uma análise de cluster para verificar se as versões maiúsculas e minúsculas das mesmas letras têm formas similares. As letras (c, x, o, w, y, z, m, k, u, j, n, f, v) apresentaram maior similaridade entre as duas versões e foram agrupadas em uma única classe. Os resultados para os casos separados e juntos são apresentados nas tabelas 3 e 4.

Tabela 3. Taxa de reconhecimento para letras maiúsculas e minúsculas separadas.

Classificador	Nº de nodos	Caso	Taxa de reconhecimento (%)
MLP	530	Maiúsculo	86.83
MLP	450	Minúsculo	82.12

Tabela 4. Taxa de reconhecimento para ambos os casos.

Classificador	Nº de nodos	Caso	Taxa de reconhecimento (%)
MLP	400	Maiúsculo + Minúsculo	82.18

6.2. Análise de Confusão

Com as redes MLP treinadas para cada caso (maiúsculo, minúsculo e maiúsculo+minúsculo) foram geradas as suas matrizes de confusão, (A figura 6 ilustra a parte referente às maiores confusões ocorridas nos experimentos com letras maiúsculas. Devido a grande quantidade de classes, 26 classes para o caso maiúsculo/minúsculo e 39 classes para o casso maiúsculo + minúsculo, é inviável exibir as matrizes de confusão completa para todos os casos) para que se possam analisar os pares de caracteres os quais a rede obteve maior número de confusão.

	A	B	C	H	K	L	N	O	R	S	U
A	308	8	4	0	1	3	8	4	5	4	0
C	1	5	370	0	0	9	1	5	1	0	0
D	1	8	2	2	1	1	3	17	0	3	0
E	1	2	9	0	2	0	0	0	9	1	0
G	0	11	5	1	0	1	1	3	1	4	0
H	4	5	0	132	3	0	11	0	0	4	0
J	1	3	0	0	0	0	6	0	0	15	1
L	0	0	4	3	0	249	1	0	4	17	0
M	1	1	0	8	1	0	25	0	2	3	0
N	6	7	1	4	4	1	545	2	3	3	0
R	11	5	5	1	1	1	4	0	277	0	1
S	1	11	4	0	0	2	2	4	2	517	1
V	0	0	0	0	1	1	4	0	0	0	14
W	0	0	1	0	0	0	10	0	0	1	1
X	2	1	0	0	6	2	4	0	7	1	0

Figura 6. Parte da matriz de confusão referente às maiores confusões ocorridas nos experimentos com letras maiúsculas.

6.3. Processo de treinamento SVM

Nos diferentes SVMs derivados dos pares de classes (EX. (U, V), (m, n), (N, n), entre outras) que constituem a maioria das confusões da MLP, como mostra a figura 6, diferentes funções de kernel (linear, polynomial and RBF) foram testadas e os melhores desempenhos foram obtidos pelos SVMs treinados com a função de kernel RBF.

6.4. Arquitetura Híbrida MLP-SVM

Baseado nas matrizes de confusão analisadas, a arquitetura híbrida para cada caso foi construída da seguinte forma:

- Caso maiúsculo: 1 MLP + 21 SVMs;
- Caso minúsculo: 1 MLP + 32 SVMs;
- Caso maiúsculo+minúsculo: 1 MLP + 33 SVMs.

A escolha dos pares de classes foi baseada na quantidade de erros tomando como valor mínimo 10% do tamanho do conjunto de teste.

A tabela 5 apresenta os resultados obtidos pelo nosso reconhecedor híbrido MLP-SVM.

Tabela 5. Resultados obtidos pelo reconhecedor MLP-SVM.

Caso	MLP (%)	SVM+MLP (%)	Melhoria (%)
Maiúsculo	86.83	90.48	3.65
Minúsculo	82.12	88.84	6.72
Maiúsculo + Minúsculo	82.18	87.12	4.94

Pode-se observar que todos os resultados obtidos pelo nosso método apresentou um ganho significativo quando comparado com qualquer par (extração de características, classificador) expostos nas tabelas 1 e 2 (as comparações realizadas com outros trabalhos tem que levar em consideração que os experimentos, apesar de serem

realizados com as mesmas bases de dados podem não ter sido realizados sobre as mesmas condições, visto que foram conduzidos em momentos diferentes e por pesquisadores diferentes). Vale ressaltar a simplicidade, a velocidade de treinamento e o baixo custo computacional do método, comparado com técnicas de extração de características mais complexas [Trier et al. 1995] que exigem, na maioria dos casos, custo computacional elevado e maior tempo de treinamento.

Os melhores resultados obtidos nos últimos anos, para a base de dados C-Cube, são apresentados na Tabela 6. O reconhecedor híbrido MLP-SVM proposto superou todos os resultados anteriores sobre o *Split B* da base de dados por uma boa margem. Os resultados usando o *Split A* não foram comparados, pois as imagens dos caracteres desta versão da base de dados não estão disponíveis no site. Também é importante observar que todos os melhores resultados do passado são baseados em modelos mais complexos, que levam mais tempo para serem treinados e têm alto custo computacional. Assim, provamos que usando uma técnica de extração de características simples combinada com um método híbrido formado por dois classificadores simples (MLP-SVM), temos as principais vantagens:

1. Processo de treinamento rápido: Em métodos de alta complexidade, que apresentam uma convergência lenta, o processo de treinamento pode parar em regiões de mínimo local onde os gradientes são nulos;
2. Baixo custo computacional: Métodos complexos, por sua vez, apresentam cálculos de alta complexidade, que podem causar perda de eficiência do sistema;
3. Melhores resultados do que os obtidos por modelos de alta complexidade apresentados na tabela 6.

Tabela 6. Taxas de Reconhecimento para a base de dados C-Cube. RBF = Radial Basis Network, HVQ = Hierarchical Vector Quantization, MDF = Modified Directional Features, SVM = SVM with Radial Basis Kernel.

Algoritmo	Taxa de Reconhecimento (%)
HVQ-32 [Thornton et al. 2008]	84.72
HVQ-16 [Thornton et al. 2008]	85.58
MDF-RBF [Thornton et al. 2009]	80.92
34D-RBF [Thornton et al. 2009]	84.27
MDF-SVM [Thornton et al. 2009]	83.60
34D-SVM + Neural GAS [Camastra 2007]	86.20
34D-MLP [Camastra 2007]	71.42
Método proposto	**87.12**

7. Análise de Erros

Alguns dos caracteres com as maiores taxas de erro são apresentados na tabela 7. Esses caracteres (g, p, V, Z e Q), são os que apresentam poucos padrões em toda a base de dados como é mostrado nas figuras 2 e 3. A rede pode não aprender a classificar esses padrões por causa do número de padrões desequilibrado entre as diferentes classes. Isso pode explicar a alta taxa de erros para essas letras. Para o caractere p, que é o caractere com o menor número de imagens na base de dados, existem apenas 9 imagens no conjunto de treinamento e 4 imagens no conjunto de teste. Todas as 4 imagens de teste

deste caractere foram classificadas erroneamente devido ao baixo número de amostras na base de dados. Para resolver este problema, uma possível solução é criar novos padrões, para as classes desequilibrada, com algumas abordagens que são amplamente utilizadas no reconhecimento de dígitos manuscritos como elastic distortions [Simard et al. 2003], [Ranzato et al. 2008] e affine transformations [Lauer et al. 2007].

Tabela 7. Caracteres com as maiores taxas de erro.

Letra	No. Erros	No. de padrões no conjunto de teste	(%)
g	39	130	30.00
p	4	4	100.00
v	35	67	52.23
z	5	9	64.43
Q	7	18	38.88

8. Conclusões

Neste artigo, um método para aumentar as taxas de reconhecimento de caracteres manuscritos combinando MLP+SVM é proposto. Os experimentos demonstraram que a combinação das redes MLPs com SVMs especialistas em pares de classes que constituem as maiores confusões da MLP, apresentam melhoria no desempenho em termos de taxa de reconhecimento. Os resultados mostram que houve melhoria, de 3,65% a 6,72%, na taxa de reconhecimento para todos os casos testados (maiúsculo, minúsculo e maiúsculo+minúsculo).

Como proposta para trabalho futuro, pretende-se utilizar diferentes técnicas de extração de características combinados com classifier ensemble em nosso sistema híbrido MLP-SVM, tendo como motivação conseguir melhores taxas de reconhecimento para o reconhecedor.

Agradecimento

Este trabalho foi apoiado pela FACEPE - Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco.

Referências

Aires, S. B. K., Reconhecimento de caracteres manuscritos baseados em regiões perceptivas. Dissertação de Mestrado. Pontifícia Universidade Católica do Paraná. Curitiba. 2005.

Bellili, A., Gilloux, M., and Gallinari, P., "An Hybrid MLP-SVM Handwritten Digit Recognizer," icdar, pp.0028, Sixth International Conference on Document Analysis and Recognition (ICDAR'01), 2001.

Camastra, F., “A svm-based cursive character recognizer,”Pattern Recognition, vol. 40, no. 12, pp. 3721–3727, 2007.

Camastra, F., Spinetti, M., and Vinciarelli, A., “Offline cursive character challenge: a new benchmark for machine learning and pattern recognition algorithms.,” Proceedings of the 18th International Conference on Pattern Recognition, pp. 913–916, 2006.

Cruz, R. M. O., Cavalcanti, G. D. C. and Ren, T. I., “An Ensemble Classifier For Offline Cursive Character Recognition Using Multiple Feature Extraction Techniques”; IEEE International Joint Conference on Neural Networks (IJCNN), Barcelona, p. 744-751, 2010.

Goonatilake, S. and Khebabal, S. (Eds.). Intelligent Hybrid Systems, vol. 1. John Wiley and Sons, 1995.

Haykin, S. Neural Networks - A Comprehensive Foundation. [S.I.]: Prentice Hall, 1996.

Ho, T. K., Hull, J. J., Srihari, S.N. (1994) Decision Combination in Multiple Classifier Systems. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(1):66–75.

Lauer, F., Suen, C. Y., and Bloch, G., "A trainable feature extractor for handwritten digit recognition," Pattern Recognition, vol. 40, no. 6, pp. 1816–1824, 2007.

Nadal, C., Legault, R., Suen, C. Y. (1990) Complementary Algorithms for the Recognition of Totally Unconstrained Handwritten Numerals. In: Proceedings of the tenth International Conference on Pattern Recognition 1990. Atlantic City, NJ, USA, pp 443–449.

Plamondon, R. and Srihari, S. N., "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.22, p. p. 63-84, 2000.

Ranzato, M., Boureau, Y., and LeCun, Y., "Sparse feature learning for deep belief networks," Advances in Neural Information Processing Systems, pp. 1185–1192, 2008.

Riedmiller, M. and Braun, H., "A direct adaptive method for faster backpropagation learning: The rprop algorithm," Proceedings of the IEEE International Conference on Neural Networks, pp. 586–591, 1993.

Rodrigues, R. J., Kupac, G. V., Thomé, A. C. G.; "Character Feature Extraction Using Polygonal Projection Sweep (Contour Detection)", IWANN2001, Granade - Spain, June 13 a 15, LINCS, 2001, pp 687-695, 2001.

Rodrigues, R. J., Silva, E., Thomé, A. C. G., "Extração de Características para o Reconhecimento de Letras Manuscritas", V Simpósio Brasileiro de Automação Inteligente - V SBAI, Canela, novembro de 2002.

Rodrigues, R. J., Silva, E., Thomé, A. C. G., "Feature Extraction Using Contour Projection"; 5th World Multiconference on Systemics, Cybernetics and Informatics SCI2001 – Orlando – USA, July 22 – 25.

Rumelhard, D.E. and Weigend, S.A., "Predicting the Future: A Connectionist Approach", International Journal of Neural Systems, pp. 193-209, 1990.

Simard, P. Y., Steinkraus, D., and Platt, J. C., "Best practices for convolutional neural networks applied to visual document analysis," International Conference on Document Analysis and Recognition, vol. 2, pp. 958–963, 2003.

Thornton, J., Blumenstein, M., Nguyen, V., and Hine, T., "Offline cursive character recognition: A state-of-the-art comparison," 14th Conference of the International Graphonomics Society, 2009.

Thornton, J., Blumenstein, M., Nguyen, V., and Hine, T., "Offline cursive character recognition: A state-of-the-art comparison,"14th Conference of the International Graphonomics Society, 2009.

Thornton, J., Faichney, J., Blumenstein, M., and Hine, T., "Character recognition using hierarchical vector quantization and temporal pooling," Proceedings of the 21st Australasian Joint Conference on Artificial Intelligence, pp. 562–572, 2008.

Trier, O. D., Jain, A. K., and Taxt, T., "Feature extraction methods for character recognition: A survey," Pattern Recognition, vol. 29, no. 4, pp. 641–662, 1995.

Vamvakas, G., Gatos, B., Perantonis, S. J., "Handwritten character recognition through two-stage foreground sub-sampling"; Pattern Recognition 43 (2010) 2807–2816.

Vapnik, V. N., "An overview of statistical learning theory"; IEEE Trans. on Neural Networks, 10(5):988–999, 1999.

Veloso, L. R., "Reconhecimento de Caracteres Numéricos Manuscritos." Dissertação de Mestrado, Universidade Federal da Paraíba, Campina Grande, 1998.

Woods, K., Jr, W. P. K., and Bowyer, K.. Combination of multiple classifiers using local accuracy estimates. IEEE Trans. on PAMI, 19(4):405–410, 1997.

Proposta de uma Metodologia para Aquisição de Conhecimento Tácito Coletivo

Cláudio Roberto do Rosário, Liane Mahlmann Kipper, Rejane Frozza, Jacques Nelson C. Schreiber

Programa de Pós-Graduação em Sistemas e Processos Industriais – Mestrado, UNISC - Universidade de Santa Cruz, Avenida Independência, 2293 - Bloco 53 - 4º andar - Sala 5340, CEP: 96815-900, Santa Cruz do Sul – RS, Brasil.

claudio.eqs@brasilata.com.br;{liane,frozza,jacques}@unisc.br

Resumo. *O artigo apresenta a proposta de uma metodologia para aquisição de conhecimento tácito coletivo, a fim de torná-lo explícito objetivando auxiliar no diagnóstico de causa de defeitos em processos produtivos. Para o processo de aquisição de conhecimento e mapeamento dos atributos e valores para posteriormente serem utilizados na base de conhecimento de um Sistema Especialista, foram utilizadas ferramentas de qualidade como Brainstorming, Gráfico de Pareto e Diagrama de Ishikawa, ferramentas normalmente utilizadas para auxiliar na resolução de problemas em processos industriais. Para estruturar a metodologia, foi utilizada a técnica da Sistemografia, que preconiza o pensamento sistêmico como forma de observação de fenômenos complexos e modelagem de objetos nas organizações.*

Palavras-Chave: *Aquisição de conhecimento, Conhecimento tácito e explícito, Sistema especialista.*

Abstract. *This paper shows the development of a methodology for the acquisition of collective tacit knowledge which aims to make it into explicit objecting in order to help diagnose cause of defects in production processes. For the process of knowledge acquisition and mapping of attributes and values for later use in the knowledge base of an Expert System, were used as quality tools Brainstorming, Pareto Chart and Diagram Ishikawa, tools normally used to assist in resolving problems in industrial processes. The methodology was structured with the systemographs technique as a way to observe complex phenomena and model objects in organizations.*

Keywords*: Acquisition of knowledge, Tacit and explicit knowledge, Expert system.*

1. Introdução

Para Bertalanffy (1973), as limitações fisiológicas e paradigmáticas, que determinam a capacidade humana, limitam a percepção da realidade dos fenômenos em sua totalidade. Daí advém a necessidade de empregar ferramentas computacionais para auxiliar no processo de compreensão dos fenômenos.

Conceitos de Inteligência Artificial, especificamente, Sistemas Especialistas (SE), serão empregados no trabalho de pesquisa para apoiar o uso da gestão do

conhecimento tácito, a fim de disseminá-lo nos processos internos e promover o aprendizado organizacional.

Segundo Rezende (2003), o objetivo da Inteligência Artificial (IA) é capacitar o computador para executar funções exercidas pelo ser humano. Desta forma, a incorporação de conhecimento é parte fundamental para o sucesso de um Sistema Inteligente. Tal afirmação vem ao encontro do objetivo proposto pelo estudo, mapear a aquisição de conhecimento tácito coletivo, gerando um sistema de gestão de conhecimento.

Para Silva (2004), Medeiros (2009) e Kim (2011), conhecimento explícito é formal, pode ser comunicado e partilhado; já o conhecimento tácito está relacionado com os modelos mentais, crenças e perspectivas dos indivíduos de forma cognitiva. Segundo Hanashiro e Cleto (2007), o conhecimento tácito é um grande potencial para trazer vantagem competitiva para as empresas, possibilitando ganhos em termos de produtividade, qualidade e custo. Ainda, segundo Hanashiro (2007), a gestão de conhecimento significa organizar e sistematizar, em todos os pontos de contato, a capacidade da empresa em captar, gerar, analisar, utilizar, disseminar e gerenciar a informação.

O objetivo do estudo é elaborar uma metodologia que permita o mapeamento dos conhecimentos tácitos coletivos relevantes que possam contribuir para a resolução de problemas na indústria. O propósito da metodologia é sistematizar a etapa de aquisição do conhecimento na modelagem de um SE.

Como metodologia para modelagem da base de conhecimento de um Sistema Especialista foi empregada a técnica de Sistemografia. A técnica auxiliará na forma pela qual o estudo está sendo estruturado, pois permite abordar, entender e interpretar o fenômeno de forma sistêmica. Durante a aplicação das etapas da metodologia serão utilizadas ferramentas de qualidade como *Brainstorming*, Gráfico de Pareto e Diagrama de Ishikawa, ferramentas normalmente utilizadas para auxiliar na resolução de problemas em processos industriais. No desenvolvimento deste estudo, o gráfico de Pareto e *Brainstorming* foram usados sistematicamente para preencher o Diagrama de Ishikawa. Já o Diagrama de Ishikawa será utilizado como forma de aquisição de conhecimento para definição de atributos e valores para modelar a rede semântica e construir as regras de produção de um Sistema Especialista.

O artigo está organizado nas seguintes seções: a seção apresenta uma breve fundamentação teórica dos assuntos envolvidos na pesquisa; a seção 3 descreve a metodologia proposta e empregada; a seção 4 discute os resultados; e a seção 5 apresenta as considerações finais.

2. Fundamentação Teórica

Esta seção apresenta a base teórica referente aos temas envolvidos no desenvolvimento do estudo, como Sistemografia (Abordagem sistêmica para observação de fenômenos complexos), Ferramentas da Qualidade, Sistema Especialista (SE) e Aquisição de Conhecimento (AC).

2.1. Abordagem Sistêmica para Observação de Fenômenos Complexos na Indústria

Segundo Neto e Fontana (2008), a modelagem de sistemas advém de um modelo geral, ou seja, modelar é conceber, para um objeto, um modelo que permita conhecê-lo, compreendê-lo, interpretá-lo e que auxilie na antecipação do seu comportamento. Segundo Bertalanffy (1973) a Teoria Geral de Sistema tem como propósito criar teoria e auxiliar na compreensão de fenômenos complexos, mediante à formação de modelos conceituais que possam representar as situações reais.

Em 1996, Le Moigne e Bartoli definiram a sistemografia como sendo um procedimento pelo qual o modelador constrói modelos de fenômenos complexos para objetos. Segundo Le Moigne e Bartoli (1996), para modelar objetos o modelador deve desempenhar papel ativo na construção do modelo.

Para Kintschener (2005) sistemografia é um método de mapeamento e reorganização de processos, possui enfoque sistêmico procurando disciplinar o bom-senso e a instuição através de um processo lógico e de uma análise formal do problema, procurando estudá-lo como um todo, preocupando-se com as interfaces entre suas partes. O papel principal da sistemografia é identificar, mapear e detalhar a análise de um processo de forma que se tenha o efetivo entendimento do problema, estuda o conjunto de elementos e as relações entre esses elementos.

Segundo Leite et al (2009), com base na Teoria Geral de Sistemas, Le Moigne (1977) desenvolveu a Sistemografia a qual tem como função auxiliar no processo de modelagem de objetos em um sistema complexo. A abordagem sistêmica pode ser empregada por meio da Sistemografia de acordo com cinco etapas, conforme a figura 1.

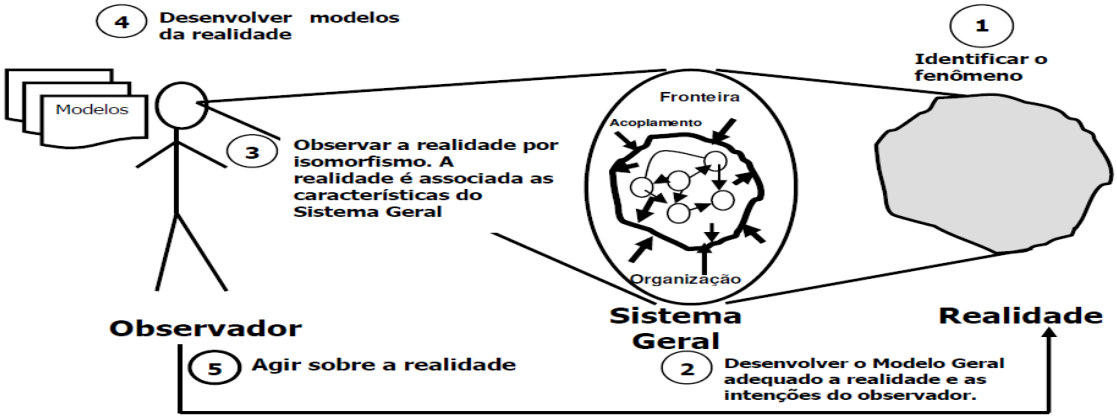


Figura 1. Etapas da Sistemografia

Fonte: Adaptado de Le Moigne (1990)

Tais etapas estão sendo adaptadas à realidade dos processos industriais, a fim de mapear, adquirir e tornar explícito o conhecimento tácito das pessoas envolvidas em processos industriais.

2.2. Ferramentas da Qualidade

Esta seção apresenta as ferramentas da qualidade estudadas e empregadas nesta pesquisa.

2.2.1. *Brainstorming*

Para Oliveira (1995) e Aguiar (2002), *Brainstorming* é uma ferramenta que auxilia a descoberta das causas de um problema utilizando o conhecimento das pessoas sobre o assunto em estudo. Destina-se à geração de idéias/sugestões que permitam avanços significativos na busca de soluções de problemas.

O *Brainstorming* será utilizado no estudo durante as reuniões com os especialistas humanos.

2.2.2. Gráfico de Pareto

Para Ramos (2000), o gráfico de Pareto é usado quando é preciso dar atenção aos problemas de uma maneira sistemática e, também, quando se tem um grande número de problemas e recursos limitados para resolvê-los. O gráfico indica as áreas mais problemáticas, seguindo uma ordem de prioridades. Tais prioridades podem recorrer entre 70 a 80% do problema em sua totalidade.

Para Kume (1993), existem duas formas de trabalhar com o gráfico de Pareto, gráfico de Pareto por Defeito e gráfico de Pareto por Causa. Para o estudo optou-se por trabalhar o gráfico de Pareto por Defeito. O gráfico por defeito apresenta o efeito causado pelo problema. No presente estudo o efeito é o defeito causado no produto ou peça que, ao ser detectado no momento da confecção, causa parada de produção.

Nesta pesquisa, o gráfico de Pareto será utilizado para mapear o tipo de conhecimento a ser priorizado para aplicar a metodologia de aquisição de conhecimento.

2.2.3. Diagrama Ishikawa (espinha de peixe)

O resultado de um processo pode ser atribuído a uma grande quantidade de fatores, e uma relação de causa-e-efeito pode ser encontrada entre esses fatores. Pode-se determinar a estrutura ou a relação de causa-e-efeito múltipla, observando o processo sistematicamente. É difícil resolver problemas complexos sem considerar esta estrutura, a qual consiste em uma cadeia de causas e efeitos, e um diagrama de causa e efeito é um método simples e fácil de representá-la (Kume, 1993).

Para Ramos (2000), o diagrama de Ishikawa representa uma relação significativa sobre um efeito e suas possíveis causas. Este diagrama descreve situações complexas, que seriam muito difíceis de serem descritas e interpretadas somente por palavras. Existem, provavelmente, várias categorias de causas principais. Frequentemente, estas recaem sobre uma das seguintes categorias: *Mão–de–Obra, Máquina, Método, Materiais, Meio Ambiente e Meio de Medição,* Ramos (2000), ver figura 2.

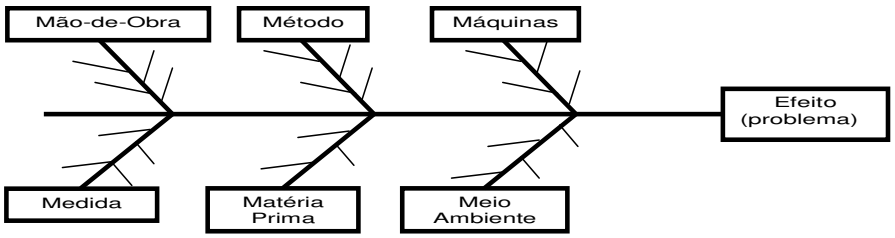


Figura 2. Diagrama de Ishikawa
Fonte: Ramos (2000)

O diagrama de Ishikawa será utilizado para dar foco na entrevista não estruturada para aquisição de conhecimento, o engenheiro do conhecimento questionará o especialista sobre os 6 m´s que envolvem o processo de manufatura.

2.3. Sistema Especialista (SE)

Para Bittencourt (2006), pode-se afirmar que os Sistemas Especialistas (SE) são técnicas de Inteligência Artificial (IA) desenvolvidas para resolver problemas em um determinado domínio cujo conhecimento utilizado é obtido de pessoas que são especialistas naquele domínio.

O sistema especialista é "informado" sobre as características do problema e decide, durante o processamento, qual o caminho mais provável de encontrar a solução. Uma das principais aplicações para os sistemas especialistas são os sistemas de diagnóstico. Esses sistemas são capazes de deduzir possíveis problemas a partir de observações ou sintomas.

A arquitetura de um sistema especialista é dividida por dois componentes básicos: um banco de informações que contenha todo o conhecimento relevante sobre o problema de uma forma organizada (base de conhecimento); um conjunto de métodos inteligentes de manipulação destes conhecimentos, os mecanismos de inferência.

2.4. Aquisição de Conhecimento (AC)

Para Cairo (1998), Mastella (2005) e Kim et al (2011) a aquisição do conhecimento é a principal atividade na gestão de Sistemas Baseados em Conhecimentos (SBC), bem como a fase crítica e o gargalo do Sistema, por isso é fundamental investigar onde e como os agentes do processo acessam e adquirem o conhecimento que necessitam. Neste sentido, as fontes de aquisição de conhecimento se tornam a parte crítica do processo de aquisição de conhecimento. Kim et al (2011) aponta um outro desafio na aquisição de conhecimento, é a aquisição de conhecimento em tempo hábil para resolução de um problema. Kim et al (2011) afirma que há três formas de fontes de conhecimento: fontes diádica, publicações e conhecimento de um grupo. Uma fonte de conhecimento diádica se trata de aquisição do conhecimento através do contato direto entre o receptor e o provedor do conhecimento. Uma fonte de conhecimento publicada se enquadra como documentos, livros e postada em comunidade virtual. Uma fonte de conhecimento de um grupo refere-se à troca de conhecimentos entre os múltiplos provedores de conhecimento, onde os receptores de conhecimento podem acessar e obter o conhecimento através de fontes de conhecimento do grupo, como uma conversa aberta ou pública entre os envolvidos, através de sistemas de perguntas e respostas, ou equipes de trabalho. Cairo (1998) apontou a necessidade de pesquisas voltadas para a forma de aquisição de conhecimento, ou seja, modelos projetados para gerenciar a aquisição de conhecimento e processo de modelagem do conhecimento. Para Cairo (1998) e Mastella (2005) os modelos para estruturar a forma de aquisição de conhecimento devem promover ferramentas que estruture a forma de raciocínio do engenheiro do conhecimento. Cairo (1998), Wagner (2003) e Mastella (2005) apontam dez formas de aquisição do conhecimento, entrevista estruturada,

entrevista não estruturada, entrevista dirigida, pesquisa, classificação de fichas, comparação triádica, grades de repertório, técnica de observação, limitação de informação, relatório verbais e análise de protocolo. Considerando o estado da arte referente à Aquisição do Conhecimento, elaborou-se o quadro 1 que demonstra a relação entre as formas de aquisição do conhecimento e as fontes de conhecimento.

Quadro 1 : Fonte de conhecimento e forma de aquisição do conhecimento.

Formas de aquisição do conhecimento	Fontes de conhecimento		
	Diádica	Publicações	Conhecimento de um Grupo
Entrevista estruturada	X		X
Entrevista não estruturada	X		X
Entrevista dirigida	X		X
Pesquisa		X	
Classificação de Fichas		X	
Comparação Triádica	X		X
Grades de Repertório	X		X
Técnica de Observação	X		X
Limitação de Informação	X		X
Relatórios Verbais	X		X
Análise de protocolo		X	X

Fontes: Adaptado de Cairo (1998), Wagner (2003), Mastella (2005), e Kim et al (2011)

O quadro 1 apresenta as possíveis formas de aquisição de conhecimento e suas respectivas fontes de conhecimento. Para o desenvolvimento deste trabalho utilizou-se as fontes de conhecimento e formas de aquisição do conhecimento apresentadas no quadro 2.

Quadro 2 : Fontes de conhecimento e formas de aquisição utilizadas.

Formas de aquisição do conhecimento	Fonte de conhecimento		
	Diádica	Publicações	Conhecimento de um Grupo
Entrevista não estruturada	X		X
Entrevista dirigida	X		X
Limitação de Informação	X		X
Análise de protocolo		X	X

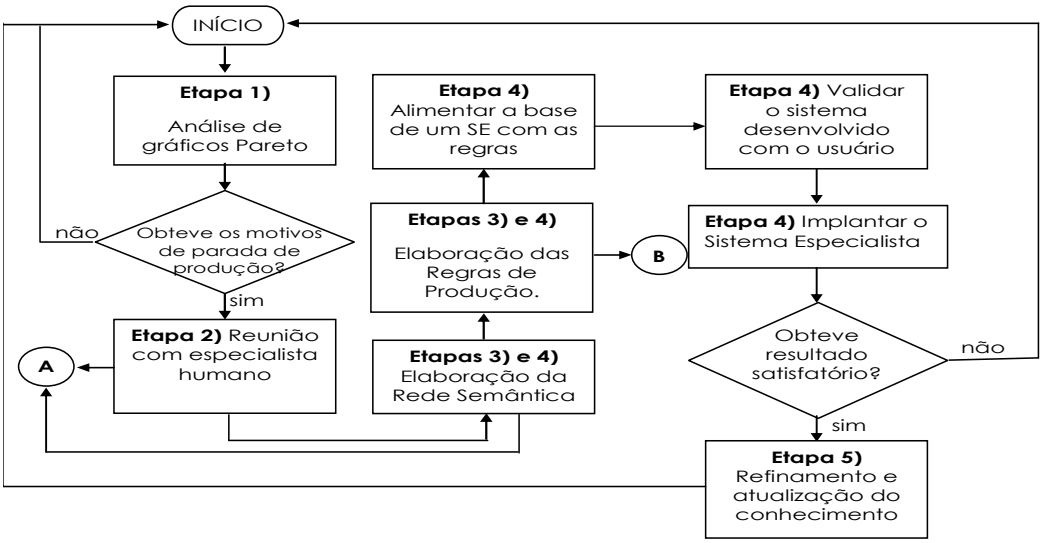
O quadro 2 apresenta-se a forma que a metodologia para aquisição de conhecimento tácito coletivo foi estruturada, observa-se que a metodologia está alinhada à prática já estabelecida pela comunidade científica.

3. Metodologia

O método empregado na pesquisa foi classificado como exploratório, realizando um estudo preliminar com a finalidade de melhor adequar a proposta da metodologia para mapeamento e aquisição de conhecimento tácito coletivo à realidade que se pretende conhecer. E, desta forma, mapeá-lo para um sistema especialista, tornando-o explícito.

3.1. Proposta de uma Metodologia para Aquisição do Conhecimento

A partir dos estudos realizados até o momento e da aplicação da técnica de sistemografia, a figura 3 representa a metodologia proposta para promover uma sequência de atividades, a fim de mapear e tornar conhecimento tácito coletivo em explícito em processos produtivos.



Legenda:
A: Utilizar o Diagrama de Ishikawa e *Brainstorming* como ferramentas de apoio para o processo de aquisição de conhecimento.
B: Utilizar a Rede Semântica como técnica de apoio representacional.

Figura 3. Metodologia proposta para aquisição do conhecimento tácito coletivo
Fonte: Elaborado pelos Autores

4. Resultados e Discussões

Esta seção apresenta a aplicação da metodologia, descrita na seção 3.1, em uma empresa Metalúrgica do Ramo de Embalagens Metálica, localizada no Sul do Brasil. A Empresa emprega a metodologia *Balanced Scorecard* (BSC) para medição e gestão de desempenho do seu processo. Processo Interno é uma das quatro perspectivas da metodologia BSC que refletem a visão e a estratégia da empresa. Para tal perspectiva, a empresa declara em seu mapa estratégico, *"manter a inovação como forma de diferenciação e criação de valor"*, como uma das estratégias adotada pela Empresa para gestão dos processos. Tal diretriz vem ao encontro do propósito e objetivos da pesquisa, que consiste em aprimorar a prática de diagnosticar as causas dos defeitos ocorridos no processo produtivo. Para aplicação da metodologia, foi escolhido o indicador de retrabalho para promover a aquisição do conhecimento tácito coletivo.

Etapa 1: Identificar o fenômeno

A empresa possui 11 linhas de montagem de embalagem metálica, lata 900 ml, Lata 1 Litro retangular, Galão 3,6 Litros, Lata 4 Litros retangular, 18 Litros Quadrada, Balde 20 Litros, Aerossol 200 ml (ABM 52), Aerossol 300 ml (ABM 57), Aerossol 400 ml (VAK 57), Aerossol 450 ml (VAA20) e Aerossol 500 ml (VAA20).

Para dar prioridade e foco no mapeamento dos motivos que contribuem para o aumento do indicador de retrabalho, foram gerados dois gráficos de Pareto, gráfico por linha de montagem e gráfico por motivo dos retrabalhos, conforme figuras 4 e 5 respectivamente. O gráfico da figura 4 evidencia que 75% dos retrabalhos estão localizados nas linhas de montagem de lata de Aerossol.

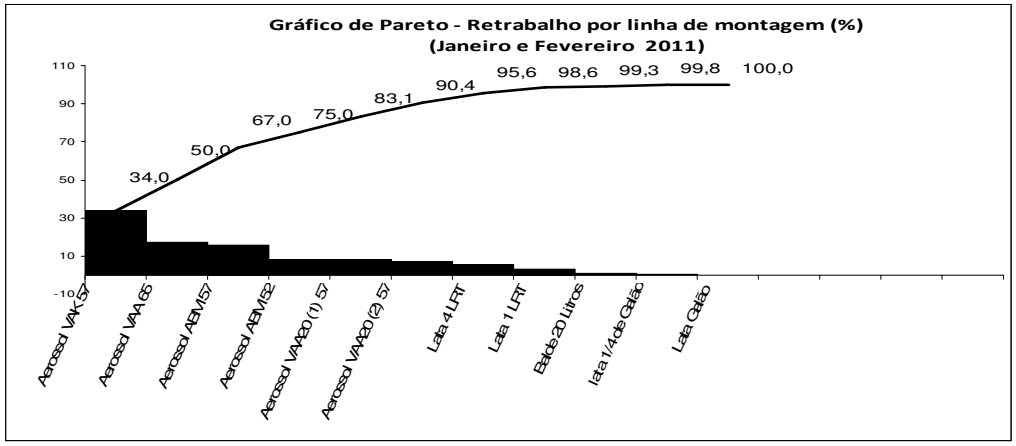


Figura 4. Gráfico de Pareto indicando as linhas com maior frequência de parada para retrabalho

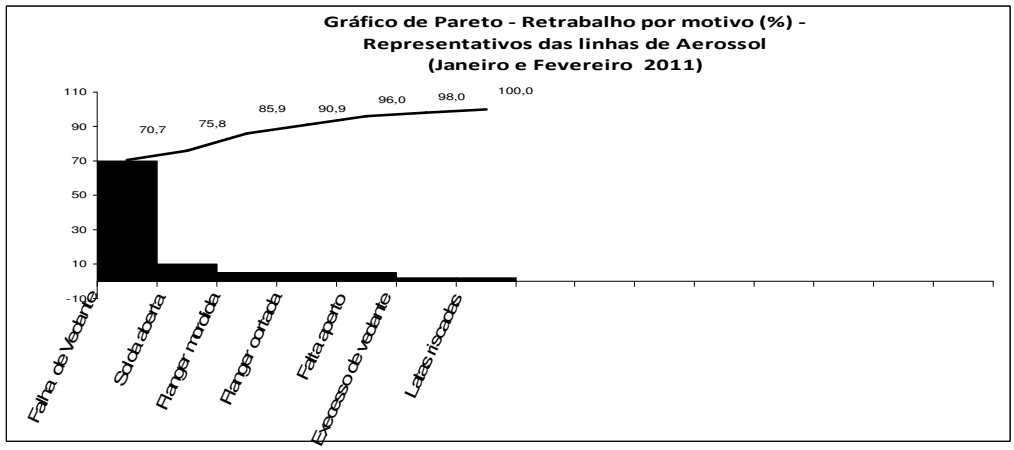


Figura 5. Gráfico de Pareto indicando os motivos com maior frequência de parada para retrabalho

O gráfico evidencia que 70,7% dos retrabalhos nas linhas de montagem de lata de Aerossol são por motivo de falha de vedante.

Nesta etapa, o conhecimento a ser adquirido se encontra mapeado, incide sobre a causa de defeito que leva à falha de vedante, a qual representa 70,7% do retrabalho no período estudado.

Etapa 2: Desenvolver o modelo Geral adequado à realidade e às intenções do observador

Foram realizadas reuniões coletivas e individuais com os envolvidos no processo produtivo. Na primeira reunião coletiva foi exposto o objetivo do estudo e apresentado

conceitos de Inteligência Artificial (IA), com ênfase na construção da rede semântica e regras de produção e o papel do especialista humano. Após a primeira reunião, de forma individual, foi realizado o mapeamento das características que poderiam contribuir para a causa do defeito de falha de vedante conforme quadro 3. Neste momento ocorreram divergências entre os entrevistados, Mecânicos e Inspetores de Qualidade. As divergências foram esclarecidas em reuniões coletivas até que foi estabelecido um padrão entre as características. Durante as reuniões coletivas foi aplicada a técnica de *Brainstorming* e o Diagrama de Ishikawa foi aplicado para dar foco na elaboração do quadro 3. O quadro 3 representa o modelo geral o qual agrega os conhecimentos acessíveis aos envolvidos no processo

Quadro 3. Mapeamento dos valores (possíveis causas) que contribuem para a ocorrência de falha de vedante

Atributo (Defeito)	Valor (possíveis causas)
FALHA DE VEDANTE	**MEDIDA** Quantidade de parafina no lubrificante <= 1,250 kg a cada 200 litros de hexano Índice de Capabilidade CP >= 1,00 Índice de Capabilidade CPK >= 1,33 Diâmetro interno do bico da aplicadora de vedante >= 0,5 mm ou <= 0,8 mm **MÃO-DE-OBRA** Treinada no procedimento I.82.029 Auditada no procedimento I.82.029 **MÉTODO** Freqüência de Inspeção <= 2 horas Tempo de agitação do tambor de vedante >= 15 min. ou < 20 min. Rotação da haste de agitação de vedante = 200 rpm. Limpeza do Filtro de vedante antes do início do turno. Limpeza do rolo aplicador de hexano. Regulagem do sensor de presença de componentes, acusa componentes. Regulagem de pressão do ar comprimido entrada tambor >= 1,0 bar ou <=2,5 bar. Pressão de entrada da bomba do vedante >= 1,5 ou <= 2,5 bar Pressão de saída da bomba do vedante <= 3,0 bar Regulagem de pressão do ar comprimido agulha >= 3 ou <= 3,5 Regulagem de pressão do ar comprimido separador >= 2 bar ou <=2,5 bar **MEIO-AMBIENTE** Temperatura ambiente estoque >= 5ºC ou <= 40ºC. **MATÉRIA PRIMA** Viscosidade do vedante >= 2150 cP ou <= 2850 cP. Densidade do vedante = 1,20 g/ml Teor de sólidos do vedante >= 50% ou <= 54% **MÁQUINA** Rotação do prato (aceleração centrípeda) >= 60 Hz ou <= 80 Hz Tempo de aplicação de vedante >= 0,080 ou <= 0,140 segundos Temperatura de set pont do estufa >= 80ºC ou <= 120ºC Última manutenção mecânica <= 12 meses

Etapa 3: Observar a realidade por isomorfismo com base no modelo geral

Através dos dados coletados sistematicamente na etapa 2, conforme quadro 3, a rede semântica é modelada a fim de representar a forma pela qual o fenômeno, neste caso, a causa do defeito, será diagnosticada, considerando a possível causa do defeito como o nodo raiz; e os 6 Ms do diagrama de Ishikawa, como os arcos dos nodos secundários. Considera-se a rede semântica, neste estudo, como uma forma isomórfica de observação do fenômeno, conforme apresenta a figura 6.

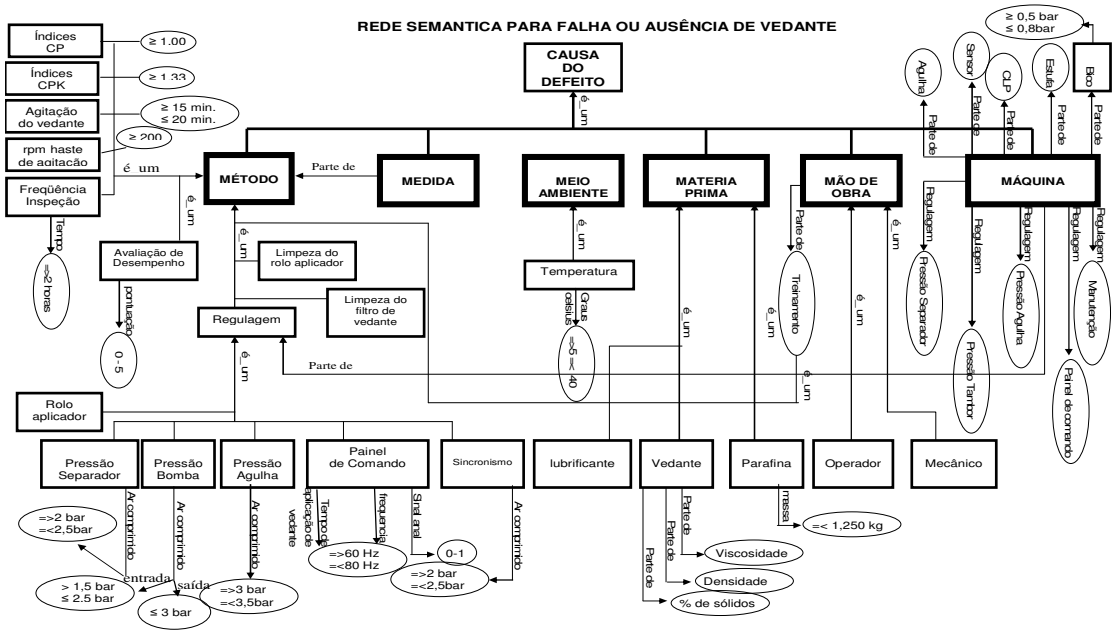


Figura 6. Rede semântica representando as possíveis causas de falha de vedante

Etapa 4: Desenvolver modelos da realidade

A realidade passa a ser representada por modelos que permitem torná-la mais clara para a comunidade envolvida. As regras de produção (conhecimento explícito) foram elaboradas a partir da rede semântica. As regras de produção devem ser discutidas de forma coletiva com o uso de *Brainstorming*. Após a elaboração das regras, as mesmas deverão ser alimentadas em uma base de conhecimento de um Sistema Especialista (SE). Para investigação da causa ou das causas de falha de vedante foram criadas 29 regras de produção. Um exemplo de regra é apresentado a seguir.

REGRA 01
SE Constatado variação de ar comprimido no sistema de < 1,0 bar
E Resíduo de vedante acumulado no interior da bomba, não
OU Resíduo de vedante acumulado no regulador do fluxo do vedante, não
E Regulagem de pressão do ar comprimido de entrada da bomba < 1,5 ou > 2,5 bar
OU Pressão de saída no tambor < 3,0 bar
E Oscilação apresentado no manômetro hidráulico < 3 PSI
ENTÃO (causa do defeito é regulagem da bomba do vedante) **FC = 90%**

Etapa 5: Agir sobre a realidade

O pesquisador irá agir sobre a realidade com base no seu modelo, de forma que aproxime dos seus objetivos.

5. Considerações Finais

O objetivo do artigo foi propor uma metodologia, portanto o estudo e pesquisa para a sua elaboração foram conduzidos de forma exploratória e empírica. Para a verificação da aplicabilidade e posterior medição da eficácia da metodologia, será desenvolvido um Sistema Especialista (SE) a partir da modelagem apresentada. Como avaliação dos resultados, será necessário utilizar uma métrica cientificamente reconhecida para avaliar a qualidade do conhecimento adquirido. Ao comparar o quadro 1 com o quadro 2, observa-se que a metodologia para aquisição de conhecimento tácito coletivo está alinhada à prática já estabelecida pela comunidade científica. Acredita-se que o uso de ferramentas de gestão de qualidade tornou o processo de aquisição do conhecimento dinâmico, bem como auxiliou o modelador nas etapas da metodologia proposta. Durante a modelagem observou-se alguns aspectos referente à cultura organizacional, conflitos de opiniões, falta de equipamentos para mensuração das características mapeadas. Também se observou que a metodologia promove algumas melhorias, as quais poderão ser utilizadas como vantagem competitiva pela Empresa na qual o estudo foi realizado, como: padronização na investigação de causa de problemas; sistemática para mapeamento de motivos de paradas de produção por priorização; banco de dados de informações de pessoas experientes; necessidade de aquisição de equipamentos de medição; aprendizado individual e coletivo; base de conhecimento que poderá fazer parte de material de treinamento.

Contudo, espera-se como resultado que após o desenvolvimento da metodologia, e ao passo que o sistema for utilizado pelos envolvidos em processos industriais, a forma de diagnosticar causas de defeitos seja padronizada e, como consequência, ocorra a redução de perdas no processo, minimização de erros operacionais de ajustes de processo, redução de custos e aumento de produtividade.

7. Referências

AGUIAR, S. (2002): Integração das Ferramentas da qualidade ao PDCA e ao Programa Seis Sigmas, Editora DG, Minas Gerais.

BERTALANFFY, Ludwig Von (1973): Teoria Geral dos Sistemas, p, 52 – 63; Petrópolis: Vozes.

BITTENCOURT G (2006). Inteligência Artificial – Ferramentas e Teorias, Editora da UFSC, Florianópolis, p. 257 – 286.

CAIRO O (1998): A comprehensive methodology for knowledge acquisition from multiple knowleddge sources. Expert Sistems with application. Volume 14, issue 1-2, pp. 1-16.

HANASHIRO Airton; CLETO, Marcelo Gechele (2007): Proposta de uma metodologia para Gestão do conhecimento no Chão de Fabrica: Um estudo de caso de Kaizen na Indústria Automotiva, ENEGEP.

KINTSCHNER F. E. (2005): Método de Mapeamento e Reorganização de Processos. *Revista Produção Online* /vol. 5/Num.1/Março.

KIM J.; Song J.; Jones D. R. (2011); The cognitive selection framework for knowledge acquisition strategies in virtual communities; International Journal of Information Management. Volume 31, issue 2, pp. 111-120..

KUME H. (1993): Métodos Estatísticos para Melhoria da Qualidade, Editora Gente, Rio de Janeiro.

LE MOIGNE, J. L.; Bártoli J. A. (1996): Organisation Intelligente et Système d´s Ínformation Stratégique, Editora Economica, Paris.

MASTELLA L. S. (2005): Um modelo de conhecimento baseado em eventos para aquisição e representação de eventos temporais, Programa de Pós-Graduação da UFRGS.

MEDEIROS, Débora Isabel (2009): Gestão de Pessoas Diante das Crises Econômicas, URFS, Porto Alegre.

NETO, A. I.; FONTANA R. M. (2008); Sistema Evolutivo de Gestão Integrada para Micro e Pequenas Empesas; XXVII Encontro Nacional de Engenharia de Produção - ENEGEP, Rio de Janeiro, RJ, Brasil, 13 a 16 de outubro.

OLIVEIRA, Teylor Sydney (1995): Ferramentas para aprimoramento, Editora Pioneira, São Paulo.

LEITE, M. S. A. et al (2009): Abordagem sistêmica como ferramenta sustentável para modelar sistemas complexos. *in* Vanderli Fava de Oliveira, Vagner Canevaghi e Francisco Soares Másculo (orgs.), *Tópicos Emergentes e Desafios Metodológicos em Engenharia de Produção: Casos, Experiências e Proposições*; V. 2; Rio de Janeiro: ABEPRO, p. 206 – 259.

RAMOS A. W. (2000): CEP para processos contínuos e em bateladas, Editora Edgard Blucher Ltda, São Paulo.

REZENDE Solange Oliveira (2003): Sistemas Inteligentes: Fundamentos e Aplicações; Editora Manole; Barueri, SP, p, 3 – 21.

WAGNER W. P (2003).: Knowledge Acquisition; Encyclopedia of Information Systems, volume 3.

Um Assistente Inteligente Fuzzy no Acompanhamento da Aprendizagem Significativa

Vandor Roberto Vilardi Rissoli[1], Giovanni Almeida Santos[2]

[1] Universidade Católica de Brasília (UCB)
QS07 Lote 01 EPCT - fone: 61 33569306 - 72002-900 Taguatinga – DF

[2] Universidade de Brasília (UnB) – Faculdade Gama
Área Especial 2 Lote 14 Setor Central - fone: 61 3484 3443 - 72405-610 Gama – DF

Apoio: Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq)

vandor@ucb.br, giovannix@unb.br

Abstract. *This work shows one different architecture of Intelligent Tutoring System that include other human agents in collaboration with one Intelligent Assistant in the teaching process and learning. This process incorporeted the Fuzzy Logic like the resource of followeed educacional methodology based in Subsumption Theory, using the Concept Maps as organized elements from the contents applied domain, reaching its prototype, promising results.*

Resumo. *Este artigo apresenta uma arquitetura diferente de Sistema Tutor Inteligente que inclui outros agentes humanos na colaboração com um Assistente Inteligente no processo de ensino e aprendizagem. Este processo incorpora a Lógica Fuzzy como recurso de acompanhamento da metodologia educacional baseada na Teoria da Aprendizagem Significativa, usando os Mapas Conceituais como elementos organizadores do conteúdo do domínio da aplicação, alcançando, seu protótipo, resultados promissores.*

1. Introdução

A evolução e a capacidade de adaptação dos recursos tecnológicos vem colaborando com as reflexões pedagógicas de âmbito educacional, ajudando a promover novas posturas e condutas na efetivação de um ensino-aprendizagem mais coerente às características atuais da sociedade e as necessidades específicas de cada área de conhecimento, assim como de seus discentes e docentes envolvidos neste processo. Essa relação tem se consolidado de tal forma que alguns pesquisadores a interpretam de maneira muito expressiva, como na afirmação de Lopes (2010) dizendo que os "*termos ensino e tecnologia estão intimamente ligados*".

Essa relação "intensa" vem aumentando nas últimas décadas, onde o auxílio da tecnologia tem ampliado seu apoio, anteriormente focado no aprendiz, para um suporte mais significativo ao processo e aos principais envolvidos com a realização adequada do ensino-aprendizagem.

Diante desta realidade os recursos tecnológicos têm subsidiado a implementação de metodologias capazes de personalizar este processo às principais necessidades dos

diferentes perfis de indivíduos envolvidos, procurando, cada vez mais, efetivar o ensino que almeja a aprendizagem como principal fator de sucesso.

Para isso, torna-se fundamental o acompanhamento de todo o processo e não somente os resultados alcançados nos momentos de avaliação. Geralmente, estas avaliações encerram um ciclo de estudo que poderia ter alcançado resultados melhores se orientações adequadas tivessem acontecido em sintonia com estes momentos da aprendizagem de cada estudante.

A possibilidade deste acompanhamento mais detalhado e personalizado às características de cada aprendiz, contribui com o processo de avaliação, seja este mais Formativo, Somativo ou de Diagnóstico [Gonçalves et al. 2009].

Embora tal acompanhamento seja relevante à formação de cada aprendiz, geralmente, ele acontece por meio da sobrecarga de trabalho do perfil docente (professores), inclusive quando tecnologias são utilizadas sem a preocupação com o objetivo de cada processo e a capacidade de assistência de cada indivíduo envolvido.

Esta situação tem promovido a difusão da tecnologia ITA (*Intelligent Teaching Assistant*) nas diferentes modalidades de ensino (presencial, semipresencial e a distância), almejando maior assistência a todos os envolvidos neste processo de formação, agilidade no acesso às informações estratégicas e relevantes à tomada de decisão, além do fornecimento de orientação pedagógica mais condizente com a situação cognitiva de cada aprendiz, sem sobrecarregar as atividades docentes.

Este trabalho propõe a integração desta tecnologia (ITA) com a Teoria dos Conjuntos Fuzzy, que será aplicada no acompanhamento contínuo da aprendizagem de cada estudante. Por meio desta teoria será possível expressar, matematicamente, valores imprecisos e incertos, que são significativos e bastante utilizados na comunicação natural entre os seres humanos (linguagem natural) [Nicoletti e Camargo 2004].

Dessa forma, se almeja avaliar o conteúdo assimilado por cada estudante durante seu processo educacional, além de fornecer assistência necessária a uma possível mudança de postura do aprendiz, assim como do professor, conforme sejam as necessidades momentâneas ao sucesso de cada perfil na efetivação coerente deste processo. Isso pode acontecer sem aumentar o comprometimento das atividades docentes, apesar do maior nível de personalização às necessidades de cada perfil (discente e docente) participante deste processo.

Este artigo está organizado em 6 seções. Na seção 2 é sintetizado o envolvimento dos Sistemas Tutores Inteligentes e o ITA na Educação. Na seção 3 são abordados os conteúdos fundamentais da Teoria da Aprendizagem Significativa e a proposta de um ITA usando Lógica Fuzzy no acompanhamento do ensino-aprendizagem mediado por esta tecnologia. Na seção 4 são mostrados alguns resultados promissores alcançados num experimento envolvendo um protótipo ITA, sendo na seção 5 apresentadas as considerações finais. As referências bibliográficas são relacionadas na seção 6.

2. Softwares Inteligentes no Apoio da Educação

A realidade das diversas comunidades terrestres (povos) é diferente em vários aspectos (cultura, economia, educação, língua, etc.), no entanto, todas estas comunidades sofrem, direta ou indiretamente, a interferência das tecnologias usadas pela sociedade atual.

Apesar de uma variedade enorme destes recursos tecnológicos, a evolução dos softwares educacionais merece certa atenção, principalmente por sua evolução e demanda contínua de novos recursos que possam satisfazer as necessidades das várias áreas de conhecimento. O uso destes recursos na Educação ainda se destaca pelo volume de usuários e investimentos de alguns países, além da aplicação de tecnologias "inteligentes" provenientes da Inteligência Artificial (IA).

Dentre estes softwares, se destacam, neste trabalho, os Sistemas Tutores Inteligentes (STI). Sua tecnologia é proveniente da década de 70, quando pesquisadores desta nova área de conhecimento (IA) se entusiasmavam com as novas perspectivas almejadas pela IA [Giraffa 2009].

Em sua evolução, os STI se consolidaram como uma arquitetura modular responsável pelo fornecimento de um ambiente interativo que contemplava os conteúdos ou domínios de estudo para os aprendizes, o armazenamento das informações resultantes da interação de cada estudante com o sistema e os aspectos pedagógicos que forneciam a direção mais adequada à realidade da aprendizagem de cada aprendiz. Um módulo de controle propiciava a interação entre estes outros módulos e gerenciava a interação individual do STI com cada um de seus aprendizes.

As características particulares de cada área de conhecimento promoveram algumas modificações ou extensões a esta arquitetura consolidada, também chamada de tradicional, gerando uma quantidade expressiva de variações desta arquitetura.

Essas variações possibilitaram a adaptação desta arquitetura a uma diversidade de necessidades, geralmente, bem específicas e inerentes a algumas áreas de conhecimento. No entanto, algumas dificuldades ainda não foram superadas por este tipo de software, o que tem limitado seu maior uso em cenários acadêmicos e profissionais.

Dentre estas dificuldades se destaca a incapacidade dos atuais recursos de hardware coletarem alguns tipos de dados relevantes ao reconhecimento de características individuais da situação cognitiva de seus aprendizes, podendo isso resultar na modelagem incompleta de alguns aprendizes pelo STI. Consequentemente a orientação pedagógica fornecida por este sistema poderá ser inadequada a real situação detectada.

Essa realidade ressalta ainda mais a grande capacidade de observação sensitiva dos seres humanos, que conseguem captar várias informações relacionadas a esta situação de aprendizagem, podendo usar delas para inferir, de forma mais segura e completa, sobre quando e qual estratégia educacional usar nos diferentes momentos que cada estudante possa estar durante sua jornada de aprendizagem [Viccari e Giraffa 2003].

Diante desta constatação, o envolvimento de mais agentes humanos (monitores e professores), participantes deste processo e como novos usuários do STI, contribuiriam com a realização de uma inferência mais completa por parte deste sistema, que, em contrapartida, poderia subsidiar as atividades destes agentes com o fornecimento de dados resultantes da interação do STI com seus aprendizes.

Estas novas características em um software educacional, que se preocupa com o processo de ensino e assiste a aprendizagem de seus estudantes, auxiliando a atuação de seus diferentes colaboradores humanos (monitores e professores), é identificada por

uma outra tecnologia, atualmente denominada de Assistentes Inteligentes (ITA-*Intelligent Teaching Assistant*).

A arquitetura ITA compreende os módulos do STI, agregando a estes responsabilidades de assistência aos colaboradores humanos envolvidos, diretamente, com o ensino-aprendizagem. Isso amplia suas alternativas de interação e propicia a participação destes agentes humanos nos diagnósticos efetuados por seus módulos, inclusive, permitindo a orientação do docente no seu módulo de controle [Yacef 2002].

Esta tecnologia procura realizar tarefas mais simples, voltadas às atividades essenciais ao acompanhamento personalizado de cada aprendiz e à assistência adequada das possíveis necessidades de seus agentes humanos colaboradores. A arquitetura ITA não trabalha com modelos tão complexos, mas provê uma interface de interação entre estes colaboradores, capazes de coletarem e compartilharem informações mais completas sobre a situação da aprendizagem dos estudantes, e os seus recursos de assistência e orientação baseados nas interações realizadas por estes estudantes e nas posturas educacionais implementadas em seu módulo pedagógico.

Geralmente, os assistentes artificiais (ITA) não adotam estratégias invasivas, se mantendo em uma conduta baseada na ideia de monitoramento de seus aprendizes. Assim, o ITA os assiste e acompanha seus resultados, sem efetuar muitas intervenções, possibilitando maior liberdade de escolha na construção de um caminho mais confortável de aprendizagem para cada estudante. A postura deste assistente, quando solicitada pelo aprendiz, procura incentivar o uso coerente dos recursos disponíveis em seu próprio ambiente, deixando o estudante também refletir sobre suas escolhas e resultados obtidos com o estudo e as atividades que foram realizadas durante sua trajetória de aprendizagem.

Esta junção de recursos, reais (agentes humanos) e artificiais (ITA), procura promover um processo educacional personalizado às características de cada perfil envolvido na eficiência do processo de assimilação dos estudantes engajados no êxito de sua própria aprendizagem. Sendo vários estes recursos, tecnológicos e humanos (docentes, monitores), ressalta-se, ainda mais, a necessidade de uma postura pedagógica harmônica entre todos eles, a fim de direcionar um apoio eficiente, com uma conduta didático-pedagógica adequada à superação de possíveis dificuldades na aprendizagem de cada estudante.

Tal harmonia é alcançada através da adoção de metodologias pedagógicas por todos estes recursos, sendo sua definição o aspecto mais importante à elaboração e uso de um ITA. Por meio destas metodologias será possível acompanhar e avaliar a ação destes recursos de apoio e a possível evolução da aprendizagem de cada estudante.

Diante destes estudos, é proposta a elaboração de um ITA que integra a Teoria da Aprendizagem Significativa (TAS), aplicada no âmbito educacional como uma metodologia de aprendizagem, com uma lógica multivalorada, denominada Lógica Fuzzy, que possibilita o acompanhamento gradual da situação de aprendizagem de cada estudante, sendo esta fundamentada pela Teoria dos Conjuntos Fuzzy.

3. Assistente Inteligente Fuzzy no Apoio à Aprendizagem Significativa

Os aspectos comuns à TAS e à tecnologia ITA atendiam aos objetivos almejados por esta proposta que utilizou o arcabouço conceitual da TAS como direcionador da

modelagem proposta para o desenvolvimento deste novo software educacional, denominado SAE (Sistema de Apoio Educacional).

Dentre as principais características desta metodologia se destaca o reconhecimento da individualidade no processo de assimilação e construção do conhecimento na estrutura cognitiva dos aprendizes. Tal construção ocorre por meio do processo de subsunção, onde os conhecimentos já estabelecidos na estrutura cognitiva de um aprendiz (conceitos subsunçores) incorporam, progressivamente, novas ideias e informações a esta estrutura, promovendo assim sua aprendizagem [Ausubel et al. 1980].

Esta teoria define a mente humana como uma estrutura hierárquica muito bem organizada, estando as ideias mais inclusivas no topo desta hierarquia, que por meio da associação de novos conteúdos promove o aumento de seus conhecimentos estabelecidos.

Além do atendimento personalizado às necessidades de cada aprendiz, esta proposta ainda objetiva satisfazer as possíveis necessidades dos agentes humanos (docentes e monitores) participantes deste processo. Para isso lhes serão fornecidos subsídios significativos sobre o estado cognitivo de cada aprendiz, suas atividades realizadas nos momentos de interação com o ITA e com os monitores estudantis, além das orientações pedagógicas fornecidas pelo professor e por este assistente inteligente.

No entanto, a análise para o acompanhamento deste ensino-aprendizagem não é adequada aos objetivos desta proposta, que deseja acompanhar a realidade da situação cognitiva de cada estudante em qualquer momento deste processo.

Uma variedade de trabalhos tem utilizado a Lógica Fuzzy no acompanhamento do ensino-aprendizagem. Em Fabri (2002), é apresentada uma solução que acompanha o desempenho dos alunos em cursos a distância, existindo somente dois conjuntos fuzzy para esta apuração (Satisfatório e Insatisfatório), que é efetuada somente sobre aspectos quantitativos dos exercícios resolvidos (certos e errados). Lopes (2010) apresenta uma outra solução mais ampla em sua análise para educação a distância, mas não envolve uma metodologia de aprendizagem e se mantém num nível alto de acompanhamento de todo o conteúdo, sem identificar em qual conceito, ou tópico deste conteúdo, o aprendiz está com dificuldade, impossibilitando ao software uma orientação bem direcionada.

Contudo, Malvezzi (2010) conclui em sua pesquisa que o emprego desta Lógica, sobre dados educacionais imprecisos, lhe possibilitou "*materializar algo tão abstrato que é o nível de evolução da aprendizagem do estudante*".

Dessa forma, propõe este trabalho a integração de um acompanhamento fuzzy com a TAS, que será incorporada como metodologia direcionadora ao desenvolvimento e uso deste novo ITA. Esta implementação conjunta (TAS e ITA) envolverá um número maior de variáveis linguísticas na averiguação de cada conceito que constitua um conteúdo a ser assimilado pelos aprendizes, respeitando os aspectos estruturantes e organizacionais propostas pela TAS.

A arquitetura destes ITA está representada na Figura 1 e engloba um novo módulo de interação entre os monitores estudantis e este assistente inteligente. Por meio deste módulo o ITA coleta mais dados resultantes da interação humana, não obtidos por suas tecnologias atuais, aumentando seu "conhecimento" sobre o estado cognitivo de cada estudante que é atendido por esta monitoria [Rissoli e Santos 2010].

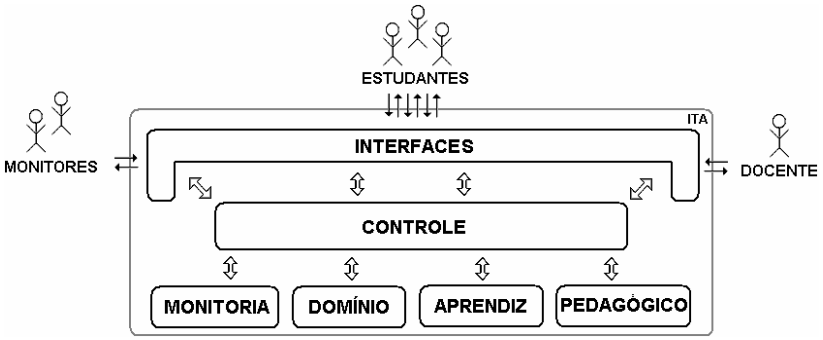


Figura 1. Representação da arquitetura desta nova proposta ITA denominada SAE.

Estes monitores são estudantes mais experientes que colaboram com a aprendizagem de estudantes novatos na exploração dos conteúdos que lhes sejam relevantes ao atual momento de sua assimilação. Tal atendimento acontece fora do horário de aula (extraclasse) e propicia um momento, usualmente, mais tranquilo para interação entre estes estudantes (experiente e novato) e o conteúdo a ser aprendido.

A atividade desta monitoria envolve uma outra metodologia de aporte aos agentes envolvidos neste processo de apoio educacional e ao ITA, sendo ela fundamentada nos aspectos principais da pedagogia proposta por Pestalozzi [Lopes 1981].

Em Viccari e Giraffa (2003), é apresentado uma outra dificuldade ao aumento do uso dos STI, que, geralmente, implementam somente uma postura pedagógica a ser utilizada pelo sistema. Esta proposta ITA se fundamenta nos pressupostos da TAS, podendo atender a várias outras metodologias, principalmente, aquelas voltadas à organização do raciocínio e à solução de problemas. Sua concepção de uso ainda incorpora os pressupostos essenciais à proposta de Pestalozzi sobre o recurso de apoio educacional conhecido, atualmente, como Monitoria Estudantil.

Estas possibilidades pedagógicas enriquecem as expectativas de atuação coerente deste ITA, sendo ele um software baseado em conhecimento com capacidade de ensinar e aprender, continuamente, por meio da interação dos seus diferentes perfis de usuários. Tal expectativa se almeja alcançar com o uso da Lógica Fuzzy no acompanhamento de todo o processo assistido pelo ITA, em colaboração com os demais agentes humanos.

3.1 Modelo de Acompanhamento Fuzzy

A Teoria dos Conjuntos Fuzzy (TCF) surgiu na década de 60, tendo como principal objetivo efetuar um tratamento matemático apropriado sobre certos termos linguísticos subjetivos. Por meio desta teoria torna-se possível a realização de análises e cálculos sobre informações vagas ou imprecisas, tais como "bom" ou "fraco".

Esta análise se fundamenta na flexibilização da pertinência de cada elemento num conjunto, sendo ela indicada por meio de um grau de pertinência que varia no intervalo [0,1]. Nos conjuntos convencionais esta pertinência é definida pela enumeração de seus elementos ou por meio de uma função característica, que identifica ou não, de maneira excludente, a pertinência de tal elemento no conjunto.

Na TCF esta pertinência não é excludente e radical, onde o elemento pertence ou não pertence ao conjunto, podendo o mesmo pertencer, parcialmente, a um dado

conjunto. Assim, para se modelar matematicamente este conjunto, esta teoria propõe a definição dos conjuntos fuzzy, que consistem na generalização da noção de conjuntos, na qual uma função de pertinência identifica o grau corresponde à pertinência de qualquer possível elemento no conjunto [Nicoletti e Camargo 2004].

A obtenção deste grau expressa, numericamente, a pertinência de informações imprecisas em determinados conjuntos, por meio de suas respectivas funções de pertinência. A definição destas funções é essencial ao tratamento fuzzy, pois representa o conhecimento e a experiência de especialistas em determinados domínios ou conteúdos.

A utilização desta teoria possibilita a identificação de pontos importantes a serem acompanhados durante o processo de aprendizagem de cada estudante, independente deles possuírem características imprecisas à sua apuração.

Esta identificação resultará na definição das variáveis linguísticas que possibilitarão o acompanhamento destes pontos pelo ITA, sendo elas formuladas por uma equipe multidisciplinar formada pelos docentes relacionados com a área de conhecimento do conteúdo envolvido nesta aprendizagem, além de especialistas em Educação, Matemática e Computação.

A apuração dos possíveis valores a serem atribuídos a estas variáveis indica o potencial de interferência de cada uma nos resultados alcançados por esta aprendizagem, sendo realizada sobre cada um dos conceitos, ou itens de estudo, pertencentes a um conteúdo a ser assimilado.

Para cada uma destas variáveis, indicadas na Tabela 1, são definidos os possíveis termos linguísticos, que lhes serão atribuídos mediante a aplicação de suas respectivas funções de pertinência. Estas funções calcularão os graus de pertinência de cada termo, lhes confiando um nível de intensidade sobre tal pertinência no conjunto fuzzy que retrata a realidade de cada uma destas variáveis.

Tabela 1. Variáveis e termos linguísticos utilizados por este ITA.

E N T R A D A			S A Í D A
ESFORÇO	**DESEMPENHO**	**PARTICIPAÇÃO**	**RESULTADO**
Baixo função pertinência: $-1/10x + 1$	**Fraco** função pertinência: $1{,}1^{-x}$	**Não Participou**	**Satisfatório** função pertinência 8º grau
Médio função pertinência: $x / 7$ para $0 <= x <= 7$ $-1/3x + 10/3$ para $7 < x <= 10$	**Razoável** função pertinência: $-1/47\ x^2 + 14/51x$	**Participou**	**Insatisfatório** recebe grau máximo quando conceito é insatisfatório
Alto função pertinência: $x / 10$	**Bom** função pertinência: $1{,}09^{(x - 10)}$	**Contribuiu**	-

Nesta proposta sintetizada na Tabela 1 são identificadas três variáveis de entrada (Esforço, Desempenho, Participação) e uma de saída (Resultado), resultante da inferência fuzzy realizada pelo ITA durante o acompanhamento da aprendizagem de cada conceito por cada estudante usuário desta tecnologia.

Na variável linguística denominada **Esforço** é analisado, quantitativamente, o número de exercícios resolvidos e a quantidade de visitas que cada estudante efetuou na monitoria estudantil, envolvendo ainda os metadados relacionados ao conteúdo, mais

especificamente, a sua organização em conceitos que serão assistidos pelo ITA. Seus possíveis termos linguísticos e suas respectivas funções de pertinência, também estão definidas na Tabela 1.

A atuação da monitoria junto aos estudantes é analisada, de maneira direta, por esta variável linguística (**Esforço**). No entanto, a atuação dos monitores interfere, indiretamente, no resultado apurado pela variável **Desempenho**, além de fornecer mais informações ao acompanhamento docente, por meio do acesso ao ambiente virtual deste ITA.

A análise da variável **Desempenho** é qualitativa e envolve o resultado obtido na solução da quantidade de exercícios apurados pela variável **Esforço**. Esta nova variável também respeita os metadados organizacionais de cada conteúdo que o ITA irá assistir, principalmente, a organização hierárquica de cada conceito contido neste conteúdo, respeitando, assim, um dos principais aspectos estruturantes estabelecidos pela TAS.

A definição destes metadados consiste na forma pela qual os docentes, de um mesmo conteúdo (disciplina), direcionarão o acompanhamento e a orientação pedagógica do ITA aos seus aprendizes, onde Mapas Conceituais são elaborados a fim de organizarem os conceitos contidos em um conteúdo, respeitando os principais pressupostos da TAS [Novak 1998].

A análise que apura o valor da variável **Desempenho** acontece sobre a quantidade de exercícios certos e errados resolvidos, distribuindo, para cada um deles, um peso diferente que respeita as características relacionadas ao tipo de questão, nível de dificuldade e categoria do exercício. Estes pesos fazem parte do que poderia ser chamado de metadados dos exercícios elaborados pelos professores e utilizados pelo ITA em interação com os aprendizes. A indicação destas definições está na Tabela 2.

Tabela 2. Definições para análise qualitativa da variável linguística Desempenho.

TIPO DE QUESTÃO (30%)	NÍVEL DE DIFICULDADE (40%)	CATEGORIA DA QUESTÃO (30%)
Verdadeiro ou Falsa (peso 1,3)	Fácil (peso 2)	Revisão (peso 2,5)
Múltipla Escolha (peso 1,8)	Médio (peso 3)	Fixação (peso 3,5)
Escolha Múltipla (peso 2,3)	Difícil (peso 5)	Avaliativa (peso 4,0)
Lacuna (peso 1,8)	-	-
Aberta/Dissertativa (peso 2,8)	-	-

Para a análise da terceira variável linguística, denominada **Participação**, este ITA averiguará a participação de cada aprendiz nas atividades interativas propostas pelo docente. Estas atividades poderão acontecer por meio de fóruns ou chats (bate papo) envolvendo assuntos relacionados aos conceitos pertinentes a cada conteúdo acompanhado pelo ITA.

No entanto, esta análise não acontece por conceitos existentes em um conteúdo, diferentemente das duas variáveis anteriores, mas sobre o nível de interação efetivada por cada estudante nestas oportunidades interativas onde há troca de experiências, esclarecimentos de dúvidas e colaboração entre os aprendizes e professores na superação de possíveis dificuldades de assimilação. A Tabela 3 sintetiza os principais aspectos relacionados a esta variável e seus possíveis termos linguísticos.

Tabela 3. As principais definições da variável linguística Participação.

TERMO	FUNÇÃO DE PERTINÊNCIA
Não Participou	$n = \begin{cases} 0 & \text{se } x \geq 1{,}013725668; \\ 0{,}5 + \dfrac{x - 0{,}0001}{-0{,}24975 + \dfrac{x - 0{,}1}{-0{,}5216481528 + 0{,}007494661219x}} & \text{se } x < 1{,}013725668. \end{cases}$
Participou	$p = \begin{cases} 1,855726754x + 14,42732455x^3 & \text{se } x < 0,1; \\ -0,0288546492 + 2,288546492x + 4,32819736642954478(x-0,1)^2 - 12,62390899(x-0.1)^3 & \text{se } x < 0,5; \\ 1,154185964 - 0,3083719287x - 10,8204934160738606(x-0,5)^2 + 39,04212707(x-0,5)^3 & \text{se } x < 0,6; \\ 1,680724081 - 1,301206801x + 0,892144695020432810(x-0,6)^2 - 0,347819232(x-0,6)^3 & \text{se } x < 1; \\ 1,254444276 - 0,7544442763x + 0,47476161646738313(x-1)^2 - 0,1203173402(x-1)^3 & \text{se } x < 2; \\ 0,4317461278 - 0,1658730639x + 0,11380959588315423(x-2)^2 - 0,03793653197(x-2)^3 & \text{se } x \leq 3,187286476; \\ 0 & \text{se } x > 3,187286476. \end{cases}$
Contribui	$c = \begin{cases} 0 & \text{se } x < 0,08386381483; \\ -0,05197573162 + 0,6197573162x - 0,1234832263(x-0,1)^3 & \text{se } x < 0,5; \\ -0,0302426838 + 0,5604853676x - 0,14817987152034259 3(x-0,5)^2 - 0,02558172733(x-0,5)^3 & \text{se } x < 1; \\ 0,0968807994 + 0,3931192006x - 0,18655246252676660 5(x-1)^2 + 0,05343326193(x-1)^3 & \text{se } x < 2; \\ 0,3893718772 + 0,1803140614x - 0,026252676659528909 7(x-2)^2 - 0,00406138477(x-2)^3 & \text{se } x < 3; \\ 0,5531263383 + 0,1156245539x - 0,038436830835117771 8(x-3)^2 + 0,01281227695(x-3)^3 & \text{se } x \leq 4,129196326; \\ 1 & \text{se } x > 4,129196326. \end{cases}$

Diante da definição destas três variáveis linguísticas é averiguado o valor mais significativo entre cada um de seus possíveis termos, sobre cada um dos conceitos relevantes ao estudo e à aprendizagem do conteúdo organizado em um Mapa Conceitual pela equipe multidisciplinar.

Para isso é aplicada a operação de *maximo*, caracterizada na TCF como uma *t-conorma*, sobre os graus de pertinência obtidos por cada um destes termos, sendo atribuído a cada uma destas variáveis o termo com maior grau entre todos os possíveis [Nicoletti e Camargo 2004]. A Figura 2 exemplifica esta operação para as variáveis linguísticas de entrada (Esforço, Desempenho, Participação).

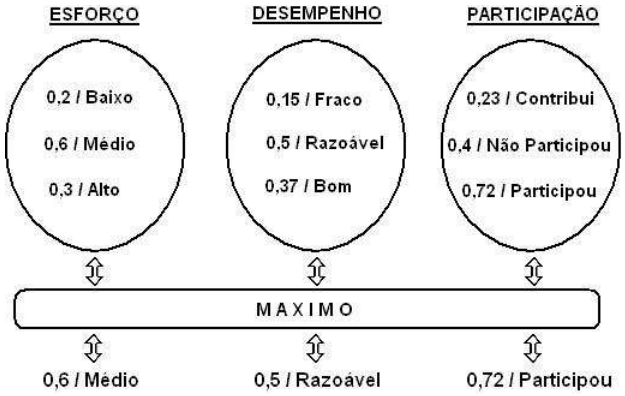


Figura 2. Exemplo da apuração do termo mais significativo às variáveis de entrada.

Após estas atribuições serão acionadas as regras de inferência que formam a base de conhecimento deste ITA. Essas regras são baseadas em conhecimento e incorporam ao assistente artificial sua capacidade de representação do raciocínio humano no fornecimento de suas orientações pedagógicas "inteligentes".

O conjunto destas regras trabalha as combinações possíveis das variáveis de entrada que definirão o termo linguístico da variável de saída, identificando se a assimilação significativa está satisfatória a aprendizagem almejada em cada conceito mapeado ao acompanhamento deste ITA.

Como pode ser observado na Figura 3, essas regras são proposições condicionais não qualificadas (***se...então***) que conferem "inteligência" a este assistente para o fornecimento de orientação aos seus estudantes sobre quais atividades interativas realizar e sobre qual conteúdo dedicar mais atenção no seu atual momento de aprendizagem.

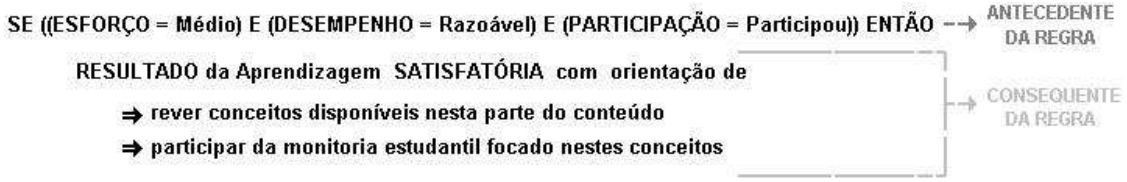


Figura 3. Exemplo de uma proposição condicional não qualificada (regra de inferência).

Estas proposições avaliam cada uma das variáveis de entrada em sua parte antecedente, indicando na consequente qual a situação da aprendizagem inferida e as orientações pedagógicas que seriam mais coerentes à aprendizagem significativa.

O número destas regras pode ser calculado pelo produto escalar das quantidades das funções de pertinências de cada conjunto fuzzy envolvido nesta apuração, sendo também de responsabilidade da equipe multidisciplinar a definição de cada uma delas.

A variável de saída deste ITA é denominada **Resultado** na Tabela 1 e possui dois termos linguísticos, Satisfatório e Insatisfatório, que são definidos pelo processo de inferência (*forward chaining*) que identifica qual é a regra mais conveniente à situação detectada pelas variáveis de entrada. A Tabela 4 mostra estes termos e a função de pertinência existente somente para a situação da aprendizagem Satisfatória, onde o ITA calcula seu grau de pertinência e o fornece como um valor de confiança sobre esta aprendizagem ter sido significativa. Na situação Insatisfatória o grau não é apurado, pois a orientação do ITA será baseada na averiguação das variáveis de entrada para o fornecimento da orientação pedagógica que seja mais indicada à superação das deficiências de assimilação inferidas.

Tabela 4. As definições da variável linguística de saída Resultado.

TERMO	FUNÇÃO DE PERTINÊNCIA
Satisfatório	*satis* = 0,002209486898 + 0,2393247106x – 0,2939870561x^2 + + 0,1447862596x^3 – 0,02634090467x^4 + 0,0005043943843 x^5 + + 0,0004038777723x^6 – 0,00004623993339x^7 + 0,000001547099395x^8 onde *satis* corresponde ao grau de pertinência desta aprendizagem com $x \in [0,10]$
Insatisfatório	**1** para qualquer indicação da regra de inferência como Insatisfatória

De acordo com a experiência multidisciplinar contida na equipe envolvida, foi definida a função de pertinência que representa a efetivação satisfatória da aprendizagem significativa. Esta função de oitavo grau foi obtida através do Método dos Mínimos Quadrados, pois foi necessário realizar cálculos de aproximação dos valores definidos pela equipe multidisciplinar, a fim de representar sua experiência e conhecimento da melhor forma possível no "raciocínio" deste ITA [Burden e Faires 2003].

Dessa forma, o ITA infere o estado cognitivo de cada aprendiz, em relação a cada conceito existente num conteúdo de estudo, além de fornecer um esclarecimento detalhado sobre a situação da aprendizagem destes estudantes, por meio da constatação adequada de cada um de seus termos linguísticos, não sendo necessário, neste processo

de inferência fuzzy, a desfuzzificação, pois estes termos já se encontram em linguagem natural para a comunicação humana e são apresentados, de maneira significativa e esclarecedora, aos seus diferentes perfis de usuários (alunos, monitores, professores).

4. Resultados Preliminares de um Experimento na área de Programação

A elaboração de um protótipo deste ITA (SAE) aconteceu num momento que possibilitou a sua utilização em um ambiente real de ensino de graduação na área de Informática (curso de Ciência da Computação).

Assim, um experimento foi realizado no segundo semestre de 2010 e envolveu disciplinas (conteúdos) da área de Programação Computacional (Algoritmo, Laboratório-1 – Linguagem C e Laboratório-2 – Linguagem Java), área esta reconhecida como uma das mais difíceis à aprendizagem nesta formação [Almeida et al. 2002].

Embora os resultados aprofundados ainda estejam sendo apurados, algumas indicações positivas conferem expectativas promissoras para esta proposta que envolveu 177 estudantes, 31 monitores e 3 professores. Entre elas se destacam o índice de satisfação de 82% dos 97 aprendizes (mais de 54% dos estudantes) que participaram de uma pesquisa no final deste período letivo (2/2010) e a aprovação de mais de 93% da metodologia usada combinando as ações dos agentes reais e o assistente artificial. Outro fator relevante consiste nos quase 95% dos estudantes que relataram que recomendariam a outros colegas cursarem as disciplinas com esta metodologia.

A postura docente não se modificou muito na condução das disciplinas, mas identificou uma satisfação em torno de 80% ao uso desta proposta pelos docentes envolvidos neste experimento, abrangendo seis turmas, sendo duas de cada disciplina. No entanto, foi possível identificar a necessidade de mais conteúdo de auxílio (*Help*) ao uso dos vários recursos disponíveis neste assistente, colaborando com o envolvimento mais seguro destes docentes na condução de suas disciplinas assistidas pelo ITA.

O índice de aprovação destes estudantes, participantes desta pesquisa, foi de 71%, porém não envolveu os aprendizes que evadiram da disciplina antes de seu encerramento. Além destes números, o próprio ITA (Figura 4) consiste em um resultado deste trabalho, onde o apoio do CNPq e o Ministério do Planejamento estarão o disponibilizando com um software público brasileiro nos próximos meses.



Figura 4. Janela de orientação pedagógica do ITA, denominado SAE, vista pelo docente.

5. Considerações Finais e Trabalhos Futuros

A elaboração deste tipo de tecnologia proporciona uma organização nos atuais ambientes educacionais, tornando-os mais eficientes no apoio ao ensino-aprendizagem.

Aspectos relevantes à realidade social também são mais bem atendidos, como a facilidade da comunicação, a autonomia de cada perfil envolvido e o respeito às características individuais de cada estudante durante sua trajetória de aprendizagem.

Apesar do acompanhamento do aprendiz ser personalizado, não houve aumento significativo na carga de trabalho docente, passando este ainda a receber maior assistência aos processos formativos, somativos e até de diagnóstico, em tempo real, sobre a situação de aprendizagem de cada um de seus estudantes.

Dessa forma, a integração entre os recursos reais e artificiais promove um espaço de aprendizagem dinâmico, com assistência baseada no acompanhamento mais realista de cada participante deste processo. Várias são as possibilidades de evolução neste trabalho, desde a criação de novas variáveis linguísticas para acompanhamento mais eficiente até a integração deste ITA com outros ambientes virtuais de aprendizagem.

6. Referências

Almeida, E. S. de, Costa, E. B., Silva, K. S., Paes, R. B., Almeida, A. A. M., Braga, J. D. H. (2002) AMBAP: Um Ambiente de Apoio ao Aprendizado de Programação. In: Workshop sobre Educação em Computação. Florianópolis: SBC, p. 691-700.

Ausubel, D. P., Novak, J. D., Hanesian, H. (1980) Psicología Educacional. Rio de Janeiro: Editora Interamericana.

Burden. B., Faires, J. D. (2003) Análise Numérica. Thompson Pioneira.

Fabri, J. (2002) Ferramenta fuzzy para acompanhamento do desempenho do estudante nos cursos à distancia. IMENSA – Universidade de São Paulo.

Giraffa, L. M. M. (2009) Uma odisséia no ciberespaço: O software educacional dos tutorias aos mundos virtuais. Revista Brasileira de Informática na Educação. v. 17, n. 1, p.20-30.

Gonçalves, J. P., Monzón, A. J. B., Aluísio, S. M. (2009) Métodos de avaliação informatizada que tratam o conhecimento parcial do aluno e geram provas individualizadas. In: Simpósio Brasileiro de Informática na Educação. Florianópolis: SBC.

Lopes, R. S., Fernandes, M. A. (2010) Lógica Fuzzy na avaliação de conhecimento e comportamento em ambientes de EAD. In: Workshop de Licenciatura em Computação, SBIE. João Pessoa: SBC.

Lopes, L. (1981) Pestalozzi e a educação contemporânea. Rio de Janeiro:Associação Fluminense.

Malvezzi, W. R., Mourão, A. B., Bressan, G. (2010) Uma ferramenta baseada em Teoria Fuzzy para o acompanhamento de alunos aplicado ao modelo de educação presencial mediado por tecnologia. In: Simpósio Brasileiro de Informática na Educação. João Pessoa: SBC.

Nicoletti, M. C., Camargo, H. A. (2004) Fundamentos da Teoria de Conjuntos Fuzzy. São Carlos: EdUFSCar.

Novak, J. (1998) Learning, Creating, and Using Knowledge: Concept Maps as Facilitative Tools in Schools and Corporations. Lawrence Erlbaum Associates, New Jersey.

Rissoli, V. R. V., Santos, G. A. (2010) A Monitoria Estudantil aliada a um Assistente Virtual Inteligente no Suporte à Aprendizagem via Internet. In: Workshop de Arquiteturas Pedagógicas para Suporte à Educação a Distância mediada pela Internet, SBIE. João Pessoa: SBC.

Viccari, R. M., Giraffa, L. M. M. (2003) Fundamentos dos Sistemas Tutores Inteligentes. In: Barone, D. (Org.). Sociedades Artificiais: A Nova Fronteira da Inteligência nas Máquinas. Porto Alegre: Bookman, p. 155-208.

Yacef, K. (2002) Intelligent Teaching Assistant Systems. In: International Conference on Computers in Education. New Zeland. Proceedings. New Zeland: IEEE. v. 1, p. 136-140.

Aplicação do algoritmo Infestação de Ervas Daninhas (IWO) ao problema multi-objetivo de alocação de objetos circulares

Michelle de Oliveira Parreira[1], Sherfis Gibran Ruwer[2], Sandra Sandri[1], José Carlos Becceneri[1], Luiz Leduíno Salles Neto[3]

[1]Laboratório Associado de Computação e Matemática Aplicada (LAC)

[2]Divisão de Mecânica Espacial e Controle (DMC)

Instituto Nacional de Pesquisas Espaciais (INPE)
Caixa Postal – 515 – 12.227-010 – São José dos Campos – SP – Brazil

[3]Universidade Federal de São Paulo (UNIFESP)
Rua Talim, 330, 12231-280, São José dos Campos, SP, Brazil

parreira.michelle@gmail.com, sherfis@yahoo.com.br,
{sandri,becce}@lac.inpe.br, luiz.leduino@unifesp.br

Abstract. *Bioinspired optimization algorithms, such as genetic algorithms and particle swarms, have been used to solve the problem of allocation of cylinders in a circular container. When the height of the cylinders is not considered, the problem is reduced to the allocation of circles, each one with a mass value. This article presents an implementation of the algorithm of infestation by weeds to solve this problem. It was used a multi-objective function that seeks to reduce the distance from the center of mass of the set to the axis of rotation of the circular container, in addition to minimize the total space allocated to the circles inside the container. Finally, it is proposed a new method that removes all the imbalance of the set.*

Resumo. *Algoritmos de otimização bioinspirados, como algoritmo genético e enxame de partículas, têm sido utilizados na resolução do problema de alocação de cilindros em um recipiente circular. Quando não se considera a altura dos cilindros, o problema se reduz à alocação de círculos, cada um com um valor de massa. Este artigo apresenta a implementação do algoritmo de infestação por ervas daninhas para resolver este problema. Utiliza-se uma função multi-objetivo que busca reduzir a distância do centro de massa dos objetos e o eixo de rotação do recipiente circular, além de minimizar o espaço total alocado aos círculos dentro do recipiente. Por fim é proposto um novo método que permite anular totalmente o desbalanceamento do conjunto.*

1. Introdução

A alocação de um conjunto de objetos em um recipiente faz parte da área de Pesquisa Operacional e tem várias aplicações significativas, dentre as quais a alocação de carga útil em satélites artificiais. Uma das principais considerações a serem tomadas no projeto de alocação de carga em satélites é a minimização do espaço total ocupado pela carga. Além disso, a disposição da carga é importante, pois ela influi no centro de massa resultante do satélite, o qual sofre efeitos do desbalanceamento quando está girando em torno de seu eixo de rotação.

Quando os objetos que compõem a carga estão todos apoiados sobre uma única superfície e suas alturas puderem ser desconsideradas, o problema pode ser modelado de forma bidimensional. Essas simplificações são válidas para as análises de minimização do espaço e desbalanceamento de massa sobre o eixo de rotação perpendicular à superfície de apoio. O modelo fica então reduzido à alocação de círculos, representando os objetos cilíndricos da carga útil do satélite, cada qual associado a um valor positivo equivalente a massa do objeto, e distribuídos dentro de um círculo maior com raio igual ao do satélite. Vários trabalhos foram propostos na literatura para resolver esse problema (AKEB e LI (2005), HUANG *et al.* (2006), XIAO *et. al.* (2006), LEI (2009)). Muitos desses trabalhos utilizam algoritmos de otimização inteligentes, dentre os quais se destacam os bio-inspirados (algoritmos baseados em simulações de ações da natureza) como Algoritmos Genéticos (BECCENERI e SILVA NETO 2009). e Colônia de Partículas (BECCENERI e SILVA NETO 2009).

Um dos mais recentes algoritmos de otimização bio-inspirados é aquele denominado Infestação de Ervas Daninhas (MEHRABIAN e LUCAS, 2006). Este algoritmo baseia-se na capacidade da infestação dessas ervas, conhecidas como plantas invasoras devido à sua alta capacidade de reprodução. Estudos demonstraram que, mesmo depois de 50 anos de existência de herbicidas, essas ervas continuam se multiplicando em um mesmo campo de análise, gerando espécies mais resistentes aos herbicidas. Isso se deve ao fato destas ervas possuírem capacidade de crescimento rápido, eficiência no uso da água, alta adaptação climática, curto intervalo entre floração e germinação, estruturas para dispersão que germinam em quase todos os substratos úmidos sem uma fertilização específica, alta dormência, alta longevidade e alta produção contínua (MEHRABIAN e LUCAS, 2006).

Além desse trabalho na área, podem ser encontradas aplicações por Tellez (2010) para o problema de Bin-Packing, que faz parte da área de Pesquisa Operacional. Esse autor sugere ainda os trabalhos de Hajimirsadeghi e Lucas (2009) para um Sistema de Vendas, e Rad e Lucas (2008) no desenvolvimento de um algoritmo híbrido IWO/PSO para otimização de parâmetros em Sistemas de Controle. Neste trabalho foi aplicada uma meta-heurística baseada no algoritmo bio-inspirado Infestação de Ervas Daninhas (MEHRABIAN e LUCAS, 2006) para otimização da alocação de objetos cilíndricos dentro de satélites.

Após estudo dos resultados, foi desenvolvido um novo método para remover o desbalanceamento das massas, isto é, ajustar o centro de massa do conjunto coincidente com o centro do recipiente. Este método é aplicado à solução do algoritmo IWO em uma fase de pós-processamento.

2. Definição do Problema – Alocação de Objetos Circulares

O problema a ser tratado neste trabalho refere-se à alocação de um conjunto de cilindros dentro de um recipiente também circular (vide AKEB e LI (2005), HUANG *et al.* (2006), XIAO *et. al.* (2006) e LEI (2009)). A altura dos cilindros é considerada irrelevante, e o problema reduz-se à alocação de círculos, cada qual associado a uma massa positiva, dentro de outro círculo.

Neste trabalho, a superfície do recipiente é denotada por *RC* e seu raio por *R*. Considera-se a alocação de *k* círculos em *RC*, cada qual com raio r_i e massa m_i. Um par

ordenado (x_i, y_i) é relacionado ao centro de cada círculo. Os valores (x_i, y_i) de todos os círculos são calculados pelo algoritmo. A Figura 1 ilustra um exemplo com quatro círculos.

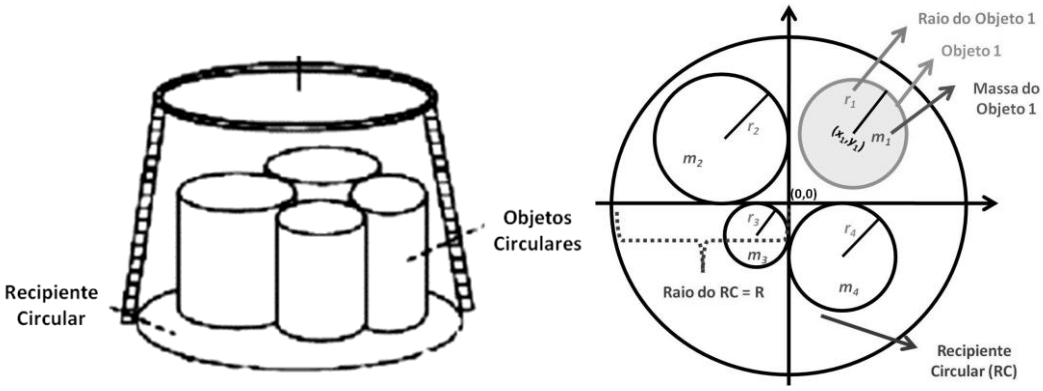


Figura 1. Objetos circulares alocados dentro de um Recipiente Circular. (Adaptação de XIAO *et. al.*, 2006).

Cada solução do problema deve obedecer algumas restrições:

- O raio de cada círculo c_i deve ser menor que o raio de *RC*:

$$\forall i \in \{1, ..., \kappa\},\ r_i < R \tag{1}$$

- Dado um círculo c_i, a soma de seu raio com a distância do centro de *RC* até o centro de c_i está limitada pelo raio de *RC*:

$$\forall i \in \{1, ..., \kappa\},\ r_i + \sqrt{x_i^2 + y_i^2} < R \tag{2}$$

- Não pode haver sobreposição de dois círculos distintos c_i e c_j ($i \neq j$). Desta forma a distância entre os centros de dois objetos c_i e c_j, com as respectivas coordenadas (x_i,y_i) e (x_j,y_j), deve ser maior ou igual à soma dos raios r_i e r_j desses objetos:

$$\forall i \in \{1, ..., \kappa\},\ \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \geq r_i + r_j \tag{3}$$

O intuito é alocar todos os objetos circulares c_i dentro da circunferência *RC* com a melhor ocupação do espaço, buscando alcançar o **menor raio para *RC***. Para isso é necessário minimizar a função f_1 apresentada em (4) abaixo, em que $L = (x_1, y_1, x_2, y_2, ..., x_k, y_k)$ é o vetor com todas as coordenadas dos círculos.

$$f_1(L) = \max_{1 \leq i \leq k} (r_i + \sqrt{x_i^2 + y_i^2}) \tag{4}$$

Além da função f_1, deve ser verificada a instabilidade de massa de *RC* após a inserção de todos os objetos c_i. Para analisar essa instabilidade é calculado o equilíbrio das massas dos objetos c_i. Segundo XIAO *et. al.* (2006) o desequilíbrio dinâmico é causado pelo deslocamento do centro de massa em relação ao centro de rotação do recipiente, que ao girar com velocidade angular ω sofre o efeito do desbalanceamento.

Porém, para o problema de alocação dos objetos no recipiente e análise da distribuição de suas massas, o cálculo do desbalanceamento independe da velocidade angular, podendo então considerar para desbalanceamento a Equação 5. Assim o desequilíbrio de massa é calculado levando-se em conta a massa de cada objeto e a distância do seu centro até o centro de *RC*. Para isso é necessário minimizar a função f_2 apresentada em (5) buscando a solução com a menor instabilidade possível.

$$f_2(L) = \sqrt{(\sum_{i=1}^{n} m_i x_i)^2 + (\sum_{i=1}^{n} m_i y_i)^2} \tag{5}$$

Por analisarmos duas funções diferentes e dependentes, denominamos este problema de função multi-objetivo. Para encontrar a melhor solução é importante que se minimize uma função global *f*, a qual dependerá de f_1 e f_2. Essa dependência pode se tornar problemática já que, algumas vezes, melhorando a solução de uma das funções objetivo pode levar a outra função para uma situação menos satisfatória. Aqui optamos por uma função de agregação linear, que utiliza as constantes λ_1 e λ_2 como fatores, para compensar a relação de magnitude dos valores finais esperados para o raio e para o desbalanceamento de massa.

Temos assim um problema com duas funções objetivos conflitantes: o raio do RC (f_1) e o desequilíbrio de massa (f_2). Trata-se, assim, de um problema de otimização multi-objetivo, ou vetorial. Neste trabalho optamos por agregar os dois objetivos em uma única função, atribuindo pesos constantes λ_1 e λ_2 para as funções f_1 e f_2, respectivamente.

$$f(L) = \lambda_1 f_1(L) + \lambda_2 f_2(L) \tag{6}$$

Neste trabalho, propõe-se realizar a eliminação da instabilidade de qualquer solução através de um ajuste, utilizando-se a Equação 7, onde dR é o acréscimo de raio devido ao ajuste feito no centro de massa.

$$dR = \frac{f_2(L)}{M} = \frac{\sqrt{(\sum_{i=1}^{n} m_i x_i)^2 + (\sum_{i=1}^{n} m_i y_i)^2}}{\sum_{i=1}^{n} m_i} \tag{7}$$

O desbalanceamento calculado na Equação 5 é conhecido como primeiro momento de inércia de um sistema de massas. Dividindo-se essa grandeza pela massa total do sistema, conforme Equação 7, encontra-se a posição do centro de massa do conjunto. O deslocamento desse centro de massa em relação ao eixo de rotação do conjunto, juntamente com o montante de massa, é o responsável, por exemplo, pela instabilidade dinâmica durante a rotação de um satélite. Aplicando esse cálculo no resultado encontrado pelo processo de otimização, estima-se que os valores de deslocamento do centro de massa sejam muito menores que o raio total do recipiente. Assim, esse valor é somado vetorialmente para deslocar a posição de todos os círculos internos, o que provoca a eliminação do desbalanceamento, já que o centro de massa do conjunto é posicionado sobre o centro do recipiente, ao custo de um aumento no raio total, que se espera ser aceitável.

3. Otimização por Infestação de Ervas Daninhas (IWO)

Para se entender o algoritmo de ervas daninhas IWO (MEHRABIAN e LUCAS, 2006), são definidas algumas inter-relações entre o problema a ser otimizado e o processo natural de infestação de ervas daninhas. Por *erva daninha* entende-se uma solução do problema. A posição da erva daninha na sua colônia é definida pelo vetor de variáveis do problema, que no caso da alocação de círculos estudada neste trabalho é representada pelo vetor *L*. A aptidão de uma erva daninha descrita na sequencia do trabalho é equivalente ao valor da função dado que a erva daninha está em uma posição na colônia que representa um conjunto de valores para as variáveis do problema.

A metodologia de infestação aplicada pode ser visualizada na Figura 2. Primeiramente é configurado um conjunto de parâmetros que regem o processo de evolução do algoritmo. O processo é iniciado com a criação da primeira geração de ervas daninhas. A partir daí é realizado um ciclo de iterações para representar a evolução da colônia. Cada iteração realiza a reprodução das ervas daninhas gerando as sementes novas, que em seguida são dispersadas e se tornam parte da população. A exclusão de ervas daninhas elimina as menos aptas dentre toda população mantendo a quantidade de individuos limitada conforme os parâmetros iniciais. O ciclo é repetido até alcançar o número de iterações definido nos parâmetros.

Ao final, com a quantidade máxima de iterações alcançada, espera-se encontrar a solução ótima global através da erva com melhor aptidão. A seguir são detalhadas cada uma das etapas do algoritmo.

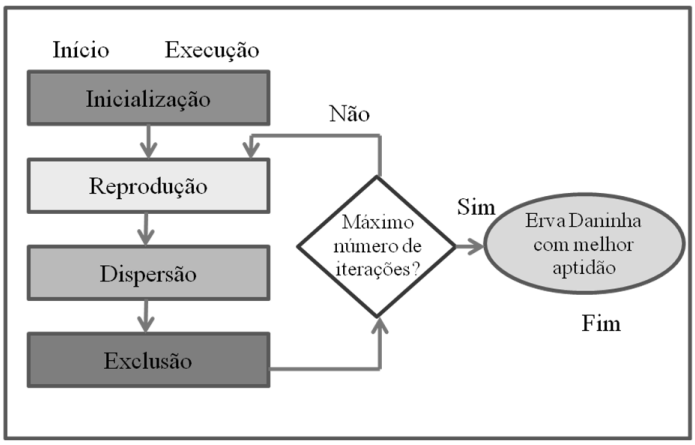


Figura 2. Metodologia desenvolvida.

3.1 Inicialização de Parâmetros

A escolha dos parâmetros de inicialização é fundamental para que se encontre uma convergência para a solução ótima global. Esses parâmetros são:

- Pinitial = (int) População inicial
- Pmax = (int) População maxima
- itermax = (int) Número máximo de iterações
- Dproblem = (int) Dimensão do problema
- Smax = (int) Número máximo de sementes que uma erva daninha pode gerar em uma reprodução
- Smin = (int) Número mínimo de sementes que uma erva daninha pode gerar em uma reprodução
- n = (int) Módulo de não linearidade para cálculo da variância

- σfinal = (double) Variância final
- σinicial = (double) Variância inicial
- A[][] = (double [][]) Área para dispersão da população inicial, onde:
- double[dimensão][0=limite mínimo; 1=limite máximo].

Os parâmetros auxiliares são informações geradas durante a execução do algoritmo e são apresentados a seguir.

- Fmin = (double) Maior aptidão encontrada na população
- Fmax = (double) Menor aptidão encontrada na população
- Fweed = (double) Aptidão de uma erva daninha
- iter = (int) Iteração corrente
- Si = (int) Número de sementes para reprodução de uma erva daninha i
- σiter = (double) Variância aplicada em uma dada iteração

3.2 Reprodução

A reprodução ocorre no momento que cada erva daninha deve replicar e gerar novas sementes. Calcula-se, através da equação da reta inscrita no retângulo da Figura 3, o número de sementes a ser gerado em cada reprodução por cada erva daninha dada sua aptidão em relação às aptidões máxima e mínima na população naquela iteração (Equação 8) (MEHRABIAN, LUCAS, 2006). Destaca-se que nesse momento as sementes ainda não possuem posição na colônia.

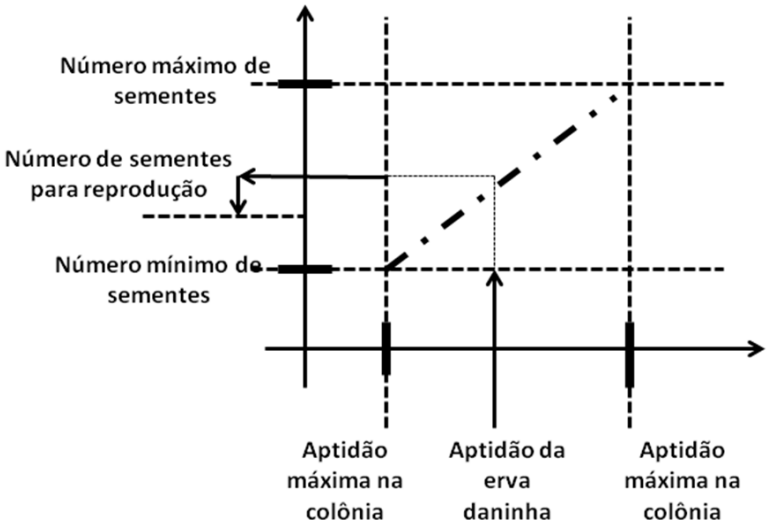


Figura 3. Procedimento de produção de sementes em uma colônia. Fonte: Mehrabian e Lucas (2006).

$$S_i = \left(\frac{F_{weed} - F_{\min}}{F_{\max} - F_{\min}} \right) . \left(S_{\max} - S_{\min} \right) + S_{\min} \tag{8}$$

3.3 Dispersão

A dispersão é baseada no cálculo do desvio padrão σ_{iter} de uma determinada iteração que dependerá de $iter_{max}$, índice de modulação não linear n, desvio padrão inicial $\sigma_{inicial}$ e desvio padrão final σ_{final} (Equação 9) (MEHRABIAN, LUCAS, 2006). Após calcular o desvio padrão de uma iteração i, é necessário gerar um vetor de valores randômicos de

uma distribuição normal centrada na posição $x_{i,0}$ da erva daninha pai. Cada um desses é o valor de uma variável do vetor de solução conforme Equação 10, onde *rnd* é um valor aleatório entre 0 e 1 que é aplicado à equação para gerar o valor de x_i de forma a seguir uma distribuição normal.

$$\sigma_{iter} = \frac{(iter_{\max} - iter)^n}{(iter_{\max})^n}(\sigma_{initial} - \sigma_{final}) + \sigma_{final} \tag{9}$$

$$x_i = x_{i,o} \pm \sigma_{iter}\sqrt{-2\ln(rnd)} \tag{10}$$

3.4 Exclusão Competitiva

O parâmetro de população máximo definido inicialmente é determinante para a continuidade de novas gerações. Após as novas ervas daninhas estarem distribuídas na colônia, elas são incorporadas à população total, o que pode tornar o número de indivíduos da população superior ao parâmetro P_{max}. Se ocorrer essa superpopulação, a exclusão das ervas com menor aptidão é realizada até que se mantenha o limite de população P_{max}. Com isto, apenas os sobreviventes poderão se reproduzir nas próximas iterações.

4. Resultados dos Experimentos

Nesta seção são apresentados os resultados dos experimentos numéricos com a implementação do algoritmo das ervas daninhas em relação a alguns trabalhos na literatura que aplicaram este algoritmo para o mesmo problema de alocação de círculos. Após algumas execuções, os melhores valores conseguidos para os parâmetros da Equação (6) foram $\lambda_1 = 100$ e $\lambda_2 = 0{,}5$.

A Tabela 2 apresenta o resultado gerado pela alocação de 10 objetos (Tabela 1). Para execução deste experimento utilizou-se dos seguintes parâmetros iniciais: $P_{initial}$=40, P_{max}=100, $iter_{Max}$=200, $D_{problem}$=20, S_{max}=40, S_{min}=3, n=3, σ_{final}=0,0001 *e* $\sigma_{inicial}$ =20. Pode-se perceber que apesar do IWO não ter encontrado uma solução melhor que sugerido por XIAO *et al.* (2006), 'Compaction+PSO' (59,93) - apesar de muito próxima (60,21) - obteve melhor resultado para a média das soluções geradas a cada 10 execuções. O valor entre parênteses (*dR*) nas tabelas a seguir é o ajuste do centro de massa realizado no pós-processamento, quando a estabilidade é diferente de 0. Este valor é acrescido ao raio mínimo da solução instável, que resulta no raio final ajustado do recipiente circular e eliminação total da instabilidade. Assim, pode-se comparar este raio final ajustado, que é parte da solução sem nenhum desbalanceamento de massas.

Tabela 1. Informações sobre raio e massa de 10 objetos. Fonte: (XIAO et al., 2006).

	Raio	Massa		Raio	Massa
1	20	35	**6**	21	80
2	22	61	**7**	11	93
3	17	49	**8**	5	82
4	17	89	**9**	23	70
5	7	68	**10**	8	20

Tabela 2. Resultados comparativos entre algumas implementações da Literatura e IWO para 10 objetos. Fonte: (XIAO et al., 2006).

LITERATURA	ALGORITMO	< Raio (+dR)	Média: Raio (+dR)	Média: Inst.
Zhou *et al.*(2005)	PSO	61,32	64,08 **(+3,1x 10^{-7})**	0,0002
XIAO *et al.*(2006)	Compaction + SA	60,96	62,01	0,0000
XIAO *et al.*(2006)	Compaction + PSO	**59,93**	**61,41**	**0,0000**
	IWO	**60,21 (+1,2x10^{-5})**	**61,14 (+1,3x 10^{-4})**	**0,0857**

A Tabela 4 apresenta o resultado gerado pela alocação de 15 objetos (Tabela 3). Para execução deste experimento utilizou-se dos seguintes parâmetros iniciais: $P_{initial}$=40, P_{max}=200, $iter_{Max}$=300, $D_{problem}$=80, S_{max}=40, S_{min}=3, n=3, σ_{final}=0,0001 *e* $\sigma_{inicial}$ =20. IWO obteve um raio mínimo de 67,62, um resultado muito semelhante à melhor solução sugerido por XIAO *et al.* (2006) ('Compaction+PSO' = 67,65). Quanto à média a cada 10 execuções, IWO também apresentou resultados melhores que os valores sugeridos pela literatura.

Tabela 3. Informações sobre raio e massa de 15 objetos. Fonte: (XIAO et al., 2006).

	Raio	Massa		Raio	Massa
1	8	75	**9**	6	76
2	14	29	**10**	18	85
3	8	36	**11**	24	59
4	15	58	**12**	13	18
5	11	75	**13**	20	85
6	17	32	**14**	10	36
7	21	98	**15**	15	12
8	16	52			

Tabela 4. Resultados comparativos entre algumas implementações da Literatura e IWO para 15 objetos. Fonte: (XIAO et al., 2006).

LITERATURA	ALGORITMO	< Raio (+dR)	Média: Raio (+dR)	Média: Inst.
Zhou *et al.*(2005)	PSO	76,58	78,52 **(+2,4x 10^{-7})**	0,0002
XIAO *et al.*(2006)	Compaction + SA	68,77	69,45	0,0000
XIAO *et al.*(2006)	Compaction + PSO	**67,65**	**68,94**	**0,0000**
	IWO	**67,62 (+2,8x10^{-5})**	**68,57 (+1,6x10^{-6})**	**0,0014**

A Tabela 6 apresenta o resultado gerado pela alocação de 40 objetos (Tabela 5). Para execução deste experimento utilizou-se dos seguintes parâmetros iniciais: $P_{initial}$=40, P_{max}=200, $iter_{Max}$=300, $D_{problem}$=80, S_{max}=40, S_{min}=3, n=3, σ_{final}=0,0001 *e* $\sigma_{inicial}$ =20. IWO obteve significativamente uma solução melhor que os outros valores encontrados para raio mínimo na literatura. IWO encontrou um raio mínimo de 740,18 com instabilidade na ordem de 0,0014, enquanto a melhor solução sugerida por LEI *et al.* (2009) foi de 769,81 com instabilidade de massa 0,0003.

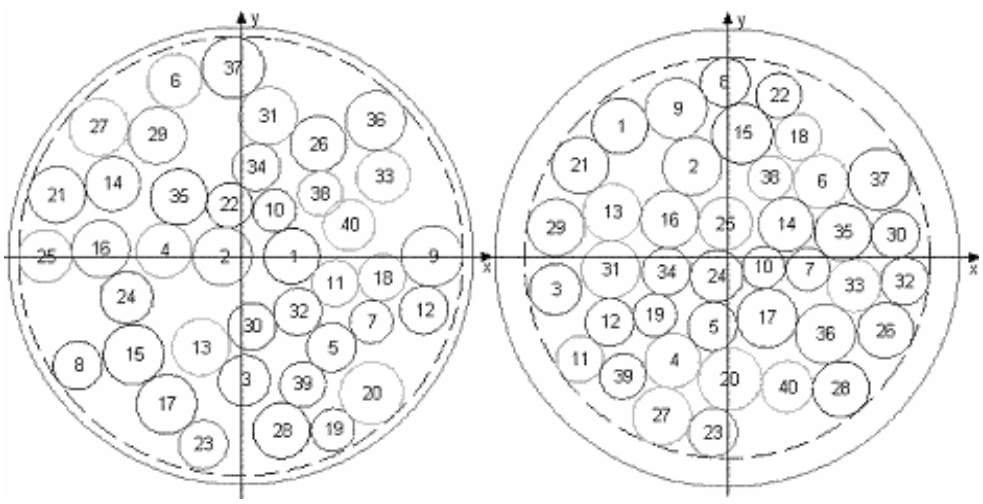
Tabela 5. Informações sobre raio e massa de 40 objetos. Fonte: (LEI et al., 2009).

	Raio	Massa		Raio	Massa		Raio	Massa		Raio	Massa
1	106	11	**11**	89	7	**21**	108	11	**31**	111	12
2	112	12	**12**	92	8	**22**	86	7	**32**	91	8
3	98	9	**13**	109	11	**23**	93	8	**33**	101	10
4	105	11	**14**	104	10	**24**	100	10	**34**	91	8
5	93	8	**15**	115	13	**25**	102	10	**35**	108	11
6	103	10	**16**	110	12	**26**	106	11	**36**	114	12
7	82	6	**17**	114	12	**27**	111	12	**37**	118	13
8	93	8	**18**	89	7	**28**	107	11	**38**	85	7
9	117	13	**19**	82	6	**29**	109	11	**39**	87	7
10	81	6	**20**	120	14	**30**	91	8	**40**	98	9

Tabela 6. Resultados comparativos entre algumas implementações da Literatura e IWO para 40 objetos. Fonte: (LEI et al., 2009).

LITERATURA	ALGORITMO	< Raio (+dR)	Inst.
TENG *et al.* (1994)		870,331 **(+1,5x10^{-5})**	0,006
FEI e HONGFEI (1999)	GA	874,83 **(+2,9x10^{-2})**	11,39
NING *et al.* (2004)	PSO	843,94 **(+7,7x10^{-6})**	0,003
LEI *et al.* (2009)	PSO	**769,81 (+7,7x10^{-7})**	**0,0003**
	IWO	**740,18 (+3,7x10^{-5})**	**0,0014**

A Figura 4 apresenta o *layout* das melhores soluções encontradas na literatura para o problema de alocação dos 40 objetos da Tabela 5. Já na Figura 5 pode ser visualizada a melhor solução encontrada por IWO neste trabalho, que demonstra ser a melhor solução até então da literatura.



(a) (b)

Figura 4. Melhores soluções encontradas na literatura para alocação de 40 objetos (Tabela 7). NING et al. (2004): 843,94 de raio mínimo. (b) LEI et al. (2009): 769,819 de raio mínimo. Fonte: (LEI et al., 2009).

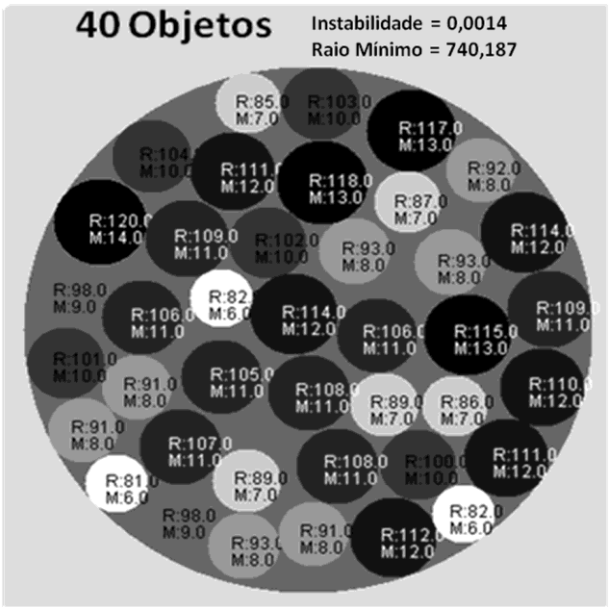


Figura 5. Melhor solução encontrada para alocação de 40 objetos (Tabela 7).

O processo de ajuste aplicado à melhor solução encontrada pelo algoritmo de ervas daninhas para o problema de alocação de círculos neutraliza o desbalanceamento em detrimento de um pequeno aumento no raio do recipiente. Entretanto, mesmo com esse aumento no raio, as soluções alcançadas por este trabalho foram as melhores dentre a literatura pesquisada. Ressalta-se que a contribuição deste trabalho, de aplicar um pós-processamento para ajuste do centro de massa do conjunto de elementos circulares, torna qualquer solução estável em termos de equilíbrio de massas, permitindo que a análise comparativa entre os algoritmos possa a partir de então ser tratada apenas como minimização de raios.

5. Conclusão e Considerações Finais

O algoritmo de otimização por infestação de ervas daninhas mostrou-se muito bom em relação à instabilidade de massa e superior a vários métodos encontrados na literatura, em relação ao raio mínimo. Com o reajuste proposto neste trabalho, a instabilidade é eliminada para qualquer solução. Nos problemas estudados, verificou-se que o acréscimo no raio de cada solução ajustada do IWO sofreu uma variação insignificante, permitindo a este algoritmo um desempenho ainda superior a seus concorrentes.

Com o aumento da quantidade de objetos a serem alocados em um recipiente circular, pode-se perceber que é fundamental aumentar o número de população máxima e número de iterações. Tendo em vista a demanda de memórias durante a realização dos testes, têm-se como trabalho futuro a aplicação das técnicas de *MultiThread* que permite executar múltiplos processos ao mesmo tempo de forma independente em computadores *multi-core*. Assim sendo, a utilização de computadores mais rápidos e com maior quantidade de memória junto com a aplicação de processamento permitirá o aumento da quantidade máxima de população e do número de iterações em relação aos valores utilizados neste trabalho. Isso deverá melhorar os resultados encontrados com menor tempo de processamento.

A utilização das técnicas de orientação a objetos gerou a possibilidade da construção de uma plataforma de *software* na utilização de problemas de otimização. Assim, qualquer problema dessa espécie pode ser facilmente implementado na plataforma de otimização por infestação de ervadas daninhas.

Uma avaliação importante para um próximo trabalho é realizar uma análise estatística dos parâmetros iniciais obrigatórios a fim de compreender as suas variações e melhores ajustes para cada problema.

A disposição da carga nos satélites do Programa Espacial Brasileiro é feita de maneira *ad hoc*, dependendo principalmente da experiência técnica da equipe responsável por este segmento. Uma ferramenta que auxilie nessa tarefa de maneira automática ou semi-automática é, portanto, de grande aplicação prática. Nesta direção, pretende-se no futuro aumentar a gama de formas geométricas suportadas para representar os objetos da carga e do próprio satélite, além de considerar três dimensões, com a existência de prateleiras. Também deverão ser consideradas restrições mais sofisticadas, como a necessidade de alguns objetos deverem estar próximos ou distantes entre si, devido a fatores como calor irradiado, minimização de cabos, etc.

Agradecimentos: ao Professor Dr. Horário Hideki Yanasse e aos órgãos financiadores CAPES e CNPq.

6. Referências

Akeb, H.; Li Y. (2005) Basic heuristics for packing a great number of equal circles. Working paper No 7, LaRIA, Université de Picardie Jules Verne, Amiens.

Becceneri, J. C. (Org.) and Silva Neto, J. A. (Org.) (2009) Técnicas de Inteligência Computacional Inspiradas na Natureza - Aplicação em Problemas Inversos em Transferência Radiativa. São Carlos: SBMAC, v. 41, p.122.

Fei, T. and Hongfei, T. (1999) A modified genetic algorithm and its application to layout optimization. In: Journal of Software 10(10), p. 1096–1102.

Hajimirsadeghi, H. and Lucas, C. (2009) A hybrid IWO/PSO algorithm for fast and global optimization. In: EUROCON 2009, EUROCON'09. IEEE, p. 1964–1971.

Huang, W. Q., Li Y. ; Li, C. M. and Xu, R. C. (2006) New heuristics for packing unequal circles into circular container. In: Journal Computers and Operations Research. v. 33, Issue 8, p. 2125–2142, August.

Lei, K. (2009) Constrained Layout Optimization Based on Adaptive Particle Swarm Optimizer. Faculty of Computer & Information Science, Southwest University, Chongqing, China. Springer-Verlag Berlin Heidelberg, LNCS 5821, p. 434–442.

Mehabian, A. and Lucas, C. (2006) A novel numerical optimization algorithm inspired from weed colonization. In: Ecological Informatics, v. 1, n. 4, p. 355–366.

Ning, L., Fei, L. and Debao, S. (2004) A study on the particle swarm optimization with mutation operator constrained layout optimization. In: Chinese Journal of Computers, 27(7), p. 8897–9039.

Rad, H. and Lucas, C. (2008) A recommender system based on invasive weed optimization algorithm. In: Evolutionary Computation. CEC 2007. IEEE Congress on, IEEE, p. 4297–4304.

Tellez, F. R. (2010) Otimização multiobjetivo por infestação de ervas daninhas em um grid cooperativo. Monografia (Bacharelado em Ciência da Computação) - Universidade Federal de São Paulo, São Paulo.

Teng, H. F., Shoulin, S. and Wenhai, G. (1994) Layout optimization for the dishes installed on rotating table. In: Science in China (Series A) 37(10), p. 1272–1280.

Xiao, R.B., Xu YC and Amos M. (2006)Two hybrid compaction algorithms for the layout optimization problem, School of Management & CAD Center, Huazhong University of Science & Technology, Wuhan, China. In: Biosystems, 90(2), p. 560-7, December.

Zhou, C., Gao, L. and Gao, H. (2005) Particle swarm optimization based algorithm for constrained layout optimization. Contr. Decis. 20(1), p. 36-40.

Agrupamento baseado em SOM com pesos adaptativos para múltiplas tabelas de dissimilaridade

Anderson B. S. Dantas[1], **Francisco de A. T. de Carvalho**[1]

[1]Centro de Informática – Universidade Federal de Pernambuco (CIn/UFPE)
Av. Prof. Luiz Freire, s/n - Cidade Universitária, CEP 50740-540, Recife – PE – Brazil

{absd,fatc}@cin.ufpe.br

Abstract. *This paper introduces a clustering algorithm based on batch Self-organizing map to partition objects taking into account their relational descriptions given by multiple dissimilarity matrices. The presented approach provide a partition of the objects and a prototype for each cluster, moreover the method is capable of learn relevance weights for each dissimilarity matrix by optimizing an adequacy criterion that measures the fit between clusters and the respectives prototypes. These relevance weights change at each iteration and are different from one cluster to another. Experiments using real-world data bases are considered to show the usefulness of the method.*

Resumo. *Este trabalho apresenta um modelo de algoritmo de agrupamento baseado no algoritmo SOM com treinamento batch para classificar objetos levando em consideração suas descrições relacionais dadas por múltiplas matrizes de dissimilaridade. O método apresentado tem como resultado uma partição dos dados de entrada, assim como um protótipo para cada agrupamento, além de adaptar pesos que calculam a relevância de cada matriz de dissimilaridade otimizando um critério de adequação entre os agrupamentos e seus respectivos protótipos. Estes pesos mudam a cada iteração do algoritmo e são diferentes de um grupo para outro. Experimentos com bases de dados reais foram realizado com o objetivo de mostrar a utilidade do algoritmo.*

1. Introdução

Algoritmos de agrupamento (*clutering*) têm sido largamente utilizados em áreas de aprendizagem de máquina como mineração de dados, recuperação de documentos, segmentação de imagens e classificação de padrões [Jain et al. 1999]. Métodos de agrupamento objetivam organizar um conjunto de dados em grupos (*clusters*), de forma que itens pertencentes a um mesmo grupo (*cluster*) tenham alto grau de similaridade, por outro lado, itens em grupos diferentes possuem alto grau de dissimilaridade.

O mapa auto-organizável de Kohonen (Self-organizing map) [Kohonen 1990] é um tipo especial de rede neural não-supervisionada e possui propriedades de agrupamento e visualização [Gopalakrishnan et al. 2008]. SOM pode ser considerado um algoritmo que faz a projeção de dados multidimensionais em um espaço de uma, duas ou, em casos especiais, três dimensões. Esta projeção possibilita uma partição das entradas em grupos similares ao mesmo tempo em que preserva a topologia original dos dados.

Existem duas representações comuns dos objetos em que os algoritmos de agrupamento podem ser baseados: dados caracterizados ou relacionais. Quando cada objeto é

descrito por um vetor de valores quantitativos ou qualitativos, os vetores que descrevem os objetos são chamados dados caracterizados. Quando cada par de objetos é representado por uma relação então temos dados relacionais. O modelo mais comum de dados relacionais é o caso de uma matriz de dissimilaridades $R = [r_{il}]$, onde r_{il} é a dissimilaridade pareada (geralmente uma distância) entre os objetos i e l.

[Golli et al. 2004] propõe uma adaptação do modelo SOM para dados de dissimilaridade. [Lechevallier et al. 2010] apresenta um algoritmo de agrupamento que realiza partição de objetos baseado nas descrições relacionais destes objetos dadas por múltiplas matrizes de dissimilaridade.

Este trabalho propõe um algoritmo baseado em SOM com pesos adaptativos para classificar dados baseados em matrizes de dissimilaridades. Estas matrizes podem ser obtidas através de diferentes conjuntos de variáveis e uma função de dissimilaridade fixa (neste caso a partição final apresenta um consenso entre diferentes conjuntos de variáveis descrevendo os objetos), através de um conjunto fixo de variáveis e diferentes funções de dissimilaridade (consenso entre diferentes funções de dissimilaridade) ou através de diferentes conjuntos de variáveis e funções de dissimilaridade. A relevância de diferentes matrizes de dissimilaridades não é igualitária na definição final dos agrupamentos. Assim, para obter uma partição significativa partindo de todas as matrizes faz-se necessário o uso de pesos adaptativos para cada matriz de dissimilaridade e diferentes para cada agrupamento.

Este trabalho está organizado da seguinte forma: a seção 2 traz uma introdução sobre o algoritmo SOM e sua definição formal. A seção 3 trata do algoritmo SOM com treinamento batch aplicado a dados de dissimilaridade. O algoritmo SOM com pesos adaptativos aplicado a dados representados por matrizes de dissimilaridade é apresentado na seção 4. Os experimentos realizados com bases de dados reais são apresentados na seção 5. Finalmente, a seção 6 traz as conclusões do trabalho.

2. Self-organizing map (SOM)

Os algoritmos chamados *self-organizing maps* [Kohonen 1990] (mapas auto-organizáveis) fazem parte da classe de métodos de aprendizado não-supervisionado, todas as propriedades de cada grupo em que são classificados os dados, são estimadas ou aprendidas sem o uso de informação a priori [Murtagh and Hernández-Pajares 1995]. O objetivo central desses modelos é encontrar uma estrutura nos dados fornecidos.

A arquitetura do algoritmo SOM consiste em um conjunto de neurônios organizados topologicamente em uma grade, geralmente com uma, duas, ou três dimensões, chamada mapa. De modo mais formal, o mapa é descrito por um grafo não-orientado (C, Γ). C é um conjunto de m neurônios interconectados com topologia definida por Γ. A estrutura de grafo permite que se defina uma função de distância entre dois neurônios no mapa. Para cada par de neurônios (*c,r*) no mapa, $\delta(c, r)$ é o comprimento do caminho mais curto entre *c* e *r* no grafo *C*. Essa função de distância permite que haja uma relação de vizinhança entre neurônios. Cada neurônio c é representado por um vetor de referência $w_c = \{w_c^1, \ldots, w_c^p\}$, onde p é igual à dimensão dos vetores de entrada. A quantidade de neurônios varia de acordo com a aplicação.

A principal característica dos mapas auto-organizáveis é a possibilidade de se comparar os agrupamentos. Cada elemento que compõe os dados fornecidos é afetado a

um grupo. Cada grupo é projetado em um neurônio do mapa. Elementos semelhantes são projetados no mesmo neurônio, enquanto que a dissimilaridade aumenta com a distância que separa duas projeções.

O conceito de vizinhança é incorporado ao algoritmo através do uso de uma função *kernel* K positiva e tal que $\lim_{|x|\to\infty} K(x) = 0$ [Badran et al. 2005]. As distâncias $\delta(c, r)$ entre os neurônios *c* e *r* do mapa permitem a definição da influência relativa dos neurônios sobre os objetos. $K(\delta(c, r))$ quantifica essa influência.

Para que o tamanho da vizinhança seja um parâmetro relevante, é utilizada a família de *kernels* K^T parametrizada por T: $K^T(\delta) = K(\delta/T)$. Quanto menor o valor de T, menor a quantidade de neurônios pertencentes à vizinhança de um determinado neurônio c. O valor de T, na iteração t, é atualizado segundo a equação:

$$T = T_{max} * (\frac{T_{min}}{T_{max}})^{\frac{t}{N_{iter}-1}} \tag{1}$$

T_{min} e T_{max} representam, respectivamente, o valor final e inicial de T. N_{iter} é o número máximo de iterações.

Seja $E = \{e_1, \ldots, e_n\}$ um conjunto de n objetos. A cada iteração do algoritmo de treinamento, são realizadas duas etapas. A primeira etapa é uma etapa de afetação, a segunda fase é de minimização de uma função de custo J. Durante a etapa de afetação, os dados são particionados de acordo com uma função de afetação f, que associa cada elemento e_i do conjunto de dados ao vetor de referência mais ”próximo”a e_i. O algoritmo de Kohonen usa a seguinte função de afetação:

$$f(e_i) = arg\min_{c} \parallel e_i - \mathbf{w}_c \parallel^2 \tag{2}$$

A função de custo é definida da seguinte forma:

$$J(f, W) = \sum_{e_i \in E} \sum_{c \in C} K^T(\delta(c, f(e_i))) \parallel e - \mathbf{w}_c \parallel^2 . \tag{3}$$

Onde W é o conjunto de vetores de referência e $\parallel \cdot \parallel$ representa uma medida de dissimilaridade que pode ser expressa em termos de uma função de distância, por exemplo, a distância Euclidiana seria uma escolha plausível. É importante notar que neste tipo de algoritmo somente um indivíduo é apresentado por vez, os neurônios do mapa se ajustam às entradas a medida em que elas são apresentadas.

A etapa seguinte, de minimização, atualiza o conjunto de vetores de referência W para minimizar a função de custo. O novo conjunto W^t é calculado de acordo com:

$$w_c^t = w_c^{t-1} - \mu^t K^T(\delta(c, f_t(e_i)))(w_c^{t-1} - e_i), \tag{4}$$

onde μ^t é a taxa de aprendizado na iteração t.

3. Batch self-organizing map aplicado em dados de dissimilaridade

O algoritmo SOM batch baseado em dados de dissimilaridade conforme proposto por [Golli et al. 2004] também é descrito por um grafo (C, Γ). A principal diferença está nos dados que serão classificados. Neste caso, os dados são representados por uma relação de dissimilaridade. O algoritmo de treinamento do tipo *batch* é um algoritmo iterativo onde todo o conjunto de exemplos (E) é apresentado ao mapa antes que qualquer ajuste seja realizado. A seguir é dada a definição formal deste modelo.

Seja $E = \{e_1, \ldots, e_n\}$ um conjunto de n objetos e uma medida de dissimilaridade $d(e_i, e_l)$ entre os objetos e_i e e_l. Cada neurônio c é representado por um protótipo $g_c = e_{j_i},\ e_{j_i} \in E$. Os protótipos iniciais são selecionados aleatoriamente. Cada iteração do algoritmo alterna entre duas fases (afetação e representação) objetivando minimizar a seguinte função custo (ou função de adequação):

$$J(f, g) = \sum_{i=1}^{n} \sum_{l \in C} K^T(\delta(f(e_i), l)) d(e_i, g_l) \tag{5}$$

Durante a etapa de afetação, a função f afeta cada indivíduo e_i ao neurônio mais próximo. O valor do parâmetro T é definido conforme a equação 1.

$$f(e_i) = arg \min_{c \in C} \sum_{l \in C} K^T(\delta(c, l))\, d(e_i, g_l) \tag{6}$$

Durante a etapa de representação, são selecionados novos protótipos que representam o conjunto de observações. Esta etapa de otimização pode ser realizada independentemente para cada neurônio. O protótipo g_l^* do grupo C_l, que minimiza o critério de adequação J é calculado segundo a equação:

$$g_l^* = arg \min_{e \in E} \sum_{i=1}^{n} \exp\{-\frac{(\delta((f(e_i))^{(t-1)}, l))^2}{2T^2}\} d(e_i, e) \tag{7}$$

4. Self-organizing map adaptativo para múltiplas tabelas de dissimilaridade

4.1. Introdução

Nesta seção apresentamos um algoritmo adaptativo baseado no algoritmo SOM para dados de dissimilaridade [Golli et al. 2004] e no algoritmo de agrupamento para múltiplas tabelas de dissimilaridade de [Lechevallier et al. 2010]. O objetivo do modelo apresentado é mapear objetos levando em conta suas descrições relacionais dadas por múltiplas matrizes de dissimilaridade. O algoritmo é capaz de associar dados de entrada com pontos representativos de um mapa formando grupos de objetos. O método utiliza pesos adaptativos diferentes de um grupo para outro com o intuito de ponderar sobre as matrizes de dissimilaridades e medir sua relevância na formação de cada agrupamento.

Seja $E = \{e_1, \ldots, e_n\}$ um conjunto de n objetos e p o número de matrizes de dissimilaridade $\mathbf{D}_j = [d_j(e_i, e_l)](j = 1, \ldots, p)$, onde $d_j(e_i, e_l)$ é a dissimilaridade entre os objetos e_i e $e_l(i, l = 1, \ldots, n)$ na matriz de dissimilaridades $\mathbf{D}_j$. Assuma que o protótipo

g_l do grupo (*cluster*) C_l é um elemento do conjunto de objetos E, i.e., $g_l \in E \; \forall l = 1, \ldots, L$.

O algoritmo SOM ponderado para múltiplas tabelas de dissimilaridade calcula uma partição $P = (C_1, \ldots, C_l)$ de E em L grupos e o respectivo protótipo g_l representando o grupo C_l em P, tal que um determinado critério de adequação (função objetivo), que mede o ajuste entre os grupos e seus repectivos protótipos é localmente otimizado. O critério de adequação é calculado de forma semelhante às equações 3 e 5:

$$J = \sum_{i=1}^{n} \sum_{l=1}^{c} \exp\{-\frac{(\delta((f(e_i)), l))^2}{2T^2} \sum_{j=1}^{p} \lambda_{lj} d_j(e_i, g_l) \tag{8}$$

onde a função kernel K é definida por: $K^T(\delta) = \exp(-\frac{\delta^2}{T^2})$. $\delta(c, r)$ calcula a distância entre dois neurônios no mapa. A matriz de pesos λ é composta por c vetores de pesos $\lambda_k = (\lambda_k^1, \ldots, \lambda_k^j, \ldots, \lambda_k^p)(k = 1, \ldots, c)$. A matriz de pesos muda a cada iteração do algoritmo, ou seja, os vetores não possuem valores determinados absolutamente e são diferentes para cada matriz de dissimilaridade. O objetivo da matriz de pesos é ponderar sobre cada matriz de dissimilaridade avaliando sua relevância na minimização do critério de adequação. O modelo segue três etapas definidas (representação, atualização de pesos e afetação):

Etapa 1: Definição dos melhores protótipos

Nesta etapa, a partição $P^{(t-1)} = (P_1^{(t-1)}, \ldots, P_c^{(t-1)})$ e $\lambda_l^{(t)}(l = 1, \ldots, c)$ são fixos. O protótipo $g_l^{(t)} = g^* \in E$ do grupo C_l, que minimiza o critério de adequação E é calculado segundo a equação:

$$g^* = arg \min_{e \in E} \sum_{i=1}^{n} \exp\{-\frac{(\delta((f(e_i))^{(t-1)}, l))^2}{2T^2}\} \sum_{j=1}^{p} \lambda_{lj}^{(t-1)} d_j(e_i, e) \tag{9}$$

Etapa 2: Definição dos melhores pesos

Nesta etapa, a partição $P^{(t-1)} = (P_1^{(t-1)}, \ldots, P_c^{(t-1)})$ e os protótipos $g_l^{(t)} \in E(l = 1, \ldots, c)$ são fixos. O elemento j do vetor de pesos $\lambda_l = (\lambda_k^1, \ldots, \lambda_k^p)$, que minimiza o critério de adequação E é calculado segundo a expressão:

$$\lambda_{lj}^{(t)} = \frac{\left\{\prod_{h=1}^{p} (\sum_{i=1}^{n} \exp\{-\frac{\delta^2(f^{(t-1)}(e_i), l)}{2T^2}\} d_h(e_i, g_l^{(t)}))\right\}^{\frac{1}{p}}}{\sum_{i=1}^{n} \exp\{-\frac{\delta^2(f^{(t-1)}(e_i), l)}{2T^2}\} d_j(e_i, g_l^{(t)})} \tag{10}$$

Etapa 3: Definição da melhor partição

Nesta etapa, os protótipos $g_l^{(t)} \in E(l = 1, \ldots, c)$ e $\lambda_l^{(t)}(l = 1, \ldots, c)$ são fixos. O grupo C_r que minimiza o critério E é determinado de acordo com a equação:

$$r = (f(e_i))^{(t)} = arg \min_{1 \leq h \leq c} \sum_{l=1}^{c} \exp\{-\frac{(\delta(h,l))^2}{2T^2}\} \sum_{j=1}^{p} \lambda_{lj}^{(t)} d_j(e_i, g_l^{(t)}) \quad (11)$$

4.2. O algoritmo

1. *Inicialização*
   Fixe: o número c de grupos, a função δ, o número de iterações N_{iter} e os valores de T_{min} e T_{max};
   Atribua $T \leftarrow T_{max}$; Atribua $t \leftarrow 0$;
   Selecione c protótipos aleatoriamente $g_l^{(0)} \in E(l = 1, \ldots, c)$;
   Configure $\lambda_l^{(0)} = (\lambda_{l1}^{(0)}, \ldots, \lambda_{lp}^{(0)}) = (1, \ldots, 1)(l = 1, \ldots, c)$;
   Configure o mapa $L(c, \mathbf{G}^0)$, onde $\mathbf{G}^0 = (g_1^{(0)}, \ldots, g_c^{(0)})$.
   Cada objeto e_i é afetado ao protótipo mais próximo com o objetivo de obter a partição $P^{(0)} = (P_1^{(0)}, \ldots, P_c^{(0)})$ de acordo com a equação 11, para $t = 0$.
2. *Passo 1: Representação*
   Atribua $t = t + 1$ e $T = T_{max}(\frac{T_{min}}{T_{max}})^{\frac{t}{N_{iter}-1}}$;
   A partição $P^{(t-1)} = (P_1^{(t-1)}, \ldots, P_c^{(t-1)})$ e $\lambda_l^{(t)}(l = 1, \ldots, c)$ são fixos.
   Calcule o protótipo $g_l^{(t)} = g^* \in E$ do grupo $P_l^{(t-1)}(l = 1, \ldots, c)$ de acordo com a equação 9.
3. Passo 2: Atualização dos pesos
   A partição $P^{(t-1)} = (P_1^{(t-1)}, \ldots, P_c^{(t-1)})$ e os protótipos $g_l^{(t)} \in E(l = 1, \ldots, c)$ são fixos.
   Atualize o vetor de pesos de acordo com a equação 10.
4. *Passo 3: Afetação*
   Os protótipos $g_l^{(t)} \in E(l = 1, \ldots, c)$ e $\lambda_l^{(t)}(l = 1, \ldots, c)$ são fixos
   $P^{(t)} \leftarrow P^{(t-1)}$
   para $i = 1$ até n faça:
   encontre o grupo $C_m^{(t)}$ ao qual o objeto e_i pertence
   encontre o grupo vencedor $C_r^{(t)}$, onde r é obtido pela equação 11

   se $r \neq m$:
   $C_r^{(t)} \leftarrow C_r^{(t)} \cup \{e_i\}$
   $C_m^{(t)} \leftarrow C_m^{(t)} \setminus \{e_i\}$

5. *Critério de parada*
   Se $T == T_{min}$ então PARE; senão volte ao passo 1

5. Experimentos e resultados

Para mostrar a utilidade do algoritmo proposto, foram realizados experimentos utilizando bases de dados obtidas do repositório UCI Machine Learning Repository [Frank and Asuncion 2010]. No algoritmo SOM adaptativo para múltiplas tabelas, cada um dos atributos que compõem as bases é representado por uma matriz de dissimilaridade. Para o modelo SOM batch não-adaptativo é considerada somente uma matriz

de dissimilaridade. Com a finalidade de avaliar os resultados dos agrupamentos realizados pelo método proposto neste trabalho foi considerado o índice de Rand corrigido (CR) [Hubert and Arabie 1985], a medida F-*measure* [van Rijsbergen 1979] e a taxa de erro global de classificação (OERC - overall error rate of classification) [Breiman et al. 1984].

O índice CR avalia o grau de similaridade entre a partição a priori e a partição fornecida pelo algoritmo de agrupamento. Além disso, o índice CR não é sensível ao número de classes nas partições ou à distribuição dos itens dentro dos grupos. O índice CR pode assumir valores no intervalo [-1,1], onde o valor 1 indica perfeita combinação entre as partições, valores perto de 0 ou negativos indicam pouca semelhança. O índice F-*measure* entre a partição a priori e a partição obtida pelo algoritmo de agrupamento assume valores no intervalo [0,1], em que 1 indica perfeita combinação entre as partições. O índice OERC mede a habilidade de um algoritmo de agrupamento encontrar classes a priori presentes na base de dados.

Além dos índices é apresentada a matriz de confusão para cada resultado. A matriz de confusão mostra a quantidade de elementos de cada classe que foi classificada em um determinado grupo. É possível, então identificar qual classe um determinado grupo representa.

Alguns parâmetros são considerados para a realização dos experimentos. São eles: topologia do mapa, o valor de T_{min}, T_{max} e o número de iterações N_{iter}. A topologia do mapa define o número máximo de grupos que serão formados. Todos os parâmetros foram definidos empiricamente. A tabela 1 mostra os valores dos parâmetros utilizados em todos os experimentos descritos a seguir. Cada experimento foi repetido 50 vezes com a mesma configuração, dentre as 50 repetições foi selecionada a que obteve menor critério de adequação.

Tabela 1. Parâmetros utilizados nos experimentos

Parâmetros	Valor
Topologia	2x5
T_{min}	0,3
T_{max}	3,0
N_{iter}	500

5.1. Base de dados Wine

A base de dados wine consiste em resultados da análise química de vinhos produzidos numa mesma região na Itália, mas derivados de três cultivos diferentes, formando três classes de vinhos. As classes 1, 2 e 3 possuem, respectivamente 59, 71 e 48 instâncias. Cada indivíduo é descrito por 13 atributos com valores reais representando os valores de 13 componentes encontrados em cada um dos três tipos de vinho.

As tabelas 2 e 3 mostram as matrizes de confusão obtidas pelos algoritmos batch SOM e batch SOM adaptativo para múltiplas tabelas de dissimilaridade, respectivamente. Analisando essas tabelas é possível perceber regiões onde dados da mesma classe a-priori são classificados em grupos próximos no mapa. Na tabela 3, os grupos (0,0), (0,1), (1,0) e (1,1), localizados na parte esquerda do mapa, possuem dados da classe 3, formando uma região bem definida. Os grupos (1,3) e (1,4) possuem dados da classe 1, os demais

grupos são da classe 2. Na tabela 2, na parte mais à esquerda do mapa, predominam dados da classe 2, na parte central observa-se dados da classe 3, enquanto que na parte mais à direita do mapa são predominantes dados pertencentes à classe 1.

Tabela 2. Wine: Matriz de confusão obtida pelo algoritmo batch SOM

Classe/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	0	1	7	10	6	0	0	5	16	14
2	20	11	3	0	0	25	6	5	1	0
3	10	16	5	0	0	2	9	6	0	0

Tabela 3. Wine: Matriz de confusão obtida pelo algoritmo batch SOM adaptativo

Classe/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	0	0	0	2	6	0	0	0	24	27
2	5	1	21	22	10	0	0	9	2	1
3	16	4	0	0	0	15	13	0	0	0

Os índices CR, F-*measure* e OERC obtiveram, respectivamente, os valores 0,42, 0,52, 0,09 para o modelo SOM adaptativo, e os valores, respectivamente 0,31, 0,45, 0,27 para o modelo SOM batch não-adaptativo. O algoritmo SOM adaptativo para múltiplas tabelas de dissimilaridade obteve melhores índices do que o modelo batch não-adaptativo.

A tabela 4 mostra a matriz de pesos final obtida pelo algoritmo batch SOM adaptativo para múltiplas tabelas de dissimilaridade. Cada peso representa a relevância de uma determinada matriz de dissimilaridade para a formação de cada grupo. Valores acima de 1 indicam uma forte influência da matriz durante a afetação de objetos em um determinado grupo. Por exemplo, na tabela 4 o grupo (0,0) tem uma influência maior da matriz 7 (3,05), enquanto que o grupo (0,1) tem maior influência da matriz 13 (8,64).

5.2. Base de dados Iris

Esta base de dados consiste em três tipos (classes) de planta íris: iris setosa, iris versicolor e iris virginica. Cada uma das classes possui 50 exemplos. Cada instância é descrita por quatro atributos.

As tabelas 5 e 6 mostram a matriz de confusão obtidas pelos algoritmos batch SOM e batch SOM adaptativo para múltiplas tabelas de dissimilaridade. É possível identificar regiões no mapa com dados da mesma classe. Por exemplo, na tabela 5, a parte inferior do mapa tem predominância de dados da classe 3 (grupos (1,0), (1,1), (1,2) e (1,3)) e na tabela 6, a classe 1 está mais presente na parte esquerda do mapa (grupos (0,0) e (1,0)).

Os índices CR, F-*measure* e OERC obtiveram, respectivamente, os valores 0,54, 0,64 e 0,04 para o modelo SOM adaptativo para múltiplas tabelas de dissimilaridade, e os valores, respectivamente 0,50, 0,62, 0,06 para o modelo SOM batch não-adaptativo. O modelo adaptativo obteve melhores índices do que o modelo não adaptativo.

A tabela 7 mostra a matriz de pesos final obtida pelo algoritmo batch SOM adaptativo para múltiplas tabelas de dissimilaridade. Pode-se perceber que os pesos para as

Tabela 4. Wine: Matriz de pesos obtida pelo algoritmo batch SOM adaptativo

Matriz/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	**1,03**	0,99	**1,27**	0,69	0,76	**1,02**	0,46	0,42	0,77	0,48
2	0,27	**1,05**	0,52	0,33	**4,19**	0,38	0,21	0,61	0,55	**6,01**
3	**1,31**	0,55	0,41	0,93	0,27	0,63	0,79	0,51	0,69	0,37
4	**1,07**	**1,09**	0,43	**2,64**	0,99	0,89	0,90	**1,35**	0,16	0,33
5	0,48	0,27	**1,47**	**1,32**	0,16	0,47	**1,85**	0,16	0,29	0,61
6	0,90	0,69	0,60	0,48	**1,92**	**1,68**	0,49	**2,93**	**1,38**	1,64
7	**3,05**	0,79	**2,42**	**2,29**	0,84	**2,42**	**5,07**	**3,12**	**3,13**	**1,65**
8	0,33	0,51	0,90	**1,22**	**1,96**	0,25	0,86	0,24	**2,14**	0,94
9	**1,87**	0,41	**1,57**	0,45	0,24	**2,79**	0,24	**2,27**	**1,12**	0,40
10	**1,25**	0,92	**5,17**	**2,39**	**3,04**	0,46	0,36	**2,96**	**1,95**	**1,50**
11	0,57	**1,80**	0,18	0,36	**3,72**	**1,43**	**2,22**	**1,33**	**1,01**	**1,74**
12	**2,07**	**3,60**	0,51	**4,02**	**2,15**	**5,12**	**5,23**	**1,72**	**3,90**	**2,79**
13	**2,02**	**8,64**	**5,65**	0,63	0,38	0,99	**3,48**	**1,01**	0,86	0,60

Tabela 5. Iris: Matriz de confusão obtida pelo algoritmo batch SOM

Classe/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	0	0	0	5	32	0	0	0	0	13
2	1	0	0	0	0	26	7	0	16	0
3	15	11	9	0	0	1	7	7	0	0

Tabela 6. Iris: Matriz de confusão obtida pelo algoritmo SOM adaptativo

Classe/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	13	0	0	0	0	37	0	0	0	0
2	0	0	17	17	4	0	2	9	1	0
3	0	9	0	1	14	0	0	0	6	20

matrizes 3 e 4 possuem valores maiores indicando maior influência na formação dos agrupamentos.

Tabela 7. Iris: Matriz de pesos obtida pelo algoritmo batch SOM adaptativo

Matriz/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	0,63	0,80	0,34	0,15	0,59	0,26	**1,59**	0,35	0,25	0,52
2	0,17	0,81	0,26	0,29	0,32	0,07	0,52	0,50	**1,53**	0,26
3	**3,40**	**1,45**	**2,95**	**10,14**	**4,99**	**5,06**	0,95	**1,50**	**1,47**	**2,59**
4	**2,62**	**1,04**	**3,79**	**2,26**	**1,05**	**11,38**	**1,26**	**3,73**	**1,76**	**2,83**

5.3. Base de dados Thyroid

Esta base possui dados sobre os estados da glândula tireóide: normal, hipertireoidismo e hipotireoidismo. As três classes (1, 2 e 3) possuem, respectivamente, 150, 35 e 30 exemplos. Cada instância é descrita por 5 atributos com valores reais.

As tabelas 8 e 9 mostram a matriz de confusão obtidas pelos algoritmos batch SOM e batch SOM adaptativo para múltiplas tabelas de dissimilaridade.

Tabela 8. Thyroid: Matriz de confusão obtida pelo algoritmo batch SOM

Classe/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	0	10	8	0	0	25	61	45	1	0
2	16	0	3	0	0	10	6	0	0	0
3	0	3	0	7	5	0	1	1	4	9

Tabela 9. Thyroid: Matriz de confusão obtida pelo algoritmo batch SOM adaptativo

Classe/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	0	33	22	0	0	29	38	24	4	0
2	34	1	0	0	0	0	0	0	0	0
3	0	1	0	6	7	0	2	1	0	13

Os índices CR, F-*measure* e OERC obtiveram, respectivamente, os valores 0,37, 0,52 e 0,023 para o modelo SOM adaptativo, e os valores, respectivamente 0,41, 0,55 e 0,11 para o modelo SOM batch não-adaptativo. O algoritmo SOM não-adaptativo para múltiplas tabelas de dissimilaridade obteve melhores índices do que o modelo batch adaptativo, porém o modelo adaptativo obteve um valor menor para o erro global.

A tabela 10 mostra a matriz de pesos final obtida pelo algoritmo batch SOM adaptativo para múltiplas tabelas de dissimilaridade.

5.4. Base de dados Ecoli

A base de dados Ecoli contém dados de proteínas classificadas em 8 classes: (1) cp (citoplasma), com 143 exemplos; (2) im (membrana interna sem sequência de sinal), com 77 exemplos; (3) pp (perisplasm), com 52 exemplos; (4) imU (inner membrane, uncleavable signal sequence), com 35 exemplos; (5) om (outer membrane), com 20 exemplos; (6) omL (outer membrane lipoprotein), com 5 exemplos; (7) imL (inner membrane lipoprotein) com 2 exemplos e (8) imS (inner membrane, cleavable signal sequence), com 2 exemplos. Cada indivíduo é descrito por 6 atributos.

As tabelas 11 e 12 mostram a matriz de confusão obtidas pelos algoritmos batch SOM e batch SOM adaptativo para múltiplas tabelas de dissimilaridade, respectivamente.

Tabela 10. Thyroid: Matriz de pesos obtida pelo algoritmo batch SOM adaptativo

Matriz/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	0,05	0,37	0,21	0,14	**5,38**	0,28	0,12	0,14	**1,15**	0,91
2	0,16	0,17	0,29	**2,96**	**3,20**	0,24	0,24	0,64	0,91	**9,47**
3	0,05	0,20	**4,45**	**3,01**	**1,64**	0,35	0,38	**1,53**	**4,18**	**4,83**
4	**27,94**	**19,29**	**2,61**	**1,27**	0,86	**9,17**	**15,75**	**27,39**	0,24	0,06
5	**85,90**	**4,14**	**1,36**	0,61	0,04	**4,59**	**5,72**	0,26	0,94	0,38

Tabela 11. Ecoli: Matriz de confusão obtida pelo algoritmo batch SOM

Classe/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,1	1,2	1,3	1,4
1	47	18	19	0	0	25	27	6	1	0
2	0	1	0	2	25	0	0	0	0	7
3	0	0	0	0	1	0	0	0	1	0
4	0	0	0	0	1	0	0	1	0	0
5	5	0	0	32	24	1	0	0	1	14
6	1	1	0	0	1	0	0	13	35	1
7	0	0	0	0	0	0	0	4	1	0
8	0	0	0	0	0	0	0	8	11	1

Tabela 12. Ecoli: Matriz de confusão obtida pelo algoritmo batch SOM adaptativo

Classe/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,2	1,3	1,4
1	0	0	3	36	57	0	0	3	44
2	10	6	1	0	0	17	0	1	0
3	0	1	0	0	0	0	0	1	0
4	1	0	1	0	0	0	0	0	0
5	23	17	0	8	0	28	0	1	0
6	0	1	5	2	2	0	13	29	0
7	0	0	5	0	0	0	0	0	0
8	0	0	1	0	0	0	11	8	0

Os índices CR, F-*measure* e OERC obtiveram, respectivamente, os valores 0,43, 0,52 e 0,24 para o modelo SOM adaptativo, e os valores, respectivamente 0,37, 0,52 e 0,24 para o modelo SOM batch não-adaptativo. O algoritmo SOM adaptativo para múltiplas tabelas de dissimilaridade obteve índices semelhantes ao modelo batch não-adaptativo, exceto para o índice CR, onde o modelo adaptativo obteve um melhor resultado.

A tabela 13 mostra a matriz de pesos final obtida pelo algoritmo batch SOM adaptativo para múltiplas tabelas de dissimilaridade.

Tabela 13. Ecoli: Matriz de pesos obtida pelo algoritmo batch SOM adaptativo

Matriz/Grupo	0,0	0,1	0,2	0,3	0,4	1,0	1,2	1,3	1,4
1	0,36	0,49	0,96	0,86	**1,26**	0,5	**2,44**	**2,55**	0,48
2	0,5	0,75	0,88	**1,46**	0,9	0,65	0,83	0,53	0,72
3	0,53	0,54	0,78	**1,12**	0,63	0,64	0,25	0,38	0,46
4	**2,55**	**2,27**	**2,09**	0,78	**1,2**	**2,15**	**1,93**	**1,53**	**2,28**
5	**3,96**	**2,16**	0,71	0,89	**1,15**	**2,2**	0,98	**1,25**	**2,69**

6. Conclusão

Este trabalho apresentou um modelo de agrupamento baseado no algoritmo SOM com treinamento batch. Este modelo realiza o agrupamento de objetos levando em consideração seus dados relacionais descritos por uma tabela de dissimilaridades. Como saída, o algoritmo produz uma partição dos dados de entrada, os protótipos de cada

grupo e pesos adaptativos que indicam a relevância de cada tabela de dissimilaridade na formação dos agrupamentos, otimizando um critério de adequação que mede o ajuste entre os grupos e seus protótipos. Tais pesos mudam a cada iteração e são diferentes para cada grupo. Experimentos utilizando bases de dados reais foram realizados para mostrar a utilidade do modelo proposto comparando-o com o modelo apresentado por [Golli et al. 2004]. Na maioria dos experimentos os resultados foram satisfatórios, mostrando a relevância do trabalho proposto, como também, seu desempenho utilizando diferentes bases de dados.

Agradecimentos

Gostaríamos de agradecer pelo suporte financeiro das agências CNPq e FACEPE, que tornaram possível o desenvolvimento e crescimento deste trabalho.

Referências

Badran, F., Yacoub, M., and Thiria, S. (2005). Self-organizing maps and unsupervised classification. In *Neural networks: methodology and applications*, pages 379–442. Springer-Verlag.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. (1984). *Classification and Regression Trees*. Chapman and Hall/CRC.

Conan-Guez, B., Rossi, F., and Golli, A. E. (2006). Fast algorithm and implementation of dissimilarity self-organizing maps. *Neural Networks*, 19:855–863.

Frank, A. and Asuncion, A. (2010). Uci machine learning repository.

Golli, A., Conan-Guez, B., and Rossi, F. (2004). A self-organizing map for dissimilarity data. In *Classification, Clustering, and Data Mining Applications*, pages 61–68. Springer Berlin Heidelberg.

Gopalakrishnan, K., Khaitan, S., and Manik, A. (2008). Enhanced clustering analysis and visualization using kohonen's self-organizing feature map networks. *International Journal of Computational Intelligence*, 4:64–71.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2:193–218.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM Comput. Surv.*, 31:264–323.

Kohonen, T. (1990). The self-organizing maps. *Proceedings of the IEEE*, 78:1464–1480.

Lechevallier, Y., de Carvalho, F. A. T., Despeyroux, T., and de Melo, F. M. (2010). Clustering of multiple dissimilarity data tables for documents categorization. *19th International Conference on Computational Statistics - COMPSTAT 2010*, pages 1263–1270.

Murtagh, F. and Hernández-Pajares, M. (1995). The kohonen self-organizing map method: An assessment. *Journal of Classification*, 12:165–190.

van Rijsbergen, C. J. (1979). *Information Retrieval*. London: Butterworths.

UMA ESTRATÉGIA HÍBRIDA PARA O PROBLEMA DE CLASSIFICAÇÃO MULTIRRÓTULO

Tiago Amador Coelho[1], **Ahmed Ali Abdalla Esmin**[2], **Wagner Meira Júnior**[1]

[1]Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte, MG – Brasil

[2]Universidade Federal de Lavras (UFLA)
Lavras, MG – Brasil

{tiago,meira}@dcc.ufmg.br, ahmed@dcc.ufla.br

Abstract. *Multi-label classification learning first arose in the context of text categorization, where each document may belong to several classes simultaneously and has attracted significant attention lately, as a consequence of both the challenge it represents and its relevance in terms of application scenarios. In this paper, we propose a new hybrid approach, Multi Label K-Nearest Michigan Particle Swarm Optimization (ML-KMPSO). It is based on two strategies. The first strategy is the Michigan Particle Swarm Optimization (MPSO), which breaks the multi-label classification task into several binary classification problems, but it does not take into account the correlations among the various classes. The second strategy is ML-KNN, which is complementary and takes into account the correlations among classes. We evaluated the performance of ML-KMPSO using two real-world datasets and the results show that our proposal matches or outperforms well-established multi-label classification learning algorithms.*

Resumo. *A aprendizagem de classificadores multirrótulo se originou na categorização de textos, onde cada documento podia pertencer a mais de uma classe simultaneamente e tem atraído muita atenção como consequência tanto do desafio de pesquisa que representa quanto da relevância dos seus cenários de aplicação. Neste artigo, propomos uma nova abordagem híbrida, Multi Label k-Nearest Michigan Particle Swarm Optimization (ML-KMPSO), que é baseada em duas estratégias. A primeira é a divisão do problema multirrótulo em diversos problemas binários, para tal foi utilizado o Michigan Particle Swarm Optimization (MPSO) para resolve-los, mais esta estratégia não leva em consideração as correlações existentes entre as classes. Já a segunda estratégia tem como objetivo considerar as correlações existentes entre as classes utilizando o Multi Label K-Nearest Neighbor (ML-KNN). Avaliamos o desempenho do ML-KMPSO utilizando duas bases de dados reais e os resultados mostram que a nossa proposta se iguala ou supera algoritmos estado-da-arte de classificação multirrótulo.*

1. Introdução

Aplicações reais como classificação semântica de cenas, classificação funcional de proteínas, categorização de textos, categorização de músicas, entre outras, vem reque-

rendo cada vez mais métodos que realizem a classificação multirrótulo, uma vez que cada instância a ser classificada nesses cenários é associada a mais de uma classe.

No problema de classificação unirrótulo, que é o mais tradicional, cada instância é associada a um sub-conjunto de rótulos unitário Y de um conjunto de rótulos $\mathcal{Y}$, $|\mathcal{Y}| > 1$. Quando $|\mathcal{Y}| = 2$, o problema é chamado de problema de classificação binária, e se $|\mathcal{Y}| > 2$ o problema é chamado de problema de classificação multiclasse. Entretanto há diversos problemas nos quais as instâncias podem estar associadas a mais de uma classe simultaneamente, $|Y| \geq 2$, em que as classes não são disjuntas, esses problemas são conhecidos como problemas de classificação multirrótulo (MLC). Essa generalização do problema de classificação o torna mais difícil de se resolver.

Uma estratégia intuitiva de se resolver esse problema é a decomposição do problema em múltiplos e independentes problemas de classificação binários, o que não considera as correlações que podem existir entre as diferentes classes, tornando-o ineficiente. Uma outra forma de se resolver esse problema é adaptar um algoritmo de classificação unirrótulo para realizar a classificação multirrótulo, como categorização de texto multirrótulo [Schapire and Singer 2000], árvores de decisão multirrótulo [Quinlan 1993], métodos de kernel multirrótulo [Elisseeff and Weston 2001], redes neurais multirrótulos [Zhang and Zhou 2006] e ML-KNN [Zhang and Zhou 2005]. Nesses casos, os resultados obtidos ainda não são muito bons e por isso essa área continua em aberto e atraindo muita atenção de pesquisadores.

Neste trabalho é proposta uma nova estratégia híbrida, ML-KMPSO, que se baseia em duas estratégias existentes. A primeira é o Michigan Particle Swarm Optimization (MPSO), que divide o problema de classificação multirrótulo em diversos problemas de classificação binário e os resolve, mas não considera as possíveis correlações existentes entre as classes. A segunda estratégia é o Multi-Label K-Nearest Neighbor (ML-KNN), que é complementar à primeira e considera a correlação existente entre as classes.

Na prática, o ML-KMPSO funciona da seguinte forma. Inicialmente, para cada instância de treinamento, identifica as instâncias vizinhas mais próximas do conjunto de treinamento e calcula a sua probabilidade *a priori*. Então, um modelo de classificação é gerado pelo MPSO e um conjunto de partículas especializadas em classificação é identificado. Baseado nas informações estatísticas obtidas do conjunto de rótulos dos vizinhos das instâncias, incluindo as partículas de classificação geradas pelo MPSO, é utilizado então o princípio do *maximun a posteriori* (MAP) [Zhang and Zhou 2007] para se determinar o conjunto de rótulos das instâncias de teste (resultado da classificação).

O trabalho está organizado da seguinte forma. A Seção 2 discute o problema de classificação multirrótulo. Na Seção 3 é feito um resumo do *Particle Swarm Optimization* ou Método de Enxame de Partículas (PSO). A Seção 4 apresenta o PSO e o MPSO (*Michigan Particle Swarm Optimization*) para o problema de classificação. Na Seção 5 são apresentados e discutidos os experimentos. Finalmente na Seção 6 são colocadas as conclusões e os trabalhos futuros.

2. Classificação Multirrótulo

Pesquisas em classificação multirrótulo foram inicialmente motivadas pelo desafio representado na categorização de textos, onde vários rótulos podem ser associados a um mesmo documento [Zhang and Zhou 2007].

No problema de classificação multirrótulo, cada instância está associada a um conjunto de rótulos Y, onde Y $\subseteq \mathcal{Y}$, sendo $\mathcal{Y}$ um conjunto de todos os rótulos do problema.

Em [Li et al. 2006], o problema de classificação multirrótulo é definido da seguinte forma: Sendo $\mathcal{X}$ um conjunto de treinamento, $\mathcal{Y} = \{1, 2, ..., k\}$ o conjunto de rótulos. Dado um conjunto de treinamento da forma $\langle x_i, Y_i \rangle, x_i \in \mathcal{X}, Y_i \in 2^{|\mathcal{Y}|}$, onde $2^{|\mathcal{Y}|}$ são todas as combinações possíveis em $\mathcal{Y}$. O objetivo é aproximar uma função $f(x)$, tal que $f(x)$ retorne valores de $2^{|\mathcal{Y}|}$ com o menor erro. A dificuldade de definir o erro em multirrótulo é que diversas combinações de rótulos são possíveis.

Na maioria dos casos, a abordagem multirrótulo induz uma ordenação dos possíveis rótulos de uma dada instância de acordo com $f(x, l_n)$. Então, formalmente, podemos definir $rank_f(x, l)$ como $rank_f$ do rótulo l para a instância x, e se $f(x, l_1) \leq f(x, l_2)$ então $rank_f(x, l_1) \leq rank_f(x, l_2)$.

Tanto [Trohidis et al. 2008] como [Zhang and Zhou 2007], em seus trabalhos, agrupam em duas categorias os métodos para resolver o problema multirrótulo:

- a primeira categoria decompõe o problema em múltiplos e independentes problemas de classificação binária ou unirrótulo, mas não leva em consideração as correlações existentes entre as classes;
- a segunda categoria é a realização de alterações nos algoritmos de classificação existentes para permitir a classificação multirrótulo.

Para ilustrar os diferentes métodos que são encontrados na literatura para o problema múltirrótulo, [de Carvalho and Freitas 2009] organizaram de maneira hierárquica todos esses métodos, como mostrado na Figura 1.

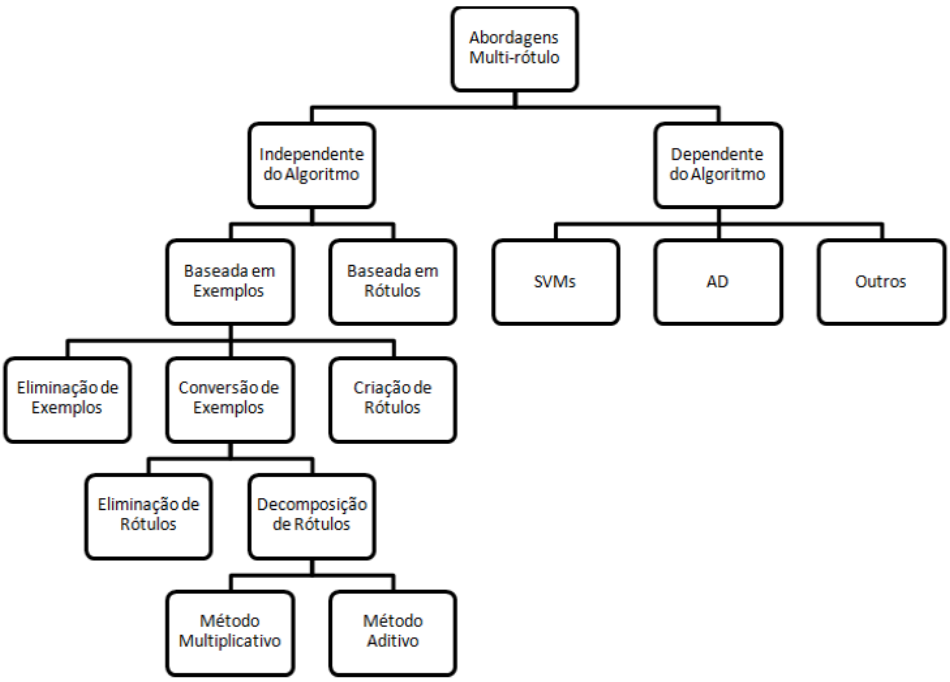


Figura 1. Técnicas para Classificação Multirrótulo (adaptado de [de Carvalho and Freitas 2009])

Um dos métodos estado da arte para classificação multirrótulo é o BoosTexter. Proposto por [Schapire and Singer 2000], o método mantém um conjunto de pesos para cada par exemplo-rótulo do conjunto de treinamento. Os pares que são difíceis de se prever corretamente têm os seus pesos incrementados e, para os pares facilmente previstos, os seus pesos são decrementados.

[Elisseeff and Weston 2001] propuseram o RANK-SVM, um método baseado em *kernels* para classificação, que maximiza a soma das margens de todas as categorias simultaneamente, ordenando as categorias relevantes de cada instância da base de treinamento e penalizando as irrelevantes.

No trabalho de [Zhang and Zhou 2007] foi proposto o método Multi Label K-Nearest Neighbors (ML-KNN), derivado do método KNN tradicional e diferenciado pelo uso de probabilidades *a priori* e *a posteriori*. Inicialmente o método identifica, para cada instância na base de treinamento, os seus k vizinhos mais próximos. Então, baseado na informação estatística *a priori* obtida do conjunto de rótulos destes vizinhos, é utilizado o princípio *Maximum a Posteriori* (MAP) para determinar o conjunto de rótulos da instância de teste.

Neste trabalho propomos uma nova abordagem híbrida para resolver o problema de classificação multirrótulo, que reúne a abordagem independente do algoritmo (MPSO) com a abordagem dependente do algoritmo (ML-KNN), como ilustrado na Figura 2.

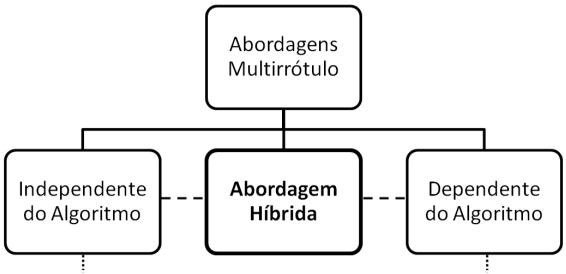


Figura 2. Nova organização dos métodos para resolver o problema multirrótulo

3. Método de Enxame de Partícula - PSO

O *Particle Swarm Optimization* (PSO) ou Método de Enxame de Partícula é um método de otimização baseado em comportamento social de um bando de pássaros, proposto por [Kennedy and Eberhart 1995]. O método PSO encontra a solução ótima do problema usando uma população de partículas, onde cada partícula é uma solução candidata do problema. Basicamente o PSO simula um bando de pássaros em um espaço multidimensional ([Kennedy and Eberhart 1995, Eberhart and Kennedy 1995]). Podemos definir o PSO como:

- Cada partícula i tem as seguintes propriedades: Uma posição atual no espaço de busca, x_i, uma velocidade atual, v_i, e uma melhor posição individual, y_i.
- A melhor posição individual, y_i, corresponde à posição no espaço de busca onde a partícula i apresenta o menor erro determinado pela função f.

- A melhor posição global é ilustrada por $\breve{y}$, representando a posição que teve o menor erro dentre todas as y_i.

Durante cada iteração do algoritmo, cada partícula no enxame é atualizada conforme as equações 1 e 2. A atualização da velocidade é dada por:

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1 r_{1,j}(t)[y_{i,j}(t) - x_{i,j}(t)] + c_2 r_{2,j}(t)[\breve{y}_j(t) - x_{i,j}(t)] \quad (1)$$

onde c_1 and c_2 são duas constantes positivas, r_1 e r_2 são variáveis aleatórias dentro do intervalo [0,l], e w é o valor da inércia.

A atual posição da partícula é atualizada com a equação:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

A equação 1 consiste em três partes. A primeira parte é a velocidade atual da partícula, o que demonstra seu estado atual; a segunda parte é o termo de cognição, que expressa o "conhecimento ou aprendizado"da própria partícula; a terceira parte é chamada de fator social, e reflete as informações compartilhadas no enxame. Estas três partes em conjunto, determinam a habilidade da partícula no espaço de busca, e sob a influência delas, as partículas podem atingir uma posição melhor e mais eficaz.

As equações 3 e 4 definem como os melhores valores individual e global são atualizados no instante t, respectivamente. É assumido que o enxame é formado por s partículas (onde $i \in 1..s$).

$$y_i(t+1) = \begin{cases} y_i(t) & se \quad f(y_i(t)) \leq f(x_i(t+1)) \\ x_i(t+1) & se \quad f(y_i(t)) > f(x_i(t+1)) \end{cases} \quad (3)$$

$$\breve{y}(t) = min f(y), f(\breve{y}(t)) \quad y \in y_0(t), y_1(t), ..., y_s(t) \quad (4)$$

O algoritmo consiste na execução iterativa das equações apresentadas. O pseudocódigo do PSO é apresentado na Figura 3.

Na próxima seção será descrito a adaptação do PSO para o problema de classificação.

4. PSO para Problemas de Classificação

Recentemente, alguns trabalhos apresentaram uma adaptação do PSO para os problemas de classificação, sendo a maioria deles baseada em regras de classificação. Em

Inicializar as posições e velocidades das partículas aleatoriamente no espaço n-dimensional;
Encontrar o $pbest$ de cada partícula;
Encontrar o $gbest$;
para cada *partícula* $i \in [1..s]$ **faça**
  se x_i *melhor que* $pbest_i(t)$ **então**
    $pbest_i(t) = x_i$
  fim se
  se $pbest_i(t)$ *melhor que* $gbest(t)$ **então**
    $gbest(t) = pbest_i(t)$
  fim se
  Atualiza a velocidade (v_i) da partícula i
  Atualiza a posição (x_i) da partícula i
fim para cada

Figura 3. Pseudo-código do PSO

[Sousa et al. 2004], o PSO é usado para extrair regras de indução para classificar dados, sendo o algoritmo do PSO executado diversas vezes, gerando em cada execução uma regra e utilizando apenas os padrões não classificados em suas próximas iterações. Em [Wang et al. 2006] e [Wang et al. 2007], o PSO é utilizado a extração de regras em problemas de classificação discretos. Já em [Esmin 2007] o PSO foi usado para extrair regras *fuzzy*.

Na estratégia padrão do PSO, a solução em potencial é codificada em apenas uma partícula, sendo que toda a informação a ser codificada é o conjunto de protótipos e as suas classes. Há um outro método, chamado de *Michigan Particle Swarm Optimization* (MPSO) que utiliza a abordagem *Michigan*. Os resultados encontrados na literatura do método MPSO são bastante competitivos em comparação com outros métodos. Na abordagem MPSO, cada membro da população não codifica toda a solução para o problema, mas apenas parte dele. Todo o enxame é uma potencial solução do problema ([Cervantes et al. 2009b, Sousa et al. 2004])

O pseudo-código do método MPSO é mostrado na Figura 4 e descrito em detalhes em [Cervantes et al. 2009b, Sousa et al. 2004]. A equação de posição foi modificada quando comparado ao método PSO padrão, no qual foi introduzido a força de repulsão e o uso de vizinhança dinâmica da partícula. Quando está se movendo, cada partícula seleciona uma outra que é chamada de ”não competidora”para ser o centro de atração, e uma segunda partícula para ser o centro de repulsão.

Durante cada iteração, as partículas ”competidoras”e ”não competidoras”são definidas dinamicamente e, para isso, considera as classes das partículas. Neste caso, as partículas ”não competidoras”são aquelas de classes diferentes e as ”competidoras”possuem a mesma classe. O MPSO utiliza o conceito de *fitness* local, sendo cada partícula é avaliada como “boa”se classifica corretamente os padrões próximos. O cálculo do *fitness* local não considera como o restante dos padrões são classificados ([Cervantes et al. 2009b]).

O MPSO é muito eficiente na geração de um enxame altamente especializado em reconhecimento de padrões e pode ser facilmente adaptado a diversas classes de problemas ([Cervantes et al. 2009a, Cervantes et al. 2009b]). Nestra trabalho, apresentamos

Carregue as instâncias de treino;
Inicializa o enxame;
Inserir N partículas para cada classe das instâncias de treino;
enquanto *(max iteração ou taxa de sucesso de 100%)* **faça**
    Calcular para cada classe, as partículas são competidoras e não competidoras ;
    para cada *partícula* **faça**
        Calcular o *Fitness* local;
        Calcular o fator de adaptação social;
        Encontrar a partícula de mesma classe e que esteja no conjunto de partículas competidoras (centro de repulsão);
        Encontrar a partícula de mesma classe e que esteja no conjunto de partículas não competidoras (centro de atração);
        Calcular a próxima posição da partícula baseada em sua velocidade anterior, sua melhor posição, centros de atração e repulsão;
    fim para cada
    Movimente as partículas;
    Atribuir classes às instâncias de treino, utilizando a partícula mais próxima;
    Avaliar a taxa de acerto da classificação feita pelo enxame;
    Se o enxame conseguiu um melhor resultado, armazena todas as posições das partículas como ”melhor enxame”;
fim enqto
Apagar do enxame de partículas aquelas partículas que, ao serem retiradas, não afetam a taxa de acerto do enxame.;

Figura 4. Pseudo Algoritmo do MPSO

uma adaptação do MPSO para resolver o problema de classificação multirrótulo chamado de ML-KMPSO. Como o próprio nome indica, o ML-KMPSO é derivado do ML-KNN e do MPSO. Na Figura 5 é demonstrado o pseudo código do algoritmo.

1. Calcula a probabilidade a priori de todos os rótulos do conjunto de treinamento
2. Encontra o enxame de partículas usando o MPSO (Figura 4)
3. Agrega o enxame de partículas aos dados de treinamento
4. Calcula a probabilidade a posteriori de todos os rótulos usando os k vizinhos
5. Determina os rótulos de cada instância de teste utilizando o MAP

Figura 5. Pseudo código do ML-KMPSO

Como é descrito na Figura 5, inicialmente é calculada a probabilidade *a priori* dos rótulos do conjunto de treinamento, então é gerado o modelo de classificação usando o MPSO (passo 2) e identificado um conjunto de partículas especializadas em classificação (o enxame). Note que no passo 5 do PSO (Figura 4), partículas que não são utilizadas são marcadas para serem retiradas da solução (enxame), iniciando por aquela que detém o pior *fitness*. Após a verificação de que esta ação não irá interferir na qualidade da classificação do enxame, a partícula é então retirada. Finalmente, utilizando as informações estatísticas obtidas dos rótulos do conjunto de vizinhos, incluindo as partículas resultantes do MPSO, é aplicado o princípio do *Maximum a Posteriori* (MAP) para determinar os rótulos que

serão aplicados à instância de teste.

Na próxima seção avaliaremos o desempenho experimental do ML-KMPSO usando bases de dados reais.

5. Experimentos

Nesta seção avaliaremos o desempenho do ML-KMPSO usando duas diferentes bases de dados: a base *Yeast* e a base *Scene*. Os resultados obtidos serão comparados com os dos métodos ML-KNN ([Zhang and Zhou 2005, Zhang and Zhou 2007]), BoosTexter ([Schapire and Singer 2000]) e RankSVM ([Elisseeff and Weston 2001]).

5.1. Base de Dados

Como mencionado, para experimentos foram utilizadas duas bases de dados reais multirrótulos: a base *Yeast*([Elisseeff and Weston 2005]) e a base *Scene* ([Boutell et al. 2004]).

Yeast é uma base de dados multirrótulos biológica obtida através de estudos realizados sobre a levedura *Saccharomyces cerevisiae*, um dos organismos mais estudados da literatura. A base é descrita como um conjunto de micro vetores que expressam as características e os perfis filogenéticos de 2417 genes, onde cada gene está associado a um conjunto de 14 classes funcionais (Figura 6). [Elisseeff and Weston 2001] realizaram o pre-processamento da base, a fim de tornar fácil o seu uso e somente as classes funcionais conhecidas serão usadas.

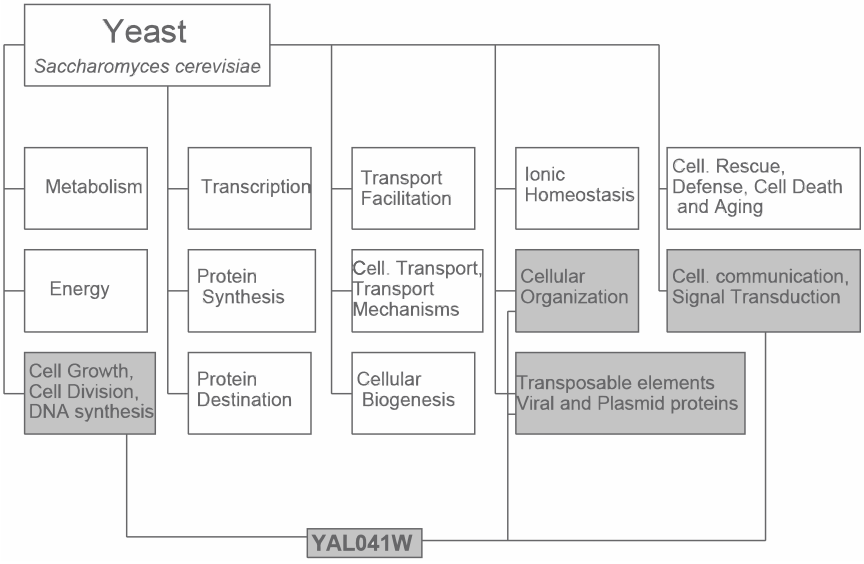


Figura 6. Classes Funcionais dos Genes da Levedura *Saccharomyces cerevisiae* [Elisseeff and Weston 2005]

A base *Scene* contém, no total, 2000 imagens naturais, todas elas são classificadas manualmente e representadas por um vetor de 294 dimensões e 5 possíveis rótulos, cada imagem possuindo em média 1.24 rótulos. O objetivo do classificador nesta base é que, após o treinamento, possa classificar automaticamente imagens que ainda serão apresentadas a ele.

A Tabela 1 mostra algumas informações estatísticas das duas bases que foram utilizadas nos experimentos.

Tabela 1. Informações Sobre as Bases Multirrótulos

Bases	Instâncias	Atributos	Rótulos	Cardinalidade	Densidade
Yeast	2417	103	14	4.327	0.302
Scene	2000	294	5	1.24	0.248

5.2. Métricas de Avaliação

Para avaliar o desempenho dos métodos de aprendizagem foram escolhidas as métricas comumente utilizadas na literatura nos problemas de classificação multirrótulo e que foram propostas por [Schapire and Singer 2000].

Hamming Loss: avalia quantas vezes o par exemplo-rótulo é erroneamente classificado, ou seja, um rótulo não pertence ao exemplo e foi previsto, ou um rótulo que pertença ao exemplo e não foi previsto. Nesta métrica, quanto menor o valor do *Hamming Loss* melhor é a acurácia do classificador.

$$HammingLoss_S(h) = \frac{1}{P}\sum_{i=1}^{P}|h(x_i)\triangle Y_i| \tag{5}$$

onde $\triangle$ representa a diferença simétrica entre os conjuntos dos rótulos previstos $(h(x_i))$ e os rótulos corretos (Y_i).

One-error: avalia quantas vezes o melhor rótulo não pertence ao conjunto de possíveis rótulos.

$$One - error_S(f) = \frac{1}{P}\sum_{i=1}^{P}\left[[\texttt{arg}_{y\in\mathcal{Y}}\texttt{max}\ \ f(x_i,y)] \notin Y_i\right] \tag{6}$$

onde $i \in \mathcal{Y}$. O termo $[\texttt{arg}_{y\in\mathcal{Y}}\texttt{max}\ \ f(x_i,y)]$ representa o valor máximo da função $f()$. Quanto menor o valor do $One - Error_S(f)$ melhor é o desempenho.

Coverage: é definido como a distância em termos de rótulos no seu *ranking* para cobrir todos os possíveis rótulos de uma instância x.

$$Coverage_S(f) = \frac{1}{P}\sum_{i=1}^{P}\texttt{max}_{y\in Y}rank_f(x_i,y) - 1 \tag{7}$$

onde $i \in \mathcal{Y}$. Quanto menor o valor do *Coverage*, melhor é o desempenho do classificador.

Ranking Loss: avalia a quantidade média dos rótulos que estão inversamente ordenados para a instância. Quanto menor o valor do *Ranking Loss* melhor é o desempenho do classificador.

$$RankLoss_S(f) = \frac{1}{P}\sum_{i=1}^{P}\frac{1}{|Y_i||\overline{Y}_i|}|(y_1,y_2)|f(x_i,y_1) \not< f(x_i,y_2),(Y_1,y_2) \in Y_i \times \overline{Y}_i| \tag{8}$$

Average Precision: avalia a proporção média de rótulos que ocupam, no *ranking*, uma posição superior a um dado rótulo, e que pertencem ao conjunto de rótulos desejados ([de Carvalho and Freitas 2009]). Quanto maior o valor desta métrica melhor é o desempenho do classificador.

$$AveragePrecision_S(f) = \frac{1}{P}\sum_{i=1}^{P}\frac{1}{|Y_i|}\times\sum_{y\in Y_i}\frac{|\{y'|rank_f(x_i,y') \not< rank_f(x_i,y'), y' \in Y_i\}|}{rank_f(x_i,y)} \tag{9}$$

5.3. Análise dos Resultados

A fim de garantir resultados mais confiáveis, todos os experimentos foram realizados utilizando o *ten fold cross validation method*. Neste método de avaliação experimental, o conjunto de dados é dividido em 10 *fold*, e são realizados 10 experimentos, onde cada experimento usa um *fold* diferente para teste e os restantes nove *fold* para treinamento.

A Tabela 2 apresenta a comparação do ML-KMPSO com os algoritmos do estado da arte para o problema de classificação multirrótulo utilizando a base *Yeast*. Para cada métrica de avaliação, a seta para baixo indica que quanto menor, melhor, enquanto a seta para cima indica que quanto maior, melhor.

Tabela 2. Resultados na Base *Yeast*

Métricas	Algoritmos			
	ML-KNN	BoosTexter	Rank-SVM	ML-KMPSO
Hamming Loss↓	**0.194±0.010**	0.220±0.011	0.207±0.013	0.198±0.008
One-Error↓	0.230±0.030	0.278±0.034	0.243±0.039	**0.225±0.045**
Coverage↓	**6.275±0.240**	6.550±0.243	7.090±0.503	6.355±0.232
Ranking Loss↓	**0.167±0.016**	0.186±0.015	0.195±0.021	0.172±0.009
Average Precision↑	**0.765±0.021**	0.737±0.022	0.749±0.026	0.762±0.012

Como visto nesta tabela, o ML-KMPSO apresentou melhor desempenho em todas as cinco métricas em relação ao BoosTexter e SVM-Rank. Comparando com o ML-KNN, foi melhor apenas na métrica *One-Error*, e similar nas demais métricas. O ranking dos melhores métodos para cada métrica é mostrada na Tabela 3.

Tabela 3. Relação de Performance dos Métodos de Classificação Multirrótulo para a Base *Yeast*

Métricas	Algoritmos
HL	ML-KNN > ML-KMPSO > Rank-SVM > BoosTexter
OE	ML-KMPSO > ML-KNN > Rank-SVM > BoosTexter
Co	ML-KNN > ML-KMPSO > BoosTexter > Rank-SVM
RL	ML-KNN > ML-KMPSO > BoosTexter > Rank-SVM
AP	ML-KNN > ML-KMPSO > Rank-SVM > BoosTexter

Os resultados para a base Scene estão na Tabela 4. O ML-KMPSO se mostrou muito promissor, sendo o melhor nas métrias *Hamming Loss* e *Coverage metrics*, o segundo em *One-Error* e terceiro no *Ranking Loss* e *Average Precision*. Os bons resultados associados ao *Coverage* (Co) e ao *Hamming Loss* (HL) são consequência da baixa cardinalidade e densidade da base, que favorece ao MPSO, que resulta em classificadores especializados e é capaz de prever com precisão um grande número de tuplas (instâncias x_i - rótulos Y_i). Por outro lado, o Rank-SVM obteve os piores resultados.

Tabela 4. Resultados na Base *Image*

Métricas	Algoritmos			
	ML-KNN	BoosTexter	Rank-SVM	ML-KMPSO
Hamming Loss↓	0.169±0.016	0.179±0.015	0.253±0.055	**0.153±0.010**
One-Error↓	**0.300±0.036**	0.311±0.041	0.491±0.135	0.311±0.036
Coverage↓	0.939±0.100	0.939±0.092	1.382±0.381	**0.872±0.274**
Ranking Loss↓	**0.168±0.024**	0.168±0.020	0.278±0.096	0.173±0.013
Average Precision↑	**0.803±0.027**	0.798±0.024	0.682±0.093	0.796±0.018

Tabela 5. Relação de Performance dos Métodos de Classificação Multirrótulo para a Base *Image*

Métricas	Algoritmos
HL	ML-KMPSO > ML-KNN > BoosTexter > Rank-SVM
OE	ML-KNN > ML-KMPSO > BoosTexter > Rank-SVM
Co	ML-KMPSO > ML-KNN > BoosTexter > Rank-SVM
RL	ML-KNN > BoosTexter > ML-KMPSO > Rank-SVM
AP	ML-KNN > BoosTexter > ML-KMPSO > Rank-SVM

Os resultados dos experimentos mostram que o algoritmo proposto supera outras abordagens como Rank-SVM, BoosTexter e empata com o ML-KNN. Acreditamos que ajustes nos parâmetros do MPSO dentro da estratégia proposta, podem resultar em resultados melhores que o ML-KNN.

6. Conclusão

O trabalho apresentou uma nova abordagem híbrida para resolver o problema de classificação multirrótulo chamado ML-KMPSO, que é uma adaptação do Michigan Particle Swarm Optimization para o problema de classificação multirrótulo. O ML-KMPSO foi comparado com outros métodos do estado da arte para esse problema (ML-KNN, RankSVM e BoosTexter) e os resultados dos experimentos, no qual foram utilizadas bases de dados reais, mostrou que o novo método superou ou ao menos se igualou aos métodos existentes, demonstrando que estratégias que representam um compromisso entre soluções globais e locais são interessantes.

Como trabalhos futuros tem-se como meta a realização de novos experimentos ajustando parâmetros e utilizando novas bases de dados multirrótulos (como a base Yahoo de categorização de páginas web) para a completa avaliação da eficiência do ML-KMPSO.

7. Agradecimentos

Os autores agradecem ao CNPq, à FAPEMIG pelo auxílio dado a esta pesquisa.

Referências

Boutell, M. R., Luo, J., Shen, X., and Brown, C. M. (2004). Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771.

Cervantes, A., Galván, I. M., and Isasi, P. (2009a). Ampso: a new particle swarm method for nearest neighborhood classification. *Trans. Sys. Man Cyber. Part B*, 39:1082–1091.

Cervantes, A., Galván, I. M., and Viñuela, P. I. (2009b). Michigan particle swarm optimization for prototype reduction in classification problems. *New Generation Comput.*, 27(3):239–257.

de Carvalho, A. and Freitas, A. (2009). *A tutorial on multi-label classification techniques*, volume Foundations of Computational Intelligence Vol. 5 of *Studies in Computational Intelligence 205*, pages 177–195. Springer.

Eberhart, R. and Kennedy, J. (1995). A new optimizer using particle swarm theory. pages 39–43.

Elisseeff, A. and Weston, J. (2001). A kernel method for multi-labelled classification. In *In Advances in Neural Information Processing Systems 14*, pages 681–687. MIT Press.

Elisseeff, A. and Weston, J. (2005). A kernel method for multi-labelled classification. In *Annual ACM Conference on Research and Development in Information Retrieval*, pages 274–281.

Esmin, A. A. A. (2007). Generating fuzzy rules from examples using the particle swarm optimization algorithm. In *HIS*, pages 340–343.

Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948 vol.4.

Li, T., Zhang, C., and Zhu, S. (2006). Empirical studies on multi-label classification. In *ICTAI '06: Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence*, pages 86–92, Washington, DC, USA. IEEE Computer Society.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Schapire, R. E. and Singer, Y. (2000). BoosTexter: A Boosting-based System for Text Categorization. *Machine Learning*, 39(2/3):135–168.

Sousa, T., Silva, A., and Neves, A. (2004). Particle swarm based data mining algorithms for classification tasks. *Parallel Comput.*, 30(5-6):767–783.

Trohidis, K., Tsoumakas, G., Kalliris, G., and Vlahavas, I. (2008). Multilabel classification of music into emotions. In *Proc. 9th International Conference on Music Information Retrieval (ISMIR 2008), Philadelphia, PA, USA, 2008*.

Wang, Z., Sun, X., and Zhang, D. (2006). Classification rule mining based on particle swarm optimization. In *RSKT*, pages 436–441.

Wang, Z., Sun, X., and Zhang, D. (2007). A pso-based classification rule mining algorithm. In Huang, D.-S., Heutte, L., and Loog, M., editors, *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, volume 4682 of *Lecture Notes in Computer Science*, pages 377–384. Springer Berlin / Heidelberg.

Zhang, M.-L. and Zhou, Z.-H. (2005). A k-nearest neighbor based algorithm for multi-label classification. In *GrC*, pages 718–721.

Zhang, M.-L. and Zhou, Z.-H. (2006). Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 18:1338–1351.

Zhang, M.-L. and Zhou, Z.-H. (2007). ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40(7):2038–2048.

Uma Nova Abordagem de Avaliação de Alinhamentos de Ontologias baseada em Consultas

Esdras Lins Bispo Jr.[1], **Renata Wassermann**[1]

[1]Instituto de Matemática e Estatística (IME) – Universidade de São Paulo (USP)
Rua do Matão, 1010 - CEP 05508-090 - Sao Paulo –SP – Brasil

{bispojr,renata}@ime.usp.br

Abstract. *This work presents a new approach to evaluate ontology alignments. In contrast to measures based on alignment, this approach proposes a measure which uses the same queries usually processed in original ontologies to assess the quality of proposed alignment. By means of a motivating example, we show some satisfactory results of our approach with regard to standard precision and recall, and relaxed precision and recall.*

Resumo. *Este trabalho propõe uma nova abordagem de avaliar alinhamentos de ontologias. Em contraponto às métricas baseadas em alinhamento, a nossa abordagem apresenta uma métrica na qual utiliza as próprias consultas normalmente já realizadas nas ontologias originais para julgar a qualidade do alinhamento proposto. Através de um exemplo de motivação, apresentamos alguns resultados satisfatórios de nossa abordagem em relação às medidas de precisão e cobertura padrão e relaxadas.*

1. Introdução

O emparelhamento de ontologias é um problema importante para o qual vários algoritmos (Anchor-PROMPT [Noy and Musen 2001], Lily [Wang and Xu 2009], RiMOM [Wang et al. 2010]) já foram propostos. Para um dado par de ontologias, estes algoritmos propõem um conjunto de correspondências entre entidades destas ontologias chamado alinhamento.

Este trabalho tem como proposta apresentar uma nova abordagem de avaliar os alinhamentos de ontologias. As abordagens usuais avaliam o alinhamento proposto utilizando uma métrica baseada em um alinhamento de referência. A nossa abordagem apresenta uma métrica na qual utilizamos as próprias consultas normalmente já realizadas nas ontologias originais para julgar a qualidade do alinhamento proposto.

O restante do trabalho está dividido como se segue. A Seção 2 faz uma breve apresentação de algumas definições importantes. A Seção 3 descreve as métricas baseadas em alinhamento e duas abordagens adotando esta perspectiva. A Seção 4 propõe uma métrica baseada em consultas como uma nova abordagem de avaliação de alinhamentos. A Seção 5 aplica a métrica baseada em consultas em contraste às duas outras métricas baseada em alinhamento para um exemplo dado. Por fim, a Seção 6 traz algumas conclusões e considerações finais.

2. Fundamentos

Na área de emparelhamento de ontologias, muitas vezes são utilizadas palavras diferentes para se referir a conceitos similares e, às vezes, diferentes conceitos são referidos pela mesma palavra [Euzenat 2001, Kalfoglou and Schorlemmer 2003, Noy and Klein 2004]. Para evitar alguns mal-entendidos, serão explicitadas a seguir algumas definições importantes neste trabalho.

Definição 1 (Alinhamento, Correspondência)**.** *Dadas duas ontologias O e O', um alinhamento entre O e O' é um conjunto de correspondências (i.e. 4-uplas): $\langle e, e', r, n \rangle$ em que $e \in O$ e $e' \in O'$ são as duas entidades emparelhadas, r é uma relação entre e e e', e n expressa uma medida de confiança em $[0; 1]$ desta correspondência.*

Como exemplo de motivação, utilizaremos ao decorrer do trabalho duas ontologias em que cada uma foi projetada para uma universidade diferente (Figura 1). Estas ontologias representam informações sobre o seu quadro de funcionários.

Podemos apresentar a correspondência $\langle$ Docente, ProfessorDE, $\equiv$, .9 $\rangle$, por exemplo, que emparelha a entidade Docente com a entidade ProfessorDE das ontologias da Figura 1. Esta correspondência determina uma relação de equivalência ($\equiv$) entre as entidades, admitindo uma confiabilidade de 90%.

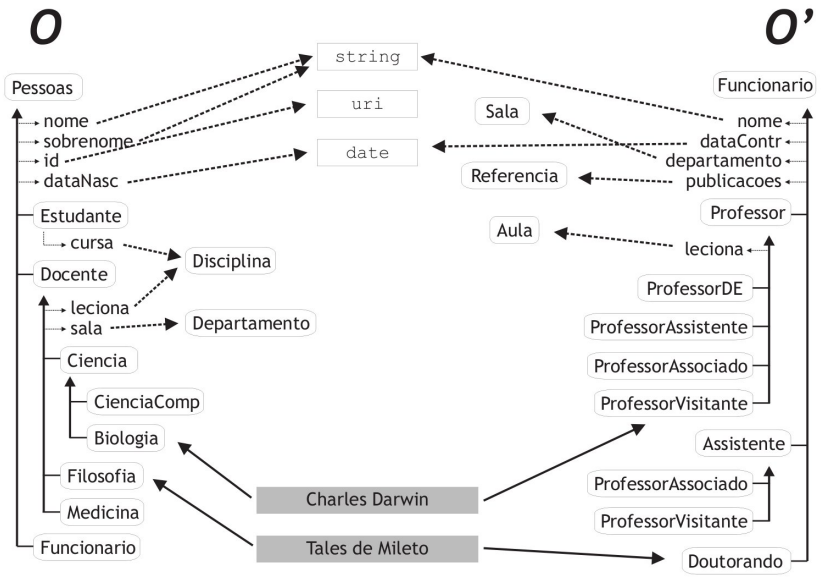


Figura 1. Duas ontologias que representam informações sobre o quadro de funcionários em duas universidades diferentes.

Existe uma forma mais concisa e elegante de representar uma correspondência. Ao invés de $\langle$ Docente, ProfessorDE, $\equiv$, .9 $\rangle$, podemos reescrever a 4-upla desta forma: Docente $\equiv_{.9}$ ProfessorDE. Na Tabela 1, apresentamos quatro alinhamentos para as ontologias da Figura 1 representando suas correspondências desta forma.

Definição 2 (Emparelhamento)**.** *Uma operação de emparelhamento determina o alinhamento A' para um par de ontologias O e O'. Existem alguns outros parâmetros que podem estender a definição de um processo de emparelhamento, a saber:*

Tabela 1. Alinhamentos A_1, A_2, A_3 e R e suas respectivas correspondências.

A_1	Departamento $\equiv$ departamento leciona $\equiv$ leciona sala $\equiv$ Sala	nome $\sqsubseteq$ nome Estudante $\sqsupseteq$ Doutorando
A_2	Docente $\sqsubseteq$ Professor Disciplina $\equiv$ Aula Pessoas $\equiv$ Funcionario	Departamento $\equiv$ Sala sala $\equiv$ departamento leciona $\equiv$ leciona
A_3	Docente $\sqsupseteq_{.9}$ ProfessorDE nome $\equiv$ nome leciona $\equiv$ leciona	Estudante $\equiv$ Doutorando Funcionario $\equiv$ Assistente
R	Estudante $\sqsupseteq$ Doutorando sala $\equiv$ departamento leciona $\equiv$ leciona	Docente $\equiv_{.9}$ ProfessorDE Disciplina $\equiv_{.7}$ Aula Departamento $\equiv$ Sala

(i) o uso de um alinhamento de entrada (A) o qual está para ser completado pelo processo;
(ii) os parâmetros do emparelhamento, p (e.g. pesos, limiares); e
(iii) recursos externos utilizados pelo processo, r (e.g. thesauri); ver Figura 2.

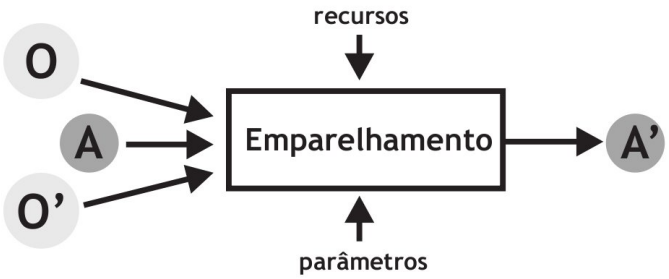


Figura 2. O processo de emparelhamento de ontologias.

3. Métricas baseadas em Alinhamento

As métricas baseadas em alinhamento têm como princípio básico confrontar um alinhamento proposto com um alinhamento de referência. Estas métricas surgiram da necessidade de avaliar a qualidade de alinhamentos produzidos por um algoritmo de emparelhamento de ontologias [Do et al. 2003].

A primeira tarefa é fazer o emparelhamento entre as ontologias manualmente. O alinhamento resultante obtido manualmente é utilizado como o padrão de referência para avaliar a qualidade do resultado determinado automaticamente pelo algoritmo. Normalmente é um especialista no domínio o qual as ontologias representam que executa esta tarefa. Este alinhamento é chamado de alinhamento de referência (R).

A próxima etapa é comparar as correspondências do alinhamento de referência em relação às correspondências do alinhamento proposto pelo sistema (A). Podemos classificar esta relação entre as correspondências destes dois alinhamentos em quatro conjuntos (ver Figura 3):

- **os falsos positivos** ($A \setminus R$): são correspondências desnecessárias em R, mas propostas em A;
- **os falsos negativos** ($R \setminus A$): são correspondências necessárias em R, mas não propostas em A;
- **os autênticos negativos** ($(A \cup R)^c$): são correspondências desnecessárias em R que não são propostas em A;
- **os autênticos positivos** ($A \cap R$): são correspondências necessárias em R que são propostas em A.

Baseadas na cardinalidade destes conjuntos, duas medidas são comumente utilizadas: precisão e cobertura. Estas medidas são originárias da área de recuperação da informação [van Rijsbergen 1975]. Elas vêm sido utilizadas de várias formas nas métricas baseadas em alinhamento. Apresentaremos a seguir, a precisão e cobertura padrão (Subseção 3.1) e a precisão e cobertura relaxada (Subseção 3.2).

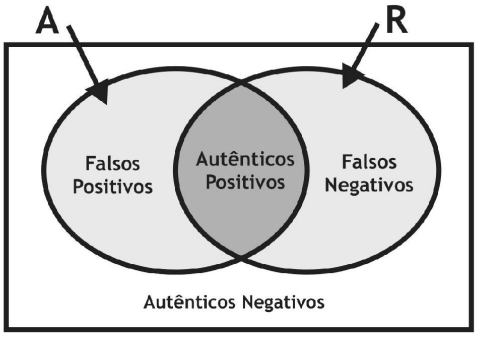


Figura 3. Sobreposição entre as correspondências do alinhamento de referência e as correspondências do alinhamento proposto.

3.1. Precisão e cobertura padrão

A precisão e a cobertura padrão são dadas a seguir:

Definição 3 (Precisão Padrão)**.** *Dado um alinhamento de referência R, a precisão de um alinhamento A é dada por*

$$P(A, R) = \frac{|A \cap R|}{|A|}$$

Definição 4 (Cobertura Padrão)**.** *Dado um alinhamento de referência R, a cobertura de um alinhamento A é dada por*

$$C(A, R) = \frac{|A \cap R|}{|R|}$$

A precisão padrão tem como objetivo indicar a presença dos falsos positivos no alinhamento proposto. Podemos reescrever a Definição 3 da seguinte forma:

$$P(A,R) = \frac{|A \cap R|}{|A \cap R| \cup |A \setminus R|}$$

Assim um alinhamento de precisão máxima (valor 1) é aquele que não apresenta falsos positivos, isto é, $|A \setminus R| = \emptyset$.

Já a cobertura padrão tem como objetivo indicar a presença dos falsos negativos no alinhamento proposto. Podemos reescrever a Definição 4 da seguinte forma:

$$C(A,R) = \frac{|A \cap R|}{|A \cap R| \cup |R \setminus A|}$$

Assim um alinhamento de cobertura máxima (valor 1) é aquele que não apresenta falsos negativos, isto é, $|R \setminus A| = \emptyset$.

Um alinhamento ideal seria aquele que não contivesse falsos positivos, nem falsos negativos. O alinhamento ideal seria composto por apenas autênticos positivos. Logo, é necessário levar em conta tanto a precisão quanto a cobertura na avaliação da qualidade do alinhamento proposto.

Por isso, utiliza-se também em conjunto com a precisão e a cobertura, uma medida combinada de ambas chamada medida-F:

Definição 5 (Medida-F)**.** *Dado um alinhamento de referência R e um número $\alpha \in [0;1]$, a medida-F de um alinhamento A é dada por*

$$M_\alpha(A,R) = \frac{P(A,R) \times C(A,R)}{(1-\alpha) \times P(A,R) + \alpha \times C(A,R)}$$

Se $\alpha = 1$, então a medida-F é igual à precisão, e se $\alpha = 0$, a medida-F é igual à cobertura. Quanto maior for o valor de α, maior importância é dada à precisão em relação à cobertura. Freqüentemente, utiliza-se o valor de $\alpha = 0,5$, isto é,

$$M_{0,5}(A,R) = \frac{2 \times P(A,R) \times C(A,R)}{P(A,R) + C(A,R)}$$

Como os autênticos negativos não são propostos em R e nem no alinhamento A, tornam-se relevantes para fins de avaliação apenas os outros três conjuntos anteriormente apresentados.

3.2. Precisão e cobertura relaxada

Existem algumas críticas às medidas de precisão e cobertura padrão. A primeira é que elas não conseguem discriminar sempre entre um alinhamento totalmente errado e um quase correto. Isto deve-se principalmente à função de sobreposição utilizada. Ao calcular a intersecção entre A e R, as medidas de precisão e cobertura padrão ignoram a proximidade entre as correspondências e aceitam somente correspondências idênticas.

A função de sobreposição adotada nas medidas de precisão e cobertura padrão trata da mesma forma estes dois pares de correspondências:

(1) (Estudante $\sqsupseteq$ Doutorando, Estudante $\equiv$ Doutorando)
(2) (Estudante $\sqsupseteq$ Doutorando, leciona $\equiv$ leciona)

embora seja óbvio que o par (1) tenha uma maior similaridade entre as correspondências do que o par (2).

Uma outra crítica é uma conseqüência da primeira. Estas métricas não conseguem estimar o esforço do usuário para adaptar o alinhamento. Após proposto, o alinhamento normalmente é refinado pelo usuário, corrigindo as eventuais inadequações que quase sempre aparecem em um alinhamento proposto automaticamente. Ao não informar qualquer indicativo da proximidade das correspondências, estas métricas podem avaliar dois alinhamentos com a mesma valoração, mesmo que um destes seja de maior qualidade do que o outro.

Com o intuito de resolver esta fragilidade, foram propostas medidas de precisão e cobertura relaxadas [Ehrig and Euzenat 2005]. A idéia é substituir a função de sobreposição utilizada na abordagem padrão por uma outra função que calcule a similaridade entre as correspondências. As medidas de precisão e a cobertura relaxadas são dadas a seguir:

Definição 6 (Precisão Relaxada)**.** *Dado um alinhamento de referência R e uma função de sobreposição ω_P, a precisão relaxada de um alinhamento A é dada por*

$$P_\omega(A, R) = \frac{\omega_P(A, R)}{|A|}$$

Definição 7 (Cobertura Relaxada)**.** *Dado um alinhamento de referência R e uma função de sobreposição ω_C, a cobertura relaxada de um alinhamento A é dada por*

$$C_\omega(A, R) = \frac{\omega_C(A, R)}{|R|}$$

A função de sobreposição ω atende às seguintes propriedades:

(1) $\forall A, B \quad \omega(A, B) \geq 0$ (positividade)
(2) $\forall A, B \quad \omega(A, B) \leq min(|A|, |B|)$ (maximalidade)
(3) $\forall A, B \quad \omega(A, B) \geq |A \cap B|$ (minimalidade)

O propósito destas propriedades é que as medidas de precisão e a cobertura relaxadas sejam uma extensão da abordagem padrão. As propriedades (1) e (2) garantem que elas variam no intervalo $[0; 1]$. A propriedade (3) garante que elas terão, na pior das hipóteses, a mesma valoração da abordagem padrão, ou contribuirão informando uma valoração maior.

4. Métricas baseadas em Consultas

Apresentamos até agora as métricas mais comumente utilizadas para fornecer a avaliação de alinhamentos de ontologias. Estas métricas são dependentes diretamente de um alinhamento de referência, isto é, são métricas baseadas em um alinhamento. Os cálculos de precisão e cobertura realizados por estas métricas são todos obtidos em relação às correspondências pertencentes a este alinhamento.

Estas métricas têm sido bastante utilizadas em competições como a *Ontology Alignment Evaluation Initiative* (OAEI) [Euzenat et al. 2010]. Os alinhamentos propostos pelos sistemas de emparelhamento participantes da competição são comparados aos respectivos alinhamentos de referência. Estes alinhamentos de referências são produzidos por especialistas no domínio de conhecimento no qual a ontologia representa.

Porém, na maioria dos casos, a realidade não funciona como nas competições. Na prática, não se julga um alinhamento proposto a partir de um alinhamento de referência. Pois se há um alinhamento de referência para as ontologias dadas, não haveria necessidade real de propor um outro alinhamento para elas. Para a lógica e estrutura de uma competição, é bem pertinente a idéia de um alinhamento de referência. Contudo isto não parece plausível para a solução de problemas de heterogeneidade em aplicações do mundo real.

É bem verdade que o espírito de uma competição como a OAEI é, na realidade, utilizar os alinhamentos de referência com o objetivo de avaliar não só os alinhamentos propostos, mas principalmente a robustez dos sistemas de emparelhamento participantes. Aquele sistema que propõe os melhores alinhamentos dentro da competição, terá maiores possibilidades de propor bons alinhamentos em aplicações do mundo real. A biblioteca de testes (*benchmark*[1]) a ser executada pelos sistemas no OAEI cobre uma gama bem variada de dificuldades críticas do processo de emparelhamento de ontologias.

O propósito de apresentar uma métrica baseada em consultas é oferecer uma abordagem alternativa ao uso de métricas baseadas em alinhamento.

4.1. Benefícios de uma métrica baseada em consultas

Existem vários benefícios do uso de uma métrica baseada em consultas. O primeiro deles é a possibilidade real do reuso da informação disponível acompanhada às ontologias existentes. Uma quantidade considerável de consultas normalmente já são realizadas para cada ontologia em seus respectivos contextos antes de serem alinhadas. Utilizar-se destas consultas com o propósito de avaliar a qualidade do alinhamento proposto é uma estratégia de abordagem coerente com procedimentos comuns, por exemplo, na área de construção de ontologias [Noy and McGuinness 2001].

Um outro benefício é que cada consulta realizada em uma das ontologias alinhadas está diretamente associada a um fragmento da ontologia. Os fragmentos associados a uma consulta revelam uma parte da ontologia bastante requisitada pela aplicação que a utiliza. Com isto, uma consulta identifica através de seu fragmento um conjunto de entidades como fortes candidatas a participarem das correspondências pertecentes ao alinhamento sugerido.

Temos por fim um problema que consiste em obter o alinhamento para ontologias de grande porte. Conforme apontado em [Gal and Shvaiko 2009], este é um dos grandes desafios para a área de emparelhamento de ontologias hoje. Para ontologias deste porte, até construir um alinhamento de referência torna-se uma tarefa bastante difícil (visto que o processo é manual, tedioso, bastante suscetível a erros e consome um tempo considerável [Noy and Musen 2000]). Além disso, nem sempre temos à disposição um especialista da área para se dedicar a uma tarefa como essa.

[1] `http://oaei.ontologymatching.org/2010/`

É bem razoável que um engenheiro do conhecimento escreva um conjunto de consultas em parceria com o especialista da área para serem realizadas sobre o conjunto alinhamento-ontologias. Parece ser bem menos custoso e coerente com os propósitos da aplicação.

4.2. Proposta de implementação

Sejam O e O' as duas ontologias alinhadas pelo alinhamento A. Sejam C_O e $C_{O'}$ os respectivos conjuntos de consultas usualmente realizadas para as ontologias O e O'. O Algoritmo 1 apresenta a solução para aplicações que exigem apenas um alinhamento que seja unidirecional e o Algoritmo 2 apresenta a solução para aplicações que exigem um alinhamento bidirecional. Os Algoritmos 4 e 5 apresentam as funções que auxiliam no cálculo da precisão e cobertura aplicado a instâncias. A Figura 4 ilustra com um esquema a proposta apresentada.

Algoritmo 1 - Métrica Baseada em Consultas (Unidirecional)

METRICACONSULTASUNIDIRECIONAL(O, O', C_O, A)
1. $precisao \leftarrow 0$
2. $cobertura \leftarrow 0$
3. $t_p \leftarrow 0$
4. $t_c \leftarrow 0$
5. $precisao, cobertura, t_p, t_c \leftarrow$ METRICACONSULTASAUX(O, O', C_O, A, $precisao$, $cobertura$, t_p, t_c)
6. $precisao \leftarrow precisao/t_p$
7. $cobertura \leftarrow cobertura/t_c$
8. **retorne** $precisao$, $cobertura$

Algoritmo 2 - Métrica Baseada em Consultas (Bidirecional)

METRICACONSULTASBIDIRECIONAL(O, O', C_O, $C_{O'}$, A)
1. $precisao \leftarrow 0$
2. $cobertura \leftarrow 0$
3. $t_p \leftarrow 0$
4. $t_c \leftarrow 0$
5. $precisao, cobertura, t_p, t_c \leftarrow$ METRICACONSULTASAUX(O, O', C_O, A, $precisao$, $cobertura$, t_p, t_c)
6. $precisao, cobertura, t_p, t_c \leftarrow$
7. METRICACONSULTASAUX(O', O, $C_{O'}$, INVERTERALINHAMENTO(A), $precisao$, $cobertura$, t_p, t_c)
8. $precisao \leftarrow precisao/t_p$
9. $cobertura \leftarrow cobertura/t_c$
10. **retorne** $precisao$, $cobertura$

Algoritmo 3 - Métrica Baseada em Consultas: Algoritmo Auxiliar

METRICACONSULTASAUX(O, O', C_O, A, $precisao$, $cobertura$, t_p, t_c)
1. **para** toda consulta c de C_O **faça**
2. $c_{trans} \leftarrow$ TRANSFORMARCONSULTA(c, A)
3. $resp_{trans} \leftarrow$ CONSULTARONTOLOGIA(c_{trans}, O')
4. $resp_{prop} \leftarrow$ TRADUZIRINSTANCIA($resp_{trans}$, INVERTERALINHAMENTO(A))
5. $resp_{ref} \leftarrow$ CONSULTARONTOLOGIA(c, O)
6. $interseccao, n_p \leftarrow$ PRECISAOINSTANCIA($resp_{prop}$,$resp_{ref}$)
7. $precisao \leftarrow precisao + interseccao$
8. $t_p \leftarrow t_p + n_p$
9. $interseccao, n_c \leftarrow$ COBERTURAINSTANCIA($resp_{prop}$,$resp_{ref}$)
10. $cobertura \leftarrow cobertura + interseccao$
11. $t_c \leftarrow t_c + n_c$
12. **fim para**
13. **retorne** $precisao$, $cobertura$, t_p, t_c

Existe uma diferença fundamental entre uma métrica baseada em alinhamento e uma métrica baseada em consultas. Para a primeira, o cálculo de precisão e cobertura é

realizado a partir das correspondências dos dois alinhamentos (A e R). Já para a segunda, o cálculo é realizado a partir das instâncias obtidas como respostas das consultas.

Para uma dada consulta de referência c_R, já são esperadas um conjunto de instâncias como respostas. Ao consultarmos c_R em uma ontologia, podemos sobrepor os resultados esperados com os resultados obtidos e construir um diagrama semelhante ao da Figura 3.

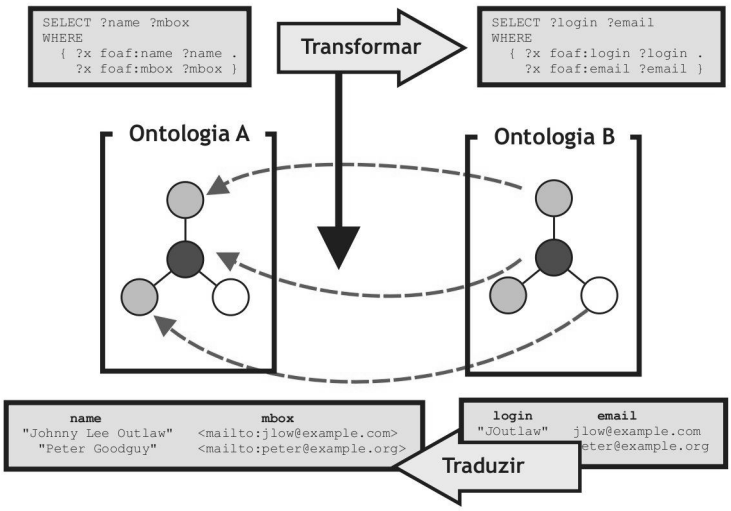


Figura 4. Esquema da abordagem de uma métrica baseada em consultas.

Algoritmo 4 - Cálculo da Precisão com Instâncias

PRECISAOINSTANCIA(A, R)

1. $interseccao \leftarrow 0$
2. $n \leftarrow 0$
3. **para** toda instância a de A **faça**
4. **para** toda instância r de R **faça**
5. **se** $a = r$ **então**
6. $interseccao \leftarrow interseccao + 1$
7. **break**
8. **fim se**
9. **fim para**
10. $n \leftarrow n + 1$
11. **fim para**
12. **retorne** $inteseccao, n$

Algoritmo 5 - Cálculo da Cobertura com Instâncias

COBERTURAINSTANCIA(A, R)

1. **retorne** PRECISAOINSTANCIA(R, A)

4.3. Dificuldades da abordagem

Para as funções utilizadas no Algoritmo 3, não há dificuldades na implementação de CONSULTARONTOLOGIA e INVERTERALINHAMENTO. A linguagem SPARQL[2], por exemplo, foi criada com o objetivo de realizar consultas em ontologias. E o processo de inversão de alinhamento é bem trivial. Para isto, invertem-se todas as correspondências do alinhamento (como Docente $\equiv_{.9}$ ProfessorDE para ProfessorDE $\equiv_{.9}$ Docente).

[2] `http://www.w3.org/TR/rdf-sparql-query/`

Porém, existem dificuldades reais para a implementação das funções TRANSFORMARCONSULTA e TRADUZIRINSTANCIA. Em [Euzenat et al. 2008], apresenta-se até uma proposta para a utilização de consultas em linguagem SPARQL com o intuito de resolver o problema de tradução de dados adicionando novas triplas com a declaração CONSTRUCT. Entretanto o problema de como criar dinamicamente tais consultas, explorando o alinhamento entre as ontologias ainda é uma problema em aberto e um campo de pesquisa relativamente pouco estudado dentro da comunidade da Web Semântica [Correndo et al. 2010].

5. Experimentos

Para os experimentos aqui realizados, utilizamos as duas ontologias apresentadas na Figura 1 e os quatros alinhamentos da Tabela 1. Para os experimentos realizados com a métrica baseada em consultas[3], populamos com indivíduos as duas ontologias já citadas, e criamos um conjunto de oito consultas para a ontologia O e traduzimos manualmente as consultas a partir dos alinhamentos A_1, A_2 e A_3 para serem realizadas na ontologia O'.

Na Tabela 2, apresentamos os resultados da avaliação dos alinhamentos A_1, A_2 e A_3 utilizando as medidas de precisão e cobertura padrão, e a medida-F correspondente. Percebe-se pelos resultados que A_2 é o melhor alinhamento dentre os três propostos. Porém não é possível diferenciar pelos resultados se o alinhamento A_1 é melhor do que o alinhamento A_3. Os valores de precisão e cobertura dos dois são idênticos.

Tabela 2. Resultados da avaliação dos alinhamentos A_1, A_2 e A_3 utilizando as medidas de precisão e cobertura padrão, e a medida-F correspondente.

Métrica	(A_1,R)	(A_2,R)	(A_3,R)
$P(A,R)$	$0,40$	$0,50$	$0,40$
$C(A,R)$	$0,33$	$0,50$	$0,33$
$M_{0.5}(A,R)$	$0,37$	$0,50$	$0,37$

Na Tabela 3, apresentamos os resultados da avaliação dos alinhamentos A_1, A_2 e A_3 utilizando as medidas de precisão e cobertura relaxadas, e a medida-F correspondente. O alinhamento A_2 é apontado ainda como o melhor alinhamento. Porém é possível agora diferenciarmos o alinhamento A_1 do A_3. A precisão e cobertura relaxada consegue apontar a diferença entre eles, indicando o A_3 como melhor do que A_1.

Tabela 3. Resultados da avaliação dos alinhamentos A_1, A_2 e A_3 utilizando as medidas de precisão e cobertura relaxadas, e a medida-F correspondente.

Métrica	(A_1,R)	(A_2,R)	(A_3,R)
$P_\omega(A,R)$	$0,40$	$0,69$	$0,58$
$C_\omega(A,R)$	$0,33$	$0,69$	$0,48$
$M_{0.5}(A,R)$	$0,37$	$0,69$	$0,53$

Na Tabela 4, apresentamos os resultados da avaliação dos alinhamentos A_1, A_2 e A_3 utilizando as medidas de precisão e cobertura baseadas em consultas, e a medida-F correspondente utilizando o Algoritmo 1. Embora os valores da precisão e cobertura

[3]As ontologias populadas e todas as consultas utilizadas estão disponíveis neste endereço: `http://www.ime.usp.br/bispojr/artigos/enia2011.tar.gz`

sejam diferentes dos apresentados na Tabela 3, é possível diferenciar os alinhamentos de forma a concluir que $A_1 < A_3 < A_2$ em termos de qualidade.

Tabela 4. Resultados da avaliação dos alinhamentos A_1, A_2 e A_3 utilizando as medidas de precisão e cobertura baseadas em consultas, e a medida-F correspondente.

Métrica	(A_1,C_R)	(A_2,C_R)	(A_3,C_R)
$P_C(A, C_R)$	$0,78$	$0,60$	$0,83$
$C_C(A, C_R)$	$0,39$	$0,80$	$0,50$
$M_{0.5}(A, C_R)$	$0,52$	$0,69$	$0,62$

Vale salientar que as consultas de referências criadas para obter os resultados da Tabela 4 foram escritas a partir do alinhamento de referência R, para podermos comparar as métricas a partir dos resultados aqui apresentados.

6. Conclusões

Inicialmente a avaliação de alinhamentos de ontologias era realizada utilizando as medidas de precisão e cobertura padrão. Porém, ultimamente, têm sido propostas as medidas de precisão e cobertura relaxadas com o intuito de discriminar de uma forma mais acurada os alinhamentos entre si. Esta diferenciação é importante para estimar o esforço do usuário para refinar o alinhamento após o processo.

Apresentamos uma métrica baseada em consultas como uma abordagem alternativa para a avaliação de alinhamentos. Esta métrica conseguiu diferenciar os alinhamentos propostos no exemplo de motivação da mesma forma que as medidas de precisão e cobertura relaxadas. Dentre os vários benefícios desta métrica, podemos destacar o reuso das consultas que já são usualmente realizadas nas ontologias emparelhadas.

Uma dificuldade significativa da métrica baseada em consultas consiste no problema de tradução de dados que está diretamente relacionado à expressividade das lógicas utilizadas. Uma possibilidade de abordar o problema seria resolvê-lo em lógicas menos expressivas e gradualmente incrementando para linguagens mais complexas.

Seria importante em trabalhos futuros propor uma maneira mais eficiente de comparação de métricas de avaliação de alinhamentos de ontologias. As comparações realizadas na área de emparelhamento de ontologias (como a utilizada neste trabalho) até então são carentes de uma validação estatística mais séria.

7. Agradecimentos

Os autores são muito gratos ao apoio fornecido pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) para a realização deste trabalho (136210/2009-0).

Referências

Correndo, G., Salvadores, M., Millard, I., Glaser, H., and Shadbolt, N. (2010). SPARQL query rewriting for implementing data integration over linked data. In *Proceedings of the 2010 EDBT/ICDT Workshops*, pages 4:1–4:11.

Do, H.-H., Melnik, S., and Rahm, E. (2003). Comparison of schema matching evaluations. In *Web, Web-Services, and Database Systems*, Lecture Notes in Computer Science, pages 221–237. Springer Berlin / Heidelberg.

Ehrig, M. and Euzenat, J. (2005). Relaxed precision and recall for ontology matching. In *Proceedings of the K-Cap Workshop on Integrating Ontologies*, pages 25–32, Banff, CA.

Euzenat, J. (2001). Towards a principled approach to semantic interoperability. In *Proceedings of the IJCAI Workshop on Ontologies and Information Sharing*, pages 19–25.

Euzenat, J., Ferrara, A., Meilicke, C., Nikolov, A., Pane, J., Scharffe, F., Shvaiko, P., Stuckenschimidt, H., Sváb-Zamazal, O., Svátek, V., and Trojahn, C. (2010). Results of the ontology alignment evaluation initiative 2010. In *Proceedings of the ISWC 2010 Workshop on Ontology Matching*, pages 85–117.

Euzenat, J., Polleres, A., and Scharffe, F. (2008). Processing ontology alignments with SPARQL. In *Proceedings of the International Conference on Complex, Intelligent and Software Intensive Systems*, pages 913–917.

Gal, A. and Shvaiko, P. (2009). Advances in ontology matching. In *Advances in Web Semantics I*, volume 4891 of *Lecture Notes in Computer Science*, pages 176–198. Springer Berlin, Heidelberg.

Kalfoglou, Y. and Schorlemmer, M. (2003). Ontology mapping: The state of the art. *The Knowledge Engineering Review*, 18(1):1–31.

Noy, N. and Klein, M. (2004). Ontology evolution: Not the same as schema evolution. In *Knoledge and Information Systems*, volume 6, pages 428–440. Springer Berlin, Heidelberg.

Noy, N. F. and McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology. Technical Report Technical Report SMI-2001-0880, Stanford Medical Informatics.

Noy, N. F. and Musen, M. A. (2000). Prompt: Algorithm and tool for automated ontology merging and alignment. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*.

Noy, N. F. and Musen, M. A. (2001). Anchor-prompt: Using non-local context for semantic matching. In *Proceedings of IJCAI Workshop on Ontologies and Information Sharing*, pages 63–70.

van Rijsbergen, C. J. (1975). *Information retrieval*. Butterworths, London, UK. Acesso em 26 de agosto de 2010. Disponvel em `http://www.dcs.gla.ac.uk/Keith/Preface.html`.

Wang, P. and Xu, B. (2009). Lily: Ontology alignment results for OAEI 2009. In *Proceedings of the ISWC 2009 Workshop on Ontology Matching*, pages 186–192.

Wang, Z., Zhang, X., Hou, L., Zhao, Y., Li, J., Qi, Y., and Tang, J. (2010). RiMOM results for OAEI 2010. In *Proceedings of the ISWC 2010 Workshop on Ontology Matching*, pages 194–201.

Algoritmos Genéticos Multi-objetivo para a Seleção de Atributos

Newton Spolaôr[1,2,3], **Ana Carolina Lorena**[1], **Huei Diana Lee**[2,4]

[1]Grupo Interdisciplinar de Mineração de Dados e Aplicações (GIMDA)
Universidade Federal do ABC (UFABC) – Santo André, Brasil

[2]Laboratório de Bioinformática (LABI)
Universidade Estadual do Oeste do Paraná (UNIOESTE) – Foz do Iguaçu, Brasil

[3]Laboratório de Inteligência Computacional (LABIC)
Universidade de São Paulo (USP) – São Carlos, Brasil

[4]Serviço de Coloproctologia, Faculdade de Ciências Médicas (FCM)
Universidade Estadual de Campinas (UNICAMP) – Campinas, Brasil

{newtonspolaor,aclorena,hueidianalee}@gmail.com

Abstract. *The occurrence of irrelevant and/or redundant features in Databases can degrade the performance of computational processes for knowledge extraction, motivating the application of a Feature Selection process. Multi-objective Genetic Algorithms can help identifying subsets of features which optimize combinations of possibly conflicting feature importance measures. This paper presents the use of Multi-objective Genetic Algorithms in Feature Selection, investigating the use of different combinations of feature importance criteria in both labeled and unlabeled datasets.*

Resumo. *A ocorrência de atributos irrelevantes e/ou redundantes em Bases de Dados pode prejudicar o desempenho de processos computacionais de extração de conhecimento, o que motiva a aplicação da tarefa de Seleção de Atributos. Os Algoritmos Genéticos Multi-objetivo podem contribuir para identificar subconjuntos de atributos que otimizam combinações entre diferentes medidas ou critérios de importância de atributos, eventualmente conflitantes. Este trabalho apresenta o uso de Algoritmos Genéticos Multi-objetivo para a Seleção de Atributos, investigando o uso de distintas combinações de critérios de importância de atributos em dados rotulados e não-rotulados.*

1. Introdução

O progresso tecnológico tem possibilitado a construção de conjuntos de dados cada vez maiores, em distintas áreas, sob a forma de Bases de Dados (BD). Processos como a Mineração de Dados (MD) podem ser aplicados nas BD para gerar modelos computacionais que representem padrões presentes nos dados e auxiliem no processo de tomada de decisão [Han and Kamber 2006]. Esses modelos podem ser gerados pelo uso de técnicas de Aprendizado de Máquina (AM) [Mitchell 1997], em que um indutor, como um algoritmo de classificação ou agrupamento (*clustering*), constrói inferências (hipóteses) sobre os padrões presentes nos dados.

A Seleção de Atributos (SA) é uma tarefa de pré-processamento dentro do processo de MD. Ela abrange algoritmos para a identificação de subconjuntos de atributos que melhor representem os padrões inerentes aos dados. Esses algoritmos podem ser genericamente categorizados conforme a interação que realizam com o algoritmo de indução usado na extração de padrões. A abordagem filtro considera propriedades inerentes aos dados, como medidas estatísticas. Algoritmos da abordagem *wrapper* avaliam a importância de atributos com o auxílio do próprio algoritmo de indução de modelos.

De modo geral, a SA pode ser vista como um processo de busca combinatorial por subconjuntos de atributos importantes conforme um ou mais critérios, o que motiva a aplicação de métodos heurísticos como os Algoritmos Genéticos mono-objetivo (AG) e os Algoritmos Genéticos Multi-objetivo (AGM) neste processo. Os AGM possibilitam definir compromissos entre distintas medidas de importância e investigar se a combinação entre medidas, destacadas isoladamente, permite identificar atributos melhores.

Estudos na literatura recente envolvem a aplicação dos AGM ao problema de SA [Bruzzone and Persello 2009, Santana et al. 2009, Wang and Huang 2009]. A partir dos avanços do estado da arte correlato, apresenta-se neste trabalho um método baseado na aplicação de AGM ao problema de SA, combinando diferentes critérios que medem a importância dos atributos considerando a abordagem filtro. O método implementado nesse trabalho apresenta flexibilidade para incorporar com facilidade distintas medidas de avaliação individual ou de subconjuntos de atributos e o suporte para otimizar combinações entre essas medidas. Três medidas são adaptadas para tratar também atributos QL. Esses aspectos possibilitam realizar neste trabalho a SA em conjuntos de dados com propriedades variadas, como a presença ou ausência do atributo-classe e a ocorrência de atributos Quantitativos (QT) ou Qualitativos (QL), diferentemente de [Spolaôr et al. 2011, Wang and Huang 2009, Zaharie et al. 2007].

Este trabalho integra o projeto Análise Inteligente de Dados [Spolaôr et al. 2011, Lee et al. 2006] (AID), o qual é desenvolvido por meio de uma parceria entre a UFABC, o LABI/UNIOESTE, o LABIC/USP e o Serviço de Coloproctologia/UNICAMP. O trabalho é organizado do seguinte modo: na Seção 2 o AGM para SA proposto é apresentado. Os materiais e métodos usados na avaliação experimental desse algoritmo são apresentados na Seção 3. As avaliações experimentais realizadas, bem como os resultados alcançados, são abordados na Seção 4. A conclusão é descrita na Seção 5.

2. Algoritmos Genéticos Multi-objetivo na Seleção de Atributos

A aplicação da SA parte usualmente da representação do conjunto de dados em uma Tabela Atributo-Valor (TAV). Cada linha da TAV refere-se a um exemplo j, enquanto cada coluna corresponde a um atributo i de tipo QT (com valores exclusivamente numéricos) ou QL (não numéricos) para $j = \{1, \cdots, n\}$ e $i = \{1, \cdots, m\}$, em que n é a quantidade de exemplos e m é a quantidade de atributos. O atributo-classe $m + 1$ contém a classe dos exemplos se o conjunto de dados for rotulado. A identificação de um subconjunto S com m' atributos ($1 \leq m' \leq m$) pela SA permite definir uma projeção da TAV original em que os exemplos são descritos apenas pelos atributos selecionados em S.

A tarefa de SA pode ser formulada como um processo de busca por um subconjunto de atributos importantes, em termos de relevância e/ou não-redundância, que auxilie na identificação de padrões inerentes aos dados. A importância de subconjuntos de atri-

butos pode ser definida a partir de diferentes critérios [Liu and Motoda 1998], possivelmente conflitantes. Neste trabalho propõe-se a aplicação de AGM para SA, otimizando simultaneamente diferentes medidas de importância de atributos.

2.1. Medidas de Importância de Atributos

As medidas de importância de atributos existentes podem ser divididas em categorias conforme o tipo de propriedade que investigam [Liu and Motoda 2008]. Medidas de consistência podem reduzir a ocorrência de exemplos com valores similares nos atributos e distintos no atributo-classe. Critérios de dependência enfatizam a identificação de atributos correlacionados. Medidas de distância valorizam atributos que destacam propriedades espaciais dos dados. Critérios de informação buscam amenizar a incerteza presente nos dados. Por fim, a categoria de precisão considera alguma medida de desempenho do algoritmo de indução para estimar a importância dos atributos, sendo em geral associadas à abordagem *wrapper*. Neste trabalho foca-se no estudo e na aplicação das medidas descritas a seguir conforme a abordagem filtro.

Medida de Consistência: a medida Pares de exemplos Inconsistentes (PI) possibilita estimar o grau de consistência inerente a uma projeção correspondente ao subconjunto de atributos investigado [Arauzo-Azofra et al. 2008]. Esse grau é dado pela razão da quantidade de PI para a quantidade total de pares de exemplos da TAV. Essa medida pode ser utilizada para atributos QT e QL por meio de adaptações simples.

Medidas de Dependência: a Correlação Atributo-classe (CC) tem por intuito a investigação das mudanças nos valores dos atributos e das diferenças nos rótulos de classe [Zaharie et al. 2007]. A medida original é apresentada na Equação 1, sendo $x_j\,(i)$ o valor do atributo i no exemplo j, $\|.\|$ a função módulo e $C\,(i)$ uma formulação que destaca atributos com valores divergentes em classes distintas. Para dados QL, essa formulação utiliza $overlap(x_{j1}\,(i)\,,x_{j2}\,(i))$ no lugar da diferença $x_{j1}\,(i) - x_{j2}\,(i)$. A distância *overlap* retorna diferença 1 ou 0, respectivamente, entre atributos com valores distintos ou iguais [Wilson and Martinez 1997]. O peso w_i assume o valor 1 ou 0, respectivamente, se i é ou não é selecionado e $\phi\,(.,.) = 1$ se j_1 e j_2 pertencem a classes diferentes ou $\phi\,(.,.)$ =-0,05 caso contrário.

$$CC = \left(\sum w_i C\,(i)\right) / \left(\sum w_i\right) \qquad (1)$$
$$em\ que\ C\,(i) = \frac{\sum_{j_1 \neq j_2} \|x_{j1}\,(i) - x_{j2}\,(i)\|\,\phi\,(x_{j1},x_{j2})}{n\,(n-1)\,/2}$$

O critério Intra-Correlação (IC) avalia a correlação de Pearson c_p existente entre os atributos de um subconjunto S [Wang and Huang 2009]. Na Equação 2 a correlação global do subconjunto S é normalizada considerando $C\,(m',2)$ como a quantidade total de combinações entre os m' atributos selecionados em S tomados dois a dois. Essa medida pode ser aplicada tanto a conjuntos de dados rotulados quanto não-rotulados.

$$IC = \frac{1}{C\,(m',2)} \sum_{i_l=1}^{m'} \sum_{i_2=i_1+1}^{m'} |c_p\,(x\,(i_1)\,,x\,(i_2))| \qquad (2)$$

Medidas de distância: a Distância Inter-Classe (IE) mensura, por meio da Equação 3, a separabilidade existente entre b classes por meio da distância média entre: (1) o centróide de cada classe c_l, isto é, o exemplo obtido pela média aritmética dos exemplos de c_l e (2) o centróide dos dados $\vec{c}$ [Zaharie et al. 2007]. Nessa equação, $d(.,.)$ denota uma medida de distância e $\vec{c_{c_l}}$ e n_{c_l} representam, respectivamente, o centróide e a quantidade de exemplos em c_l. Para dados com atributos QT e QL empregou-se, respectivamente, a distância Euclidiana e a *overlap*.

$$IE = \frac{1}{n} \sum_{c_l=1}^{b} n_{c_l} d\left(\vec{c_{c_l}}, \vec{c}\right) \tag{3}$$

O *Laplacian Score* (LS) inspira-se no fato de que os exemplos da mesma classe em geral situam-se relativamente próximos uns aos outros. O poder de preservação de localidade pode ser explorado nesses casos para a SA [He et al. 2005]. A medida LS propõe identificar atributos que melhor atendem a estrutura geométrica local dos dados, modelada em um grafo de vizinhos mais próximos conforme a Equação 4. Nessa equação, $\vec{x}\left(i\right) = \left[x_1\left(i\right), x_2\left(i\right), \cdots, x_n\left(i\right)\right]^T$, $\mathbf{1} = \left[1, \cdots, 1\right]^T$ e a matriz D e o grafo Laplaciano L são definidos como $D = diag\left(G\mathbf{1}\right)$ e $L = D - G$. Nessas formulações, a função $diag(.)$ extrai a diagonal de uma matriz e G corresponde à matriz de pesos das arestas do grafo.

$$LS\left(i\right) = \frac{\tilde{x}\left(i\right)^T L \tilde{x}\left(i\right)}{\tilde{x}\left(i\right)^T D \tilde{x}\left(i\right)} \tag{4}$$
$$em\ que\ \tilde{x}\left(i\right) = \vec{x}\left(i\right) - \frac{\vec{x}\left(i\right)^T D\mathbf{1}}{\mathbf{1}^T D\mathbf{1}}\mathbf{1}$$

A definição de vizinhança de cada exemplo pode ser realizada entre k vizinhos mais próximos ou entre todos os exemplos de sua classe. O primeiro tipo de vizinhança é aplicável tanto em dados rotulados quanto não-rotulados pois não requer a informação fornecida pelo atributo-classe.

Medida de Informação: a medida *Representation Entropy* (RE), apresentada na Equação 5, mensura a redundância de atributos de um subconjunto S com auxílio de formulações envolvendo os auto-valores extraídos de uma matriz de covariância [Yan 2007]. Esta matriz é construída a partir da projeção do conjunto de dados investigada sem utilizar o atributo-classe, o que agrega a esta medida a aplicabilidade a dados não-rotulados. A informação é distribuída uniformemente na projeção correspondente a S e há pouca redundância nos dados quando todos os auto-valores obtidos são iguais. O atributo i pode representar toda a informação dessa projeção se apenas o auto-valor λ_i for diferente de 0.

$$RE = -\sum \tilde{\lambda}_i \log \tilde{\lambda}_i \tag{5}$$
$$em\ que\ \tilde{\lambda}_i = \frac{\lambda_i}{\sum \lambda_i}$$

2.2. Algoritmos Genético Multi-objetivo

Os AGM vêm sendo considerados como uma alternativa para tratar o problema de SA. Neste processo, busca-se subconjuntos de atributos que otimizem um ou mais critérios

de importância. Uma grande parte dos trabalhos encontrados na literatura usam alguma medida de precisão dos modelos obtidos com o uso de atributos selecionados, configurando uma abordagem *wrapper*. Este trabalho apresenta como principal diferencial a investigação de múltiplas combinações multi-objetivo de medidas de importância de atributos conforme a abordagem filtro. Também diferentemente de trabalhos correlatos [Spolaôr et al. 2011, Wang and Huang 2009, Zaharie et al. 2007], foram analisados conjuntos de dados, rotulados e não-rotulados, com presença de atributos QT e QL. A adaptação das medidas PI, CC e IE para tratar atributos QL também contribui na flexibilidade do método. Esses fatores evidenciam a generalidade do método apresentado frente a diferentes tipos de dados e análises.

Utilizou-se para a otimização multi-objetivo o AGM *Non-Dominated Sorting Genetic Algorithm* (*NSGA-II*) [Deb et al. 2000], o qual se baseia na teoria de Pareto. Os indivíduos das populações do AGM são iniciados randomicamente e representam distintos subconjuntos de atributos, por meio da codificação binária. Cada subconjunto corresponde portanto a um cromossomo binário com m genes, de modo que um valor 1 no gene i indica a seleção do atributo i e um valor 0 no mesmo gene implica que este atributo não é selecionado. Os operadores genéticos utilizados foram o cruzamento de um ponto e a mutação por *bit flip*. O procedimento de seleção realizado foi o torneio binário.

O *NSGA-II* identifica um conjunto de soluções equivalentemente ótimas ao final do processo evolutivo, ou seja, diferentes subconjuntos de atributos representando distintos compromissos entre as medidas de importância combinadas. Para selecionar um único subconjunto de atributos entre eles, empregou-se a técnica relativamente simples *Compromise Programming* [Zeleny 1973], que propõe a seleção do indivíduo que possui a menor distância para uma solução ideal.

Conforme apresentado na seção anterior, foram selecionadas seis medidas de importância de atributos para serem investigadas como objetivos a serem otimizados. A escolha dessas medidas foi realizada de maneira a se ter ao menos um representante de cada categoria de importância de atributos, com exceção da de precisão. Todas são diretamente aplicáveis a dados rotulados com atributos QT. Três delas (PI, CC e IE) foram adaptadas para lidar também com atributos QL. E as medidas IC, LS e RE são aplicáveis a conjuntos de dados não-rotulados. Nota-se também que as medidas RE e IC oferecem suporte para a análise de redundância de atributos durante a SA, enquanto medidas como PI e IE permitem investigar a relevância dos atributos em relação à classe.

Todas as medidas, exceto a LS, medem a importância de subconjuntos de atributos. A medida LS avalia cada atributo individualmente. Optou-se então em aplicar esta medida para cada atributo e utilizar o valor médio obtido como o valor final para um determinado subconjunto de atributos.

Convém ressaltar que a otimização das medidas PI, IC e LS envolve a minimização dos valores objetivo correspondentes, enquanto que os critérios CC, IE e RE devem ser maximizados por definição. Transformou-se os três problemas de maximização em minimizações equivalentes, de maneira a obter uma uniformidade do método.

O método utilizado suporta atualmente a otimização de combinações multi-objetivo tomadas dois a dois e três a três entre os critérios mencionados na Seção 2.1. A sua flexibilidade também permite adicionar novas medidas e gerar combinações consi-

derando mais objetivos com simplicidade. Desse modo, torna-se possível a investigação experimental de complementaridades variadas entre esses critérios de importância.

3. Materiais e Métodos

Um conjunto de experimentos foi realizado para avaliar o AGM implementado na tarefa de SA. A seguir são apresentados os materiais e métodos empregados neste processo.

3.1. Conjuntos de Dados

Selecionou-se conjuntos de dados investigados em trabalhos correlatos [Spolaôr 2010] do repositório UCI [Asuncion and Newman 2007]: *Australian* (A), *Crx* (C), *Dermatology* (D), *German* (G), *Ionosphere* (I), *Lung cancer* (L), *Promoter* (P), *Sonar* (S), *Soybean small* (Y), *Vehicle*[1] (V), *Wisconsin Breast cancer* (B) e *Wine* (W). As informações sobre cada um desses conjuntos, apresentadas na Tabela 1, incluem a quantidade de atributos QT e QL e de classes (b) e o Erro da Classe Majoritária aproximado (ECM).

Tabela 1. Informações dos conjuntos de dados utilizados nos experimentos.

	A	**C**	**D**	**G**	**I**	**L**	**P**	**S**	**Y**	**V**	**B**	**W**
n	690	653	358	1000	351	32	106	208	47	846	569	178
m	14	15	34	20	34	56	57	60	35	18	30	13
QT	14	6	34	7	34	56	0	60	35	18	30	13
QL	0	9	0	13	0	0	57	0	0	0	0	0
b	2	2	6	2	2	3	2	2	4	4	2	3
ECM (%)	45	45	69	30	36	59	50	47	64	74	37	60

Os conjuntos de dados foram divididos conforme a Validação Cruzada Estratificada (VCE) [Han and Kamber 2006], com 10 partições.

3.2. Método Experimental

Para avaliação dos subconjuntos de atributos selecionados pelos AGM, foram induzidos modelos preditivos (de classificação) e descritivos (de agrupamento) com técnicas de AM nas projeções obtidas. Verificou-se em seguida possíveis ganhos de desempenho dos modelos derivados dos AGM frente àqueles gerados com o uso de todos os atributos. A SA foi aplicada apenas às partições de treinamento dos conjuntos de dados, reservando as partições de teste para uso na construção e avaliação dos modelos. Sendo os AGM estocásticos, eles foram aplicados cinco vezes em cada partição de treinamento, resultando em cinco subconjuntos de atributos por configuração multi-objetivo.

Os modelos de classificação foram induzidos com os métodos J48[2], *Support Vector Machines* (SVM), *k-Nearest Neighbour* (NN) com $k = 1$ e *Naïve Bayes* (NB). Foram utilizados vários algoritmos com o intuito de minimizar a influência que um deles poderia ter nos resultados [Witten and Frank 2005]. A construção de modelos descritivos foi realizada com o popular método de agrupamento *K-Means* (KM) [Jain and Dubes 1988].

Cada algoritmo para SA é avaliado por: (1) desempenho dos modelos derivados e (2) Porcentagens médias de Redução na quantidade original de atributos (PR). Para os

[1]Com apoio do Instituto Turing de Glasgow, Escócia.

[2]A SA embutida do J48, uma implementação do método C4.5, não foi investigada neste trabalho.

modelos de classificação, a medida de desempenho verificada é a taxa de acerto média no procedimento de VCE, em conjunto com seu desvio-padrão. Os modelos descritivos gerados foram avaliados por meio dos métodos de comparação Validação Cruzada Não-supervisionada (VCN) [Filho 2003] e Análise de Replicação (AR) [Morey et al. 1983] nas partições definidas pela VCE. Obteve-se em ambos os casos os valores médios do índice de validação *Rand* Corrigido (RC) [Jain and Dubes 1988]. O RC indica uma perfeita concordância entre agrupamentos pelo valor 1 e uma concordância gerada pelo acaso por valores negativos. Salienta-se que os rótulos não são utilizados para a SA e para a construção dos agrupamentos, apenas para a VCN.

A AR busca aferir a replicabilidade de um algoritmo de agrupamento por meio da técnica *nearest centroid*, a qual utiliza os centróides dos grupos identificados pelo algoritmo em cada partição de treinamento para agrupar os exemplos da partição de teste correspondente. Os grupos assim construídos são comparados aos que são gerados convencionalmente pelo algoritmo na partição de teste. A VCN utiliza o *nearest centroid* inicialmente e então compara os grupos obtidos em uma partição de teste com as classes dos exemplos nele.

3.3. Ferramentas e Configurações Adicionais

Os AGM são aplicados com os parâmetros: (1) *tamanho da população = 50 indivíduos*; (2) *probabilidade de cruzamento = 0,8*; (3) *probabilidade de mutação = 0,01*; (4) *critério de parada = 50 gerações*; (5) *quantidade de pais e de filhos após alteração = 50 indivíduos*. Esses valores foram selecionados a partir da literatura correlata [Spolaôr et al. 2011, Spolaôr 2010, Wang and Huang 2009]. A implementação do método baseou-se em módulos da ferramenta *PISA* [Bleuler et al. 2003]. Um desses módulos, originalmente proposto para um problema específico de otimização, foi adaptado neste trabalho para abstrair as medidas de importância de atributos como os objetivos a serem otimizados pelo AGM. A medida RE foi implementada com o apoio da *GSL*[3].

Os modelos preditivos foram gerados no trabalho com o auxílio da ferramenta *Weka* [Witten and Frank 2005], com valores de parâmetros como *default*. O KM utilizado foi o fornecido na ferramenta *R*[4] com valores dos parâmetros mantidos como *default*, com exceção da quantidade de grupos K. Esse parâmetro foi definido como $K = b$, isto é, K assume valor equivalente à quantidade de classes b no emprego do KM nos conjuntos de dados A, I, S, B, W, D e V.

Em cada conjunto de dados, o desempenho dos modelos gerados por um determinado classificador, obtidos antes e após a realização da SA, foi comparado pelo teste não-paramétrico de Kruskal-Wallis (KW) [Kruskal and Wallis 1952] a 95% de confiança. Ao todo, esse teste foi aplicado aos 12 conjuntos de dados rotulados da Tabela 1 e aos sete não-rotulados mencionados anteriormente. Foi utilizado um único controle, correspondente aos modelos construídos com todos os atributos (c_a), em cada avaliação para amenizar o efeito da multiplicidade [Salzberg 1997].

[3]http://www.gnu.org/software/gsl

[4]http://www.r-project.org

4. Resultados Experimentais

Os experimentos descritos a seguir foram organizados de acordo com o tipo de análise efetuada, ou seja, se sobre dados rotulados ou não-rotulados.

4.1. Seleção de Atributos em Dados Rotulados

Inicialmente, realizou-se um estudo acerca das medidas individuais e combinações de objetivos em duplas e triplas, envolvendo conjuntos de dados com atributos Quantitativos [Spolaôr 2010]. As combinações que se destacaram nesses experimentos foram então investigadas em um número maior de conjuntos de dados. Neste trabalho são apresentados os resultados das combinações IE+CC e IE+PI em conjuntos de dados com atributos QT e QL. Especificamente para dados com presença de atributos QL utilizam-se as medidas adaptadas PI, CC e IE.

A quantidade de modelos derivados de AGM com desempenho estatisticamente não inferior, isto é, superior (entre parênteses) ou equivalente, segundo o teste de KW, aos modelos construídos com todos os atributos é exibida na Tabela 2. Na Tabela 3 são apresentados, para cada modelo baseado no NN derivado de cada algoritmo de SA, as médias e os desvios-padrão, entre parênteses, das taxas de acerto (linha superior) e da PR (linha inferior), respectivamente. O valor médio e desvio-padrão das taxas de acerto e PR para todos os conjuntos de dados também é considerado no fim da tabela. A escolha por esse classificador justifica-se pela sensibilidade que apresenta à presença de muitos atributos (maldição da dimensionalidade) [Han and Kamber 2006] e pela sua relativa simplicidade. Os resultados correspondentes a outros algoritmos de classificação estão descritos em [Spolaôr 2010].

Tabela 2. Quantidade de modelos derivados de AGM para SA em dados rotulados com desempenho estatisticamente equivalente ou superior (entre parênteses) aos modelos c_a.

	IE+CC	IE+PI	**Total**
J48	10 (1)	12 (0)	22 (1)
SVM	8 (0)	12 (0)	20 (0)
NB	9 (1)	12 (0)	21 (1)
NN	9 (0)	11 (0)	20 (0)
Total	36 (2)	47 (0)	83 (1)

4.2. Discussão

Na Tabela 2, $87{,}5\%$ dos 96 modelos (12 conjuntos de dados $\times$ 2 AGM $\times$ 4 indutores) derivados dos AGM apresentam qualidade estatisticamente não inferior à atingida pelos modelos gerados com todos os atributos. Desse modo, nota-se a manutenção do desempenho com uma redução na quantidade de atributos e, consequentemente, no custo computacional envolvido na construção dos modelos.

A competitividade dos modelos derivados dos dois AGM em termos de desempenho preditivo ocorre não apenas em relação a c_a, como indicado anteriormente, mas também para modelos derivados de algoritmos filtro como *Correlation-based Feature Subset Selection*, *Consistency Subset Eval* e AG otimizando cada medida de importância descrita na Seção 2.1 [Spolaôr et al. 2011, Spolaôr 2010].

Tabela 3. Taxas médias de acerto e desvio-padrão (entre parênteses) dos modelos NN derivados e PR média e desvio-padrão dos algoritmos de SA, em cada conjunto e entre todos os conjuntos de dados.

	IE+CC	IE+PI	c_a
A	81,01 (3,73)	80,00 (4,08)	80,00 (4,26)
	57,14 (0,45)	0,00 (0,00)	0,00 (0,00)
C	70,45 (6,50)	81,94 (4,50)	81,32 (3,40)
	60,00 (0,00)	9,07 (0,69)	0,00 (0,00)
D	93,26 (3,50)	94,42 (2,16)	94,42 (2,26)
	22,94 (1,46)	0,00 (0,00)	0,00 (0,00)
G	61,16 (5,19)	70,72 (4,46)	71,90 (4,38)
	89,90 (0,14)	8,10 (1,21)	0,00 (0,00)
I	87,69 (3,84)	87,75 (4,49)	87,75 (4,68)
	32,59 (1,40)	1,41 (0,50)	0,00 (0,00)
L	52,17 (26,82)	46,67 (25,86)	46,67 (26,99)
	35,36 (1,96)	1,04 (0,73)	0,00 (0,00)
P	71,64 (10,91)	73,75 (9,08)	79,27 (7,30)
	24,60 (2,63)	11,68 (2,22)	0,00 (0,00)
S	85,05 (8,69)	85,61 (8,92)	86,98 (7,96)
	56,47 (2,46)	5,73 (1,70)	0,00 (0,00)
Y	100,00 (0,00)	100,00 (0,00)	100,00 (0,00)
	59,60 (0,50)	20,4 (1,65)	0,00 (0,00)
V	68,95 (3,04)	69,74 (2,99)	69,74 (3,12)
	7,44 (0,72)	0,00 (0,00)	0,00 (0,00)
B	93,46 (3,54)	95,25 (3,18)	95,25 (3,32)
	69,40 (0,44)	0,00 (0,00)	0,00 (0,00)
W	95,56 (4,89)	94,97 (4,67)	94,97 (4,87)
	37,69 (0,95)	0,00 (0,00)	0,00 (0,00)
Média	80,03 (15,07)	79,54 (16,83)	84,81 (9,05)
	46,09 (23,21)	4,16 (6,40)	0,00 (0,00)

Em termos de PR, nota-se na Tabela 3 que o AGM IE+CC obteve redução média entre todos os conjuntos de dados superior a 45%. Assim, esse algoritmo permitiu obter vários modelos com desempenho comparável a c_a e um custo computacional menor para a geração de modelos preditivos, pois menos atributos são utilizados durante a indução. O destaque geral da combinação IE+CC é compatível com o observado em relação ao uso de outros AGM em dados QT e QL [Spolaôr 2010].

Contudo, o AGM relacionado a IE+PI apresentou PR média próxima a 4%. Esse aspecto motiva estudos futuros sobre mecanismos que estimulem a identificação de subconjuntos com menos atributos nessa combinação, como a minimização explícita da quantidade de atributos em um objetivo adicional.

Em geral, o uso de algoritmos de classificação distintos pouco influencia na quantidade total de modelos estatisticamente não inferiores, obtida em cada classificador. A não interferência direta do classificador durante a SA, característica da abordagem filtro, pode auxiliar a justificar esse comportamento.

4.3. Seleção de Atributos em Dados Não-rotulados

Na Tabela 4 são quantificados os modelos derivados dos AGM que obtiveram índice RC estatisticamente não inferior, segundo o teste de KW, ao atingido por c_a nos métodos VCN e AR. Na Tabela 5 são apresentados, para cada modelo baseado no KM derivado de

cada algoritmo de SA e avaliado pelo método VCN, as médias e os desvios-padrão, entre parênteses, dos índices RC (linha superior) e da PR (linha inferior), respectivamente. Os resultados correspondentes para o método AR estão descritos em [Spolaôr 2010].

Tabela 4. Quantidade de modelos derivados de AGM para SA em dados QT não-rotulados com desempenho estatisticamente equivalente ou superior (entre parênteses) ao modelo c_a.

	IC+LS	RE+IC	RE+LS	**Total**
VCN	2 (1)	1 (2)	1 (2)	4 (5)
AR	2 (1)	4 (0)	4 (0)	10 (1)
Total	4 (2)	5 (2)	5 (2)	14 (6)

Tabela 5. Taxas médias de RC e desvio-padrão (entre parênteses) dos modelos KM derivados e PR média e desvio-padrão dos algoritmos de SA, em cada conjunto e entre todos os conjuntos de dados, avaliados pelo método VCN.

	IC+LS	RE+IC	RE+LS	c_a
A	0,00 (0,00)	0,19 (0,21)	0,20 (0,17)	0,00 (0,00)
	85,71 (0,00)	58,86 (0,82)	64,29 (0,00)	0,00 (0,00)
D	0,03 (0,04)	0,58 (0,12)	0,59 (0,13)	0,04 (0,04)
	94,12 (0,00)	66,35 (0,54)	58,88 (0,59)	0,00 (0,00)
I	0,07 (0,10)	0,14 (0,20)	0,05 (0,13)	0,17 (0,11)
	94,06 (0,14)	36,94 (0,64)	31,12 (0,54)	0,00 (0,00)
S	0,05 (0,15)	0,00 (0,07)	0,01 (0,07)	-0,01 (0,06)
	92,97 (1,66)	42,40 (1,63)	34,97 (2,13)	0,00 (0,00)
V	0,02 (0,02)	0,02 (0,03)	0,02 (0,02)	0,12 (0,05)
	88,89 (0,00)	88,89 (0,00)	88,89 (0,00)	0,00 (0,00)
B	0,20 (0,11)	0,00 (0,03)	0,00 (0,03)	0,48 (0,10)
	93,33 (0,00)	73,33 (0,29)	56,73 (0,25)	0,00 (0,00)
W	0,22 (0,15)	0,23 (0,15)	0,17 (0,12)	0,36 (0,15)
	84,62 (0,00)	76,92 (0,00)	62,31 (0,30)	0,00 (0,00)
Média	0,10 (0,10)	0,17 (0,19)	0,15 (0,19)	0,19 (0,19)
	89,79 (4,32)	65,08 (17,98)	57,44 (18,06)	0,00 (0,00)

4.4. Discussão

Na Tabela 4, $47{,}62\%$ dos 42 agrupamentos ([7 conjuntos de dados $\times$ 3 AGM para VCN] $+$ [7 conjuntos de dados $\times$ 3 AGM para AR]) derivados de AGM apresentam nível de concordância estatisticamente não inferior a c_a. Esses agrupamentos estão bem distribuídos entre os algoritmos de SA, mas o método VCN concentra os modelos com RC superior.

Observa-se na Tabela 5 que todos os algoritmos obtiveram uma PR média, entre todos os conjuntos de dados, superior a 55%. A agressividade na PR e o desempenho razoável dos modelos derivados, relacionados aos AGM RE+IC e RE+LS, também haviam sido observados anteriormente na SA supervisionada [Spolaôr 2010]. Experimentos complementares envolvendo mais algoritmos de SA e de agrupamento possibilitariam reforçar a competitividade desses AGM em relação a mais *baselines*.

Parte dos agrupamentos obtidos após a aplicação do AGM IC+LS atingiram taxas de concordância aproximadamente equivalentes às obtidas por c_a para ambos os métodos de comparação, apesar da PR ser superior a 85%. Assim, subconjuntos com poucos atributos podem contribuir para a geração de bons modelos, o que não ocorreu em relação a SA supervisionada [Spolaôr et al. 2011, Spolaôr 2010].

5. Conclusão

Neste trabalho apresentou-se um estudo da influência de combinações de medidas de importância de atributos em AGM conforme a abordagem filtro em dados QT e QL. O desempenho de modelos preditivos e descritivos derivados de AGM é comparado à qualidade dos modelos gerados com todos os atributos. A Porcentagem de Redução na quantidade de atributos obtida pela SA também é considerada.

Em geral, uma grande parte dos modelos construídos após a SA foram considerados competitivos em termos de desempenho, enquanto a PR variou conforme o AGM utilizado. Assim, a adaptação dos critérios para tratar dados com atributos QL e a aplicação de medidas para dados não-rotulados foram relativamente bem-sucedidas. O método implementado neste trabalho demonstrou flexibilidade para incorporar distintas medidas para SA em dados com propriedades distintas e suporte para estudar combinações entre elas.

O uso dos AGM para SA em dados artificiais e reais com mais atributos está em andamento. Para dados artificiais, pretende-se realizar análise de relevância e de redundância dos atributos. Trabalhos futuros incluem uma análise ampla da complementaridade entre medidas de importância envolvendo, por exemplo, a intensidade de conflito entre os critérios, e a inclusão de algoritmos *wrapper* e outras heurísticas como *baselines*.

Agradecimentos

A UFABC, CAPES, FAPESP e CNPq pelo apoio financeiro recebido para este trabalho e a Ronaldo C. Prati, Ivan G. C. Filho e colaboradores do projeto AID pela contribuição.

Referências

Arauzo-Azofra, A., Benitez, J. M., and Castro, J. L. (2008). Consistency measures for feature selection. *Journal of Intelligent Information Systems*, 30(3):273–292.

Asuncion, A. and Newman, D. (2007). UCI machine learning repository. http://www.ics.uci.edu/∼mlearn/MLRepository.html.

Bleuler, S., Laumanns, M., Thiele, L., and Zitzler, E. (2003). PISA — a platform and programming language independent interface for search algorithms. In *Evolutionary Multi-Criterion Optimization*, pages 494–508.

Bruzzone, L. and Persello, C. (2009). A novel approach to the selection of spatially invariant features for the classification of hyperspectral images with improved generalization capability. *IEEE transactions on geoscience and remote sensing*, 47:3180–3191.

Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. Technical report, Indian Institute of Technology Kanpur - India.

Filho, I. G. C. (2003). Comparative analysis of clustering methods for gene expression data. Dissertação de mestrado, Universidade Federal de Pernambuco.

Han, J. and Kamber, M. (2006). *Data mining: concepts and techniques*. Morgan Kaufmann.

He, X., Cai, D., and Niyogi, P. (2005). Laplacian score for feature selection. In *Advances in Neural Information Processing Systems*, pages 507–514.

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc., New Jersey, Estados Unidos.

Kruskal, W. and Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *American Statistical Association*, 47:583–621.

Lee, H. D., Monard, M. C., and Wu, F. C. (2006). A fractal dimension based filter algorithm to select features for supervised learning. In *Ibero-American Conference on Artificial Intelligence - Brazilian Symposium on Artificial Intelligence*, pages 278–288.

Liu, H. and Motoda, H. (1998). *Feature Selection for Knowledge Discovery and Data Mining*. Kluwer Academic Publishers.

Liu, H. and Motoda, H. (2008). *Computational Methods of Feature Selection*. Chapman & Hall/CRC.

Mitchell, T. M. (1997). *Machine Learning*. Hardcover.

Morey, L. C., Blashfield, R. K., and Skinner, H. A. (1983). A comparison of cluster analysis techniques within a sequential validation frame work. *Multivariate Behavioral Research*, 18:309–329.

Salzberg, S. L. (1997). On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery*, 1:317–328.

Santana, L. E. A., Silva, L., and Canuto, A. M. P. (2009). Feature selection in heterogeneous structure of ensembles: a genetic algorithm approach. In *International Joint Conference on Neural Networks*, pages 1491–1498.

Spolaôr, N. (2010). Aplicação de algoritmos genéticos multiobjetivo ao problema de seleção de atributos. Dissertação de mestrado, Universidade Federal do ABC.

Spolaôr, N., Lorena, A. C., and Lee, H. D. (2011). Multiobjective genetic algorithm evaluation in feature selection. In Takahashi, R. H. C., Deb, K., Wanner, E. F., and Greco, S., editors, *Lecture Notes in Computer Science (Evolutionary Multi-criterion Optimization Proceedings)*, pages 462–476. Springer-Verlag.

Wang, C.-M. and Huang, Y.-F. (2009). Evolutionary-based feature selection approaches with new criteria for data mining: A case study of credit approval data. *Expert Systems with Applications*, 36(3):5900–5908.

Wilson, D. R. and Martinez, T. R. (1997). Improved heterogeneous distance functions. *Journal of Artificial Intelligence Research*, 6:1–34.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.

Yan, W. (2007). Fusion in multi-criterion feature ranking. In *International Conference on Information Fusion*, pages 01–06.

Zaharie, D., Holban, S., Lungeanu, D., and Navolan, D. (2007). A computational intelligence approach for ranking risk factors in preterm birth. In *International Symposium on Applied Computational Intelligence and Informatics*, pages 135–140.

Zeleny, M. (1973). An introduction to multiobjetive optimization. In Cochrane, J. L. and Zeleny, M., editors, *Multiple criteria decision making*, pages 262–301. University of South Carolina Press.

Meta-aprendizado para otimização de parâmetros de redes neurais.

Tarcísio D. P. Lucas[1], **Teresa B. Ludermir**[1], **Ricardo B. C. Prudêncio**[1], **Carlos Soares**[2]

[1]Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Av. Jornalista Anibal Fernandes – Cidade Universitária – 50740-560 – Recife – PE – Brasil

[2]Faculty of Economic
University of Porto (LIACC) – Porto – Portugal

{tdpl,rbcp,tbl}@cin.ufpe.br, csoares@fep.up.pt

Abstract. *The optimization of Artificial Neural Networks (ANNs) is an important task to the success of using these models in real-world applications. The solutions adopted to this task are expensive in general, involving trial-and-error procedures or expert knowledge which are not always available. In this work, we investigated the use of meta-learning to the optimization of ANNs. Meta-learning is a research field aiming to automatically acquiring knowledge which relates features of the learning problems to the performance of the learning algorithms. The meta-learning techniques were originally proposed and evaluated to the algorithm selection problem and after to the optimization of parameters for Support Vector Machines. However, meta-learning can be adopted as a more general strategy to optimize ANN parameters, which motivates new efforts in this research direction. In the current work, we performed a case study using meta-learning to choose the number of hidden nodes for MLP networks, which is an important parameter to be defined aiming a good networks' performance. In our work, we generated a base of meta-examples associated to 93 regression problems. Each meta-example was generated from a regression problem and stored: 16 features describing the problem (e.g., number of attributes and correlation among the problem's attributes) and the best number of nodes for this problem, empirically chosen from a range of possible values. This set of meta-examples was given as input to a meta-learner which was able to predict the best number of nodes for new problems based on their features. The experiments performed in this case study revealed satisfactory results.*

Resumo. *A otimização de Redes Neurais Artificiais (RNAs) é uma tarefa importante para o sucesso do uso desses modelos em aplicações reais. As soluções para essa tarefa em geral são custosas, envolvendo procedimentos de tentativa e erro ou conhecimento especializado nem sempre disponível. Nesse trabalho, investigamos o uso de meta-aprendizado para otimização de parâmetros de RNAs. Meta-aprendizado é uma área de pesquisa que tem como objetivo adquirir conhecimento de forma automática relacionando características dos problemas de aprendizado com o desempenho dos algoritmos candidatos. As técnicas de meta-aprendizado foram originalmente propostas e avaliadas para o problema de seleção de algoritmos e posteriormente para otimização de parâmetros de Support Vector Machines. No entanto, meta-aprendizado pode ser adotada*

como uma estratégia mais geral para otimização de parâmetros de RNAs, o que motiva novos esforços nessa direção. No presente trabalho, realizamos um estudo de caso usando meta-aprendizado para a escolha do número de neurônios ocultos de redes MLP, que é um parâmetro importante a ser definido visando um bom desempenho das redes. No nosso trabalho, construímos uma base de meta-exemplos associada a 93 problemas de regressão. Cada meta-exemplo foi gerado a partir de um problema de regressão e armazenou: 16 características descrevendo o problema (e.g. número de atributos e correlações entre atributos do problema) e o melhor número de neurônios para o problema, escolhido empiricamente de um conjunto de valores possíveis. Esse conjunto de meta-exemplos foi fornecido como entrada para um meta-aprendiz que foi capaz de predizer o melhor número de neurônios para novos problemas a partir de suas características. Os experimentos realizados nesse estudo de caso apresentaram resultados satisfatórios.

1. Introdução

Redes Neurais Artificiais (RNAs) vêm sendo utilizadas na literatura para resolver problemas nas mais diferentes áreas do conhecimento [Braga et al. 2007]. Apesar do potencial desses modelos, seu desempenho é dependente da escolha dos seus parâmetros de treinamento e arquitetura.

A definição dos parâmetros depende do problema (ou tarefa de aprendizagem) sendo resolvida. Considere por exemplo a definição do número de neurônios da camada escondida de redes *Multilayer Perceptron* (MLP). Se esse número for exageradamente pequeno, pode ser insuficiente para a rede neural aprender os padrões existentes nos dados (i.e. *underfitting*). Já se o número de neurônios for exageradamente grande, a rede pode decorar o conjunto de treinamento (i.e. *overfitting*), perdendo sua capacidade de generalização [Braga et al. 2007].

A otimização dos parâmetros de RNAs é um problema complexo e várias soluções já foram propostas para resolvê-lo. O método manual de tentativa e erro é a mais simples, no entanto, esse método pode exigir muito do usuário e pode não levar a bons resultados. Abordagens mais sofisticadas e sistemáticas, como o uso de algoritmos de busca meta-heurística, têm sido bastante exploradas [Zanchettin et al. 2011] [El-Henawy et al. 2010] [Bin et al. 2010] [Li et al. 2010]. Nesses casos, os melhores parâmetros são definidos por um processo de otimização, em que o desempenho da rede neural corresponde à função objetivo da otimização, e o conjunto de configurações possíveis de parâmetros corresponde ao espaço de busca. Uma limitação dessa abordagem é que o conhecimento adquirido ao lidar com um problema não é usado para ajudar a resolver novos problemas. Isso faz com que a busca comece sempre de soluções definidas aleatoriamente do espaço de busca para um dado novo problema, levando a uma convergência mais lenta.

Meta-aprendizado é uma área que tenta extrapolar o conhecimento adquirido em problemas passados para resolver problemas novos. Meta-aprendizado pode ser definido como o processo de aquisição automática de conhecimento que relaciona o desempenho dos algoritmos de aprendizado com as características dos problemas [Giraud-Carrier et al. 2004]. O conhecimento em meta-aprendizado é adquirido a partir de um conjunto de meta-exemplos. Em geral, cada meta-exemplo é associado a um

problema de aprendizado e é formado por: (1) meta-atributos que descrevem o problema (e.g., número de atributos, número de exemplos, entropia de classe,...) e (2) meta-rótulo (rótulo que indica o algoritmo que obteve melhor desempenho no problema) [Prudêncio and Ludermir 2009]. Diversos trabalhos em meta-aprendizado têm sido desenvolvidos para seleção de algoritmos com sucesso em diferentes domínios de aplicação [Smith-Miles 2008].

As técnicas de meta-aprendizado, originalmente propostas e validadas para seleção de algoritmos, têm sido exploradas para otimização de parâmetros, mais especificamente para otimização de parâmetros de *Support Vector Machines* (SVM) [Alia and Smith-Miles 2006] [Soares et al. 2004]. O bom desempenho do meta-aprendizado nesses trabalhos motiva o uso dessa abordagem para outros modelos de RNAs. Neste trabalho, investigamos o uso de meta-aprendizado para otimização de parâmetros de RNAs. Dessa forma, desenvolvemos um estudo de caso que aplica meta-aprendizado para escolha do número de neurônios escondidos para redes MLP com uma camada escondida. Para tal, construímos uma base de meta-exemplos a partir da avaliação de redes MLP em 93 problemas de regressão, variando sistematicamente o número de neurônios da camada escondida. Cada meta-exemplo é associado a um problema de regressão e armazena (1) 16 características descrevendo os dados dos problemas; (2) o melhor número de neurônios para a rede MLP no problema, definido de forma empírica. Um algoritmo de regressão usado como meta-aprendiz recebeu esse conjunto de meta-exemplos como entrada, sendo usado para predizer o melhor número de neurônios para um novo problema a partir de suas características. No nosso trabalho, avaliamos quatro algoritmos de regressão diferentes: regressão linear, k-NN, M5 e SVM. O estudo de caso validou o meta-aprendizado como alternativa na otimização do número de neurônios da camada escondida. Verificamos nos experimentos que as características dos problemas (meta-atributos) podem ser utilizadas como uma forma de se indicar o número de neurônios da camada escondida de redes MLP.

O restante desse artigo é organizado da seguinte forma. Na Seção 2, é feita uma revisão bibliográfica sobre o problema de otimizar RNAs. A Seção 3 mostra os conceitos básicos de meta-aprendizado. Em seguida, na Seção 4, o trabalho desenvolvido é detalhado. Por fim, na Seção 5, encontram-se as conclusões e trabalhos futuros.

2. Otimização de redes neurais

Redes Neurais Artificiais (RNAs) vêm sendo utilizadas na literatura para resolver problemas nas mais diferentes áreas de conhecimento [Mar et al. 2011] [Gomperts et al. 2011] [Patra and Chua 2010] [Achkar and Nasr 2010] [Harun et al. 2010]. O sucesso desse tipo algoritmo em aplicações motivou o desenvolvimento crescente de novos modelos com diferentes níveis de complexidade. No entanto, a qualidade de um modelo neural é fortemente dependente da definição adequada dos valores de seus parâmetros (e.g., tipo de rede, número de camadas, número de neurônios, taxas de aprendizagem, dentre outros). Para um dado novo problema, existe uma parametrização ótima e específica a ser aplicada no modelo neural. Definir os valores de tais parâmetros não é uma tarefa fácil, pois o número de configurações possíveis costuma ser muito alto. Além disso, existe um custo computacional para avaliar o quão boa é certa parametrização, que varia de acordo com o problema, modelo neural e algoritmo de treinamento escolhido. Obviamente, uma busca exaustiva no espaço de parâmetros não é viável na prática.

Existem várias abordagens para se otimizar os parâmetros de uma RNA. Em muitos casos, a otimização dos parâmetros é feita por tentativa e erro, de forma não sistemática pelo usuário das redes. Essa abordagem pode não levar a bons resultados se o usuário tiver pouca experiência para guiar o processo de busca. Nesse contexto, diferentes autores têm tratado a otimização de RNAs como uma tarefa de otimização, adotando algoritmos de busca meta-heurística para explorar sistematicamente o espaço de configurações possíveis de parâmetros. Diferentes algoritmos já foram usados nesse contexto como Simulated Annealing, Tabu Search, Algoritmos Genéticos e Otimização por Enxame de Partículas [Zanchettin et al. 2011] [El-Henawy et al. 2010] [Bin et al. 2010] [Li et al. 2010].

A Figura 1 ilustra o uso de um algoritmo de busca para otimizar parâmetros para uma rede MLP. Os algoritmos de busca partem de uma ou mais soluções normalmente geradas aleatoriamente. No caso da Figura 1, são geradas várias soluções, cada uma armazenando dois parâmetros: número de neurônios ocultos (*ne*) e taxa de aprendizagem (*ta*). Cada solução é ainda associada a um valor de *fitness* que nesse caso corresponde ao desempenho empírico (e.g., taxa de acerto estimada por validação cruzada) da rede neural quando os parâmetros são adotados. Operadores de busca são adotados iterativamente sobre as soluções atuais para encontrar novas soluções que otimizem a função objetivo desejada.

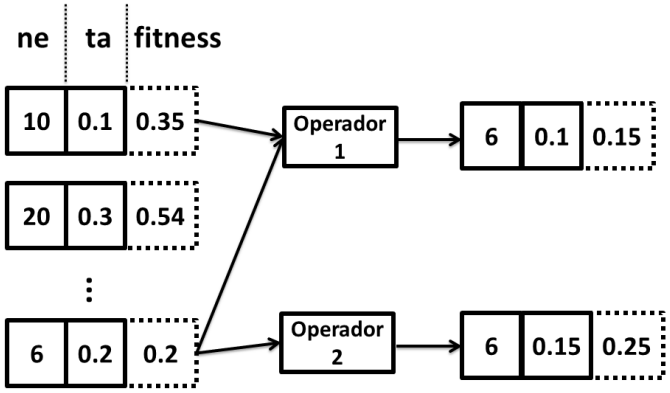


Figura 1. Modelagem de um algoritmo de busca para selecionar parametrização de uma rede MLP com uma camada escondida, onde *ne*, *ta* e *fitness* representam respectivamente número de neurônios escondidos, taxa de aprendizagem e o retorno da função objetivo para tal.

Embora tragam bons resultados, tais abordagens ainda são custosas uma vez que, para o cálculo da função objetivo, é necessário avaliar empiricamente todas as configurações de parâmetros geradas pelo algoritmo ao longo da busca. Além disso, os algoritmos de busca têm por sua vez uma quantidade expressiva de parâmetros a definir (e.g., taxas de *crossover* e mutação, tamanho da população e número de gerações), que têm impacto alto na qualidade do processo de busca. Finalmente, citamos que o conhecimento adquirido ao lidar com um problema não é extrapolado para ajudar a resolver outros problemas. Com isso, para cada novo problema, o processo de busca começa sempre de soluções aleatórias e o algoritmo pode demorar a convergir.

Uma alternativa para a otimização de RNAs é o uso de conhecimento sobre os modelos. De fato, existem uma quantidade grande de regras e heurísticas que podem ser

usadas para guiar a escolha de alguns parâmetros de modelos específicos [Sarle 1997]. Por exemplo, em [Blum 1992], é sugerido que o número de neurônios ocultos de uma rede MLP seja menor que o número de entradas (i.e., número de atributos do problema) e maior que o número de saídas da rede. Embora essas regras possam ser úteis em alguns contextos, elas são limitadas e pouco extensíveis. De fato, em geral é difícil levar em consideração características mais complexas dos dados como correlações entre atributos e ruído nos dados. Além disso, é difícil elicitar conhecimento para modelos de RNAs mais complexos uma vez que especialistas nem sempre estão disponíveis. Nesse contexto, meta-aprendizado surge como uma alternativa para adquirir automaticamente conhecimento que leva em consideração características dos dados para recomendar os melhores algoritmos e parâmetros. Essa abordagem é menos custosa computacionalmente quando comparada com as técnicas de busca e ao mesmo tempo fornece uma estratégia sistemática para aquisição de conhecimento que pode ser adotada para otimização de parâmetros em diferentes modelos de RNAs.

3. Meta-aprendizado

Meta-aprendizado pode ser definido como o processo de aquisição automática de conhecimento, que relaciona o desempenho dos algoritmos de aprendizagem com as características dos problemas [Giraud-Carrier et al. 2004]. O conhecimento em meta-aprendizado é adquirido a partir de um conjunto de meta-exemplos. Cada meta-exemplo, de uma forma geral, é formado por: (1) meta-atributos que definem o problema (número de atributos, número de exemplos, entropia do rótulo, etc.) e (2) meta-rótulo (algoritmo que obteve melhor resultado no problema) [Prudêncio and Ludermir 2009].

Para aplicar meta-aprendizado é necessário construir uma base de meta-exemplos (ou meta-base). Cada meta-exemplo é composto por um conjunto de meta-atributos e um meta-rótulo (variável alvo). Os meta-atributos são, em geral, estatísticas que descrevem o conjunto de dados, tais como número de exemplos, número de atributos, correlação entre atributos, entropia média dos atributos, entre outros [Brazdil et al. 2003] [Kalousis et al. 2004]. Já o meta-rótulo, de uma forma geral, é o melhor algoritmo aplicado ao problema, dentre um conjunto de algoritmos candidatos [Prudêncio and Ludermir 2009]. O atributo alvo é determinado normalmente de forma experimental (por exemplo, validação cruzada).

A maioria dos trabalhos de meta-aprendizado foi proposta para seleção de algoritmos. No entanto, desde 2004, o meta-aprendizado também tem sido aplicado para otimização de parâmetro de algoritmos. Os trabalhos feitos utilizando meta-aprendizado para otimização de parâmetros foram aplicados a máquina de vetores de suporte (do inglês Support Vector Machine - SVM). Num desses trabalhos, tentou-se encontrar a melhor função de *kernel* de um SVM para problemas de classificação [Alia and Smith-Miles 2006]. Nesse caso, a tarefa de meta-aprendizado é uma classificação em que o meta-rótulo é a melhor função *kernel* dentre 5 candidatas, para um determinado problema de classificação. Nesse trabalho foram construídos 112 meta-exemplos, a partir de 112 bases de dados de classificação.

Em outro estudo, procurou-se encontrar a relação entre a largura do *kernel* Gaussiano e os meta-atributos para problemas de regressão [Soares et al. 2004]. Embora o parâmetro de largura do *kernel* Gaussiano seja contínuo, o problema foi tratado como

sendo de classificação, tendo como variável objetivo 11 diferentes valores discretos. Nesse trabalho, foram construídos 42 meta-exemplos, a partir de 42 bases de dados de regressão.

Os resultados promissores obtidos na aplicação de meta-aprendizado para otimização de parâmetros de SVMs [Soares et al. 2004] [Alia and Smith-Miles 2006] motivaram a expansão desse tipo de abordagem para otimização de parâmetros de redes neurais. Este trabalho é um estudo de caso que lança a ideia de aplicar meta-aprendizado na otimização dos parâmetros de RNAs. A rede escolhida para tal foi a MLP com uma camada escondida.

4. Meta-aprendizado para otimização de redes neurais artificiais

Este estudo de caso investiga a utilização de meta-aprendizado na otimização de parâmetros de redes neurais artificiais (RNAs). A construção da meta-base tem, normalmente, um alto custo computacional. Após a geração da base, ainda é necessário treinar o meta-aprendiz com os meta-exemplos gerados. No entanto, o custo computacional de utilização do modelo desenvolvido é muito baixo. Neste trabalho, a investigação se dará em cima da otimização de um dos parâmetros de uma rede MLP com uma camada escondida. O algoritmo de treinamento utilizado para treinar a rede foi o Extreme Learning Machine (ELM).

O ELM é um algoritmo de treinamento para RNAs MLP com uma camada escondida. Sua principal vantagem, com relação a outros algoritmos de treinamentos clássicos, como o *Backpropagation*, é a velocidade de treinamento, que é extremamente veloz. Outra vantagem é que o ELM possui apenas dois parâmetros: número de neurônios escondidos e função de ativação, que deve ser infinitamente diferenciável. Informações detalhadas sobre o funcionamento do ELM podem ser encontradas em [Huang et al. 2004] e [Huang et al. 2006]. Tais vantagens motivaram o uso do ELM para o estudo de caso. A rapidez do treinamento do ELM torna a construção dos meta-exemplos uma tarefa menos custosa do que se fosse utilizado um algoritmo mais lento, como, por exemplo, o *Backpropagation*.

O parâmetro escolhido para o estudo de caso foi o número de neurônios escondidos da rede MLP. Dessa forma, foi fixada uma função de ativação e testou-se todos os valores de neurônios escondidos no intervalo $[1, 300]$ para 93 diferentes bases de dados de regressão. Foram extraídos também os meta-atributos de cada uma das bases de dados. Com isso, foi possível montar uma meta-base com 93 meta-exemplos. Por fim, alguns algoritmos de regressão foram testados como opções de meta-aprendiz.

4.1. Meta-exemplos

A construção dos meta-exemplos passa primeiro pela coleta das bases de dados. Foram selecionadas 93 bases de regressão obtidas através do site do WEKA [1] . Foram selecionadas bases com no mínimo 100 exemplos, e que o atributo alvo (ou rótulo) tivesse pelo menos 10 valores distintos. Tais restrições tiveram como objetivos aumentar a probabilidade de obter problemas que: (1) tivessem conhecimento a ser aprendido (mínimo de 100 exemplos) e que (2) tivessem uma diversidade mínima na variável a ser predita (pelo menos 10 valores distintos)

[1] Site do WEKA - http://www.cs.waikato.ac.nz/ml/weka/

Cada uma das 93 bases teve seus dados normalizados. As variáveis objetivo foram normalizadas entre zero e um. Já as variáveis de entrada contínuas foram normalizadas entre -1 e 1. Atributos simbólicos sofreram transformação para representação binária, onde um atributo com k diferentes valores passou a ser representados por k-1 bits zeros e um bit 1.

Cada meta-exemplo, no nosso trabalho, foi criado a partir de um problema de regressão e armazenou: (1) 16 meta-atributos descritivos do problema; e (2) o melhor número de neurônios para rede MLP no problema, que será o meta-rótulo. Esses dois aspectos serão definidos a seguir.

4.1.1. Meta-atributos

Os meta-atributos coletados tiveram o intuito de representar a complexidade de cada problema. Isso foi feito porque existe uma relação entre a complexidade de um problema e a quantidade de neurônios suficiente para aprender os exemplos que constituem a base. No nosso trabalho foram usados 16 meta-atributos. A escolha dos meta-atributos teve por base trabalhos anteriores de meta-aprendizado aplicado a parametrização do SVMs [Soares et al. 2004] [Alia and Smith-Miles 2006], são eles:

- média de assimetria dos atributos;
- média de curtose dos atributos;
- assimetria média absoluta dos atributos;
- máxima distância média entre cada valor *target* e seus dois vizinhos;
- máxima correlação atributo/*target*;
- número de atributos;
- número de exemplos;
- número de atributos contínuos com *outliers*;
- proporção de atributos contínuos com *outliers*;
- R^2 coeficiente de regressão linear múltipla (apenas atributos numéricos são usados);
- R^2 coeficiente de regressão linear múltipla (atributos simbólicos são binarizados);
- largura do coeficiente de variação absoluta do *target*: 0 se não escasso, 1 se esparsas e 2 se extremamente escasso;
- largura do coeficiente de variação do *target*: 0 se não escasso, 1 se esparsas e 2 se extremamente escasso;
- existência de *outliers* no atributo alvo: 1 se tiver, 0 caso contrário (calculado para os atributos contínuos);
- valor do *outlier*;
- verificar se o desvio padrão é maior do que a média: 1 se for, 0 caso não seja.

4.1.2. Meta-rótulo

Coletados os meta-atributos, o objetivo agora é encontrar o melhor número de neurônios escondidos para as redes MLP em problemas de regressão. No nosso trabalho, utilizamos o algoritmo Extreme Learning Machine (ELM) para o treinamento das redes MLP. O ELM não utiliza conjunto de validação como critério de parada em seu treinamento. Dessa

forma, cada base de dados foi dividida em dois conjuntos: treinamento (70% dos dados) e teste (30% dos dados). O conjunto de treinamento foi utilizado para ajustar os pesos da MLP. Já o conjunto de teste foi utilizado para avaliar a precisão do modelo desenvolvido no treinamento. Foram testados todos os valores de neurônios escondidos no intervalo $[1, 300]$. A função de ativação utilizada foi a *sigmoidal* $g(x) = 1/(1 + exp(-x))$.

Cada experimento foi repetido 10 vezes, com diferentes pesos e bias iniciais. O número de neurônios escondidos considerado ideal para cada problema foi o que obteve o menor erro médio no conjunto de teste nas 10 repetições. O tipo de erro utilizado foi a raiz quadrada média dos erros (do inglês, Root Mean Square Error - RMSE), descrito na Equação 1, onde n é o número de exemplos no conjunto de teste e P_j e T_j são, respectivamente, a saída da rede e a resposta correta (rótulo). Todas as simulações com o ELM foram realizados no MATLAB 7.9, a partir da implementação do ELM do próprio autor[2]. A Tabela 1 ilustra a base de meta-exemplos da meta-base construída. Já a Figura 2 mostra um histograma com a distribuição do meta-rótulo na meta-base. O espalhamento dos valores do meta-rótulo ao longo do histograma indica a complexidade de se aprender o meta-problema de seleção do número ideal de neurônios na camada escondida.

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(P_j - T_j)^2} \tag{1}$$

Tabela 1. Exemplos dos meta-exemplos desenvolvidos para meta-aprendizado do número de neurônios escondidos ideal para um dado problema.

Meta-atributos				**Meta-rótulo**
Número exemplos	Número atributos	Número outliers	...	Neurônios Escondidos
100	3	0	...	4
209	6	5	...	32
950	9	0	...	215
...	...	...	...	...
13750	40	0	...	142

Desenvolvida a meta-base, a etapa seguinte é submetê-la ao meta-aprendiz para avaliar a precisão da predição do modelo.

4.2. Meta-aprendiz

Nesse estudo de caso, a tarefa de meta-aprendizado é prever o número de neurônios da camada escondida ideal para uma rede MLP, baseado nas características do problema ao qual a rede será submetida. O responsável pelo aprendizado dessa tarefa é o meta-aprendiz. Como o meta-rótulo é um inteiro no intervalo $[1, 300]$, optamos por utilizar algoritmos de regressão como meta-aprendiz. O uso de classificadores como meta-aprendiz

[2]Site com a implementação do ELM: http://www.ntu.edu.sg/home/egbhuang/

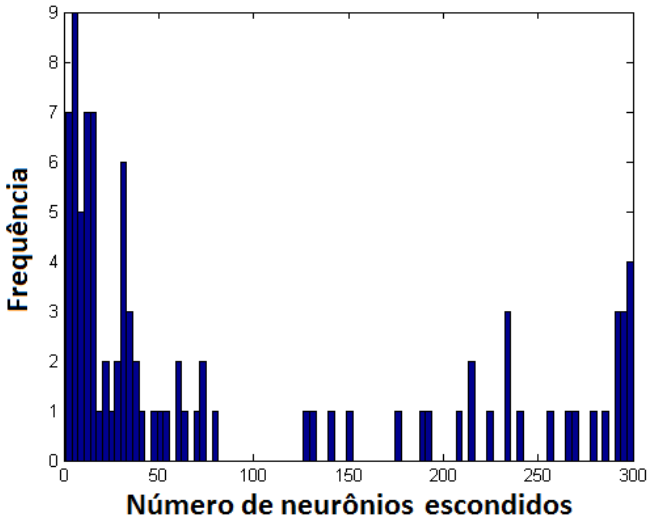


Figura 2. Histograma que mostra a distribuição das melhores opções de neurônios escondidos para cada um dos 93 problemas de regressão.

poderia tornar o problema de meta-aprendizado mais complexo, uma vez que seriam 300 classes a serem aprendidas a partir de apenas 93 exemplos.

Nesse trabalho, foram testados meta-aprendizes utilizados anteriormente na literatura [Guerra et al. 2008]. Todos os algoritmos estão implementados na ferramenta WEKA, são eles:

- 1-NN: um caso especial do algoritmo de aprendizado baseado em instância (do inglês *Instance-based learning algorithms*) [Aha et al. 1991].
- M5: algoritmo proposto por Quinlan para induzir árvores de regressão [Quinlan 1992].
- Regressão Linear
- SVM: utilizando dois diferentes *kernels*: polinomial e RBF.

Todos os meta-aprendizes foram utilizados na sua parametrização *default* do WEKA, exceto o SVM com *kernel* RBF, onde se testaram outras duas parametrizações baseado nos resultados obtidos com esse meta-aprendiz em [Guerra et al. 2008].

4.3. Experimentos e resultados

Nesta seção, é avaliado o desempenho dos meta-aprendizes citados na seção anterior. Para avaliação dos meta-aprendizes, foi adotado um procedimento de *leave one out* com os 93 meta-exemplos disponíveis. Nesse procedimento, cada meta-exemplo é utilizado por vez para testar a precisão do meta-aprendiz, enquanto todos os demais são utilizados em seu treinamento. Dessa forma são feitos 93 treinamentos distintos. A partir desse procedimento, foi observado o erro relativo absoluto (do inglês, relative absolute error - RAE) obtido das respostas dos meta-aprendizes em comparação com os valores reais do melhor número de neurônios em cada problema.

A medida tem valores altos próximos de 100% quando as respostas dos algoritmos têm precisão similar a precisão obtida pela média do atributo-alvo (que foi de 93,8 neurônios escondidos) para os meta-exemplos da base. Nesse caso, valores próximos a

100% indicam que o regressor não está sendo útil e tem desempenho similar a um preditor inocente (predição pela média do atributo-alvo). Além da medida RAE, observamos também a correlação entre as respostas dos meta-aprendizes e as saídas desejadas.

Os resultados obtidos com os meta-aprendizes estão resumidos na Tabela 2. Todos os meta-aprendizes obtiveram RAE abaixo de 100% (a partir de 68,11%). A correlação observada entre as saídas da rede e as respostas corretas variaram entre 0,51 e 0,72. Esses resultados indicam que de fato existem regularidades nos meta-atributos dos problemas que podem ser usados para predizer o número de neurônios ocultos, i.e., que existe conhecimento que pode ser adquirido a partir da base de meta-exemplos. O SVM com função *kernel* obteve a maior precisão e correlação entre os meta-aprendizes testados, com 52,85% de RAE e correlação de 0,72. O SVM já mostrou ser uma boa opção para meta-aprendiz em outras trabalhos de meta-aprendizado, como em [Guerra et al. 2008].

Tabela 2. Medida (RAE) e coeficiente de correlação para 5 diferentes meta-aprendizes na predição do número ideal de neurônios na camada escondidos em redes MLP.

Método	RAE	Correlação
Regressão Linear	68,11%	0,66
SVM (Polinomial)	60,25%	0,66
SVM (RBF)	**52,84%**	**0,72**
1-NN	66,26%	0,51
M5	61,73%	0,62

5. Conclusão

Este trabalho é um estudo de caso que investiga a aplicação de meta-aprendizado na otimização de redes neurais artificiais (RNAs). A rede neural escolhida para o estudo foi a MLP. O algoritmo de treinamento utilizado no treinamento da MLP foi o *Extreme Learning Machine*. O parâmetro escolhido para otimização foi o número de neurônios escondidos.

Uma meta-base com 93 meta-exemplos foi construída e diferentes meta-aprendizes foram aplicado para aprender o meta-problema proposto. Os erros RAE encontrados pelos meta-aprendizes testados na meta-base foram todos menores que 100%, o que assegura que a otimização do número de neurônios escondidos é um meta-problema.

O melhor erro, como em trabalhos anteriores [Guerra et al. 2008], foi obtido tendo o SVM como meta-aprendiz (52,85%). O alto erro reflete a complexidade do meta-problema proposto. Testes realizados por algoritmos de busca provavelmente obteriam soluções mais precisas. No entanto, o custo de se obter uma resposta a parti do modelo de meta-aprendizado desenvolvido será frequentemente inferior a abordagens com algoritmo de busca. Existe ainda uma alternativa intermediária, que seria utilizar o meta-aprendizado para gerar parametrizações promissoras, e atribuir a um algoritmo de busca a tarefa de refiná-las. Dessa forma, o algoritmo de busca já começaria de uma região promissora, e provavelmente chegaria a uma boa solução num tempo mais satisfatório que a busca iniciada aleatoriamente.

Existem ainda alguns estudos que podem melhorar a precisão obtida com meta-aprendizado. A seleção de meta-atributos foi pouco explorada e pode contribuir nesse sentido. Além disso, outros meta-aprendizes e conjuntos de dados podem ser adicionados a fim de encontrar resultados melhores e mais bem embasados. Outro trabalho futuro seria aplicar meta-aprendizado para otimização de redes neurais que possuem o treinamento mais lento como a MLP treinada com o *Backpropagation*.

Referências

Achkar, R. and Nasr, C. (2010). Real time application of an amb using MLP: Study of robustness. In *Computational Intelligence, Modelling and Simulation (CIMSiM), 2010 Second International Conference on*, pages 9–14.

Aha, D. W., Kibler, D., and Albert, M. K. (1991). Instance-based learning algorithms. *Machine Learning*, 6:37–66.

Alia, S. and Smith-Miles, K. A. (2006). A meta-learning approach to automatic kernel selection for support vector machines. *Neurocomputing*, 70:173–186.

Bin, Z. Y., Zhong, L. L., and Ming, Z. Y. (2010). Study on network flow prediction model based on particle swarm optimization algorithm and RBF neural network. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, volume 2, pages 302–306.

Blum, A. (1992). *Neural networks in C++*. John Wiley e Sons, New York.

Braga, A. d. P., Carvalho, A. P. d. L. F., and Ludermir, T. B. (2007). *Redes Neurais Artificiais: Teoria e Aplicações*. LTC, Rio de Janeiro.

Brazdil, P., Soares, C., and da Costa, J. (2003). Ranking learning algorithms: Using IBL and meta-learning on accuracy and time results. *Machine Learning*, 50:251–277.

El-Henawy, I., Kamal, A., Abdelbary, H., and Abas, A. (2010). Predicting stock index using neural network combined with evolutionary computation methods. In *Informatics and Systems (INFOS), 2010 The 7th International Conference on*, pages 1–6.

Giraud-Carrier, C., Vilalta, R., and Brazdil, P. (2004). Introduction to the special issue on meta-learning. *Machine Learning.*, 54:187–193.

Gomperts, A., Ukil, A., and Zurfluh, F. (2011). Development and implementation of parameterized FPGA-based general purpose neural networks for online applications. *Industrial Informatics, IEEE Transactions on*, 7(1):78–89.

Guerra, S. B., Prudêncio, R. B. C., and Ludermir, T. B. (2008). Predicting the performance of learning algorithms using support vector machines as meta-regressors. In *Proceedings of the 18th international conference on Artificial Neural Networks, Part I*, ICANN '08, pages 523–532. Springer-Verlag.

Harun, N., Dlay, S., and Woo, W. (2010). Performance of keystroke biometrics authentication system using multilayer perceptron neural network (MLP NN). In *Communication Systems Networks and Digital Signal Processing (CSNDSP), 2010 7th International Symposium on*, pages 711–714.

Huang, G., Zhua, Q., and Siew, C. (2004). Extreme learning machine: A new learning scheme of feedforward neural networks. *International Joint Conference on Neural Networks*, 2:985–990.

Huang, G., Zhua, Q., and Siew, C. (2006). Extreme learning machine: Theory and applications. *Neurocomputing*, 70:489–501.

Kalousis, A., Gama, J., and Hilário, M. (2004). On data and algorithms: Understanding inductive performance. *Machine Learning*, 54(3):275–312.

Li, H., Cai, M., and Xia, Z.-Y. (2010). Algorithm research of RBF neural network based on improved PSO. In *Multimedia Communications (Mediacom), 2010 International Conference on*, pages 87–89.

Mar, T., Zaunseder, S., Martinez Cortes, J. P., Llamedo Soria, M., and Poll, R. (2011). Optimization of ECG classification by means of feature selection. *Biomedical Engineering, IEEE Transactions on*, PP(99):1.

Patra, J. and Chua, K. (2010). Neural network based drug design for diabetes mellitus using QSAR with 2D and 3D descriptors. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1–8.

Prudêncio, R. B. and Ludermir, T. B. (2009). Active generation of training examples in meta-regression. In *Proceedings of the 19th International Conference on Artificial Neural Networks: Part I*, ICANN '09, pages 30–39. Springer-Verlag.

Quinlan, J. R. (1992). Learning with continuous classes. In *Proceedings of the Australian Joint Conference on Artificial Intelligence*, pages 343–348. World Scientific.

Sarle, W. S. (1997). Neural network faq, part 1 of 7: Introduction. ftp://ftp.sas.com/pub/neural/FAQ.html.

Smith-Miles, K. A. (2008). Cross-disciplinary perspectives on meta-learning for algorithm selection. *ACM Comput. Surv.*, 41:1–25.

Soares, C., Brazdil, P. B., and Kuba, P. (2004). A meta-learning method to select the kernel width in support vector regression. *Machine Learning*, 54:195–209.

Zanchettin, C., Ludermir, T. B., and Almeida, L. M. (2011). Hybrid training method for MLP: Optimization of architecture and training. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 41:1–13.

e-Science - V e-Science Workshop

PREFÁCIO

As últimas décadas têm testemunhado uma revolução na maneira em que o desenvolvimento científico e de engenharia vem sendo conduzido. O aparecimento de instrumentos e equipamentos capazes de produzir grandes volumes de dados, de máquinas com múltiplos núcleos multiplicando a capacidade de processamento, e de redes de comunicação de alta velocidade, abriu as portas para novos desafios científicos na biologia, astronomia, meteorologia e ciências em geral. A utilização destes novos recursos, fundamentais para o avanço científico, modificou o modo de se fazer ciência nas áreas fins e coloca grandes desafios para a área de computação, que deve preparar hardware e software capazes de atender às exigências de aplicações complexas e com crescente demanda de recursos computacionais. Tal simbiose entre ciência e computação criou uma nova área conhecida como eScience posicionando-se, para alguns, como um terceiro ramo da ciência em complemento à ciência teórica e experimental.

No âmbito do que se passou a se identificar como eScience, encontram-se grandes desafios para computação. Dados produzidos por instrumentos e calculados em simulações numéricas alcançam taxas impressionantes de gigabytes por segundo e precisam ser armazenados e processados por workflows científicos. Projetos internacionais de pesquisa compartilham dados e programas que se encontram distribuídos, requerendo sistemas eficientes integrados à grades computacionais e, mais recentemente, à ambientes em nuvem. O volume de dados a ser analisado requer novas técnicas de mineração que considerem milhões de fatos e centenas de atributos qualificadores. Em muitos casos, técnicas de visualização científica se tornam um recurso valiosíssimo na observação de simulações e análises de dados complexos. A lista de novo requisitos é grande e se complementa com camadas de software que ofereçam uma visão transparente, extensível e eficiente da plataforma de hardware e software disponível.

Em particular, uma das áreas que se encaixa perfeitamente nesse contexto de e-Science é a bioinformática. Nesta área, o grande avanço das tecnologias para adquisição de sequências biológicas causou um enorme aumento no volume de dados produzidos. Diversas iniciativas estão em andamento para organizar e disponibilizar esses dados em bancos de dados. Estes bancos contemplam tipos de dados de natureza variável, como sequências nucleotídicas, protéicas, estruturas tridimensionais, padrões, ontologias, perfis probabilísticos, etc. Assim, a e-Science se apresenta como uma área importante no tratamento de dados de bioinformática, utilizando a tecnologia da informação para gerenciar o “dilúvio de dados” e aperfeiçoar o acesso à informação científica.

O V eScience Workshop vem continuar o sucesso de edições anteriores mostrando solucões brasileiras para enfrentar estes grandes desafios na área de eScience. Este ano tivemos a submissão de 25 trabalhos, sendo 20 artigos técnicos e 5 posteres. Foram aceitos 8 artigos técnicos e 4 posteres. Os trabalhos foram organizados em três seções de apresentação: aplicações científicas em eScience; Computação de alto desempenho em eScience; e Serviços de apoio a eScience. O evento contará,

igualmente, com a palestra convidada do Prof. Luis Nicolaci da Costa, astrônomo do Observatório Nacional, que apresentará os eforços que têm sido dedicados pelo grupo do projeto Linea em sua participação de grandes projetos internacionais na área de astronomia.

A qualidade do programa deste ano é fruto da dedicação dos Comitês Diretivo e de Programa, que se empenharam na organização que antecede ao evento e nas revisões dos trabalhos submetidos, contribuindo com os autores na preparação final de seus trabalhos. Em particular, gostaríamos de agradecer à professora Vanessa Braganholo que muito nos ajudou passando sua experiência na organização do IV eScience, bem como aos demais membros do comitê diretivo.

O V eScience workshop contou de forma valiosa com o apoio do web designer Leandro Tinoco na preparação e manutenção do site do evento, e com o constante auxílio da professora Ana Maria Moura na preparação e organização. Gostaríamos, igualmente, de agradecer o suporte do comitê local, principalmente do professor Edgard Correa, garantindo uma comunicação ágil e precisa entre os organizadores do workshop e do evento principal.

Finalmente, esperamos que o programa agrade aos colegas e que possamos estabelecer dicussões profícuas e interessantes que nos levem a contribuir com a eScience no país. Considerando-se as qualidades de Natal, sede do evento, desejamos que os colegas possam aproveitar as belas praias e os deliciosos pratos da culinária natalense.

Um excelente evento para todos!!

Natal, Julho de 2011

Fabio Porto, Marisa Nicolas
Coordenardores do V eScience

COMISSÕES

Coordenação Geral

Fábio André Machado Porto, LNCC (coordenador geral)
Marisa Fabiana Nicolas, LNCC (coordenadora geral)
Edgard de Faria Correa, DIMAp-UFRN (coordenador local)

Comitê de Programa

Alexandre Plastino (UFF)
Ana Maria de Carvalho Moura (LNCC)
Antônio Tadeu Azevedo Gomes (LNCC)
Bruno Schulze (LNCC)
Daniel Guariz Pinheiro (FUNDHERP)
Fabio André Machado Porto (LNCC)
Fernanda Baiao (UNIRIO)
Fernanda Campos (UFJF)
Gilberto Z. Pastorello (University of Alberta)
Israel Tojal da Silva (FMRP)
João Carlos Setubal (Virginia Tech)
João Eduardo Ferreira (IME/USP)
José Antonio Macêdo (UFC)
Jose Miguel Ortega (UFMG)
Leonardo Murta (UFF)
Luciano Digiampietri (USP)
Maria Claudia Cavalcanti (IME-RJ)
Maria Emilia Walter (UnB)
Marisa Nicolas (LNCC)
Marta Mattoso (COPPE/UFRJ)
Mauricio Egídio Cantão (LNCC)
Regina Braga (UFJF)
Ricardo Torres (Unicamp)
Rodrigo Matheus Pereira (UFGD)
Sandra Fabbri (UFSCar)
Sergio Lifschitz (PUC-Rio)
Vanessa Braganholo (UFF)
Vania Vidal (UFC)

ARTIGOS

Predição de Fluidos em um Reservatório Petrolífero Utilizando Métodos de Previsão de Séries Temporais

Harold I. A. Bustos[1], **Marcelino P. S. Silva**[1], **Cláubio L. L. Bandeira**[2], **Ewerton C. Ribeiro**[1]

[1]Departamento de Informática
Universidade do Estado do Rio Grande do Norte (UERN)

[2]Mestrado em Ciência da Computação
Universidade do Estado do Rio Grande do Norte (UERN)
Universidade Federal Rural do Semi-Árido (UFERSA)
Mossoró – RN – Brasil

haroldivan@hotmail.com, {prof.marcelino,claubiobandeira,ewertonce}@gmail.com

Abstract. *The production decline of an oil reservoir is a common and natural thing. Thus, techniques are used with the aim of maintaining and even increasing the recovery of oil and gas from the reservoirs. Among these techniques, it can be highlighted the use of recovery methods that seek to preserve reservoir pressure and move the oil to the producing well. However, most of the oil tends to move to other layers and not just to the wells. Therefore, a solution capable of monitoring this movement and even predict future positions of these fluids is a process very important. We present two algorithms that execute this task using time series forecasting methods. Pre-processed real field data by co-registration were used to validate them.*

Resumo. *O declínio da produção de um reservatório é algo comum e natural. Com isso, técnicas são usadas objetivando manter e até mesmo aumentar a recuperação de petróleo e gás dos reservatórios. Dentre elas podemos destacar o uso dos métodos de recuperação que buscam manter a pressão do reservatório e deslocar o óleo para o poço produtor. Porém, a maior parte do óleo tende a deslocar-se para outras camadas e não só para os poços. Assim, uma solução capaz de monitorar este deslocamento e mesmo prever posições futuras desses fluidos é um processo bastante relevante. Apresentamos dois algoritmos que executam tal tarefa usando métodos de previsões de séries temporais. Dados de campo reais pré-processados por co-registro foram utilizados para validá-los.*

1. Introdução

O declínio da produção de um reservatório de petróleo é algo comum e natural. Isso ocorre devido à redução da pressão interna do reservatório. Visando manter e até mesmo aumentar a produção são empregadas técnicas que objetivam otimizar a recuperação dos hidrocarbonetos (petróleo e gás) contidos no interior do reservatório. Com isso, pode-se identificar os melhores locais para a perfuração dos poços [Ostrander 1984].

Dentre estas técnicas uma bastante comum é o uso de métodos de recuperação, que têm como finalidade tentar manter a pressão do reservatório e deslocar o óleo para o poço produtor. Isso pode ser feito, por exemplo, injetando-se água ou gás dentro do

reservatório. Todavia, grande parte do óleo tende a deslocar-se para outras camadas e não só para os poços.

Dessa forma, tarefas como monitorar esse deslocamento e mesmo prever posições futuras desses fluidos em rochas reservatórios é uma etapa bastante relevante, evitando assim usos desnecessários dos dispendiosos processos de prospecção. Para tanto, existem metodologias que permitem realizar esta tarefa como o co-registro de imagens, também conhecido como alinhamento geométrico. Este, por sua vez, pode ser utilizado para rastrear o vapor d'água injetado nos poços entre as sequências sísmicas, possibilitando o mapeamento do deslocamento de fluidos no reservatório.

Dentre as técnicas utilizadas para este fim, está a conhecida sísmica 4D que baseia-se na aquisição de dados em lapso de tempo (*time-lapse*) de sucessivas imagens 3D sobre uma mesma área [Nur 1989]. Segundo [Lumley 2010], a sísmica 4D pode ser extremamente útil para monitorar o dióxido de carbono (CO_2) injetado e armazenado em reservatórios geológicos de subsuperfície. Um trabalho neste sentido pode ser visto, por exemplo, em [Risso and Schiozer 2008] que proporam um método integrando o histórico de produção e de pressão com os mapas das frentes de saturação obtidos através da sísmica 4D, permitindo ajustá-los de forma que estes modelos possam ser usados na previsão de produção com maior confiabilidade.

Diante deste cenário, o objetivo deste trabalho é realizar a predição de fluidos que, por sua vez, visa estimar o comportamento futuro de um reservatório. Para tanto, é gerado um modelo numérico de reservatório utilizando métodos de previsão de séries temporais baseados em modelos de regressão linear e polinomial do sinal de interesse a ser monitorado. Dados reais de reservatórios obtidos da base de dados do software Madagascar são submetidos aos algoritmos desenvolvidos para validá-los.

2. Predição de Fluidos

Nos poços em regime de produção o volume a ser recuperado tende a cair principalmente em virtude da redução da pressão interna do reservatório. Além disso, da quantidade de petróleo existente nos reservatórios, na prática, apenas uma pequena fração consegue ser retirada [Curbelo 2006].

Com isso, várias técnicas são utilizadas para otimizar a recuperação dos hidrocarbonetos. Dentre as mais utilizadas estão os métodos de recuperação que, por sua vez, têm a finalidade de tentar manter a pressão do reservatório e deslocar o óleo para o poço produtor. Para tanto, isso pode ser feito injetando-se no poço água, gás, vapor d'água dentre outros. No entanto, grande parte do óleo tende a deslocar-se para outras camadas e não só para os poços.

Desta forma, monitorar esse deslocamento e mesmo prever posições futuras desses fluidos em rochas de reservatórios é um processo bastante relevante, uma vez que pode-se assim, direcionar inteligentemente o uso de processos dispendiosos de prospecção. Para abordar este problema, existe a sísmica 4D que, por sua vez, baseia-se na aquisição repetitiva de dados sísmicos em intervalos de tempo. Uma vez coletados, os dados podem ser processados com algoritmos específicos de forma a possibilitar o mapeamento do deslocamento de fluidos no reservatório e até mesmo predizer sua posição futura. A modelagem de fluxo de fluido e sua previsão no tempo pode ser aproximada

através de técnicas de extrapolação, empregando métodos de previsão de séries temporais como regressão linear e polinomial descritos a seguir.

2.1. Métodos de Previsão de Séries Temporais

2.1.1. Regressão Linear

A equação ajustada pelo modelo de regressão linear simples é dada por:

$$I_1(x, y, t) = a_1(x, y)t + b_1(x, y) \tag{1}$$

Onde $I_1(x, y, t)$ representa o tom de cinza do pixel associado à amplitude acústica de cada sequência na t-ésima tentativa, ou observação; (x, y) correspondem às coordenadas do pixel de interesse para cada sequência sísmica; t = 1,2,...,n corresponde à variável independente do tempo; o subíndice 1 indica que um ajuste linear entre duas sequências sucessivas da Amplitude versus Afastamento (AVO, do inglês *Amplitude Versus Offset*) foi usado para calcular a extrapolação deste sinal em um tempo futuro posterior à segunda sequência do sinal AVO; a_1 é o coeficiente de regressão (coeficiente angular); e, por fim, b_1 representa o coeficiente linear. Assim, a_1 e b_1 representam os parâmetros do ajuste da regressão linear no tempo. Sendo este um modelo de regressão baseado em um ajuste linear da serie temporal associada ao sinal AVO, então ele encaixa dentro do grande modelo Auto-Regressivo generalizado chamado de (AR) [Wiener 1964], uma vez que o valor futuro da série pode ser obtido a partir de uma combinação linear dos valores passados da mesma.

Para encontrar uma imagem do sinal AVO extrapolada no tempo é necessário ter uma sequência em um determinado período de tempo – a qual corresponde ao instante de tempo inicial, ou seja, $t = 0$ – e outra sequência em um instante de tempo posterior. Também, é preciso determinar os valores de a_1 e b_1 . Para tanto, b_1 pode ser definido assumindo que t é igual a 0 sobre a Equação 1. Da mesma forma, a_1 pode ser estimado através da Equação 1 assumindo que t corresponde ao período de tempo posterior em que é amostrada a segunda sequência do sinal AVO e substituindo o valor de b_1.

2.1.2. Regressão Quadrática

A equação ajustada pelo modelo de regressão quadrática é demonstrada abaixo:

$$I_2(x, y, t) = a_2(x, y)t^2 + b_2(x, y)t + c_2(x, y) \tag{2}$$

Onde: $I_2(x, y, t)$, do mesmo modo que $I_1(x, y, t)$, representa o tom de cinza do pixel associado à amplitude acústica de cada sequência na t-ésima tentativa, ou observação; da mesma forma, t = 1,2,...,n corresponde à variável independente no tempo; porém, o subíndice 2 indica que um ajuste polinomial quadrático entre três sequências sucessivas foi usado para calcular a extrapolação em um tempo futuro posterior à terceira sequência do sinal AVO; a_2, b_2 e c_2 são parâmetros da equação quadrática e representam os parâmetros do ajuste da regressão quadrática.

Para encontrar uma imagem extrapolada do sinal AVO usando a regressão polinomial quadrática é necessário ter: uma sequência em um determinado período de tempo, a qual corresponde ao instante de tempo inicial, ou seja, t=0; e outras duas sequências em instantes de tempos posteriores diferentes. Da mesma forma, é preciso encontrar os valores de a_2, b_2 e c_2. Para tanto, c_2 pode ser definido assumindo que t é igual a 0 sobre a Equação 2. Para definir a_2 e b_2, é necessário resolver o sistema linear que se forma ao substituir os diferentes instantes de tempo posteriores na Equação 2 que estão associados às amostras das séries temporais conhecidas nestes períodos de tempo e substituir o valor de c_2 nelas.

3. Resultados

O co-registro (alinhamento geométrico) pode ser definido como a sobreposição de duas ou mais imagens – uma imagem base e uma imagem monitor. Isso pode ser feito mediante algumas operações tais como rotação, translação e escala que posicionam alinhadamente as imagens, gerando uma matriz dessa transformação [Rodrigues 2010].

Sobre as aplicações desenvolvidas, foram utilizados dados de campo reais pré-processados por co-registro. Em aplicações sísmica, ele pode permitir o monitoramento dinâmico de fluidos, realçando somente o comportamento do reservatório e possibilitando eliminar grande parte dos artefatos que não pertencem a ele. Tal monitoramento pode ser bastante útil para predizer o fluxo futuro dos fluidos nas rochas reservatórios.

A Figura 1 mostra dados *onshore* (de terra) que serão utilizados neste trabalho. Eles representam dados de campo reais para monitoramento de fluidos em Duri, Indonésia após a aplicação do método de co-registro – sobre os registros de 2 a 19 meses [Fomel and Long 2007]. Neste campo, um registro base foi adquirido antes da injeção de vapor d'água e outros cinco registros foram feitos em uma média de intervalos de 4 meses, que varia de 2 a 19 meses após o início da injeção do vapor. A parte superior do reservatório principal está em cerca de 1,6 s e a base está um pouco abaixo dos 2 s [Lumley 1995].

Estes dados correspondem à variação da amplitude acústica refletida pelas camadas do subsolo para a superfície (onde são captadas pelos sensores) com o afastamento (distância entre a fonte e os sensores). Este sinal é denominado AVO, onde nele, normalmente a amplitude registrada decresce com o afastamento, e na presença de gás ocorre, porém, um crescimento anômalo. Esta anomalia é produzida pela descontinuidade e diminuição brusca da impedância acústica geradas devido à passagem das ondas sísmicas das camadas sólidas para as camadas contendo gás no reservatório. Mediante este comportamento, dá-se a importância da análise de AVO como instrumento para detecção direta de hidrocarbonetos [Ostrander 1984].

A última sequência sísmica conhecida é a de 19 meses após a injeção do vapor. Dada a importância de se estimar o comportamento futuro do reservatório, torna-se preciso estimar uma sequência que ainda não se conhece, por exemplo, 23 meses após a injeção de vapor d'água.

Inicialmente, visando verificar a eficiência e o desempenho do algoritmo proposto, buscou-se extrapolar sequências sísmicas conhecidas. Desta forma, procurou-se obter extrapolações para as sequências de 13 e 19 meses utilizando os métodos de regressão

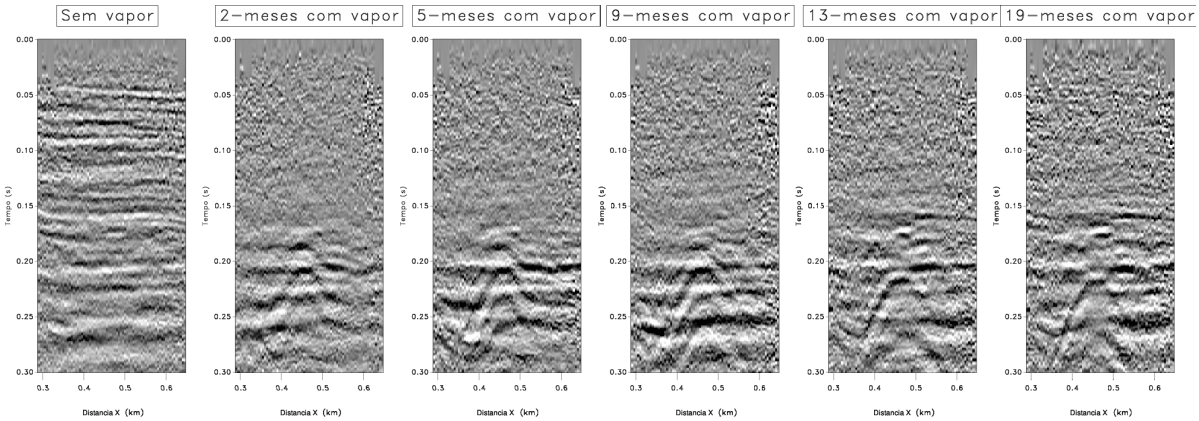


Figure 1. Sequências sísmicas co-registradas. Adaptado de [Fomel and Long 2007]

linear e quadrática, definidos pelas equações 1 e 2, respectivamente. Com isso, é possível quantificar numericamente o quão exato é o processo adotado, pois permite comparar as imagens extrapoladas com as próprias.

Assim, as Figuras 2(a), (b) e (c) correspondem aos processos realizados sobre a sequência sísmica de 13 meses, na qual a Figura 2(a) exibe a imagem de 13 meses original, enquanto que na Figura 2(b) tem-se a sequência sísmica obtida pelo processo de extrapolação regressão linear e a Figura 2(c) representa a sequência sísmica obtida pelo processo de extrapolação regressão quadrática.

Já as Figuras 2(d), (e) e (f) correspondem aos processos efetuados sobre a sequência sísmica de 19 meses, onde na Figura 2(d) tem-se a imagem de 19 meses original, na Figura 2(e) tem-se a sequência sísmica obtida pelo processo de extrapolação regressão linear e na Figura 2(f) a sequência sísmica obtida pelo processo de extrapolação regressão quadrática.

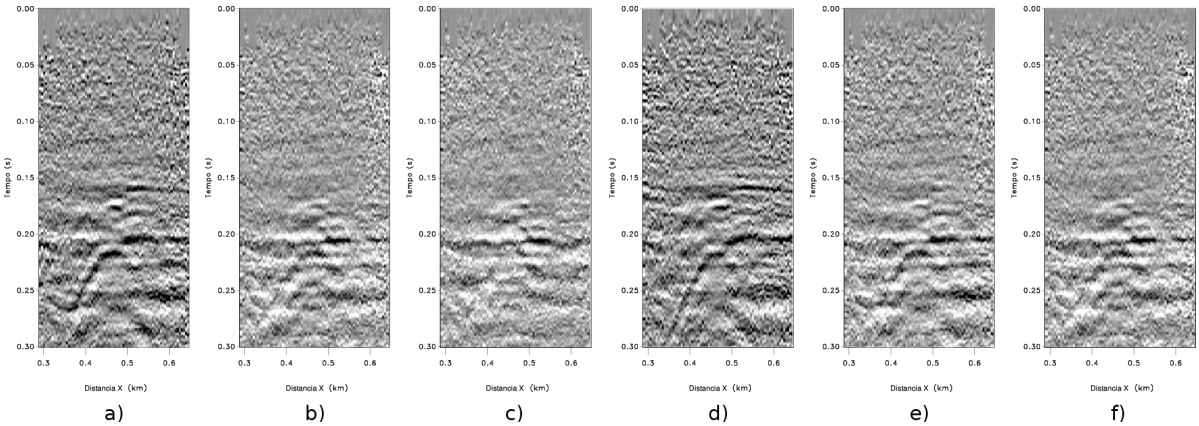


Figure 2. Sequências sísmicas de 13 meses a) original, extrapolada por b) regressão linear e por c) regressão quadrática; e sequências sísmicas de 19 meses d) original, e) extrapolada por regressão linear e f) por regressão quadrática

Para cacular o nível de semelhanças entre as imagens extrapoladas e a sequência

sísmica real, foi utilizado a biblioteca ProEikon [Kim 2011]. O ProEikon é um pacote de *software* com algoritmos de Processamento e Análise de Imagens e Vídeos (PAIV). Esta biblioteca possui um programa chamado DistG que calcula a diferença entre duas imagens em níveis de cinza, possibilitando assim, quantificar numericamente o quão exatas são as imagens geradas.

O DistG possui dentre outros estimadores o *Mean Absolute Error* (MAE, do inglês Erro Médio Absoluto). Na Estatística, ele é uma grandeza utilizada para medir o quão próximas as previsões e as predições estão dos resultados finais, além de ser uma medida comum de erro de previsão em análise de séries temporais. Desta forma, ele permite avaliar o erro entre os dados previstos pela extrapolação numérica e os dados reais utilizados para gerar o sinal AVO da sequência sísmica amostrada. O MAE é representado pela Equação 3.

$$MAE = \frac{1}{n}\sum_{i=1}^{N}|x_i - y_i| \tag{3}$$

Sendo x_i é o valor real obtido, y_i corresponde aos dados previstos e n o número de avaliações consideradas.

A Figura 3 exibe as saídas após a execução do algoritmo DistG. Na Figura 3(a) tem-se o resultado da análise entre a imagem de 13 meses após a injeção do vapor d'água e sua extrapolada obtida por regressão linear. Nela, é possível observar que o algoritmo MAE calculou 13,07% de diferença entre as duas imagens, isto é, elas possuem 86,93% de semelhança. Já a Figura 3(b) mostra a saída do DistG sobre a imagem de 13 meses e sua extrapolada por regressão quadrática, onde o algoritmo MAE calculou 14,72% de diferença entre estas duas imagens, ou seja, elas possuem 85,28% de semelhança, resultado inferior ao método de regressão linear.

A Figura 3(c) mostra o resultado da execução deste algoritmo sobre a imagem de 19 meses e sua extrapolada pelo método regressão linear. Nela, é possível observar que o algoritmo MAE calculou 14,19% de diferença entre as duas imagens, ou seja, elas possuem 85,81% de semelhança. Por fim, a Figura 3(d) mostra a saída após a execução do algoritmo DistG sobre a imagem de 19 meses e sua extrapolada pelo método regressão quadrática. Desta vez, o algoritmo MAE calculou 15,64% de diferença entre estas duas imagens, ou seja, elas possuem 84,36% de semelhança. Este resultado também mostra-se inferior ao método de regressão linear.

Intuitivamente, é de conhecimento que o processo de ajuste polinomial da série poderia permitir obter resultados mais exatos na extrapolação do sinal AVO, em relação aos gerados pelo ajuste linear, pois ele utiliza três sequências de dados de entrada quando comparados com apenas as duas sequências de entrada do ajuste linear. Porém os erros calculados para ambas as aproximações dos ajustes são quase similares, tal como é mostrado na Figura 3.

Uma vez testado os métodos para extrapolar uma sequência conhecida, foram utilizados tais métodos para estimar o comportamento do reservatório 23 meses após a injeção do vapor, isto é, uma sequência futura e que ainda não se conhece. Assim, a Figura 4a) a seguir, exibe a extrapolação por regressão linear e a Figura 4b) exibe o resultado da

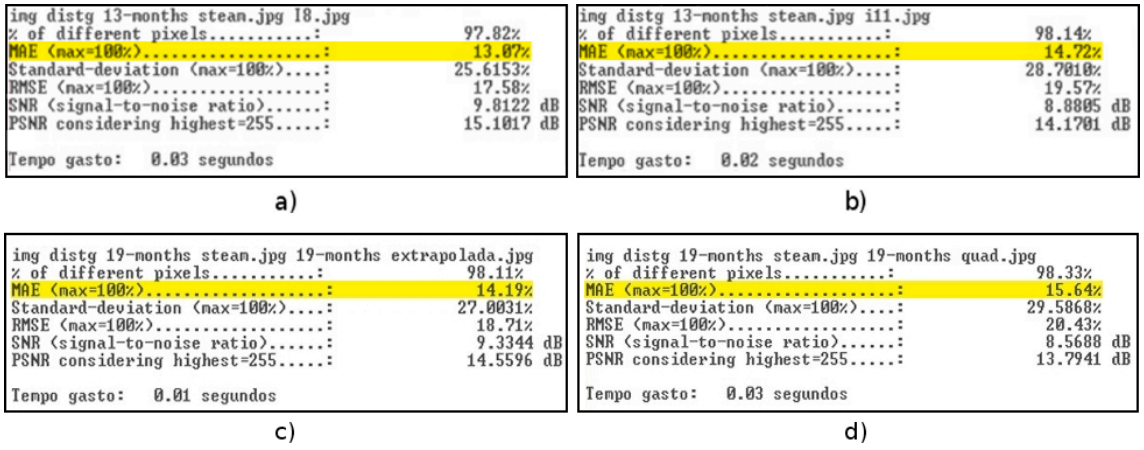


Figure 3. Saídas do algoritmo DistG

extrapolação por regressão quadrática desta sequência.

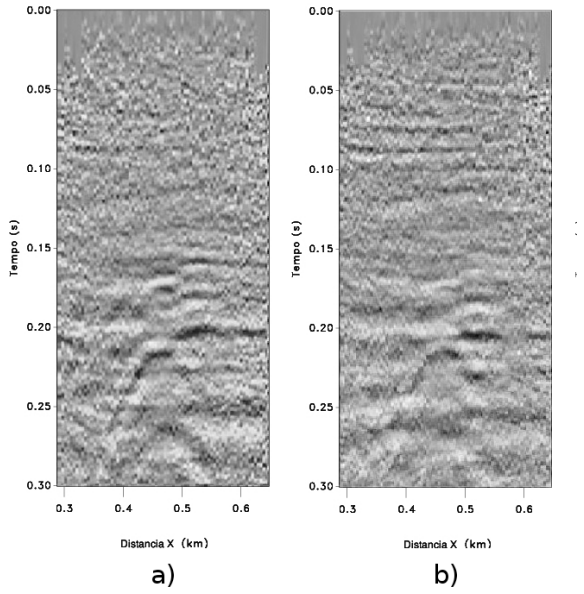


Figure 4. Sequências sísmicas correspondendo a 23 meses após a injeção do vapor extrapoladas por a) regressão linear e b) quadrática

4. Conclusões

A predição do comportamento futuro de um reservatório é uma questão em aberto e é uma área extremamente importante para a indústria petrolífera, pois ela almeja o aumento na recuperação final de hidrocarbonetos. Técnicas para realizar tal tarefa foram propostas neste trabalho empregando métodos de previsões de séries temporais, especificamente regressão linear e regressão quadrática.

Os erros mostrados na Figura 3 e computados por ambos os métodos, evidenciam que o método de ajuste linear descreve melhor o modelo de previsão feito sobre a série espaço temporal do sinal AVO, uma vez que o erro do ajuste linear é menor para ambos os casos se comparado ao erro do ajuste polinomial. Consideramos que tal fato pode ser

explicado devido às sequências do sinal AVO terem sido registradas em intervalos, ou lapsos, de tempo relativamente próximos uns dos outros. Assim, tais sequências apresentam tendências no tempo relativamente idênticas uma das outras. Nesta hipótese, pode ser explicado então porque um ajuste baseado em uma simples regressão linear adquire melhores resultados do que os obtidos por um ajuste polinomial não linear, na tentativa de prever a tendência no tempo do reservatório.

Um outro trabalho já em andamento visa incorporar técnicas de inferência estatística Bayesiana para previsão inteligente de sequências sísmicas em sinais AVO, obtidas a partir de reservatórios de hidrocarbonetos.

Agradecimentos

Os autores agradecem à CAPES e ao CNPq pela concessão das bolsas de pesquisa e pelo apoio financeiro para realização da mesma.

References

Curbelo, F. D. S. (2006). *Recuperação avançada de petróleo utilizando tensoativos*. PhD thesis, Universidade Federal do Rio Grande do Norte (UFRN), Departamento de Engenharia Química.

Fomel, S. and Long, J. (2007). Time-lapse image registration using the local similarity attribute. *77th Annual International Meeting, SEG, Expanded Abstracts*, pages 2979–2983.

Kim, H. Y. (2011). Proeikon - rotinas e programas em c++ para processamento de imagens e visão computacional. Diponível em: *<http://www.lps.usp.br/ hae/software>*. Acesso em 14 de Fevereiro de 2011.

Lumley, D. E. (1995). 4-d seismic monitoring of an active steamflood. *65th Ann. Internat. Mtg, Soc. of Expl. Geophys.*, pages 203–206.

Lumley, D. E. (2010). 4d seismic monitoring of co2 sequestration. *The Leading Edge*, 29(2):150–155.

Nur, A. (1989). Four-dimensional seismology and (true) direct detection of hydrocarbons: the petrophysical basis. *The Leading Edge*, 8(9):30–36.

Ostrander, W. J. (1984). Plane-wave reflection coefficients for gas sand at non normal angles of incidence. *Geophysics*, 49:1637–1648.

Risso, V. F. and Schiozer, D. J. (2008). Utilização de sísmica 4d e de mapas de saturação no ajuste de histórico de reservatórios petrolíferos. *Revista Brasileira de Geociências*, 38(1):172–187.

Rodrigues, E. P. (2010). *Avaliação de métricas para o corregistro não rígido de imagens médicas*. PhD thesis, Universidade de São Paulo (USP).

Wiener, N. (1964). Extrapolation, interpolation, and smoothing of stationary time series. *The MIT Press*.

Managing Animal Sounds - Some Challenges and Research Directions

Daniel Cintra Cugler[1], **Claudia Bauzer Medeiros**[1], **Luís Felipe Toledo**[2]

[1]Institute of Computing
University of Campinas – UNICAMP
13.083-970 – Campinas – SP – Brazil

[2]Museum of Zoology "Prof. Adão José Cardoso", Institute of Biology
University of Campinas – UNICAMP
13.083-862 – Campinas – SP – Brazil

{danielcugler, cmbm}@ic.unicamp.br, toledolf2@yahoo.com.br

Abstract. *For decades, biologists around the world have recorded animal sounds. As the number of records grows, so does the difficulty to manage them, presenting challenges to save, retrieve, share and manage the sounds. This paper presents our preliminary results concerning management of large volumes of animal sound data. The paper also provides an overview from our prototype, an online environment focused on management of this data. This paper also discusses our case study, concerning more than 1 terabyte of animal recordings from Fonoteca Neotropical "Jacques Vielliard", at UNICAMP, Brazil.*

1. Introduction

Earlier animal recordings were commonly stored in magnetic tapes. These recordings are of priceless value, and therefore, special attention must be spent in order to maintain these media clean, free of humidity and fungus infection. More recently, recordings use devices that save data in digital format, such as ATRAC, WAV and MP3.

One initial hurdle is data conversion. In UNICAMP, for instance, researchers have amassed the largest collection of animal recordings in the Neotropics. All of the records are offline, and only recently has there been a concentrated effort in creating digital metadata files for the recordings and converting them to digital media - a painstaking work from expert curators. Converting magnetic tapes to digital format is just part of the problem.

A major challenge is how to conveniently store the data so that other scientists can use them [Gorder 2006]. Besides archival issues, there are many directions that need to be considered in managing animal sound collections. Some such challenges concern the target domain of biology studies - e.g. biologists need to analyze and compare several physical characteristics of the recordings, such as frequency (lower, higher, dominant), call/note/pulse duration, number of pulsers/note, number of notes/call and number of harmonics. Other challenges involve engineering issues - e.g., spatial sound systems, analysis, classification and separation of sounds, automatic speech recognition or even audio compression [Cobos and Lopez 2010].

Still other challenges are directly related to sound management – e.g., data structures, indexing, retrieval, querying and mining. Additional topics involve content-based

retrieval (e.g., the input to a query is a sound recording), or web-based data management (to allow interoperability across distinct collections). This whole scenario is typical of eScience research, in which many kinds of scientists need to get together to advance in their fields.

When one considers possible scenarios of research involving animal recordings, it is possible to define at least two groups of interest:

- Biologists – to study countless aspects of animal communication
- Citizens – to identify animal sounds, or just learn to recognize them

In this context, this paper discusses preliminary research results involving researchers from the Institutes of Computing and Biology at UNICAMP, concerning challenges on managing animal recordings, specially birds. Section 2 discusses related work. Section 3 describes features and main challenges faced on starting to digitalize bird sounds from Fonoteca Neotropical "Jacques Vielliard". Section 4 shows a prototype developed to manage sound data. Section 5 presents final considerations and ongoing work.

2. Related Work and Some Challenges

There are several research directions concerning sound management, as mentioned in the introduction. This section concentrates on issues related to sound databases, and some systems developed to manage animal sounds.

First of all, sound is a wave, and as such there have been studies to perform sound mining – e.g., [Weihs et al. 2007]. As pointed out by [Gerhardt and Huber 2002], sound representation is at least bidimensional, requiring considering pitch and duration, and thus traditional series mining techniques must be extended. Similarly, [Uitdenbogerd and Zobel 1999] propose an algorithm for query collections of MIDI files, searching music by similarity. A range of techniques were used in order to compare whether two pieces of MIDI sounds are similar - including combining melody extraction, standardization and similarity functions to improve the results.

Sound archival is also part of the problem, as mentioned in the introduction [Gorder 2006]. This involves issues of data structures, but also recording formats and compression. Data heterogeneity also plays a major role, a common problem in eScience. The recording is just the tip of the iceberg. It is the result of the scientific methodology used to decide what to record (and when, and where, and with which device). Hence, designing animal sound management platforms is a hard task, especially if different sound collections are to be somehow integrated.

Management of animal sounds requires cooperation from domain experts - the biologists. Different animal groups present distinct forms of sound production and, therefore, different sounds are produced. In general, these sounds evolve under the pressure of sexual selection, promoting significant differences between different species. As a result, almost each animal species in the world produces a distinct sound. However, there is a complication: there are multiple levels of complexity in sound production. For example, crickets and frogs generally present simple advertisement calls (i.e. just one type of note) with well conserved structure (even between individuals of different populations). On the other hand, birds and mammals may present different notes (i.e. complex calls) in the same call, may present differences between populations (i.e. the calls from individuals

of the same species from the northern range may be different from that from its southern distribution), between sex (females *vs*. males), age (juveniles *vs*. adults), social context (e.g., isolated individuals *vs*. grouped individuals; or territorial context *vs*. reproductive context). The social context may also influence sound emission of frogs and crickets [Gerhardt and Huber 2002] [Wells 2007].

As such, standard similarity techniques may not suffice – queries have to consider additional factors, such as geographic location. Also, it is possible that specific algorithms need to be developed for distinct taxonomic families. Indexing is also complicated by such particularities – multidimensional structures might be adequate, the problem is to select the appropriate features to index. Spectrograms can be an important resource to help researchers visualize animal sound. Another interesting challenge related to acoustic data is the automatic creation of metadata, to help reinterpretation and re-analysis of the data [Wimmer et al. 2010].

If users know some metadata (e.g., taxon, location) then a possible approach is to perform the query in multiple steps, starting with textual filtering. However, even text-based queries also pose open problems – e.g., [Daltio and Medeiros 2008]. Not only does metadata content has to be individually curated, for each recording, but there are other domain-specific concerns, such as species naming (and distinct versions of taxonomic classifications), ambiguity in location identification, and diversity imposed by the variety of collection methodologies, affecting data and metadata.

One must also consider sound management software. [Barrington et al. 2007] have proposed a system for audio information retrieval using semantic similarity . The system retrieves generic sound records through ranking items in a database using semantic similarity, rather than acoustic similarity. In order to improve the results, queries are performed using both acoustic algorithms and semantic metadata queries.

In the past, biologists did not always have access to devices to analyze sounds. With the increase of computational power of personal computers, many softwares for acoustic analysis were developed. One example is the research performed by [Wimmer et al. 2010], a workbench that helps ecologists to automatically recognize animal sounds (ASR - Automatic Speech Recognition), annotate and remove sound noise from sounds that can be saved in a centralized database. [Bardeli 2009] created a method for similarity search in animal sound databases. Bardeli also proposed algorithms for feature extraction, indexing and retrieval for animal sound databases. Similarly, [Gunasekaran and Revathy 2010] proposed an algorithm for classification and retrieval of animal sounds. The algorithm is based on fractal dimensions analysis to compute k-nearest neighbors. Such systems, however, are not generic and have their limitations – e.g., the fact of proposing algorithms for specific groups, usually does not cover important features that are essential in groups from similar domains. Then, there are lack of systems that are focused on general domains, covering the necessities of different groups.

A final direction concerns the so-called citizen science (in which citizens contribute to scientific efforts by recording observations, or helping annotate data [Dickinson et al. 2010]). This has given rise to several major data management projects in which citizens provide lots of primary and secondary data - with consequent research on cleaning, indexing and processing such data. Examples are bird-watcher citizen sci-

ence sites (e.g., the projects in the Cornell Ornithology Laboratory[1]) or the FrogWatch project [2] in which people learn to recognize frog calls, and then report hearing them in nature, to find out about wetland ecology. Most of these projects involve entering text, sometimes pictures, but to the best of our knowledge there is no project involving sound management, or sound mining. Some sites, like FrogWatch, publish frog call sounds so that volunteers may learn how to recognize them in nature. The number and variety of possibilities is limited, and there is no linkage to more advanced data management tools.

Analyzing these research efforts, it is clear that animal sound recordings need specific technologies to treat them. For instance, traditional data mining techniques are not able to be employed for these data. Then, although the research cited in this section solves specific issues, all of them have common challenges: how to manage large amounts of heterogeneous animal sound data.

3. Managing Large Amounts of Sounds - A Case Study

Between 1970 and 2010, Prof. Jacques Vielliard *(in memoriam)* and students recorded about 20.000 sounds, mainly birds in South America, mostly recorded in magnetic tapes – now the Fonoteca Neotropical "Jacques Vielliard", UNICAMP. These recordings are undergoing digital conversion. Although this conversion started many years ago, due to its complexity, about 12.000 sounds from the collection have been saved in digital format so far.

This collection - the largest of its kind in the Neotropics - is frequently visited by biologists for their research. It does not support online access, which hampers its widespread use. Thus our first step was to develop a web system to publish sound record metadata, and to allow scientists to upload and download recordings. Our ultimate goal is to increasingly enhance the functions offered at this site, to support novel query and access operations.

Sound management systems, however, are not new - e.g. see Motorola/Shazam Music Recognition Software[3]. What is then so different about management of animal recordings? First, they are often made under difficult conditions, presenting lots of background noise. Also, when recorded in their natural habitats, many animal sounds appear in a single recording, there being a need to identify individuals (or at least individual species). Even if one assumes recordings are performed under ideal conditions, such sounds are very much season, geographic distribution, and social context sensitive (see section 2).

The design of our web system is not a simple task. Biologists primarily use spreadsheets in order to help them manage the collection of sounds, mostly recording metadata. However, these are not adequate for more advanced resources to manage the sound records. Using spreadsheet helps them manage the sounds' metadata, but do not help the management of sound recording, or advanced query processing.

Here, interoperability is a major problem, if one wants to give access to several sound repositories. There arise many issues such as the definition of metadata standards

[1]http://www.birds.cornell.edu/citsci/projects

[2]http://www.aza.org/frogwatch/

[3]http://www.shazam.com/music/web/pages/shazamid.html

or common query vocabularies (and thus ontologies). At present, most recordings are exchanged via email or even by the post office. Second, sound analysis is often based on statistical studies, and queries for records are based on metadata. Query by content is limited, due to the particular characteristics presented by distinct animal groups - see section 2.

4. Publishing Animal Sounds on the Web

This section describes our prototype available at http://proj.lis.ic.unicamp.br/BioProj. It was developed as a web environment to help the management of the sound collection at Fonoteca Neotropical. Its long term goal, however, is to support two kinds of users and usages. First, it is intended to help biologists around the world who want to upload and share bird sounds recorded by them. Thus, it aims to support cooperative research projects that involve animal recording analysis - e.g. in biodiversity, behavior and ecology. Second, it will also contribute to development of citizen science initiatives [Dickinson et al. 2010], allowing non-experts to contribute to enhance data on sound collections, and to learn more about species. The prototype was developed in Java and JSF (Java Server Faces). It was also used Hibernate framework (to perform data persistence), Ajax (Rich Faces Framework) and MySQL DBMS.

Currently, the prototype has three main functionalities:

- Navigate the whole collection of sounds
- Perform metadata-based queries
- Upload bird sounds

4.1. Navigate the whole collection of bird sounds

This functionality allows users to see all recordings, ordered by scientific taxonomy. Though simple, this kind of functionality is a first step towards a more sophisticated navigation system, whose specification requires user feedback. Figure 1 shows the main screen, where biologists are able to view taxonomy data associated with each recording. For instance, record with identification 7 concerns a bird song – Phylum Chordata, Class Aves, and so on. Users can also download non-copyrighted sound files, and view recording metadata by clicking on the ”details” button. Figure 2 shows extra metadata of a selected sound record - e.g., information such as the time and date that the sound was recorded, country, temperature of the air at the moment of the recording. Metadata is extracted from the Excel files.

4.2. Metadata-based textual queries

This functionality allows biologists to retrieve only the desired sound record. Figure 3 shows part of the query form, with multiple textual filters, including specific taxonomy classification terms, country or city where the sound was recorded. For example, Figure 3 shows a query that retrieves all recordings that were recorded in Brazil, whose Phylum is *Chordata*, Class *Aves* and Order *Tinamiformes*. Users can subsequently retrieve more information and download sounds – see Figure 1.



Figure 1. Page to navigate the whole collection of bird sounds

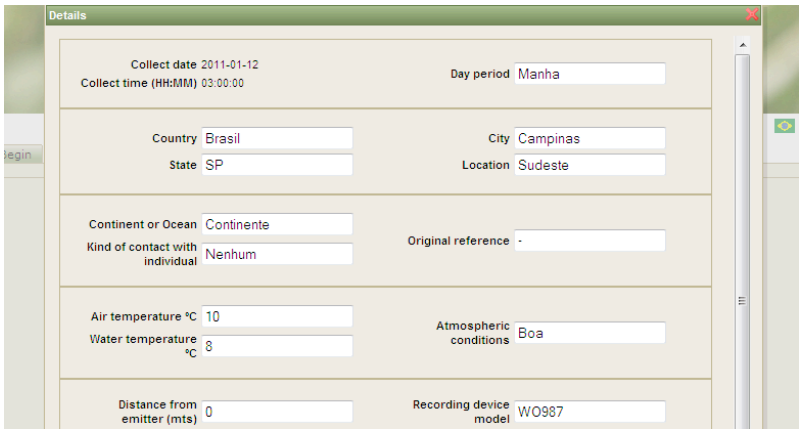


Figure 2. Form containing details about one sound record metadata

4.3. Upload sounds

This functionality allows biologists from Fonoteca Neotropical to upload sounds. Another goal is to allow biologists around the world to upload and share bird sounds recorded by them. The deposit of recordings must be mediated by expert curators. Figure 4 shows the form to upload sounds. It may eventually be used to support citizen science, e.g. people who like to record animal sounds as a hobby will be also able to publish and share their recordings. In this case, submitted data must be differentiated from curated scientific data, posing challenges in distinct management functions.

4.4. Rationale Behind the Prototype

In this project, we have adopted a collaboration policy that has proved to be successful in previous multidisciplinary projects of ours [4]. Instead of designing and developing a complex system with many query possibilities (e.g., sound mining, or indexing), we developed a first prototype to help scientists manage their data. Once they test and approve a prototype, then we can improve on it, on a continuous evolution lifecycle, rapid prototyping style. This has the advantage of ensuring fast feedback and in attracting a larger number of users. We point out that, even if relatively simple, this kind of web system is very much in demand by experts and, in our case, it has the advantage of the richness of the underlying data collection.

[4]http://www.lis.ic.unicamp.br/projects/biocore

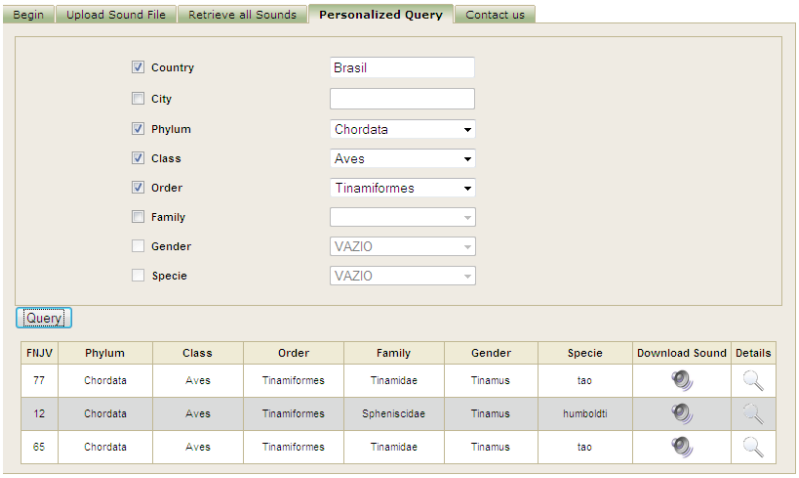


Figure 3. Form to perform metadata-based queries

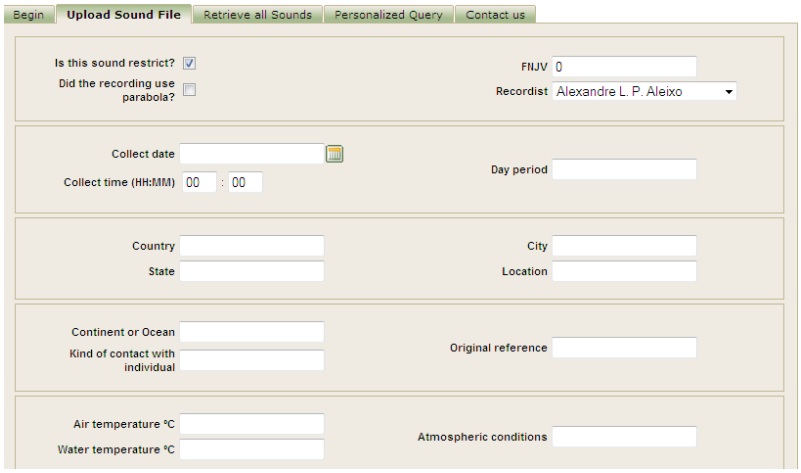


Figure 4. Page to upload bird sounds

5. Final Considerations and Ongoing Work

This research is being motivated by challenges faced by Fonoteca Neotropical on managing large amounts of animal recordings. The current collection, which has more than 1 terabyte of digital sound data, is growing up. Only half the recordings have been digitalized. We are now working on publishing all metadata online, linked to the recordings.

Ongoing and future work involve attacking some of the challenges mentioned, with the ultimate goal of designing and developing a full-fledged animal sound exploratory system for biologists and citizen science. One issue is to make copyright sounds available for biologists that want to use them for research. To do this, it will be necessary to provide distinct access control mechanisms. This control may be also exercised for all downloads, and will create new data files that can be further used for data mining, for example scientists can find other scientists that are working with the same species calls, improving interaction among scientific community and improving worldwide science.

6. Acknowledgments

We thank CAPES, CNPQ and FAPESP for financial support. We also thank biologists from Fonoteca Neotropical "Jacques Vielliard", who provide us support in this research.

References

Bardeli, R. (2009). Similarity search in animal sound databases. *Multimedia, IEEE Transactions on*, 11(1):68–76.

Barrington, L., Chan, A., Turnbull, D., and Lanckriet, G. (2007). Audio information retrieval using semantic similarity. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE Int. Conf. on*, volume 2, pages II–725 –II–728.

Cobos, M. and Lopez, J. (2010). Listen up - the present and future of audio signal processing. *Potentials, IEEE*, 29(4):40 –44.

Daltio, J. and Medeiros, C. (2008). Aondê: An ontology web service for interoperability across biodiversity applications. *Information Systems*, 33(7-8):724–753.

Dickinson, J., Zuckerberg, B., and Bonter, D. (2010). Citizen Science as an Ecological Research Tool: Challenges and Benefits. *Annual Review of Ecology, Evolution, and Systematics*, 41(1).

Gerhardt, H. and Huber, F. (2002). *Acoustic communication in insects and anurans: common problems and diverse solutions*. University of Chicago Press.

Gorder, P. (2006). Not just for the birds: archiving massive data sets. *Computing in Science Engineering*, 8(3):3 –7.

Gunasekaran, S. and Revathy, K. (2010). Content-based classification and retrieval of wild animal sounds using feature selection algorithm. *Machine Learning and Computing, Int. Conf. on*, 0:272–275.

Uitdenbogerd, A. and Zobel, J. (1999). Melodic matching techniques for large music databases. In *Proc. of the 7th ACM Int. Conf. on Multimedia (Part 1)*, MULTIMEDIA '99, pages 57–66, New York, NY, USA. ACM.

Weihs, C., Ligges, U., Morchen, F., and Mullensiefen, D. (2007). Classification in music research. *Advances in Data Analysis and Classification*, 1(3):255–291.

Wells, K. (2007). *The ecology & behavior of amphibians*. University of Chicago Press.

Wimmer, J., Towsey, M., Planitz, B., Roe, P., and Williamson, I. (2010). Scaling Acoustic Data Analysis through Collaboration and Automation. In *2010 IEEE 6th Int. Conf. on e-Science*, pages 308–315. IEEE.

High Performance Computing for Protein Sequence Modeling

Raquel Melo Minardi[1], Karine Bastard[234], François Artiguenave[2345]

[1]Department of Computer Science, Federal University of Minas Gerais (UFMG)
Belo Horizonte, Brazil

[2]Genoscope, Institut de Génomique, CEA, Evry cedex

[3]UMR 8030, CNRS, Evry cedex

[4]University Evry Val d'Essonne, Evry cedex, France

[5]Medical Sciences Faculty, UNICAMP State University, Campinas, Brazil

raquelcm@dcc.ufmg.br, kbastard@genoscope.cns.fr,
francois@fcm.unicamp.br

Abstract. *Common bioinformatics approaches for protein function prediction are based on sequence classification and annotation transfer from known proteins to their closest homologous. These approaches are restricted to homogeneous superfamilies and are not able to predict new activities. Structural biology offers a new insight to overcome this problem by adding protein structure information. Using a 3D modeling approach, we developed a method to predict evolution of catalytic sites in superfamilies. We present results obtained during a computational Grand Challenge on the 350 Tflops CCRT French Supercomputing Facility that illustrate how high performance computing provide new perspectives for understanding protein evolution and function.*

Resumo. *As abordagens mais utilizadas para predição de função de proteínas são baseadas na classificação de sequências e na transferência de funções de proteínas conhecidas para seus homólogos mais próximos. Estas abordagens são restritas às super-famílias homogêneas e não são úteis na predição de novas atividades. A biologia estrutural oferece novos meios de superar esta limitação através da agregação da informação sobre as estruturas de proteínas. Neste trabalho, apresentamos os resultados do Grand Challenge, um desafio do supercomputador francês CCRT de 350TFlopss que ilustra as novas perspectivas que este tipo de tecnologia juntamente com as técnicas de e-Science nos fazem vislumbrar rumo ao entendimento das funções e evolução de proteínas.*

1. Introduction

With the increasing number of genomes being sequenced, a critical challenge concerns functional prediction of proteins encoded by these genes. Several methods are widely used, including sequence and gene context analysis. While these methods are efficient for closed homologous, they are limited to proteins sharing homologies with known proteins. To overcome such problems, structural information can be used, in combination with sequence data, in order to provide a more successful understanding of protein function on molecular basis. To gather as much as possible structural information, recent international initiatives are developing high throughput technologies for experimental resolution of protein structures. However, costs and technical difficulties still explain the huge gap between number of sequences and number of structures available. Homology Modeling is a reliable approach to bridge this gap as long as the target sequence has a minimum sequence similarity with at least one experimentally solved protein structure [Tramontano *et al*., 2001].

To evaluate the potential target of structural approach for annotation, we examined PFAM database [Finn *et al*., 2008] and estimated the size of data eligible to modeling in addition to sequence analysis (table 1). Looking at PFAM v. 23.0 reveals the high number of families without annotation, DUFs (Domain of Unknown Function) and UPFs (Uncharacterized Protein Family) representing 21% of the overall database. While the number of available structures represents only 10% of unknown families, about 200 families are potentially tractable by structural analysis. To complete this view, objectives of the PSI (Protein Structure Initiative) is to solve at least one protein structure for all PFAM families.

Table 1. PFAM (v. 23.0) families lackingq annotation with structural information

Total PFAM entries	DUF	UPF	DUF/UPF with PDB entries
10340	2156	92	208

Depending on the accuracy of the structural information, one can reach different level of elucidation of the relationships between structure and function. At high level of resolution, biochemical reaction may be predicted using molecular modeling and substrate docking methods. At low-resolution level, fold assignment and 3D motif searching can support functional annotation. For example, conserved structural cavities in a protein family are an indicator of active sites. Residues in these cavities are subject to different selective pressures so that multiple alignments can reveal conserved profiles. Hidden Markov Models (HMMs) provide a coherent statistical theory for this analysis.

In this study, we applied a recent developed methodology to 83 DUFs families present in the Cloaca meta-genome data studied at Genoscope (1 million prokaryote sequence proteins from Evry Waste Water Anaerobic Plant). The modelling of 60,000 sequences (1,000 models for each sequence) were obtained during a Computational Grand Challenge proposed by the French Super-Computing Facility CCRT of the CEA (Commissariat à l´Energie Atomique et aux Energies Alternatives). The 60.10^6 generated models are stored in a new structural database, integrating information on

sequence, structure and conserved pockets. Initial analysis of some families allowed the identification of new enzymatic functions and specificities.

2. Active Site Modeling and Clustering

In a recent work [Melo-Minardi *et al.*, 2010], we developed ASMC (|Active Site Modeling and Clustering), a methodology for analysis of residues of protein cavities to detect determinant ones involved in catalytic or enzyme specificity. Briefly, the ASMC method is an unsupervised method for the classification of protein sequences based on structural information of protein pockets. ASMC combines homology modeling of family members, structural alignment of modeled active sites and a subsequent hierarchical conceptual classification. Comparison of profiles obtained from computed clusters allows the identification of residues correlated to subfamily function divergence, called specificity determining positions. Instead of using a global MSA, we use structural alignments of the predicted cavity residues. From theses alignments, we are able to divide the protein families into groups of similar profiles using conceptual clustering [Fisher, 1987]. The analysis detects intra-family variations that can be responsible for function and/or specificity.

The different steps of the ASMC method are summarized in figure 1:

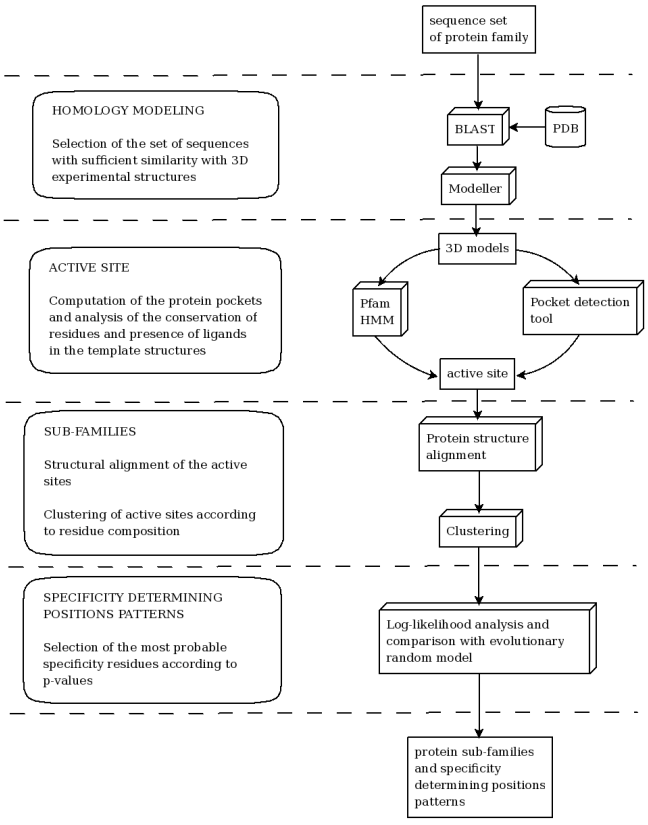


Figure 1. ASMC method diagram

3. Results

We scaled up the methodology to propose a systematic functional annotation of new families, focusing on poorly annotated ones. From 208 families identified in PFAM, we restricted our analysis to 83 families present in *in-house* data from the Evry wastewater Metagenomics project, which are of specific interest for the study of anaerobic prokaryote metabolism (main research project of Genoscope). For the 83 families, the 60.10^6 modelling jobs (Modeller v. 9.6) were run on the "Platine" supercomputing facility at CEA/CCRT. Platine is a cluster of Novascale server including 932 computing nodes and 26 administration or IO nodes. Each node is composed of 4 1.6 Ghz bi-cores Intel® Itanium. Each node has a 24 Gb memory. The Novascale servers, running linux, are interconnected by a Voltaire network (InfiniBand DDR). The LSF batch system is provided by the Platform Computing company, and use resources of Slurm®. The jobs were dispatched on 4000 processors for a total of 280,000 CPU hours. The results were obtained in 70 hours. From the computed models, cavities were detected and conservation profiles were computed. Cavities' key residues were identified and families were clustered into subfamilies. All results are available at (http://bioinfo.speed.dcc.ufmg.br/3dbio/raquelcm/dufs/index.jsp?idioma=ingles).

In order to illustrate perspectives opened by these results, we will present a detailed study on a family for which a new enzymatic activity has been characterized [Bellinzoni et al., 2011]. For this family, DUF849, data from structural modelling provided key insights for enzymatic mechanism elucidation and for analysis of evolution of the activity inside the protein superfamily. The main predicted clusters (figure 2) were tested for enzymatic activities and presented different response.

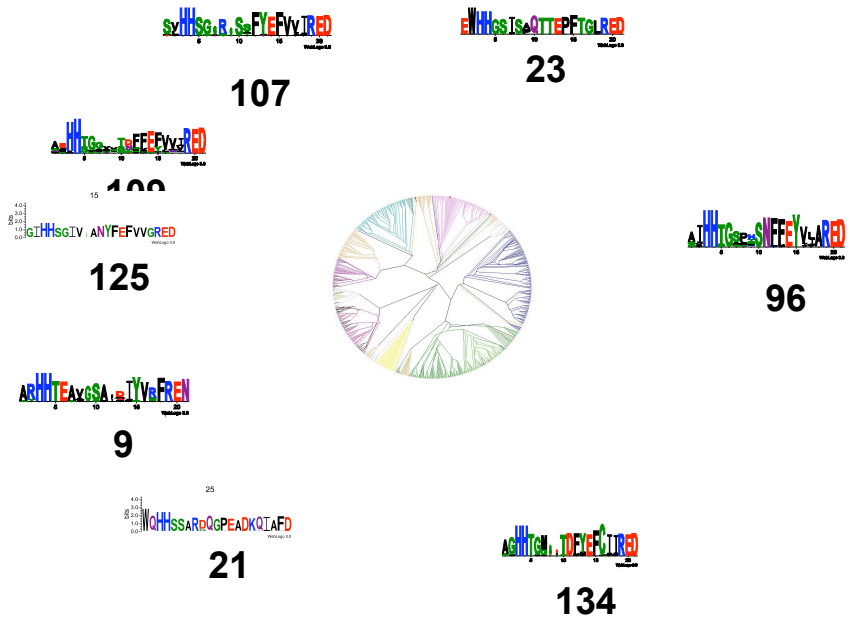


Figure 2. Mapping of catalytic site clusters and profiles on the evolutive tree of the DUF 849 family.

4. Conclusion

For the recent years, many efforts are provided to understand complex molecular and cellular systems including new computations analysis, modelling and simulation capabilities. This work presented an initiative to approximate two computational disciplines of the biology, bioinformatics and chemo-informatics. While bioinformatics focuses mainly to large scale and genomics sequence analysis, traditional chemoinformatics works focused on specific molecular and enzymatic aspects. Current development of informatics infrastructure and the development of supercomputing facilities open new perspectives which combine both statistical and data mining approaches to mathematical modelling approaches. Such approaches can bring new impulses to biology by providing *in silico* tools able to predict *de novo* biological functions.

References

Bellinzoni, M., Bastard, K., Perret, A., Zaparucha, A., Perchat, N., Vergne, C., Wagner, T., Melo-Minardi, R., Artiguenave, F., Cohen, G., Weissenbach, J., Salanoubat, M., Alzari, P. (2011) 3-keto-5-aminohexanoate cleavage enzyme: a common fold for an uncommon reaction, Submitted to PNAS.

Eswar,N. et al. (2006) Comparative protein structure modeling using modeller. Curr. Protoc. Bioinformatics, Chapter 5, Unit 5.6. Eswar,N. et al. (2008) Protein structure modelling with Modeller. Methods Mol. Biol., 426, 145–159.

Finn R.D. et al. (2008) The Pfam protein families database. Nucleic Acids Res., 36, D281–D288.

Fisher,D. (1987) Knowledge acquisition via incremental conceptual clustering. Mach .Learn., 2, 139–172.

Guilloux,V.L. et al. (2009) Fpocket: an open source platform for ligand pocket detection. BMC Bioinformatics, 10.

Holmes, G. et al. (1994) Weka: a machine learning workbench. In Proceedings of the Second Australia and New Zealand Conference on Intelligent Information Systems.

Madhusudhan,M. et al. (2009) Alignment of multiple protein structures based on sequence and structure features. Protein Eng. Des. Sel., 22, 569–574.

Melo-Minardi, R.C., Bastard, K. and Artiguenave, F. (2010) “Identification of subfamiliy-specific sites based on active sites modeling and clustering”. Bioinformatics, 26, 3075-3082.

Pei,J. et al. (2006) Prediction of functional specificity determinants from protein sequences using log-likelihood ratios. Bioinformatics, 22, 164–171.

Shatsky,M. et al. (2004) A method for simultaneous alignment of multiple protein structures. Proteins, 56, 143–156.

Sol,A.D. et al. (2003) Automatic methods for predicting functionally important residues. J. Mol. Biol., 326, 1289–1302.

Sonnhammer,E. et al. (1997) Pfam: a comprehensive database of protein families based on seed alignments. Proteins, 28, 405–420.

Tramontano,A. and Morea,V. (2003) Assessment of homology-based predictions in CASP5. Proteins, 53 (Suppl. 6), 652–368.

Whelan,S. and Goldman,N. (2001) A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. Mol. Biol. Evol., 18, 691–699.

Scalable Simulation of 3D Wave Propagation in Semi-Infinite Domains Using the Finite Difference Method on a GPU Based Cluster

Thales Luis Rodrigues Sabino[1], Marcelo Zamith[1], Diego Brandão[1]
Anselmo Montenegro[1], Esteban Clua[1], Maurício Kischinhevksy[1]
Regina C.P. Leal-Toledo[1], Otton T. Silveira Filho[1], André Bulcão[2]

[1]Instituto de Computação – Universidade Federal Fluminense (UFF)
Rua Passo da Pátria 156 - Niterói - Rio de Janeiro - Brazil

[2]Cenpes, Petrobrás
Ilha do Fundão, Rio de Janeiro - Brazil

{tsabino, mzamith, dbrandao, anselmo}@ic.uff.br

{esteban, kisch, leal, otton}@ic.uff.br, bulcao@petrobras.br

Abstract. *The scattering of acoustic waves has been of practical interest for the petroleum industry, mainly in the determination of new oil deposits. A family of computational models that represent this phenomenon is based on finite difference methods. The simulation of these phenomena demands a high computational cost and large amounts of available memory. In this work we employ GPU based cluster environment for the development of scalable solvers for a 3D wave propagation problem with finite difference methods.*

1. Introduction

The scattering of acoustic waves has been of practical interest for oil and gas industries, mainly the determination of new oil deposits. The scattering process is commonly described with Hyperbolic Partial Differential Equations (PDEs). This kind of equation describes a large variety of physical phenomena governed by wave behavior.

In order to solve such PDEs, the use of numerical methods is employed. Try to solve the scattering problem with a minimum level of precision requires a huge computational effort. Solutions based on multi core CPU clusters have been widely used with successful results [Balevic et al. 2008]. With the advent of GPU computing technology, the numerical solution of PDEs can be done from 20 to 200 times faster than a traditional CPU implementation depending on the problem ([Zamith et al. 2010] and [Brandão et al. 2010]).

GPU computing has become an important choice for many parallel computational problems, since GPUs are potentially more powerful for massively parallel computations than CPUs. Cluster based environments usually needs to work with some kind of communication system. Message interfaces such MPI standard have been extensively used with numerical methods to simulate large complex scattering problems. The use of GPUs in MPI based clusters is almost automatic, since MPI deals with communication steps when GPUs deals with intense computational steps.

One needs to be aware that GPUs are made for graphical purposes and have some limitations. GPU works faster with single precision floating point numbers. Double precision support only became available with recent releases. In order to achieve maximum performance, GPUs dedicates more transistors to arithmetic units than control units so it is important to coalesce memory reads and writes. GPUs have a limited memory (6 GB for NVIDIA Tesla C2070).

Industry scale problems may require use of hundreds of GB of memory and a single GPU may not be suffice to solve such problems timely, not for computational power, but for memory limitations. Taking, for instance, a 3D scattering of acoustic wave problem with a domain discretized in a cube of side 700 requires, at least, 4GB of storage capacity and this requirement grows by a cubic factor.

Many different non-graphical computation, simulation and numerical problems, including Protein Structure Prediction [Langdon and W.Banzhaf 2008], Solution of Linear Equation Systems [Bolz et al. 2003], Options Pricing [Abbas-Turki and Lapeyre 2009], Flow Simulation [Rozen et al. 2008], Wave Propagation [Balevic et al. 2008], [Michea and D.Komatitsch 2010], have been solved in GPUs.

This paper presents a parallel implementation for the scattering of 3D acoustic waves in semi-infinite non-homogeneous domains in a heterogeneous GPUs based cluster capable of divide the problem domain in a way that each domain's partition is handled by one GPU making industry scale wave propagation problems solvable by up two orders of magnitude faster.

2. Acoustic Wave Equation

The wave equation is a second order linear differential equation that describes the behavior of sound waves over time. The acoustic wave field is described by $P(x, y, z, t)$ and $u(x, y, z, t)$, where P is the pressure field the u is the particle's displacement. The relation between P and u is given by $P(x, y, z, t) = -k\nabla^2 u(x, y, z, t)$. Thus, the 3D wave equation is given by:

$$\frac{\partial^2 P(x, y, z, t)}{\partial t^2} = c^2(x, y, z)\nabla P(x, y, z, t) + f(x, y, z, t) \tag{1}$$

where x, y and z are cartesian coordinates, t is time, c is the velocity acoustic wave and $f(x, y, z, t)$ is the source term.

To numerically solve the partial differential equation (Eq. 1), we first discretize it into a set of finite-difference (FD) equations by replacing partial derivatives with central differences. A central-difference approximation can be derived from the Taylor series. Thus, using a second order approximation for space and time, assuming $h = \Delta x = \Delta y = \Delta z$ and $t = n\Delta t$, Eq. 1 is rewritten as:

$$\begin{aligned} P^{n+1}_{(i,j,k)} = 2P^{n}_{(i,j,k)} - P^{n-1}_{(i,j,k)} + A\left[P^{n}_{(i-1,j,k)} + P^{n}_{(i+1,j,k)}\right] - \\ - A\left[6P^{n}_{(i,j,k)} + P^{n}_{(i,j-1,k)} + P^{n}_{(i,j+1,k)} + P^{n}_{(i,j,k-1)} + P^{n}_{(i,j,k+1)}\right] \end{aligned} \tag{2}$$

where $A = \left(\frac{c(x,y,z)\Delta t}{h}\right)^2$ and $n = 1, 2, \ldots$ represents the time slice. Since the velocity field does not vary with time, c is not a function of time.

3. CUDA and GPU Computing

Since 2006 with NVIDIA's CUDA release, the world of high-performance computing became more accessible. A GPU with hundreds of cores is now available by a tenth of the price one need to build a CPU cluster with the same number of processor cores and this cost tends to fall further.

Few years ago, scientific computing based on GPU architecture was developed using graphical shader languages together with some graphics API that allow the execution of such shaders. The programmer needed to map the problem as a set of vertices and fragments to generate a texture representing the final solution. One of the advantages of programming using CUDA API instead of conventional Shader Language is that it allows one to work with familiar concepts while developing kernels that run on GPUs.

CUDA extends existing programming languages with a set of instructions that allows code execution on NVIDIA GPUs. Such extensions expose the GPU hardware memory hierarchy. The GPU memory structure is divided as global, texture and shared. Shared memory is a small but extremely fast memory. This speed comes from its physical proximity to the processing core. Texture memory is a read only memory slower than the shared one, but is almost twice as fast as global memory.

4. Finite Difference Method on GPU

As described in [Brandão et al. 2010], finite difference methods implemented on GPUs are more efficient when it take advantage of shared memory. Using a single GPU one need to allocate memory that is suffice to store tree times the domain size. To compute the next step, one needs the actual and the past steps stored. Figure 1 shows a 3D finite difference explicit scheme with a second order spatial FD operator.

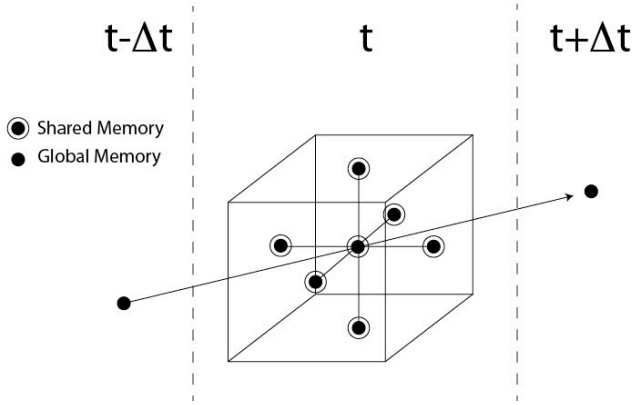


Figure 1. Second order 3D finite difference operator. To compute the next step ($t + \Delta t$) it is necessary to fetch the actual (t) and the past one ($t - \Delta t$).

A typical CUDA application workflow consists in 4 basic steps: (1) Initialize the necessary data on the host, (2) copy the data from host to device, (3) invoke the kernel that

will process the data in device and (4), read the data back from device to the host. Initially, the velocity field is allocated and copied to the device memory. Memory is allocated to contain the values for the past, present and future instants wave amplitude values. The amplitude values for $P^0_{(x,y,z)}$ and $P^1_{(x,y,z)}$ are set with a forward explicit approximation scheme. After initialization, Eq. 2 is then used for subsequent time steps.

A GPU can run millions of lightweight threads. To help the management of these threads, CUDA works with the concept of grid of thread blocks. Basically the threads are organized into thread blocks, which in turn are organized into a grid of blocks. Our approach considered that each new value $P^{n+1}_{(x,y,z)}$ is computed by only one thread. Hence, the 3D domain is divided in a set of 3D blocks, observing the constrains described in [NVIDIA 2010]. The maximum number of blocks that can be allocated is $65,535$ with 512 threads each, giving a total of $33,553,920$ points that can be processed by a single GPU.

Shared memory is used to hold the values of some $P^n_{(x,y,z)}$ step. This avoids unnecessary reads from the global memory. Figure 1 shows that a thread must access seven amplitude values of the same instant $P^n_{(x,y,z)}$. Bringing this values to shared memory makes this accesses much faster. The values of instants $P^{n-1}_{(x,y,z)}$ are fetched direct from global memory, since all the threads of the same block will access only one position, we guarantee a coalesced memory read taking maximum performance of GPU architecture.

4.1. Work Division Across Multiple GPUs

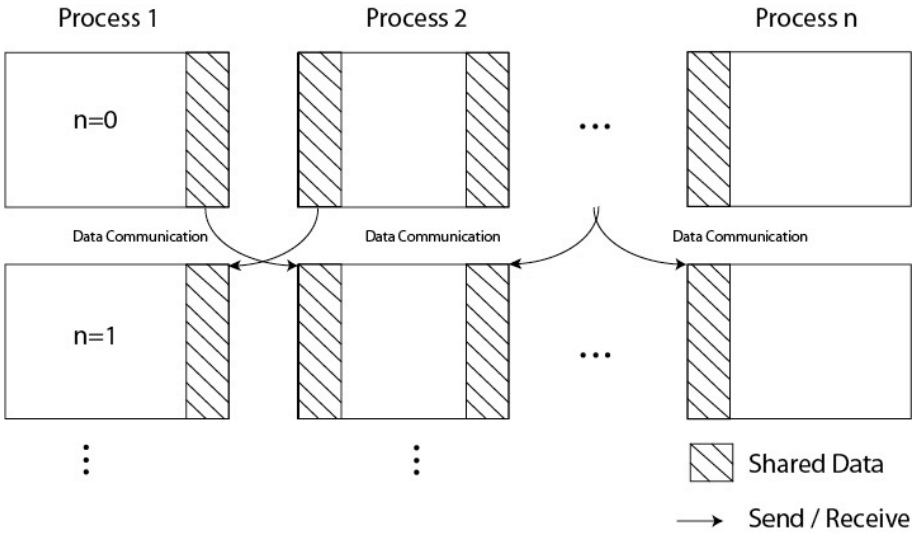


Figure 2. Problem division across multiple devices. Each device is assumed to be handled by one CPU. Each CPU is running its own process and communication of necessary data is done using MPI when all devices finish computation of associated data.

As we discussed before, a considered small problem for industry standards might not be handled entire by a single GPU. Our work implements a solution that consists in split problem domain in a way that each partition fits in a single GPU. This allows the use of a GPU based cluster to solve industry scale problems with the power of GPU.

Figure 2 shows how we organized the devices in an array with the data that need

to be communicated between time steps. For 3D problems, each device handles a parallelepiped whose borders need to be sent to other processes in order to have the entire array of devices solving one single problem. The division planes are chosen in way which communication data are reduced at minimum. This division ensures a maximum of two communication interfaces. Taking for instance a division scheme where a internal subportion of parallelepiped is chosen for a device, there is a need to communicate data up to six interfaces, which can become impractical for both communication costs and by memory consumption since data near communication interfaces are redundant.

The "data communication" stage is done at the of each simulation step. Processes need to communicate the edge values of each $P^{n+1}_{(x,y,z)}$ state, like the diagram shown in Figure 2. The amount of data traffic relies on the order the FD spatial operator chosen. Practical simulations uses 6^{th} up to 10^{th} order spatial operators, which can turn the communication time larger than computation time. In such situations, the final time of simulation may not be smaller as one that use a pure CPU implementation. Network switched technologies such Infiniband turn communication time almost negligible, which allows our scheme for use in practical situations.

The user might want to store the state of simulation at any instant. Take any instant n, the state of the entire simulation is scattered across the devices on the cluster. Knowing that a state can be composed of hundreds of GB of memory, such state may need to be written directly to a secondary memory device.

We present a communication scheme for 3D simulation of scattering of acoustic wave equation using a finite differences method that is capable of deal with industry scale problems that may not fit entire in a single GPU.

5. Results and Analysis

As discussed before, a considered small problem for industry standards may easily bypass a single GPU memory capacity. In order to run large finite difference simulations, we implement a communication scheme like the one shown in Figure 2. The amount of data transferred from one process to another is given mainly by the spatial order of the FD method being used. The amount of data that need to be transferred / received from one process to another is given by $x \times y \times d$, where d is the method discretization order. We choose a domain division in a way that a GPU is fulfilled on its maximum capacity. One should note that doing a more refined division not necessarily generates more traffic in the network, since one process do not need to broadcast its data, only send / receive directly to / from another process.

Our test environment consists in a cluster composed by 64 quad-core CPUs and 128 NVIDIA Tesla C1060 GPUs. Each process uses a single GPU. We run simulations with different number of processes in order to evaluate communication bottlenecks. We run simulations with different domain sizes and kernel configurations. We measure the performance in giga samples per second (GSamples), meaning the number of calculations that can be performed per second, that is, $10^{-9}(nx \times ny \times nz)/(computing_time)$, where nx, ny and nz are the number of discretized points in x, y and z axis, respectively. Our application was implemented using NVIDIA CUDA API and the communication stages are done with MPI.

The simulation configurations used are listed in Table 1. We choose these con-

figurations in a way that each GPU is used on its full capacity. For each configuration, the sizes of xy planes are kept varying only the z-axis value. The size of z-axis chosen determine the number of GPUs necessary to solve the problem. This values are listed in Table 1 too.

Simulation Configurations					
z	**Hosts**	**Processes**	**z**	**Hosts**	**Processes**
64	1	1	1280	10	20
128	1	2	1408	11	22
256	2	4	1536	12	24
384	3	6	1664	13	26
512	4	8	1792	14	28
640	5	10	1920	15	30
768	6	12	2048	16	32
896	7	14	4096	32	64
1024	8	16	8192	64	128
1152	9	18			

Table 1. Simulation configurations

Figure 3 shows a theoretical GSample values. For each configuration, the theoretical GSample values are obtained taking the time spent for running the simulation in a single GPU times the number of GPUs that is going to be used. Figure 4 shows the measured GSamples values for the same simulations. The discrepancy between the theoretical and measured GSample values occurs because theoretical model does not counts the communication time. But, as one can see in Figure 4 the simulations scales linearly with the number of processes being used.

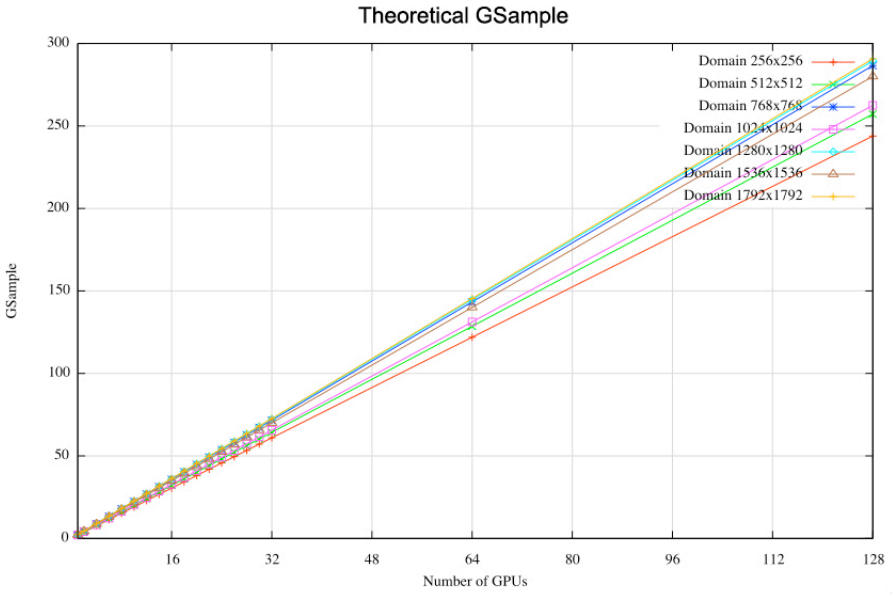


Figure 3. Theoretical GSample result for various domain sizes.

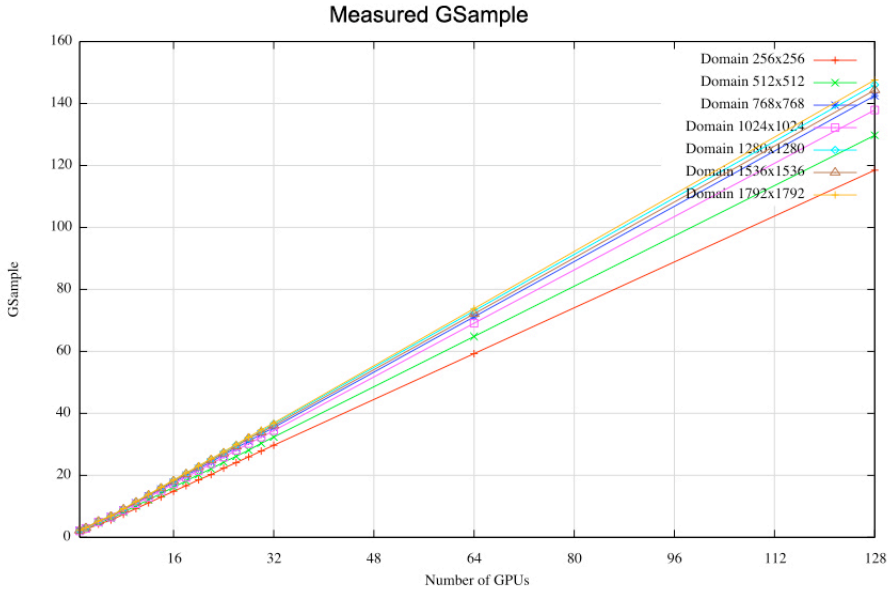


Figure 4. Measured GSample result for various domain sizes.

6. Conclusion and Future Works

Even GPUs being powerful arithmetic high-parallel multithreaded processors, its hardware memory and programming languages are limited. Due to such limitations, strategies must be designed to solve large-scale problems that may require massive storage capacity.

We present a communication scheme for 3D simulation of scattering of acoustic wave equation using a finite differences method that is capable of deal with industry scale problems that may not fit in a single GPU. We show that our implementation scales linearly with the number of GPUs being employed minus the communication time generated by MPI.

As a forthcoming work we propose the use of OpenCL with MPI in order to scale the application in a heterogeneous environment and a study on the impact of the discretization order in the final distributed execution time.

7. Acknowledments

The authors gratefully acknowledge Petrobás, CNPq, CAPES and FAPERJ for the financial support of this work.

References

Abbas-Turki, L. A. and Lapeyre, B. (2009). American options pricing on multi-core graphic cards. *International Conference on Business Intelligence and Financial Engineering*, 0:307–311.

Balevic, A., Rockstroh, L., Tausendfreund, A., Patzelt, S., Goch, G., and Simon, S. (2008). Accelerating simulations of light scattering based on finite-difference time-domain method with general purpose gpus. *Computational Science and Engineering, IEEE International Conference on*, 0:327–334.

Bolz, J., Farmer, I., Grinspun, E., and Schroder, P. (2003). Sparse matrix solvers on the gpu: conjugate gradients and multigrid. In *ACM Transactions on Graphics: Proceedings of ACM SIGGRAPH*, pages 917–924.

Brandão, D., Zamith, M., Clua, E., Montenegro, A., Bulcão, A., Madeira, D., Kischinhevsky, M., and Leal-Toledo, R. (2010). Performance evaluation of optimized implementations of finite difference method for wave propagation problems on gpu architecture. In *Computer Architecture and High Performance Computing Workshops (SBAC-PADW), 2010 22nd International Symposium on*, pages 7 –12.

Langdon, W. B. and W.Banzhaf (2008). A simd interpreter for genetic programming on gpu graphics cards. In *Lecture Notes in Computer Science: Genetic Programming*, pages 73–85. Springer Berlin-Heidelberg.

Michea, D. and D.Komatitsch (2010). Accelerating a three-dimensional finite-difference wave propagation code using gpu graphics cards. *Geophysical Journal International*, 182:389–402.

NVIDIA (2010). *NVDIA - CUDA Programming Guide*. NVIDIA.

Rozen, T., Boryczko, K., and Alda, W. (2008). A gpu-based method for approximate real-time fluid flow simulation. *Machine Graphics and Vision International Journal*, 17(3):267–278.

Zamith, M. P. M., Brandão, D. N., Kischinhevsky, M., Leal-Toledo, R. C. P., Filho, O. T. S., Montenegro, A. A., and Bulcão, A. (2010). Simulation of wave propagation in semi-infinite domains using the finite difference method on a gpu based on cluster. pages 7147–7157.

Uma avaliação da Distribuição de Atividades Estática e Dinâmica em Ambientes Paralelos usando o Hydra*

Vítor Silva[1], Fernando Chirigati[1], Eduardo Ogasawara[1, 2], Jonas Dias[1], Daniel de Oliveira[1], Fábio Porto[3], Patrick Valduriez[4], Marta Mattoso[1]

[1] Programa de Engenharia de Sistemas e Computação – COPPE / UFRJ

[2] Centro Federal de Educação Tecnológica Celso Suckow da Fonseca – CEFET/RJ

[3] LNCC – Laboratório Nacional de Computação Científica

[4] INRIA & LIRMM

{silva,fernando_seabra,ogasawara,jonasdias,danielc,marta}@cos.ufrj.br,
fporto@lncc.br, Patrick.Valduriez@inria.fr

Abstract. *Scientific Workflows are used as a basic tool to design and execute scientific experiments on different computational environments. These workflows can be computational and data intensive, requiring high performance computing. The way in which workflow activities are parallelized and distributed over these environments affects the overall performance of the workflow. This work evaluates two different strategies for activity distribution (static and dynamic) using Hydra middleware integrated with VisTrails. Our experiments show that using the right strategy decreases elapsed time for activity distribution in 30%.*

Resumo. *Workflows Científicos são usados como uma abstração básica para estruturação e execução de experimentos científicos em diferentes ambientes computacionais. Estes workflows podem ser intensivos tanto computacionalmente quanto na manipulação de dados, exigindo computação de alto desempenho. A paralelização e distribuição adequada de atividades nestes ambientes influenciam o desempenho do workflow científico como um todo. Este trabalho avalia duas estratégias de distribuição (estática e dinâmica) de atividades utilizando o middleware Hydra integrado ao VisTrails. Nossos experimentos observaram que usar a estratégia adequada diminui a duração para distribuição de atividades em 30%.*

1. Introdução

Muitos experimentos científicos são baseados em simulações computacionais complexas que consomem e produzem conjuntos de dados extensos e utilizam grandes quantidades de recursos computacionais. Conforme a complexidade do experimento cresce, executar tais simulações torna-se um desafio. Portanto, é comum a utilização de workflows científicos com o objetivo de auxiliar os cientistas na gerência de recursos envolvidos em simulações computacionais de larga escala (Mattoso et al. 2010). Um workflow pode ser definido como um modelo de um processo, que consiste em uma série de atividades e suas dependências (Aalst and Hee 2002). Um workflow científico estrutura o processamento de uma simulação científica como um grafo de atividades, nos quais os nós correspondem a atividades que processam dados e as arestas representam os fluxos de dados entre eles. Atividades de workflow estão associadas a programas científicos que preparam, processam e analisam dados.

* Esse trabalho foi parcialmente financiado pela CAPES, CNPq e INRIA (projeto SARAVA, equipe associada).

Sistemas de Gerência de Workflows Científicos (SGWfC) (Deelman et al. 2009) são *softwares* que apoiam a definição, execução e monitoramento de workflows científicos. Existem diversos SGWfC, tais como VisTrails (Callahan et al. 2006) e Swift (Zhao et al. 2007). Cada um deles tem sua própria linguagem (Deelman et al. 2009) e foca em diferentes aspectos, tais como execução paralela, suporte semântico, características específicas de domínio e proveniência dos dados.

Em um workflow científico, uma mesma atividade pode ser executada repetidas vezes variando-se os valores de seus parâmetros. Assim, a distribuição de atividades de um workflow na execução paralela se apresenta como uma característica importante para os SGWfC em experimentos de longa duração. Essa distribuição em ambientes de Processamento de Alto Desempenho (PAD) é essencial para obter os resultados de experimentos em larga escala em um tempo hábil. Existem soluções que visam a apoiar essa distribuição, tais como o Swift, o MapReduce (Wang et al. 2009) e o Hydra (Ogasawara et al. 2009).

Nas soluções de distribuição de atividades, um fator que precisa ser levado em consideração é a estratégia adotada. Diante da grande quantidade de valores de parâmetros que precisam ser processados pelas atividades envolvidas na execução de um workflow científico, a estratégia de distribuição pode afetar o desempenho da solução. Existem basicamente dois extremos no que diz respeito a estratégias de distribuição. A primeira estratégia é a distribuição estática, onde cada *core* recebe o conjunto de instâncias de atividades que ele deve processar antes de começar a execução. A segunda estratégia é a distribuição dinâmica, onde cada *core* requisita uma nova instância de atividade conforme sua disponibilidade. Assim, o objetivo deste trabalho é avaliar a variação de desempenho na utilização dessas duas estratégias.

A avaliação foi realizada utilizando o VisTrails com o Hydra. O Hydra foi escolhido por apoiar diferentes estratégias de distribuição e poder ser utilizado em diversos SGWfC. O Hydra fornece um conjunto de componentes que podem ser incluídos na especificação do workflow de um SGWfC para controlar a distribuição de atividades em ambientes de PAD. O Hydra também permite a configuração da estratégia de distribuição e a coleta de proveniência distribuída.

Este artigo está organizado em cinco seções, além desta introdução. A Seção 2 apresenta a arquitetura do middleware Hydra. Os métodos de distribuição de atividades em ambientes de PAD são discutidos na Seção 3. A Seção 4 apresenta os resultados experimentais. Os trabalhos correlatos são detalhados na Seção 5 e a Seção 6 conclui o artigo.

2. Middleware Hydra

O Hydra (Ogasawara et al. 2009) é um middleware capaz de distribuir, controlar e monitorar a execução paralela das atividades de um workflow científico em ambientes de PAD. Ele é desenvolvido em Java 1.6 e utiliza MPJ (http://mpj-express.org) para a comunicação paralela. Para o armazenamento de dados de proveniência, o PostgreSQL 8.4 (http://www.postgresql.org/) é utilizado.

O Hydra oferece ao cientista a distribuição automática de uma determinada atividade em ambientes de PAD, através de poucas alterações na definição do workflow. Pode-se citar como características positivas do Hydra: (i) não associação a uma máquina de execução do SGWfC específica; (ii) simplificação do processo de distribuição de atividades em PAD; e (iii) coleta de proveniência distribuída.

Um dos tipos de paralelismo suportado pelo Hydra é a varredura de parâmetros (Walker and Guiang 2007). Neste modelo de paralelismo, os cientistas exploram o comportamento do seu modelo computacional com diferentes entradas, configuradas no workflow. A varredura de parâmetros é muito comum e pode representar uma barreira para os cientistas controlarem e monitorarem a execução do experimento, sendo então importante fornecer apoio para tal paralelismo. Para explicar a varredura de parâmetros, apresentaremos uma formalização de workflow científico adaptada de Meyer et al. (2007).

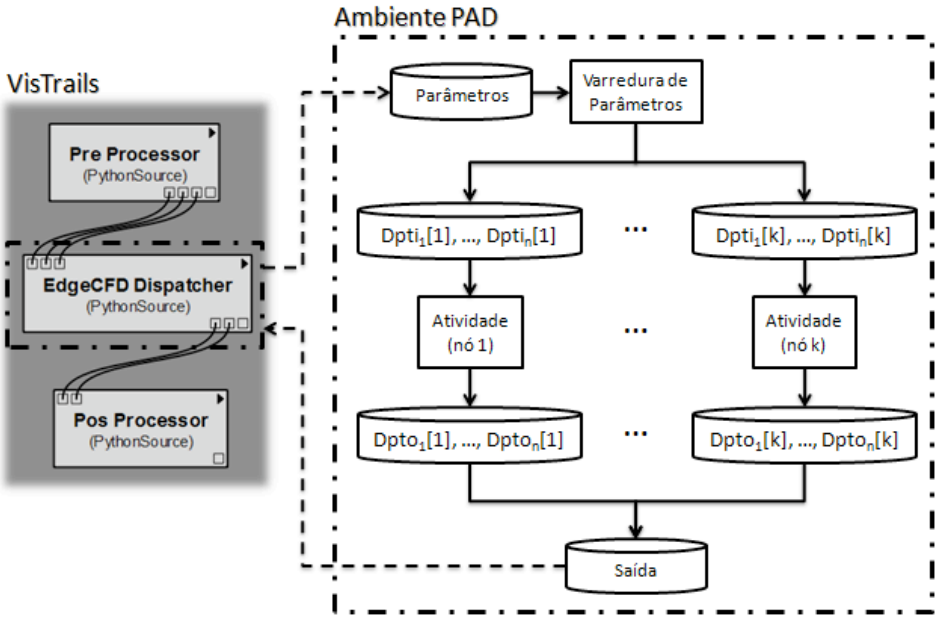


Figura 1. Varredura de parâmetros realizada pelo Hydra

Um workflow científico *Wf* é representado por uma tripla (*A*, *PtI*, *PtO*), onde: *A* é um conjunto $\{a_1, a_2, ..., a_n\}$ de atividades que constituem *Wf*; *PtI* é o conjunto $\{pti_1, pti_2, ..., pti_m\}$ de parâmetros de entrada de *Wf*; e *PtO* é o conjunto $\{pto_1, pto_2, ..., pto_r\}$ de parâmetros de saída de *Wf*. Cada parâmetro em *PtI* e em *PtO* pode assumir valores no domínio associado a ele. Assim, $Dpti_j$ representa o domínio de possíveis valores para o parâmetro pti_j, enquanto que $Dpto_j$ representa o domínio de possíveis valores para o parâmetro pto_j.

A varredura de parâmetros efetuada pelo Hydra é caracterizada por diversas e simultâneas execuções paralelas de uma atividade a_j do workflow, em que cada execução usa diferentes valores para os parâmetros de *PtI*. Por exemplo, suponha que a atividade a_j do workflow *Wf* possua parâmetros de entrada pti_1 e pti_2, e que os domínios de possíveis valores para esses parâmetros sejam $Dpti_1 = \{x, x'\}$ e $Dpti_2 = \{y, y'\}$. Uma instância da atividade a_j de *Wf* pode consumir os valores *x* e *y* para os parâmetros pti_1 e pti_2, respectivamente, enquanto outra instância da mesma atividade pode consumir os valores *x'* e *y'*. O paralelismo para essa varredura de parâmetros é realizada alocando-se diferentes configurações de valores para os parâmetros de *PtI*, em que cada configuração está associada a um diferente *core* do ambiente de PAD. A Figura 1 apresenta o esquema desse paralelismo aplicado a um workflow científico desenvolvido no SGWfC VisTrails, em que a atividade a_j é representada pela atividade EdgeCFD Dispatcher, onde a mesma realiza a varredura de parâmetros.

3. Distribuição estática e distribuição dinâmica no Hydra

Dado um determinado cenário de paralelização de uma atividade por meio de varredura de parâmetros, pode-se estabelecer um conjunto de instâncias desta atividade. Uma instância da atividade é uma execução da atividade que consome uma determinada combinação de valores de parâmetros. As instâncias podem ser distribuídas dentre os *cores* computacionais que irão processá-las num ambiente de PAD, tal como um cluster. No Hydra, o conjunto de instâncias da atividade paralelizada pode ser distribuído seguindo as estratégias de distribuição estática e dinâmica.

Na estratégia estática, as instâncias da atividade de um dado workflow são pré-alocadas para cada *core* antes de serem consumidas. Ou seja, a distribuição do número de instâncias por *core* é feita *a priori*. A Figura 2(a) representa um esquema da distribuição estática de atividades. Nesta figura, cada instância da atividade A (as instâncias estão representadas pelos quadrados *a*) é referente a um mesmo programa com um conjunto diferente de valores de parâmetros. No início do processo paralelo, o escalonador atribui para cada *core* um número fixo de quatro instâncias da atividade A para serem processadas. O escalonador apresentado na figura é o componente do Hydra responsável pela distribuição. Na estratégia de distribuição estática, o escalonador do Hydra utiliza a abordagem *gather* e *scatter* do MPJ para transmitir as tarefas aos *cores*.

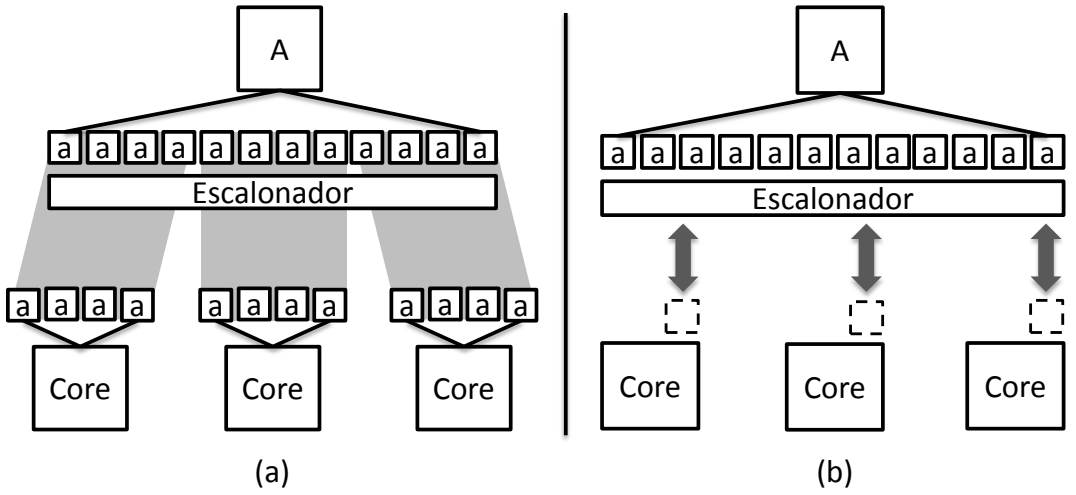
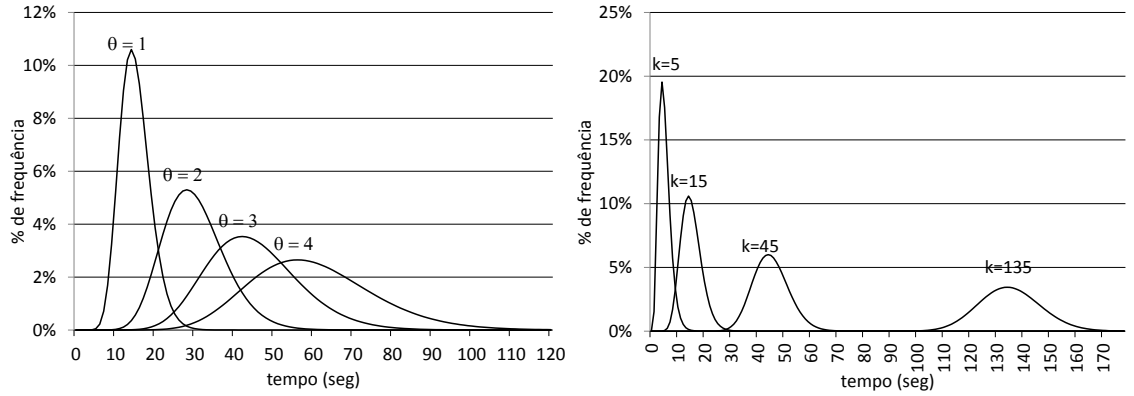


Figura 2. Estratégias de distribuição estática (a) e dinâmica (b)

Por outro lado, na estratégia dinâmica, cada instância da atividade é alocada para um *core* conforme este esteja disponível e requisite uma instância para processar. Ou seja, as instâncias da atividade permanecem em uma fila única e cada núcleo obtém e processa apenas uma instância da atividade por vez. Na Figura 2(b) é apresentado um esquema da distribuição dinâmica, onde cada *core* requisita ao escalonador uma instância *a* da atividade A por vez. Neste caso, o escalonador do Hydra precisa manter uma *thread* que aguarda requisições feitas pelos *cores* através de primitivas de comunicação do tipo *send* e *receive* do MPJ.

4. Avaliação Experimental

Nesta seção, apresentamos uma comparação de desempenho entre as distribuições estática e dinâmica de atividades em cenários de varredura de parâmetros. Os estudos foram conduzidos utilizando o Hydra como middleware de distribuição de instâncias de atividades em um cluster SGI Altix ICE 8200 de 256 cores (32 nós, cada nó contendo dois processadores quad-core Xeon).



(a) (b)

Figura 3. Duração de uma atividade seguindo uma distribuição Gamma com k = 15 fixo e θ variando de 1 a 4 (a); Duração de uma atividade seguindo uma distribuição Gamma, com θ = 1 fixo e k variando em 5, 15, 45 e 135 (b).

Inicialmente, realizamos um estudo em que as atividades possuíam duração fixa de um segundo. Constatou-se que a estratégia estática obteve um desempenho superior em aproximadamente 30%, quando comparada à estratégia dinâmica.

Em outro estudo, utilizamos dados sintéticos para avaliar atividades de duração variada. Consideramos oito tipos de atividades diferentes, cada uma delas seguindo uma distribuição Gamma (Freedman et al. 2007) específica, conforme apresentado nos cenários (a) e (b), da Figura 3. A distribuição Gamma, representada por $\Gamma(k, \theta)$ nesses cenários, tem k como parâmetro de forma e θ como parâmetro de escala ($k, \theta > 0$). Cabe ressaltar que a distribuição Gamma é adequada, por frequentemente ser utilizada como modelo de probabilidade para tempo de espera (Weisstein 2011). No cenário (a), temos quatro tipos de atividades, todas com o mesmo k fixo de 15 e cada uma delas com um θ diferente (1, 2, 3 e 4). Já no cenário (b), temos quatro tipos de atividades, todas com o θ fixado em 1 e cada uma delas com um k diferente (5, 15, 45 e 135).

O estudo mediu a duração da distribuição das atividades para cada tipo de atividade, ou seja, o tempo necessário para processar todas as instâncias de um determinado tipo. Em cada uma das medições foi realizada uma varredura de parâmetros de 1024 combinações de valores de parâmetros numa distribuição de 64 cores. A Figura 4 apresenta os resultados para os dois cenários mostrando o desempenho das distribuições estática e dinâmica. A estratégia dinâmica mostrou-se superior em todas estas situações, especialmente onde a variância da distribuição é maior, *i.e.*, o valor de θ é maior. No cenário (b), de variância fixa (θ fixo), observamos que ambas as estratégias escalam de maneira muito similar, apesar de a estratégia dinâmica apresentar melhores resultados.

Para se tentar justificar esta diferença, fizemos uma segunda análise verificando a porcentagem de recursos ociosos durante a execução. A Figura 5(a) apresenta os resultados de ambas as estratégias. Os resultados explicitam a razão da superioridade da estratégia dinâmica. Ela minimiza a ociosidade dos recursos, pois funciona com um mecanismo de fila única enquanto a estratégia estática funciona com o mecanismo de uma fila para cada *core*, o que pode causar um desbalanceamento. Com menos ociosidade, a duração da execução paralela tende a ser menor.

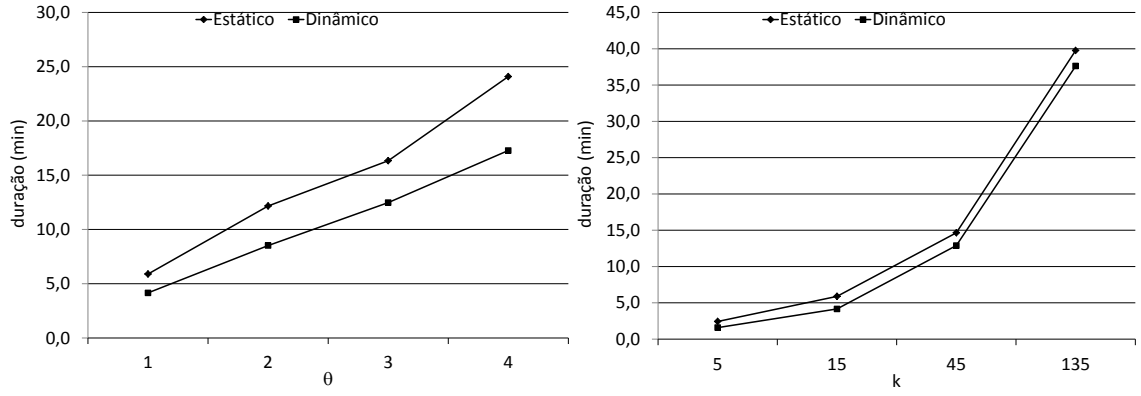


Figura 4. Duração de distribuição de atividades segundo distribuição Gamma com k = 15 e θ variando de 1 a 4 (a); Duração de distribuição de atividades segundo distribuição Gamma com θ = 1 fixo e k variando (5, 15, 45 e 135) (b).

Uma última análise apresentada na Figura 5(b) mostra a melhora do desempenho com o aumento do número de recursos. Pode-se observar que o comportamento da estratégia dinâmica permanece melhor, embora a diferença de desempenho para a estratégia estática diminua com o aumento da quantidade de recursos. Entretanto, essa diferença de desempenho também está associada à quantidade de instâncias de atividades para serem consumidas.

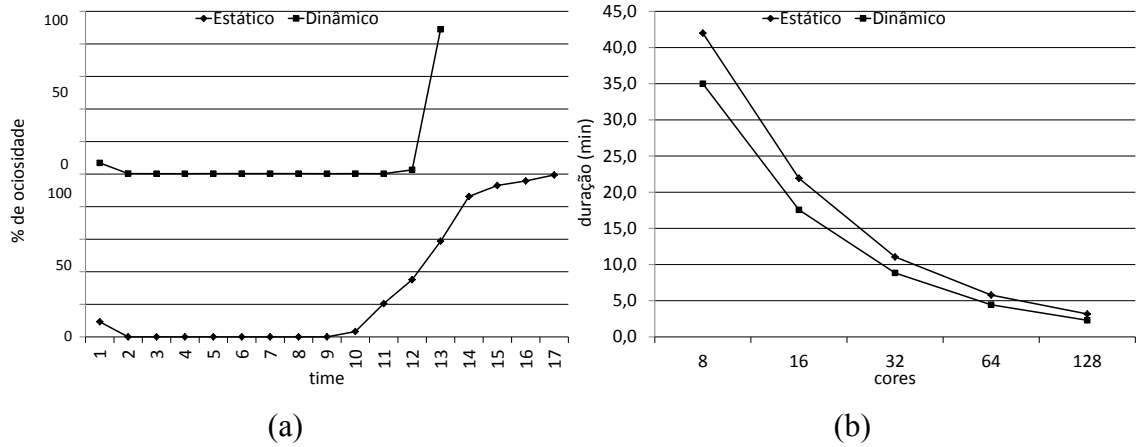


Figura 5. % de ociosidade de cores para distribuição dinâmica e estática (a); Comparação de desempenho variando-se o número de core (b).

5. Trabalhos Relacionados

As soluções atuais voltadas ao paralelismo em workflows científicos, como o SGWfC Swift, são capazes de executar atividades paralelas em diferentes ambientes computacionais como clusters, grades computacionais e nuvens. Em geral, tais soluções podem ser acopladas a escalonadores existentes como o Condor (Thain et al. 2002). Embora o Condor seja capaz de distribuir as atividades de maneira dinâmica dentre os *cores*, ele demora cerca de meio segundo para disparar cada atividade (Raicu et al. 2008). Em um cenário com centenas ou milhares de instâncias de atividade, essa taxa de distribuição é indesejada. Além disso, a política de filas do Condor pode limitar o número de atividades submetidas por usuário. Mais especificamente no caso do Swift, o mesmo pode ser também acoplado ao Falkon (Raicu et al. 2007). O Falkon é um

escalonador que pode ser usado em substituição ao Condor, mas a sua estratégia de distribuição adotada não é divulgada pela equipe de desenvolvimento.

O Nimrod/K é um conjunto de complementos para o SGWfC Kepler cujo objetivo é orquestrar a execução do workflow realizando a varredura de parâmetros em clusters, grades e nuvens. Entretanto, o Nimrod/K só funciona com o Kepler e limita-se a apoiar a varredura de parâmetros por meio de distribuição estática das instâncias de atividades do workflow. Uma solução baseada no paradigma MapReduce, como o Hadoop (http://hadoop.apache.org), integrada a um SGWfC tradicional, como o Kepler, também pode realizar a distribuição de atividades de workflow. O Hadoop é capaz de mapear diferentes execuções de atividades do workflow em ambientes distribuídos como clusters, grades e nuvens. Após a execução, os resultados são agrupados. Existe integração do Kepler com o Hadoop, mas o workflow fica dependente do SGWfC e da capacidade do cientista em programar as funções de Map e Reduce no Kepler. O Hadoop também realiza somente a distribuição estática das instâncias da atividade paralela do workflow.

6. Conclusão

Experimentos científicos em larga escala são caracterizados por cálculos computacionais complexos envolvendo simulações. A utilização de workflows científicos para a concepção, execução e controle destes experimentos é muito comum e a longa duração das atividades dos workflows implica na necessidade de utilizar a computação paralela em ambientes de PAD, a fim de diminuir o tempo de execução dos workflows científicos. A varredura de parâmetros de uma determinada atividade do workflow pode ser paralelizada de modo a se obter os resultados em tempo hábil. Cada instância da atividade pode ser alocada para executar em paralelo em diferentes *cores*. Porém, a estratégia de distribuição pode influenciar o desempenho total da execução. As soluções atuais para processamento paralelo em SGWfC usam escalonadores, como o Condor, para distribuir uma determinada atividade. Nesses escalonadores, porém, não é possível configurar a estratégia de distribuição a ser utilizada. Dessa forma, este trabalho apresenta um estudo comparativo das estratégias estática e dinâmica utilizando o Hydra e o SGWfC VisTrails, com o intuito de apoiar a tomada de decisões em experimentos que demandem paralelismo em workflows.

O estudo mostrou que a estratégia estática apresenta desempenho superior quando a atividade distribuída possui uma curta duração. Em outros casos, porém, a estratégia dinâmica apresenta um desempenho superior, pois ela minimiza a ociosidade dos *cores*. A abordagem dinâmica também é menos sensível a variância do tempo de execução da atividade. Essa superioridade ocorre em função do mecanismo de fila única utilizado pela estratégia dinâmica que evita o desbalanceamento presente no mecanismo de filas múltiplas na distribuição estática. Mesmo com o aumento do número de *cores* presentes na execução, a estratégia dinâmica manteve o melhor desempenho. Esses resultados apontam a importância na escolha da estratégia de distribuição, embora a maioria das soluções atuais como o Swift, Nimrod/K e o Hadoop desconsiderem tal opção.

Agradecimentos

Os autores agradecem ao NACAD/UFRJ pelo uso da infraestrutura computacional.

Referências

Aalst, W. van der, Hee, K. van, (2002), *Workflow Management: Models, Methods, and Systems*. 1 ed. The MIT Press.

Callahan, S. P., Freire, J., Santos, E., Scheidegger, C. E., Silva, C. T., Vo, H. T., (2006), "VisTrails: visualization meets data management". In: *Proc. SIGMOD*, p. 745-747, Chicago, Illinois, USA.

Deelman, E., Gannon, D., Shields, M., Taylor, I., (2009), "Workflows and e-Science: An overview of workflow system features and capabilities", *Future Generation Computer Systems*, v. 25, n. 5, p. 528-540.

Freedman, D., Pisani, R., Purves, R., (2007), *Statistics, 4th Edition*. 4 ed. W. W. Norton.

Mattoso, M., Werner, C., Travassos, G. H., Braganholo, V., Murta, L., Ogasawara, E., Oliveira, D., Cruz, S. M. S. da, Martinho, W., (2010), "Towards Supporting the Life Cycle of Large Scale Scientific Experiments", *International Journal of Business Process Integration and Management*, v. 5, n. 1, p. 79–92.

Meyer, L., Scheftner, D., Vöckler, J., Mattoso, M., Wilde, M., Foster, I., (2007), "An Opportunistic Algorithm for Scheduling Workflows on Grids", *VECPAR 2006*, 1 ed, p. 1-12.

Ogasawara, E., Oliveira, D., Chirigati, F., Barbosa, C. E., Elias, R., Braganholo, V., Coutinho, A., Mattoso, M., (2009), "Exploring many task computing in scientific workflows". In: *MTAGS 09*, p. 1-10, Portland, Oregon.

Raicu, I., Foster, I. T., Yong Zhao, (2008), "Many-task computing for grids and supercomputers". In: *Workshop on Many-Task Computing on Grids and Supercomputers*, p. 1-11, Austin, Texas.

Raicu, I., Zhao, Y., Dumitrescu, C., Foster, I., Wilde, M., (2007), "Falkon: a Fast and Light-weight tasK executiON framework". In: *SC07*, p. 1-12, Reno, Nevada.

Thain, D., Tannenbaum, T., Livny, M., (2002), "Condor and the Grid", *Grid Computing: Making the Global Infrastructure a Reality. John Wiley & Sons Inc*

Walker, E., Guiang, C., (2007), "Challenges in executing large parameter sweep studies across widely distributed computing environments". In: *Workshop on Challenges of large applications in distributed environments*, p. 11-18, Monterey, California, USA.

Wang, J., Crawl, D., Altintas, I., (2009), "Kepler + Hadoop: a general architecture facilitating data-intensive applications in scientific workflow systems". In: *Proc. of the 4th Workshop on Workflows in Support of Large-Scale Science*, p. 1-8, Portland, Oregon.

Weisstein, E. W., (2011), *Gamma Distribution*, Text, http://mathworld.wolfram.com/GammaDistribution.html.

Zhao, Y., Hategan, M., Clifford, B., Foster, I., von Laszewski, G., Nefedova, V., Raicu, I., Stef-Praun, T., Wilde, M., (2007), "Swift: Fast, Reliable, Loosely Coupled Parallel Computation". In: *Proc. of the 3rd IEEE World Congress on Services*, p. 206, 199, Salt Lake City, USA.

eScience-as-a-Service: Desafios e Oportunidades para a Criação de Nuvens Científicas

Rostand Costa[1, 2], Francisco Vilar Brasileiro[2], Guido Lemos de Souza Filho[1], Dênio Mariz Sousa[1]

[1] Laboratório de Aplicações em Vídeo Digital – LAVID
Universidade Federal da Paraíba (UFPB) - João Pessoa, PB – Brasil

[2] Laboratório de Sistemas Distribuídos – LSD
Universidade Federal de Campina Grande (UFCG) - Campina Grande, PB - Brasil

rostand@lavid.ufpb.br, fubica@dsc.ufcg.edu.br,
{guido, denio}@lavid.ufpb.br

Abstract. *The cloud computing approach is based on the provision of computing resources through higher levels of abstraction. The results of some pioneering initiatives indicate that the possibility of dynamic provisioning of computing infrastructure has many advantages for e-science, but its effective use and control by the researcher still presents some challenges. This paper examines some of these challenges and points out ways for effective provision of infrastructure for e-science in the form a service. Among them, special attention is given to the creation of different layers of abstractions atop the underlying infrastructure that really meet the specific requirements of scientific computing and construction of experimental scientific clouds as a catalyst in the cultural adaptation and understanding about the potential of the clouds for e-science.*

Resumo. *A abordagem de computação na nuvem (cloud computing) baseia-se na oferta de recursos computacionais através de níveis de abstração mais altos. Os resultados de algumas iniciativas pioneiras indicam que a possibilidade de provisionamento dinâmico da infraestrutura computacional tem inúmeras vantagens para e-science, mas a sua efetiva utilização e controle por parte do pesquisador ainda apresenta alguns desafios. Este trabalho analisa alguns desses desafios e aponta caminhos para a efetiva oferta de infraestrutura para e-science na forma de um serviço. Dentre eles, atenção especial é dada à criação de diferentes abstrações sobre a camada de infraestrutura que realmente atendam aos requisitos específicos de computação científica e a construção de nuvens científicas experimentais como um catalisador do aculturamento e entendimento sobre o potencial das nuvens para e-science.*

1. Introdução

É reconhecido que muitos dos avanços recentes em pesquisas científicas somente foram possíveis devido à habilidade dos cientistas em usar eficientemente computadores para gerar e processar grandes quantidades de dados.

Ao traduzir infraestrutura de Tecnologia da Informação (TI) em serviços elásticos e ilimitados, utilizados sob demanda e pagos de acordo com a quantidade de serviço consumida, o paradigma de computação na nuvem (do inglês *cloud computing*) oferece inúmeras possibilidades novas para o planejamento de capacidade das instituições que utilizam TI de forma intensiva. A comunidade científica não está indiferente a este fenômeno e inúmeras iniciativas em todo o mundo já investigam a aplicabilidade do novo ambiente para computação científica (*e-science*) [Evangelinos 2008][Juve 2009][Keahey 2010][Oliveira 2011][Osterman 2008][Walker 2008].

Os resultados preliminares indicam que embora sejam ambientes muito flexíveis e simples de se configurar, ativar *e-science* nas nuvens não é tão automático, considerando-se as implementações disponíveis, e a sua efetiva utilização e controle por parte do pesquisador ainda apresenta alguns desafios [Lee 2010]. Dentro desta perspectiva, é possível encontrar algumas situações peculiares e, até mesmo, conflitantes na migração de *e-science* para nuvens.

As principais dificuldades de endereçamento das necessidades da comunidade científica residem principalmente no fato de que as propriedades mais atrativas do paradigma de *cloud computing* para o contexto acadêmico: conveniência e facilidade de uso, ampla elasticidade e níveis adequados de abstração, ainda não são plenamente endereçados pelo estado-da-prática do segmento.

Este trabalho analisa alguns desses obstáculos e aponta caminhos para a efetiva oferta de infraestrutura para *e-science* na forma de serviço. Dentre eles, o provisionamento de nuvens usando recursos amortizados ou não convencionais, a criação de camadas de abstração sobre *infrastructure-as-a-service* (*IaaS*) que realmente atendam aos requisitos específicos de computação científica e a construção de nuvens científicas experimentais como um catalisador do aculturamento e entendimento sobre o potencial das nuvens para *e-science*.

O restante do documento está organizado como segue. A Seção 2 analisa a limitação na alocação de recursos imposta pelos provedores de nuvens atuais. A Seção 3 discute alguns dos desafios culturais e de usabilidade para fazer *e-science* em nuvens. São discutidos algumas ações e caminhos que possam conduzir à oferta de nuvens específicas para *e-science* na Seção 4. A Seção 5 contém as nossas considerações finais.

2. Melhorias na Amplitude da Elasticidade

A abordagem de computação na nuvem baseia-se na oferta de recursos computacionais através de altos níveis de abstração e na forma de serviços que os clientes adquirem sob demanda e pagam apenas pela quantidade de recursos que realmente consomem. Isso dá aos clientes a possibilidade de aumentar e diminuir a capacidade da sua infraestrutura à vontade, sem incorrer em um custo adicional. A elasticidade ilimitada teórica que é associada aos serviços oferecidos por provedores de *clouds* deveria, potencialmente, permitir aos usuários decidir livremente se desejam usar 1 recurso computacional (ex. uma máquina virtual com uma determinada configuração) por 1.000 horas ou 1.000 recursos por 1 hora e pagar o mesmo preço em ambos os casos. Essa propriedade singular de computação na nuvem é chamada de *associatividade de custos* [Fox 2011].

Tal capacidade de instanciar concomitantemente um grande número de recursos por um período de tempo relativamente curto é um requisito fundamental para um modelo de programação de aplicações paralelas chamado computação de alta vazão (HTC, do inglês *High-Throughput Computing*) [Litzkow 1988]. Essas aplicações têm cargas de trabalho altamente paralelizáveis e quanto mais cedo a sua execução possa ser concluída, melhor. Assim, idealmente, elas poderiam ser executadas simultaneamente pela totalidade dos recursos necessários para terminar o mais rapidamente possível e, ainda, com um custo que só dependeria da carga de trabalho que tiver sido realmente processada. Desta forma, muitas aplicações HTC, científicas ou comerciais, poderiam potencialmente obter um enorme benefício a partir da elasticidade dos fornecedores de computação na nuvem.

Infelizmente, os provedores públicos atuais de IaaS precisam limitar o número máximo de instâncias que podem ser adquiridas simultaneamente por um dado cliente e permitem somente que poucas máquinas virtuais sejam instanciadas automaticamente. Por exemplo, o serviço EC2 da Amazon Web Services (aws.amazon.com) um dos mais

importantes provedores de IaaS em operação, permite que apenas 20 instâncias possam ser iniciadas concomitantemente pelo mesmo cliente. Para alocar grandes quantidades é necessária uma negociação paralela. Embora, na prática, este limite possa ser visto como praticamente "infinito" pela maioria das aplicações, isso não é o caso para muitas aplicações HTC e, possivelmente, para outras classes de aplicações.

Dentre as razões pelas quais os fornecedores tradicionais necessitam impor um limite na elasticidade, a principal é que quanto maior é o limite, maior é a capacidade da infraestrutura que os fornecedores precisam manter e, considerando uma taxa fixa de ociosidade, menor será a sua rentabilidade [Costa 2011]. Assim, os provedores públicos atuais de IaaS precisam limitar a quantidade de recursos para que possam garantir uma disponibilidade suficientemente elevada para seus serviços e, ao mesmo tempo, manter os seus lucros em um nível aceitável.

Como já existem serviços de alta demanda hospedados em provedores de IaaS públicos e privados (ex. Gmail, Twitter, Bing etc.) e também a possibilidade de se negociar alocações superiores com provedores públicos, é possível inferir que o limite serve como um regulador do uso intensivo de recursos por períodos curtos, ou seja, o alvo do limite não é o volume da requisição em si, mas o exercício extremo da elasticidade através de grandes alocações com liberações logo em seguida [Costa 2011]. Desta forma, não é fácil atingir computação de vazão extremamente alta nelas, considerando as implementações disponíveis. Em particular, nenhum dos provedores de IaaS atuais é capaz de aceitar a instanciação de um sistema com milhares de máquinas virtuais pelo curto período de tempo requerido para executar uma aplicação HTC típica.

Lidar com as demandas por escalabilidade extremamente alta de aplicações HTC ou mesmo com *flash crowds* [Jung 2002], quando um grande número de usuários acessa simultaneamente um sítio Web que adquire uma popularidade instantânea, não é uma tarefa trivial. Proporcionar tal nível de flexibilidade traz desafios enormes para o planejamento de capacidade que precisa ser realizado pelos provedores de IaaS. Para dar suporte a este tipo de utilização, esses provedores provavelmente teriam que enfrentar níveis de ociosidade de suas estruturas maiores do que os que são observados hoje, com forte impacto em sua lucratividade. O resultado desta limitação é que existe uma faixa inteira de aplicações que ainda não está sendo bem atendida pelos serviços oferecidos atualmente pelos provedores de computação na nuvem.

3. Abstração, Controle e Outros Desafios

Em uma classificação bastante ampla, notadamente quando se considera as diferenças entre as características das cargas de trabalho (*workload*), a computação científica é normalmente dividida em Computação de Alta Performance (HPC, do inglês *High Performance Computing*) e Computação de Alta Vazão (HTC). Entretanto, há algumas premissas relevantes que trafegam entre estas categorizações de forma transversal, dentre elas a busca pelo equilíbrio na relação entre controle e abstração. Tanto em HPC quanto em HTC, a simplicidade e facilidade proporcionada para os usuários por cada camada de abstração adicionada precisam ser permanentemente equilibradas com o atendimento dos requisitos de desempenho das aplicações, o que implica na necessidade de um maior controle sobre a infraestrutura computacional [Lee 2010]. Dentro dessa perspectiva, é possível encontrar algumas situações peculiares e, até mesmo, conflitantes na migração de *e-science* para nuvens.

Em ambientes de computação em grade (*grid computing*), por exemplo, a heterogeneidade dos recursos disponíveis requer o gerenciamento da compatibilidade entre sistemas operacionais, bibliotecas, executáveis e parâmetros do ambiente para a distribuição

de aplicações enquanto que um gerenciamento de configuração similar é muito mais facilitado na abordagem *infrastructure-as-a-service*. Em *IaaS*, o uso de imagens de máquinas virtuais permite empacotar conjuntamente tanto a aplicação quanto a configuração exata do ambiente necessário para a sua execução. Por outro lado, os *middleware* de *grids* normalmente são sensíveis ao contexto das aplicações científicas e oferecem inúmeras facilidades para a submissão e controle de tarefas, enquanto que boa parte do trabalho de configuração, instanciação e encerramento das instâncias de máquinas virtuais em *IaaS* deve ser feita pelo próprio usuário. Além de *grids*, vários outros modelos de execução proporcionam abstrações que facilitam o uso das infraestruturas computacionais para finalidades científicas, como *workflows*, paradigmas de programação (ex. *dataflow*, map-*reduce*, *data streaming*, *parameter sweep*), dentre outras. Estes modelos possuem *frameworks* que os tornam amigáveis e eficientes e que podem ser adaptados para ambientes de computação na nuvem, tornando-os mais adequados para computação científica. Entretanto, até o momento, poucos desses modelos possuem implementações comerciais para computação na nuvem, como o *Hadoop* (http://hadoop.apache.org) para o *Map-Reduce* (http://http://labs.google.com/papers/mapreduce.html). Coincidentemente, boa parte dos relatos atuais de experiências reais de *e-science* em nuvens estão relacionadas com o uso do *Hadoop* ou do modelo de programação *Map-Reduce* [Simalango 2010]. Outras iniciativas mais recentes como as plataformas Azure Blast [Lu 2010], de bioinformática, e Cirrus [Lu 2010], voltada para *parameter sweep*, ratificam a diversidade e singularidade dos modelos de programação necessários.

Há ainda os obstáculos relacionados com o modelo global corrente de financiamento de projetos de pesquisa que, salvo raras exceções, não contempla a contratação em provedores privados da infraestrutura computacional necessária para a realização de experimentos. Principalmente quando esta contratação envolve a transferência de recursos financeiros para outros países e continentes. Até mesmo aspectos culturais da execução de pesquisas são impactados pela novidade da valoração monetária de cada experimento ou simulação realizada - ela se torna indiretamente evidente em um contexto *pay-as-you-go* onde cada sessão de uso é bilhetada e faturada. Estamos preparados para uma investigação empírica de resultados? Quanto vale uma refutação de hipótese?

Além disso, os atuais serviços públicos de computação na nuvem foram concebidos tendo em vista o atendimento de requisitos bem específicos como, por exemplo, o suprimento de infraestrutura para pequenas e médias empresas e o suporte computacional para aplicações do estilo Web 2.0 fazendo com que alguns contextos, inclusive o de *e-science*, se ressintam de uma maior identificação com as suas reais necessidades.

4. A Necessidade de uma Agenda para *e-Science-as-a-Service* (*eSaaS*)

É fundamental evoluirmos o entendimento de como o paradigma de computação na nuvem pode ser aplicado na ciência e quais requisitos são relevantes para a construção de nuvens científicas (*e-science clouds*). As iniciativas e ações neste sentido podem ser consolidadas em uma agenda colaborativa que permita a evolução tanto do estado-da-arte quanto do estado-da-prática do segmento e conduza ao desenvolvimento de uma plataforma de computação na nuvem que seja plenamente adequada para *e-science*. Chamamos tal categoria de serviços especializados, abstraída na forma de modelos de negócio e modelos de programação próprios, de *e-Science-as-a-Service* (*eSaaS*).

4.1. Identificação e Disponibilização das Abstrações Adequadas

Já existem diversas alternativas de código aberto para suporte a IaaS como Eucalyptus (http://open.eucalyptus.com), Nimbus (http://www.nimbusproject.org), OpenNebula (http://www.opennebula.org) e OpenStack (http://openstack.org) que oferecem

funcionalidades básicas como suporte a virtualização, armazenamento distribuído de dados (*storage*) e o gerenciamento de filas assíncronas, dentre outros. Entretanto, são ainda necessários mecanismos adicionais para o seu uso mais eficiente em nuvens voltadas para o uso científico, incluindo o suporte a grupos de pesquisa, o controle de reputação e utilização de usuários e também facilidades para o intercâmbio de experiências e resultados.

Sobre a estrutura de IaaS convencional provida pelos *middleware open source* podem ser construídas camadas de abstração que facilitem o seu uso para *e-science*. Uma destas camadas pode ser composta por recursos que permitam, de forma automática, a instanciação coordenada e eficiente de múltiplas cópias de uma mesma máquina virtual. Outra camada pode ser voltada para a adição de funcionalidades, como as existentes em *grids*, de suporte à execução de grandes lotes de tarefas *bag-of-tasks* (BoT) [Cirne 2003]. Uma terceira camada pode envolver mecanismos para integração e uso mútuo de recursos entre nuvens científicas federadas e assim por diante.

Além disso, os modelos de programação comumente adotados em experimentos científicos precisam ser fatorados e incorporados às nuvens científicas como mecanismos de incremento da usabilidade e produtividade dos pesquisadores.

4.2. Construção de Nuvens Científicas Nacionais ou Voluntárias

Na nossa visão, a existência de nuvens nacionais especificamente voltadas para a ciência, mantidas e operadas por agências de fomento e órgãos federais, pode oferecer um suporte adequado e *top-down* para diversas classes de aplicações científicas, facilitando o ingresso das instituições acadêmicas no segmento de *cloud computing* e promovendo padronização, integração e otimização de investimentos. Neste momento de amadurecimento do nosso entendimento sobre as possibilidades de uso de computação na nuvem na ciência, uma plataforma aberta pode atuar como agente de disseminação e catalisador de uma evolução colaborativa no estabelecimento dos requisitos que *e-science clouds* devem atender.

Algumas iniciativas já começam a surgir buscando suprir estas lacunas através de nuvens mais temáticas e direcionadas. Como exemplo de abordagem *top-down*, o CIO do governo americano anunciou recentemente a formação de uma *National Cloud* para atender às necessidades das agências governamentais e racionalizar os investimentos governamentais na área [Lee 2010]. Em paralelo e de forma *bottom-up*, algumas instituições de pesquisa americanas também começam a montar as suas estruturas de nuvem privadas e discutir aspectos de interoperabilidade.

Em paralelo, a existência de condições permitindo que instituições acadêmicas possam criar e manter nuvens voluntárias de pequena escala oferecendo recursos computacionais no estilo IaaS para iniciação no uso científico tanto pelos seus membros quanto pela comunidade em geral pode atuar como um incentivo para que requisitos e especificidades reais possam ser identificados e endereçados. Os aspectos de padronização e interoperabilidade podem ser investigados através da federação destas nuvens voluntárias, permitindo uma abordagem alternativa (*bottom-up*) para a criação de nuvens científicas regionais ou nacionais. Mecanismos voltados a dar visibilidade à iniciativa podem ajudar a reunir e integrar uma comunidade de usuários e doadores.

4.3. Uso de Recursos Amortizados para Construção de Nuvens Públicas

A baixa amplitude da elasticidade dos provedores atuais de nuvens reflete um pequeno paradoxo. Enquanto da perspectiva do cliente, o modelo de computação na nuvem permite que este aplique aos seus investimentos em TI os mesmos princípios do *Toyota Production System* (TPS) [Toyota 2010]. Criada pela Toyota nos anos 50, a filosofia de sistema de produção "*Just in Time*" (*JiT*) é baseada em uma idéia muito simples: "*what is needed,*

when it is needed, and in the amount needed". Os provedores de IaaS, por sua vez, não possuem as mesmas facilidades quando estão montando a infraestrutura sobre as quais eles irão prover os seus serviços, tendo que lidar com a complexidade e riscos associados com o planejamento de capacidade de longa duração.

Para lidar com esta limitação, foi proposto o conceito de *Just in Time (JiT) Clouds* [Costa 2010], cujos prestadores apenas incorrem em custos de propriedade quando os recursos que eles usam para fornecer os seus serviços são demandados pelos seus clientes e apenas durante o período que eles são necessários. Isto alivia os riscos e custos do planejamento de capacidade envolvidos tanto com sub-provisionamento quanto com excesso de provisionamento de recursos. Para tal, provedores de *JiT Clouds* utilizam apenas o poder de processamento ocioso de recursos cujos custos de montagem e operação já foram amortizados pela finalidade principal do negócio que eles dão suporte.

Do ponto de vista da escala, os detentores de recursos computacionais amortizados podem ser classificados em duas categorias principais: a) os que possuem capacidade excedente suficiente para poderem atuar como provedores públicos de IaaS, oferecendo os seus recursos ociosos para terceiros, como fez a Amazon Bookstore (www.amazon.com), por exemplo, dando origem à Amazon Web Services; e b) os que não possuem, sozinhos, recursos amortizados ociosos suficientes para uma atuação solo.

A última categoria, que chamamos de recursos amortizados de pequena escala, envolve todo o espectro de escala imediatamente inferior ao nível esperado para a categoria *a)*, incluindo desde as empresas de grande porte, passando por *data centers* de pequeno porte e chegando até servidores e recursos individuais, convencionais ou não convencionais, pertencentes a instituições ou a indivíduos. Explorando tais recursos amortizados, um fornecedor de *JiT Cloud* é capaz de oferecer IaaS de uma forma altamente escalável e com uma elasticidade muito mais ampla, uma vez que é baseada na descoberta, federação e revenda de recursos ociosos já amortizados por outras finalidades. Contando com modelos alternativos de provisionamento que permitam custos menores ou irrelevantes para a disponibilidade de recursos, os provedores de *JiT Clouds* podem proporcionar aos clientes com aplicações científicas em geral e HTC em particular os benefícios da associatividade de custos: o menor tempo de processamento possível por um mesmo custo.

4.4. Promoção do Impacto de Investimentos através da Convergência de Iniciativas

Alguns caminhos alternativos e complementares para prover recursos computacionais sob demanda para experimentos científicos também podem ser explorados. A elaboração simultânea de Planos Nacionais de Banda Larga (PNBL) por diversos países além do Brasil [Faria 2010] é uma afirmação mundial de que massificar o acesso adequado à Internet por meio de políticas públicas é insumo estratégico para o desenvolvimento, de modo a promover oportunidades, desconcentrar renda e incorporar os cidadãos e instituições hoje excluídos desse serviço, reduzindo assim as desigualdades sociais e regionais. Entretanto, esta iniciativa requer ações complementares para atingir uma inclusão digital concreta como, por exemplo, a viabilização de equipamentos para os cidadãos de baixa renda que lhe permitam um uso efetivo e autônomo dos canais de comunicação subsidiados.

Abordagens recentes, como a TV Conectada [InternetTV 2011] e a TV Social [Social Television 2011], indicam que os receptores de TV Digital interativos, dotados de poder de processamento e de comunicação, podem desempenhar um papel relevante neste cenário de convergência de mídias, conteúdos e serviços. Nesta direção, o Google TV (www.google.com/TV), o Ginga AppStore (www.lavid.ufpb.br), o Sticker Center (www.stickercenter.com.br) e outras iniciativas similares ampliam o escopo de atuação dos receptores interativos, os quais, com a possibilidade de execução de aplicações

independentemente do conteúdo televisivo, passam a atuar também como verdadeiros integradores de entretenimento, comunicação e interação digital.

No Brasil, a expectativa é que o Sistema Brasileiro de TV Digital (SBTVD) atinja um número total de 15 milhões de receptores digitais até o ano de 2012. Esta influência não pode ser ignorada e a utilização dos receptores interativos de TV Digital (TVD) como agentes complementares do Plano Nacional de Banda Larga pode realmente promover uma inclusão digital efetiva, notadamente nos segmentos mais carentes e desprovidos de outros equipamentos para acesso à Internet. Para tal, é necessário estabelecer os mecanismos adequados de incentivo para a sua adoção e proliferação, incluindo subsídios para os fabricantes e parcerias com a iniciativa privada. Nesse sentido, uma possibilidade é a utilização do poder de processamento e comunicação ociosos de receptores interativos de TVD para permitir o processamento de aplicações que demandam alto poder computacional.

Na esteira destes movimentos e visando o aproveitamento do poder computacional que este enorme contingente de receptores interativos representa, abrem-se novas oportunidades técnicas, principalmente com a possibilidade de uso destes equipamentos para resolver problemas de grande escala nas áreas da ciência, engenharia, medicina, física etc. Os principais atrativos desta idéia são a possibilidade de alocar uma enorme quantidade destes recursos não convencionais para o processamento distribuído de uma aplicação paralela em uma abordagem baseada em serviços e provisionamento dinâmico de recursos. Uma arquitetura para o uso de recursos não convencionais, notadamente os organizáveis sob demanda em redes de *broadcas*t, para a construção de infraestruturas computacionais distribuídas foi proposta por Costa et al [Costa 2009] e teve os seus aspectos de segurança para o contexto de TVD discutidos por Maia et al [Maia 2010]. Infraestruturas similares também podem ser construídas sobre *smartphones*, PDAs e *tablets*.

Na nossa visão, este é um exemplo de cenário de sinergia entre governo, empresas, academia e sociedade organizada. Por exemplo, é possível integrar investimentos e benefícios governamentais em inclusão digital através do PNBL e receptores interativos de TVD com as demandas por recursos computacionais de várias classes de aplicações da comunidade científica. De forma similar, a iniciativa privada pode ser incentivada a fornecer receptores em comodato em troca da possibilidade de uso compartilhado destes recursos computacionais para as suas finalidades específicas. Finalmente, os cidadãos que adquirem os seus próprios receptores podem ser estimulados a ceder os seus equipamentos de forma voluntária e participar de esforços coletivos, científicos ou comunitários, com os quais se identifique.

5. Conclusão

Considerando um contexto onde possam convergir os três principais aspectos da onda tecnológica ("*perfect storm*") que tem impactado a humanidade nos últimos anos [Hemsoth 2011]: o massivo uso de dispositivos móveis inteligentes, a conectividade ubíqua em alta velocidade e o paradigma de computação na nuvem, este trabalho aponta algumas iniciativas que endereçam os desafios ao uso de computação na nuvem pela comunidade científica: a) criação de camadas de abstração sobre IaaS que realmente atendam aos requisitos específicos de computação científica; b) criação de nuvens científicas nacionais ou voluntárias que permitam o amadurecimento do conhecimento do potencial do paradigma para *e-science*; c) ampliação da elasticidade de provedores de computação na nuvem através do uso de recursos amortizados e d) promoção da convergência e sinergia entre iniciativas e cenários, incluindo os não convencionais, que possam alavancar o escopo e a abrangência do uso de nuvens científicas.

Estas e outras ações podem ser consolidadas em uma agenda colaborativa na busca da oferta de serviços e modelos de programação adequados para a comunidade científica em uma abordagem efetiva de *e-Science-as-a-Service* (eSaaS).

Referências

Cirne, W., Paranhos, D., Costa, L., Santos-Neto, E., Brasileiro F. et al. (2003) *Running Bag-of-Tasks Applications on Computational Grids: The MyGrid Approach*, Proc. of the ICCP'2003 - Int. Conference on Parallel Processing.

Costa, R., Brasileiro, F., Lemos, G., and Mariz, D. (2009) *OddCI: on-demand distributed computing infrastructure*. In Proceedings of the 2nd Workshop on Many-Task Computing on Grids and Supercomputers (Portland, EUA). ACM MTAGS '09.

Costa, R., Brasileiro, F., Lemos, G. and Mariz, D. (2010) *Just in Time Clouds: Enabling Highly-Elastic Public Clouds over Low Scale Amortized Resources*. Technical Report. Disponível em http://www.lsd.ufcg.edu.br.

Costa, R., Brasileiro, F., Lemos, G., Mariz, D. (2011) *Sobre a Amplitude da Elasticidade dos Provedores Atuais de Computação na Nuvem*. In-press to SBRC 2011.

Evangelinos, C., and Hill, C. (2008) *Cloud Computing for parallel Scientific HPC Applications: Feasibility of running Climate Models on Amazon's EC2*.

Faria, F. (2010) *Visão & Tecnologia: A forma para tornar as nossas cidades mais inteligentes*. Em Conferência Internacional de Cidades Inovadoras.

Fox, A. (2011) *Cloud Computing—What's in It for Me as a Scientist?*. In Science Vol. 331 no. 6016 pp. 406-407.

Hemsoth, K. (2011) *IEEE Declares War on Cloud Computing Challenges*, http://www.hpcinthecloud.com/features/IEEE-Declares-War-on-Cloud-Computing-Challenges-119208959.html

Internet Television (2011) Disponível em http://en.wikipedia.org/wiki/Internet_television.

Juve, G. et. al. (2009) *Scientific Workflow Applications on Amazon EC2*. In 5th IEEE International Conference on e-Science.

Keahey, K. (2010) *Another Barrier Goes Down*, http://sciencecloud s.org/blog/

Lee, C. A. (2010) *A Perspective on Scientific Cloud Computing*. In Proc. of the 19th ACM International Symposium on High Performance Distributed Computing, 451-459.

Litzkow, M., Livny, M. and Mutka, M. (1988) *Condor - a hunter of idle workstations*. In: Proc. 8th Int'l Conf. Distributed Computing Systems. 1988.

Lu, W., Jackson, J., Ekanayake, J., Barga, R.S., Araujo, N. (2010) *Performing Large Science Experiments on Azure: Pitfalls and Solutions*. In CloudCom(2010) 209-217.

Maia, G., Costa, R., Lemos, G., Mariz, D., Brasileiro, F. (2010) *Um Modelo de Segurança para Infraestruturas Computacionais Distribuídas Baseadas em Receptores de TV Digital*. II Workshop de TV Digital Interativa/WebMedia 2010. Belo Horizonte MG

Oliveira, D., Baião, F. and Mattoso, M. (2011) *Migração de Experimentos Científicos para a Nuvem*. SBC Horizontes, abril 2011.

Ostermann, S., Iosup, A., Yigitbasi, N. et al. (2008) *An Early Performance Analysis of Cloud Computing Services for Scientific Computing*. 2008:1-24.

Social Television (2011) Disponível em http://en.wikipedia.org/wiki/Social_television.

Simalango, M., and Sangyoon O. (2010) *Handbook of Cloud Computing*. In Compute, edited by Borko Furht and Armando Escalante, 535-551. Boston, MA: Springer US.

Toyota (2010). *Just in Time - Toyota Production System (TPS).* Disponível em http://www2.toyota.co.jp/en/vision/production_system/just.html.

Walker, E. (2008) Benchmarking Amazon EC2 for high-performance scientific computing. Login, vol 33 no. 5, Usenix.

Métricas para ontologias no formato OBO: Um estudo utilizando o Cytoscape

Ana Carolina Gama e Silva Assaife[1], Fernanda Bruno dos Santos[1], Jonice de Oliveira Sampaio[1], Maria Luiza Machado Campos[1]

[1]Programa de Pós Graduação em Informática (PPGI)
Universidade Federal do Rio de Janeiro (UFRJ)
Rio de Janeiro– RJ – Brazil

{assaife,fernanda.bruno}@ufrj.br

Abstract. *The importance of the role of ontologies is growing and also its size and complexity. Therefore, the evaluation of ontologies becomes essential because it makes possible to determine some of their fundamental characteristics that are important for designing ontologies, improving quality of existing ones, preventing and reducing the need for future maintenance. This paper aims, through literature review, some methods for evaluating ontologies and through them, suggests a set of metrics in order to relate their theoretical meaning to the meaning in the ontology through an experiment.*

Keywords: *ontology, metrics, bioinformatics, Cytoscape*

Resumo. *A importância do papel das ontologias vem crescendo e, consequentemente também, seu tamanho e complexidade. Portanto, a avaliação de ontologias se torna essencial pois torna possível a determinação de algumas de suas características fundamentais que são importantes para projetar ontologias, melhorar qualidade das já existentes, prevenir e reduzir a necessidade de manutenção futura. Este trabalho aponta, através de revisão de literatura, alguns métodos de avaliação de ontologias e através deles, sugere um conjunto de métricas com o objetivo de relacionar o seu significado teórico ao significado na ontologia através de um experimento.*

Palavras chave: *ontologia, métricas, bioinformática, Cytoscape*

1. Introdução

O rápido crescimento da Internet impulsiona o compartilhamento de informação entre os usuários. É neste contexto que a Web Semântica se expandiu tendo como principal objetivo criar uma estrutura de representação do conteúdo de páginas da Web, possibilitando atingir esse compartilhamento de forma que os computadores consigam interpretar dados e, portanto, facilitando a utilização da informação pelos usuários. Nesse contexto, as ontologias são utilizadas no compartilhamento de conhecimento e visam a apoiar a representação explícita e formal de uma conceituação sendo um mecanismo importante no suporte à interoperabilidade de sistemas.

Com o aumento da popularidade das ontologias, o desenvolvimento de atividades que utilizam essa representação de domínio cresceu e se expandiu para diversas áreas. Em especial, as comunidades internacionais da área da genômica proporcionaram esse crescimento de informações biológicas descentralizadas. A disponibilidade crescente e necessidade de gerência e compartilhamento de ontologias

favoreceram a criação do consórcio de terminologias conhecido como Open Biological and Biomedical Ontologies, OBO, (Smith et al. 2007) possibilitando assim, a troca e reuso do conhecimento pelas comunidades acadêmicas e científicas. Diante de uma grande massa de dados e, consequentemente, de um volume muito grande de informações disponíveis para os usuários, torna-se fundamental que estas informações possam estar descritas ou contextualizadas de forma a facilitar suas localizações, usos e interoperabilidades.

Com o crescente número de ontologias e expansão de seu reuso se torna importante conhecer, analisar e comparar elementos destas ontologias. Além disso, é necessário também avaliar sua estrutura para que seu entendimento seja mais completo e, portanto, melhor. Em especial, no processo de integração e mapeamento entre ontologias, uma avaliação mais acurada das características estruturais, dentre outras, constitui valioso instrumento de apoio.

Outra questão que deve ser destacada é o pouco conhecimento de ferramentas que atendam a necessidade de analisar a ontologia e seus elementos. Algumas vezes, recorre-se a uma análise manual de termos. Tendo em conta que essas tarefas demandam tempo e muito esforço, corre-se um risco de o resultado obtido não ser preciso o suficiente quando comparado com um método automatizado. Este problema pode ser causado pela falta de profissionais de Ciência da Computação capacitados em Bioinformática que buscariam automatização do processo necessário para análise.

O objetivo deste trabalho é levantar, experimentar e analisar um conjunto de ferramentas livres e de fácil manuseio que realizam análises de ontologias utilizando-se de métricas variadas que possam ser úteis aos interesses dos pesquisadores. Além disso, visa apontar uma série de métodos para avaliação de ontologias disponíveis, no domínio da biomedicina, bem como uma discussão de métricas que permitem avaliar e conhecer essas estruturas. Serão propostos, também, experimentos que buscam evidenciar o significado obtido através das métricas no contexto da ontologia selecionada, a GO Slim, utilizando a ferramenta Cytoscape e seus plugins.

2. Revisão de Literatura

Esta seção irá tratar de métodos já existentes para avaliar ontologias disponíveis na literatura com uma visão voltada para integração deles em um único *framework* com foco teórico.

De acordo com Gangemi et al. (2005), é possível identificar três tipos principais de métricas para avaliação de ontologias: métricas estruturais, que estão relacionadas com a representação da ontologia como um grafo, métricas funcionais que estão relacionadas com a utilização da ontologia e seus componentes, e por fim as métricas relacionadas a usabilidade que dependem do nível de anotação da ontologia considerada.

O trabalho de Yao et al. (2005) traz um conjunto de medidas que avaliam o grau de relacionamento entre elementos em uma ontologia, chamadas métricas de coesão. Uma ontologia tem alto grau de coesão se suas entidades são fortemente relacionadas. A utilidade dessas métricas vem da idéia de que conceitos agrupados em uma ontologia devem estar relacionados a um determinado domínio de modo a atingir objetivos semelhantes.

Os estudos de Gómez-Pérez (2003) discutem, dentre outros assuntos, avaliação de conteúdo de ontologias, que tem como principal objetivo detectar inconsistências ou redundâncias antes de utilizá-las em aplicações. Ela está relacionada com o paradigma da representação de conhecimento que possui ligação com a linguagem em que a ontologia é implementada.

A pesquisa de Noy (2004) é focada nas necessidades dos consumidores de ontologias que necessitam saber quais delas são adequadas às suas necessidades. É mencionado, também, que é importante não só um sistema para avaliar ontologias de um ponto de vista genérico mas também modos práticos para que esses consumidores possam descobrir e avaliar as ontologias em questão.

A aplicação OntoClean, que foi proposta por Welty et al. (2003), tem como objetivo detectar inconsistências formais e semânticas nas propriedades definidas por uma ontologia. Sua função principal é a avaliação formal de propriedades definidas na ontologia por meio de uma estrutura taxonômica pré definida de meta propriedades.

Através da revisão dos trabalhos citados foi selecionado um conjunto de métricas candidatas para a realização de um experimento apresentado na seção 4. Neste experimento, será possível visualizar não só o significado teórico das métricas mas o que elas significam no contexto da ontologia em questão.

3. Ferramentas e métricas para ontologias

Esta seção irá tratar das ferramentas utilizadas para realização do experimento bem como as métricas que cada uma delas trata.

De acordo com Sure e Vrandecic (2007), métricas são necessárias para se avaliar ontologias no decorrer das fases de construção e de aplicação, possibilitando uma compreensão rápida e simples a respeito do que está sendo modelado por essas estruturas, desse modo, facilitando o controle de sua futura evolução. Com o objetivo de calcular essas métricas e, consequentemente, seus significados será utilizado o Cytoscape, um software de bioinformática de código aberto e gratuito usado para visualizar interações de redes moleculares (Shannon et al. 2003).

A ontologia GO Slim será utilizada para realizar o estudo. Ela é uma versão editada da ontologia GO (Ashburner et al. 2000), também chamada de *GO Full*., e contém um subconjunto de termos pertencentes a GO completa. Sendo assim, ela contuma visão de alto nível dos processos biológicos, funções moleculares e/ou localizações celulares sem todo o detalhamento contido na GO completa. Vale lembrar que ela pertence a Gene Ontology que faz parte do Consórcio OBO.

Para está análise, serão utilizados três plugins do Cytoscape para realizar o cálculo das métricas e a visualização da ontologia como um grafo. O primeiro deles é o ClusterViz que é um *plugin* desenvolvido para visualização e análise de *clusters* de uma rede.Três importantes algoritmos de criação de grafos de *clusters* que são FAG-EC, EAGLE e MCODE foram implementados no *plugin*. Para esse estudo, foi utilizado o algoritmo MCODE, visando encontrar regiões fortemente conectadas em ontologias. A partir do resultado da análise de *clusters* nas ontologias propostas, é possível esperar que os termos das ontologias encontrados no grupo tenham similaridade entre eles, explicando, dessa forma, a relação forte entre esses conceitos e apontando a presença de possíveis redundâncias.

Outro *plugin* útil que fornece métricas de uma ontologia é o CytoHubba. Com ele, é possível fazer uma classificação de nós, escolhendo o algoritmo dentre os disponíveis. Ou seja, em uma ontologia importada, a ferramenta é capaz de disponibilizar o *ranking* dos termos, usando algoritmos topológicos.

Dentre os critérios de ordenação disponíveis, para o experimento, foi selecionado o *degree*, ou grau de um nó em um grafo. Essa medida é simples de ser calculada. Para grafos não direcionados, a medida é correspondente ao número de ligações do vértice. Já para grafos direcionados, o grau é correspondente a soma das arestas que entram com as arestas que saem do vértice. No contexto das ontologias, *degree* é a soma das relações que envolvem aquele termo analisado. O grau de um nó em uma ontologia pode trazer o sentido de ambigüidade. Um termo que possui somente uma relação para outro termo, possui um significado único que pode ser descrito por esse relacionamento. Portanto, baixos valores de *degree* facilitam o entendimento das relações entre os conceitos analisados. Já um termo com alto valor dessa medida, possui um alto valor de relações associadas. Dessa forma, há a dificuldade de estabelecer um significado único para um termo em estudo, dependendo da qualidade das muitas relações associadas ao termo, trazendo uma possível ambigüidade.

O terceiro e último plugin utilizado neste estudo é o Network Analyzer (Assenov 2008). Ele será útil para analisar e visualizar ontologias computando parâmetros que descreverão a topologia para redes direcionadas e não direcionadas com o objetivo de coletar estatísticas sobre uma única ontologia selecionada.

Uma qualidade deste *plugin* é que ele pode ser utilizado em dois momentos no experimento. O primeiro envolve o cálculo de estatísticas da ontologia como um todo, permitindo assim que o usuário visualize um resumo do conteúdo da ontologia. Esse sumário torna-se importante na medida em que as ontologias são atualizadas, modificadas e reutilizadas para o uso em conjunto ao longo do tempo. Com o grande número de versões que passam a estar disponíveis, é necessário comparar as mudanças entre versões e, se possível, acessar rapidamente o tipo de mudança efetuada. Dessa forma, os valores obtidos nos cálculos facilitam a visão do usuário e o entendimento de uma determinada ontologia que está em constante desenvolvimento e manutenção. Em um segundo momento, o *plugin* é utilizado com uma abordagem diferente. Isto é, o cálculo passa a ser feito para um determinado nó isolado e não mais para a ontologia inteira. Uma qualidade da ferramenta que deve ser ressaltada é a possibilidade do tratamento de centralidades como atributos, permitindo o enriquecimento da análise do Cytoscape. De acordo com Scardoni et al. (2009), centralidades são parâmetros de nós que permitem identificar nós que possuem posição relevante em comparação ao universo da rede analisada.

Medidas relacionadas a caminhos mais curtos são úteis nos experimentos propostos, pois fornecem um alto grau de conhecimento sobre o conceito e caracterizam a posição do termo na ontologia, disponibilizando mais dados que o *degree*. Nesse estudo, duas dessas medidas foram selecionadas: *closeness centrality* e *betweenness centrality*.

Deve-se destacar que a *closeness centrality* pode ser obtida através do inverso da média dos caminhos mais curtos entre um nó *n* e todos os outros nós do grafo. Ela representa a influência que o nó em questão exerce sobre as interações dos outros nós na rede como um todo. No contexto da ontologia, os termos que possuem altos valores

dessa medida possuem os menores caminhos para os outros da ontologia. Ou seja, estão mais pertos dos outros termos e são partes essenciais do fluxo da rede.

No que se refere à *betweenness centrality*, considerando três nós distintos *a, b, c*, a *betweenness centrality* do nó *a* é calculada a partir da razão entre o número de caminhos mais curtos entre *b* e *c* e o número de caminhos mais curtos entre *b* e *c* que passam por *a*. É possível observar que quanto maiores os valores encontrados, maior será a quantidade de caminhos mais curtos entre outros termos que passam por esse nó. A *betweenness centrality* é uma forma de obter um alto grau de informação sobre o nó, ja que é capaz de ilustrar a sua posição em um sentido global. Essa medida mostra que o nó com alto valor exerce uma função de controle e influência sobre a rede, funcionando como uma espécie de ponte entre complexos da ontologia.

4. Experimento

Nesse trabalho, experimentos foram propostos a fim de ilustrar a importância da métrica tirada de uma ontologia. Sendo assim, a GO Slim foi usada como entrada dos experimentos por ser uma ontologia com poucos conceitos, de alto nível e que possibilita a visão do nó em comparação com o todo. Os três ramos da GO Slim, *cellular component*, *biological process* e *molecular function*, são tratados como ontologias distintas nesse experimento. Para apoiar essa análise, um conjunto de ferramentas foi selecionado. O Cytoscape é a ferramenta de edição e visualização de grafos selecionada, já para calcular estatísticas e análises os plugins Clusterviz, Network Analyzer e CytoHubba foram utilizados.

4.1 – Análise de *Clusters*

Essa etapa do experimento foi apoiada pelo *plugin* ClusterViz e foram mantidos os parâmetros sugeridos pela documentação da ferramenta, objetivando encontrar regiões fortemente conectadas.

Após concluir essa parte do experimento, foi possível observar que somente no ramo *cellular component* foi encontrado um *cluster*, utilizando os parâmetros propostos. Ao permitir a criação de *clusters* maiores, a partir da alteração dos parâmetros, foram encontradas regiões conectadas também nos outros dois ramos da GO Slim. Porém, esses não foram analisados, pois, ao aumentar o tamanho do *cluster,* a análise perdeu o propósito. Sendo assim, o *cluster* do ramo *cellular component* possui três termos: *organelle*, *nucleolus* e *nucleus*. Pode-se concluir que ou esses elementos trazem uma redundância por estarem tão fortemente ligados ou representam uma inconsistência de representação ao compará-los com os outros termos da ontologia.

4.2 – Análise de *Degree*

Essa etapa do experimento foi feita usando o *plugin* CytoHubba visando obter uma classificação dos termos da ontologia que possuem mais ligações com outros termos.

A partir dos resultados obtidos, pode-se concluir que quanto mais ligações, mais relações envolvem o conceito analisado e mais significado está agregado ao conceito.

No ramo *biological process,* o termo com maior valor de grau é a própria raiz da ontologia, com valor 32. Observa-se que aproximadamente 50% das relações dessa ontologia são com a raiz, o que mostra uma falta de detalhamento na descrição dos

termos dessa ontologia. O mesmo acontece com o ramo *molecular function,* evidenciando termos que são folha diretamente ligados por uma relação *part-of* ou *is_a* com a raiz.

Já no ramo *cellular component,*a raiz também possui o maior valor de grau, mas os termos *cytoplasm* e *organelle* também têm um alto valor, podendo ser interpretados assim como subdomínios da ontologia *cellular component,* já que os outros termos da ontologia possuem muitas ligações com os dois destacados.

Vale ressaltar que ao analisar somente o *out-degree,* deve-se perceber se o excesso de arestas de saída dá significado da descrição dos relacionamentos entre os conceitos de difícil compreensão.

4.3 – Análise da Ontologia

Esta etapa do experimento foi realizada utilizando o *Network Analyzer* no intuito de obter uma visão geral da estrutura da ontologia, usando a abordagem de enxergar a ontologia como um grafo.

O resumo da ontologia permite que o usuário tenha noção, principalmente, do seu tamanho e suas características como número de nós, quantidade de ligações entre termos, a média de termos a que cada nó está ligado. A cada nova versão dessas ontologias é possível perceber, comparar os valores obtidos e concluir seu diâmetro, que significa a maior distância entre dois nós da ontologia, seu raio, correspondente à menor distância entre dois nós da ontologia. Além disso, é possível verificar algumas possíveis inconsistências na sua construção como um nó com a ligação para ele mesmo e termos isolados, isto é, sem ligações.

O resumo de cada ontologia analisada pode ser visto na Tabela 1. É possível perceber, a cada nova versão dessas ontologias, pode-se comparar os valores obtidos e concluir se as modificações feitas foram a nível estrutural ou não.

Tabela 1. Resumo das ontologias usando o Network Analyzer

	Molecular Function	*Biological process*	*Cellular Component*
Número de Componentes Conexos	1	1	1
Diâmetro	4	4	4
Raio	1	1	1
Número Médio de Vizinhos	2,049	2,36	3,611
Número de nós	41	50	36
Número de arestas	42	63	65
Densidade	0,051	0,048	0,103
Número de nós isolados	0	0	0
Número de Loops	0	0	0

4.4 – Análise das Centralidades

Esta etapa do experimento foi realizada utilizando o *Network Analyzer* objetivando a obtenção da relevância de um nó em relação a ontologia em questão.

Após a geração dos resultados das métricas *betweenness centrality* e *closeness centrality,* foi percebido que os termos que possuiam maiores valores dessas métricas eram termos mais complexos da ontologia, ou seja, os mais gerais. Esses termos são importantes uma vez que possuem caminhos mais curtos passando por eles o que indica que esses termos possuem relação com grande parte dos outros termos da ontologia. Logo, são termos centrais em questão de estrutura.

Foi notado, também, que para os três ramos da GO Slim esta idéia se repetiu, ou seja, termos com maiores valores de centralidade eram termos mais gerais da ontologia representados pelas raízes *cellular component, molecular function* e *biological process.*

5. Conclusão

Este artigo tratou da importância de se avaliar ontologias no cenário atual de crescimento do uso e relevância das ontologias nas mais diversas áreas do conhecimento. O estudo apontou um conjunto de métodos de avaliação disponíveis na literatura e selecionou algumas métricas a fim de realizar um experimento que apontou ferramental apropriado para os cálculos.

Pode-se concluir que o ramo de avaliação de ontologias está em crescente expansão uma vez que as ontologias estão se tornando cada vez mais importantes e complexas demandando métricas para que seja possível conhecê-las melhor. Desse modo, pode-se trabalhar na qualidade e estimar custos de sua utilização, por exemplo.

As métricas selecionadas se mostraram úteis para determinar características da ontologia e dos termos pertencentes à ela. Foi possível, também, identificar como estão dispostos os relacionamentos entre termos e, consequentemente, perceber seus papéis nos ramos da ontologia selecionada.

Segundo Cross e Pal (2005), o resumo da ontologia é necessário, por exemplo, no resultado de buscas de programas como o Swoogle (Ding et al. 2004). As estatísticas sobre as ontologias buscadas orientariam os usuários na escolha de uma terminologia apropriada ao uso. O trabalho sugere mecanismos e mostra aplicações de como obter essas informações.

Como trabalho futuro, seria interessante ampliar o domínio do experimento, sem fixar a análise à área da bioinformática a fim de analisar se os resultados teriam comportamento diferenciado.

References

Assenov, Y., Ramírez, F., Schelhorn, S., Lengauer, T., Albrecht, M., (2008), "Computing topological parameters of biological networks", Bioinformatics (Oxford, England), v. 24, n. 2 (Jan.), p. 282-284.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., et al., (2000), "Gene ontology: tool for the unification of biology. The Gene Ontology Consortium", Nature Genetics, v. 25, n. 1 (Maio.), p. 25-29.

Cross, V., Pal, A., (2005), "Metrics for Ontologies". In: NAFIPS 2005 - 2005 Annual Meeting of the North American Fuzzy Information Processing SocietyNAFIPS 2005 - 2005 Annual Meeting of the North American Fuzzy Information Processing Society, p. 448-453, Detroit, MI, USA.

Gangemi, A., Cantenacci, C., Ciaramita, M. And Lehmann, J. (2005) "A theoretical framework for ontology evaluation and validation", Semantic Web Applications and Perspectives SWAP2nd

Gómez-Pérez, A. (2003) "Ontology Evaluation". In Handbook on Ontologies, pages 251-254

Ding,L , Finin, T., Joshi, A. , Pan, R. , Scott Cost, R. , Peng, Y., Reddivari, P., Doshi ,V., and Sachs,J. (2004). Swoogle: a search and metadata engine for the semantic web. In Proceedings of the thirteenth ACM international conference on Information and knowledge management (CIKM '04). ACM, New York, NY, USA, 652-659.

Noy, N. (2004) "Evaluation by Ontology Consumers". IEEE Intelligent Systems 1541-1672

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski, B., and Ideker, T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res*, 13(11):2498–2504.

Sure, Y. e Vrandecic, D. (2007) "How to Design Better Ontology Metrics". In: 4th European Conference on The Semantic Web, pages 311–325

Welty, C., Kalra, R. and Chu-Carrol, J.(2003) "Supporting ontological analysis of taxonomic relashionships". In *Proceedings of the ISWC-03* Workshop on Semantic Integration

Yao, H., Orme, A. M., and Etzkorn, L. (2005). Cohesion metrics for ontology design and application. *Journal of Computer Science*, 1(1):107–113.

Tratamento de Inferência em Banco de Dados Ecológicos

Maira Poltosi[1], Daniele Palazzi[1], Ana Maria de C. Moura[1], Fabio Porto[1], Frederic Valentin[1], Jean Louis Valentin[2]

[1]Extreme Data Lab (DEXL Lab)
Laboratório Nacional de Computação Científica (LNCC)
Petrópolis – RJ – Brasil

[2]Departamento de Biologia Marinha – Instituto de Biologia
Universidade Federal do Rio de Janeiro (UFRJ) – Rio de Janeiro, RJ – Brasil

{maira,dpalazzi,anamoura,fporto,valentin}@lncc.br,
jlv@biologia.ufrj.br

Abstract. *This paper presents the development of a database to support ecological data management and treatment. This database, developed in the context of the PELD project, is provided with inference capability, achieved from the relational model mapping to RDF. Taxonomical, spatial and trophic relations are explored by means of nested sets and explicit transitive-closure, which enable retieving interesting information about existing marine species of Guanabara bay, illustrated by SPARQL queries.*

Resumo. *Este artigo apresenta o desenvolvimento de um de banco de dados para auxiliar no gerenciamento e tratamento de dados ecológicos. Este banco, desenvolvido no contexto do projeto PELD, é dotado da capacidade de inferência, fruto de um mapeamento do modelo relacional para RDF. Relações taxonômicas, espaciais e de cadeia trófica são exploradas por meio de modelos de conjuntos aninhados e de fecho-transitivo explícito, que permitem a descoberta de informações interessantes sobre espécie marinhas da baia da Guanabara, ilustradas por consultas SPARQL.*

1. Introdução

A forte preocupação com a preservação do meio-ambiente no planeta e a necessidade de organizar e catalogar as reservas naturais existentes, tanto minerais como biológicas, a exemplo fauna, flora e recursos hídricos, tem dado origem a diversos sistemas para o repertoriamento e controle destas informações.

As descobertas observadas e relatadas em eventos científicos (ISEST[1], ECOTHEE[2], etc.) e publicações (Nature, Biotropica, Ecological Informatics, etc.) no domínio da Ecologia, juntamente com os avanços tecnológicos na área da Ciência da Computação, vêm contribuir significativamente para o desenvolvimento de sistemas de software adequados para lidar com dados de natureza ecológica e espaço-temporais, abrangendo os mais diversos aspectos como: modelagem, metadados, ontologias, estruturas multi-dimensionais, armazenamento, acesso, distribuição, mineração, visualização, interfaces gráficas, interoperabilidade, consultas, etc.. Segundo [Manzi 2009], os sistemas de informação para dados ecológicos ainda se apresentam em estágios iniciais de desenvolvimento e enfrentam alguns desafios técnicos, como o de criar

1 2011 International Symposium on Environmental Science and Technology: http://www.isest11.com/

2 2nd International Conference on ECOLOGICAL THEOLOGY AND ENVIRONMENTAL ETHICS: http://www.conferencealerts.com/seeconf.mv?q=ca1i0xh6

estruturas de dados verdadeiramente genéricas, podendo acomodar dados de qualquer tipo de levantamento.

No Brasil, encontra-se em desenvolvimento o projeto PELD[3] (Pesquisas Ecológicas de Longa Duração) que tem entre as principais missões o avanço no conhecimento ecológico de forma a fornecer dados para embasar decisões relacionadas ao gerenciamento de recursos naturais além de disponibilizar estas informações aos diversos setores da sociedade. O PELD atualmente possui 29 sítios de coletas distribuídos em diversos biomas do Brasil, a fim de consolidar o conhecimento existente sobre a composição e funcionamento destes ecossistemas. Os dados provenientes das pesquisas do sítio PELD Amazônia estão sendo migrados para o padrão de metadados[4] EML, adotado pelas redes de pesquisa mundiais LTER e ILTER, prevendo a futura integração de dados entre os sítios PELD.

Entre estes, está em desenvolvimento o PELD Guanabara[5], que visa melhorar o entendimento da estrutura e do funcionamento do ecossistema da Baia de Guanabara e de suas respostas aos impactos antrópicos e climáticos. Além disso, esse projeto permitirá avaliar a adequação das medidas que estão sendo tomadas para a despoluição da Baia de Guanabara, no que diz respeito à ecologia marinha. De forma a viabilizar um estudo contínuo de longo prazo, e de prover meios computacionais que permitam realizar um prognóstico da tendência evolutiva desse ecossistema, está em desenvolvimento o projeto de um banco de dados para a organização, gerenciamento e recuperação de informações de dados ecológicos da Baia de Guanabara.

A grande contribuição desse trabalho em relação a outros sistemas de gerenciamento de dados ecológicos existentes (Ecoflora, [Manzi 2009], [Cavalcanti 2005]) é dotá-lo da capacidade de raciocínio. Por meio de uma representação ontológica do modelo, novos relacionamentos e instâncias podem ser inferidos a partir de propriedades transitivas e de hierarquias sobre os conceitos do modelo, oferecendo aos pesquisadores da área a descoberta de dados interessantes, tais como, os predadores, de diferentes níveis, de uma determinada espécie.

O restante desse trabalho está organizado da seguinte forma: a seção 2 apresenta um panorama geral sobre os trabalhos relacionados na área. A descrição do cenário de aplicação está na seção 3. A seção 4 formaliza e descreve o processo de mapeamento do modelo relacional para a representação RDF. Na seção 5 encontram-se exemplos de consultas à base realizadas em SPARQL. Finalmente, a seção 6 finaliza o artigo, com sugestões para trabalhos futuros.

2. Trabalhos relacionados

Vários trabalhos têm sido propostos com a finalidade de gerenciar e compartilhar dados ecológicos. Em especial, no contexto do projeto PELD [Manzi 2009], é proposta a criação de um banco de dados (BD) para gerenciamento de dados biológicos e ambientais geo-referenciados, de modo a prover a recuperação simples e avançada de dados, incluindo a análise espacial dos mesmos.

Ecoflora[6] é um BD ecológico sobre 2200 espécies de plantas das ilhas britânicas, construído a partir de inúmeras fontes e classificadas segundo uma taxonomia envolvendo 130 características ecológicas e morfológicas contendo informações sobre família, gênero, nome específico, autor, sinônimos, etc. Os dados incluem ainda informações sobre espécies distribuídas no Reino Unido e demais países da Europa, além de informações sobre insetos e doenças provenientes de fungos, os quais são obtidos da literatura.

Em [Cavalcanti 2005] é descrito um sistema de BD relacional para o gerenciamento e análise dos dados coletados da Reserva de Desenvolvimento Sustentável do Lago Tupé. O

3 http://ppbio.inpa.gov.br/Port/projetosassociados/peld/
4 http://knb.ecoinformatics.org/community.jsp
5 http://www.lncc.br/peldguanabara/index.php
6 http://www.ecoflora.co.uk/

objetivo do banco é a sua integração com um sistema de informação distribuído sobre biodiversidade, que possa fornecer apoio às atividades de pesquisa e extensão em andamento na área. Dados específicos sobre inventários biológicos são coletados dos sistemas ECOLOG [Cavalcanti 1993], que contém dados sobre espécies, indivíduos, descritores morfológicos e variáveis ambientais; e ALINE [Cavalcanti 1998], que reúne informações sobre flora e a fauna da área de estudo, incluindo dados sobre taxonomia e nomenclatura, usos econômicos, nomes vulgares, ecologia, morfologia, distribuição geográfica e bibliografia relevante para cada espécie. Além disto, conta com o sistema DELTA [Cavalcanti 1996] que permite codificar e processar a informação taxonômica sobre os dados coletados, que posteriormente podem ser analisados com o auxílio de pacotes estatísticos.

Em [Campos et al 2009] é proposta uma modelagem para dados de zoneamento ecológico econômico do estado de Minas Gerais, com o objetivo posterior de criar um BD. Nesse banco serão armazenadas características ecológicas envolvendo temas mais genéricos tais como zonas, riscos ambientais, prioridade de conservação e prioridade de recuperação, todos geo-referenciados, mas que são a seguir expandidos em temas mais específicos, envolvendo Vulnerabilidade Natural, que é composta pelos subtemas Biótico e Abiótico.

Dentre os sistemas analisados, observou-se que a grande preocupação dos mesmos refere-se ao armazenamento e recuperação dos dados ecológicos. Interfaces flexíveis e amigáveis permitem consultar informações, cujos tópicos são indexados por categorias com base em taxonomias. No entanto, nenhum deles explora a recuperação de informações implícitas que podem ser obtidas através de um tratamento semântico e de inferência sobre os dados ali representados. O aspecto de inferência é sem dúvida uma questão interessante na análise das informações ecológicas do ponto de vista dos pesquisadores. Quando essas informações são descritas e representadas por uma linguagem ontológica, o mecanismo de raciocínio ali embutido pode trazer contribuições relevantes para área, como mostram alguns trabalhos que lidam com sistemas de anotação de documentos em bio-informática [Fontes 2011], área geo-espacial [Macário e Medeiros 2009], [Macário et al 2009] e de meio ambiente [Crivelenti et al 2009].

3. Cenário da Aplicação PELD Guanabara

O PELD Guanabara visa obter dados bióticos e abióticos a partir de amostras extraídas da água da baia e de comunidades planctônicas e bentônicas, assim como de recursos pesqueiros. Este estudo utiliza um modelo parcial considerando apenas os aspectos relativos ao acompanhamento de biomassa de plancton.

De uma amostra de plancton são armazenadas informações temporais (data e hora), espaciais (latitude, longitude e profundidade), método de coleta da amostra e de conservação utilizados, assim como as condições atmosféricas e marítimas no momento da coleta. Para cada análise realizada, são registradas a data, amostra e o método aplicado.

As medições de biomassa dos organismos encontrados nas amostras podem ser realizadas em nível de espécie ou em nível mais alto da taxonomia. A taxonomia classifica os organismos em uma hierarquia, cujo primeiro nível é Reino, que é dividido em Filos e sucessivamente em classes, ordens, famílias, gêneros e espécies. Cada nível destes tem, por sua vez, suas respectivas subdivisões. Qualquer nível desta classificação é chamada taxon e tem como plural taxa. Há diferenças nos níveis preenchidos de um e outro organismo. Alguns organismos têm sido reclassificados, e neste caso, ambas classificações são mantidas, sendo estabelecida uma relação de sinônimos entre elas.

Do ponto de vista de análise, o BD ecológico apresenta algumas dimensões importantes. A informação de região geográfica permite selecionar e classificar eventos segundo sua localização. Por outro lado, as relações tróficas são fundamentais para o estudo do ecossistema. Finalmente a taxonomia permite uma análise hierarquizada das espécies. O ambiente de

inferência apresentado na seção a seguir explora de forma integrada estratégia de banco de dados e técnicas da Web Semântica possibilitando estas análises.

4. Mapeamento do modelo relacional para representação RDF

A disponibilização em um formato homogêneo dos dados produzidos pelos projetos PELD é um fator fundamental na formação de uma visão integrada dos ecossistemas brasileiros.
Neste contexto, vêm se utilizando cada vez mais do RDF[7] (Resource Description Framework) como meio de interoperar, inferir e publicar dados, além de possibilitar que estes possam posteriormente se conectar a outros recursos de domínios similares. Permite ainda a exploração de dados e associações utilizando-se da linguagem de consulta SPARQL[8]. Optou-se pela representação do modelo através de assertivas RDF para incorporar expressividade semântica na camada de dados.

A utilização de RDF pode trazer, todavia, uma preocupação com o desempenho no tratamento de grandes volumes de dados, como é o caso do banco de dados ecológico. Nestas situações, uma estratégia comumente adotada é armazenar os dados em bancos de dados relacionais.

Deste modo, o banco de dados do PELD Guanabara está sendo armazenado no sistema PostgreSql v. 8.4.5. e é mapeado para um grafo virtual RDF com auxílio da ferramenta D2RQ[9], uma plataforma desenvolvida em Java que permite executar consultas SPARQL e acessar a informação do BD como *LinkedData*[10] sobre a Web. Esta plataforma é constituída pelos seguintes componentes: D2RQ Mapping Language, linguagem declarativa que descreve o mapeamento; D2R Server, servidor HTTP que fornece uma visão Linked Data e permite consultas SPARQL; e D2RQ Engine, um plug-in para uso de raciocinadores Jena[11] e Sesame[12]. A Figura 1 apresenta a representação em RDF do BD PELD Guanabara.

No processo de mapeamento é criado um arquivo padrão que analisa o esquema do BD. Todas as tabelas do modelo relacional são então transformadas em classes no grafo RDF virtual e todas as colunas em propriedades. Este arquivo pode ainda ser personalizado utilizando vocabulários conhecidos, como por exemplo, rdf, rdfs, foaf, dc, owl entre outros. Uma vez gerado o arquivo de mapeamento, é possível navegar sobre o grafo RDF, através do servidor D2R Server.

4.1. Formalização da arquitetura

Define-se um BD relacional como BD = R ∪ C, onde R={R_1, R_2,..., R_n} é um conjunto de relações, de esquema E={$\mathcal{R}_1$,$\mathcal{R}_2$,..,$\mathcal{R}_n$}, e C= {C_1, C_2,..., C_m} é um conjunto de restrições do tipo chave primária e chave estrangeira.

Um domínio de aplicação pode ser modelado em RDF segundo um conjunto de assertivas do tipo <s,p,o>, onde um predicado *p* descreve um recurso *s*, segundo um objeto *o*. Definimos $s \subseteq URI$, $p \subseteq URI$ e $o \subseteq URI \cup L$, onde *URI* é um conjunto de referências à *Unified Resource Identifiers*[11], identificando recursos do domínio, e *L* é um conjunto de literais. O conjunto *A* de assertivas RDF é chamado de grafo RDF, onde $A = (N, E)$, e $N \subseteq S \cup O$ e $E \subseteq P$, onde S é o conjunto de recursos, O é o conjunto de objetos e P é o conjunto de predicados. Recursos definidos em S ou em O podem ser agrupados em tipos através do predicado rdf:type, que associa recursos a Classes do domínio. Adicionalmente, predicados em P são instâncias do tipo *rdf:Property*.

7 www.w3.org/RDF
8 www.w3.org/TR/rdf-sparql-query/
9 http://www4.wiwiss.fu-berlin.de/bizer/d2rq/spec/
10 http://linkeddata.org/
11 http://jena.sourceforge.net/
12 http://www.openrdf.org/

O mapeamento entre os dois modelos, implementado pela ferramenta D2RQ, considera um conjunto de correspondências entre o esquema relacional E e elementos do grafo RDF A. Neste mapeamento, cada esquema de relação $\mathcal{R}$ corresponde a uma classe RDF K, do tipo *rdf:class* e cada atributo $at_i \in \mathcal{R}$ corresponde a uma propriedade p_i do tipo *rdf:property*. O domínio da propriedade p_i é a classe K e seu contra-domínio é um tipo de dado da classe *rdf:Datatype* ou uma classe K_i. Neste último caso, o atributo at_i correspondente seria chave primária ou estrangeira de $\mathcal{R}$.

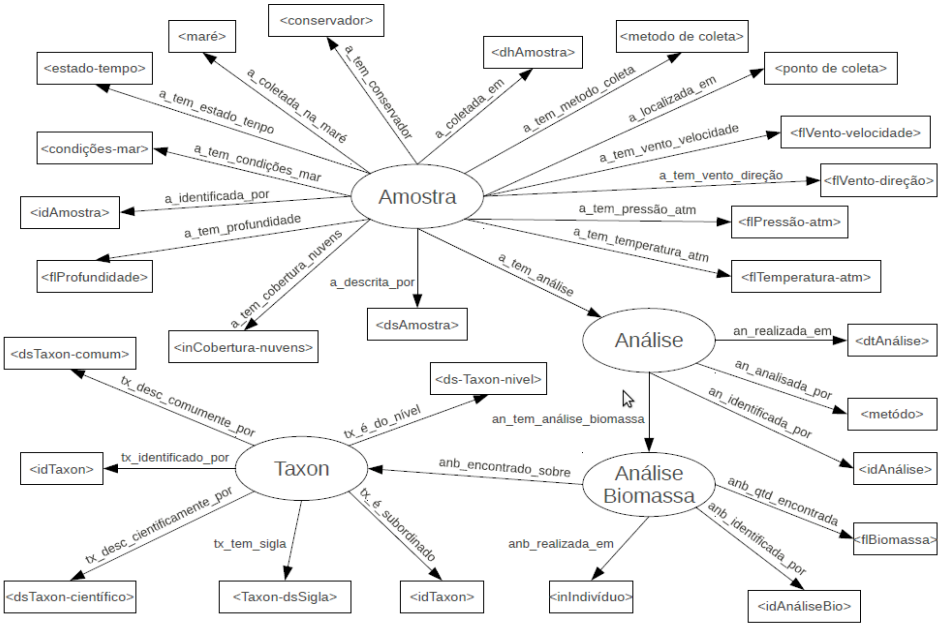


Figura 1. Representação RDF do BD ecológico

4.2. Implementando inferência em visões RDF

A capacidade de inferência durante o processamento de consultas sobre grafos RDF, considera a semântica explícita de transitividade associada a alguns predicados. Assim, predicados RDF como *rdf:subClassOf* tem semântica transitiva permitindo a aplicação do fecho transitivo sobre assertivas deste tipo. A possibilidade de inferência é interessante, pois permite que relacionamentos sejam estabelecidos de maneira implícita. Em particular, no contexto da aplicação sendo estudada, as relações Taxonomia e Região apresentam os relacionamentos transitivos *é_subordinado* e *é_subregião*, respectivamente, e a relação CadeiaTrófica apresenta o relacionamento *é_presa,* que apesar de não se caracterizar como uma relação transitiva, é utilizada no modelo para implementar uma lista ordenada nesta relação.

Considerando-se a implementação do BD em PostgreSQL, que oferece nativamente o tratamento de transitividade, o recurso de inferência assim obtido, poderia ser oferecido pela visão RDF. A ausência de um tratamento para a transitividade em D2RQ requer a extensão do esquema relacional. Utilizaram-se duas abordagens de representação explícita do fecho transitivo. Para os relacionamentos *é_subordinado* e *é_subregião,* utilizou-se o modelo de conjuntos aninhados [Celko 2005], uma vez que o fecho transitivo nestes exemplos expande-se para uma estrutura arborescente. Por outro lado, o relacionamento *é_presa* expande-se em um grafo, uma vez que uma espécie pode ser *presa* de mais de uma outra espécie, e pode igualmente ser predadora de várias espécies, impedindo a representação em conjuntos aninhados. Adota-se então um modelo de *fecho-transitivo-explícito,* que neste trabalho é aplicado em nível de instância.

4.3. Modelo de conjuntos aninhados

No modelo de conjuntos aninhados, considera-se o conjunto *Is* de instâncias de uma relação R, e uma função booleana é_pai(*R*, *R*), tal que $\forall\ i_i\ e\ i_j \in Is$, é_pai($i_i$, i_j) se e somente se, i_i tem uma chave estrangeira para i_j [Navathe 2005].

A implementação do modelo de conjuntos aninhados é realizada construindo-se uma árvore que obedeça a relação é_pai(R,R) para todas as instâncias de R. Cada nó nesta árvore representa uma instância dessa relação. Processa-se a árvore em pré-ordem. A cada nó associam-se dois valores, esquerda e direita. O valor esquerda corresponde à posição em que o elemento foi lido. Adicionalmente, uma vez que todos os filhos de um nó tenham sido visitados, associa-se o valor *direita*, correspondendo à posição seguinte a de seu descendente mais à direita. A navegação continua em um irmão, a partir do valor à direita recém associado.

Como resultado do processamento acima descrito, adiciona-se ao esquema da relação R os atributos esquerda e direita. Assim, dada duas instâncias $i_i\ e\ i_j \in R$, é_pai(i_i, i_j) se e somente se $i_i.esquerda > i_j.esquerda$ e $i_i.direita > i_j.direita$, e não existe $i_k \in R$, $i_k \neq i_j$, tal que $i_i.esquerda > i_k.esquerda$ e $i_i.direita > i_k.direita$.

As seguintes relações foram implementadas utilizando o modelo de conjuntos aninhados para representação do fecho transitivo:

```
Região (idRegiao, …, Esquerda, Direita)
Taxon (idTaxon, …, Esquerda, Direita)
```

4.4. Modelo de fecho transitivo explícito

O relacionamento transitivo expresso pela associação *é_presa* produz uma ordenação em grafo, impedindo sua implementação através do modelo de conjuntos aninhados. Nesta implementação decidiu-se explicitar todo o fecho transitivo. Define-se uma relação R'(R,R), tal que, $\forall\ i_i\ ,\ i_j \in R$, se *é_presa*($i_i$, i_j) então existe $i_k \in R'$ e $i_k < i_i\ ,\ i_j>$.

A relação Cadeia_trofica foi implementadas utilizando o modelo de fecho transitivo explícito:

```
Cadeia_trofica (idCadeia_trofica, idTaxon_presa, idTaxon_predador)
```

5. Consultas em SPARQL

A partir da modelagem apresentada anteriormente, algumas consultas SPARQL foram implementadas, com seus respectivos resultados. Elas visam a análise dos taxa sobre a região em que foram coletados, relacionando-os à cadeia trófica e aos sinônimos entre eles.

5.1. Consultas com inferência sobre BD PELD Guanabara

As consultas a seguir foram implementadas utilizando SPARQL, cujo código é apresentado na Figura 2. Os resultados correspondentes são listados na Figura 3.

(i) Quais são os animais que correspondem aos predadores do peixe “Manjuba” na cadeia trófica?

(ii) Obter as presas do taxon “Bagre”, incluindo os taxa sinônimos.

(iii) Quais os taxa subníveis de um dado taxon? Recuperar a sub-árvore a partir do taxon informado, indicando o nível da hierarquia taxonômica.

(iv) A ocorrência de um acidente ecológico em uma dada região pode causar a morte de plancton que são componentes dos primeiros níveis da cadeia trófica. É possível identificar quais os predadores das espécies de plancton encontrados nesta região para dimensionar o impacto deste evento.

(i)
```
SELECT DISTINCT ?nome_pres
                ?nome_predador ?grau
WHERE {
?s <URI/taxon_dstaxon_comum> "Manjuba";
<URI/taxon_dstaxon_comum> ?nome_pres.
?w <URI/cad_trof_Taxon_idTaxon_predador>
?pred.
?pred <URI/taxon_dstaxon_comum>
?nome_predador.
?resource <URI/cad_trof_inGrau_relacao>
?grau ;
<URI/cad_trof_Taxon_idTaxon_presa> ?e.
FILTER ( ?w = ?resource && ?s = ?e) }
ORDER BY ?grau ?nome_pred
```

(ii)
```
SELECT DISTINCT ?nome_predador
                ?nome_presa ?sinonimo
WHERE {
{?s <URI/taxon_dstaxon_comum> "Bagre".
?w <URI/cad_trof_Taxon_idTaxon_predador>
?s.
?s <URI/taxon_dstaxon_comum>
?nome_predador.
?w <URI/cad_trof_Taxon_idTaxon_presa>
?presa.
?presa <URI/taxon_dstaxon_cientifica>
?nome_presa.
}UNION{
?s <URI/taxon_dstaxon_comum> "Bagre".
?w <URI/cad_trof_Taxon_idTaxon_predador>
?s.
?s <URI/taxon_dstaxon_comum>
?nome_predador.
?w <URI/cad_trof_Taxon_idTaxon_presa>
?presa.
?presa <URI/taxon_dstaxon_cientifica>
?nome_presa.
?id_sinonimo
<URI/taxon_taxon_idtaxon_sinonimo> ?presa.
?id_sinonimo
<URI/taxon_dstaxon_cientifica>?sinonimo.}}
```

(iii)
```
SELECT ?nome_taxom_nivel ?nivel
WHERE {
?s          <URI/taxon_dstaxon_cientifica>
"Calanoida".
?s <URI/taxon_inleft> ?left.
?s <URI/taxon_inright> ?right.
?taxon <URI/taxon_inleft> ?left_t.
?taxon <URI/taxon_inright> ?right_t.
?taxon
<URI/taxon_taxon_nivel_idtaxon_nivel>
?taxon_nivel.
?taxon <URI/taxon_dstaxon_cientifica>
?nome_taxom_nivel.
?taxon_nivel
<URI/taxon_nivel_intaxon_nivel_ordem>
?ordem.
?taxon_nivel
<URI/taxon_nivel_dstaxon_nivel> ?nivel
FILTER ( ?left_t > ?left &&
         ?right_t < ?right )}
ORDER BY ?ordem
```

(iv)
```
SELECT DISTINCT ?id_regiao
?id_ponto_coleta ?name_taxon ?nome_predador
WHERE {
?id_regiao <URI/regiao_dsregiao> "da barra
a ponte".
?id_ponto_coleta
<URI/ponto_coleta_regiao_idregiao> ?regiao.
?id_amostra
<URI/amostra_ponto_coleta_idponto_coleta>
?id_ponto_coleta.
?id_analise <URI/analise_amostra_idamostra>
?id_amostra.
?id_analise_bio
<URI/analise_biomassa_analise_idanalise>
?id_analise;
<URI/analise_biomassa_taxon_idtaxon>
?id_taxon.
?id_taxon <URI/taxon_dstaxon_cientifica>
?name_taxon;
<URI/taxon_dstaxon_cientifica> ?nome_presa.
?w <URI/cad_trof_Taxon_idTaxon_predador>
?predador.
?predador <URI/taxon_dstaxon_cientifica>
?nome_predador.
?w <URI/cad_trof_inGrau_relacao> ?grau;
<URI/cad_trof_Taxon_idTaxon_presa> ?e.
FILTER ( ?id_regiao = ?regiao &&
         ?id_taxon = ?e )}
ORDER BY ?nome_taxon
```

Figura 2. Consultas SPARQL sobre BD Peld Guanabara

6. Conclusão

O projeto PELD Guanabara visa oferecer melhor entendimento da estrutura e do funcionamento do ecossistema da Baia de Guanabara e de suas respostas aos impactos antrópicos e climáticos. Como parte desta iniciativa, está sendo desenvolvido um sistema computacional para armazenar os dados ecológicos da Baia de Guanabara e permitir sua análise integrando informações geográficas, taxonômicas e sobre a cadeia alimentar das espécies deste ecossistema. O sistema utiliza um banco de dados relacional para armazenamento dos dados e uma visão RDF que homogeniza a publicação de hierarquias, conceitos e relacionamentos do domínio em uma estrutura do tipo grafo. De modo a oferecer a capacidade de inferência sobre relacionamentos transitivos, técnicas de conjuntos aninhados e de fecho-transitivo explícito estendem o esquema relacional adotado e são exploradas em consultas SPARQL.

Finalmente, como trabalhos futuros, pretende-se avaliar a adoção do padrão EML como

terminologia para a integração das bases PELD, bem como integrar a visão RDF ao contexto de dados associados conhecidos como *LinkedData*.

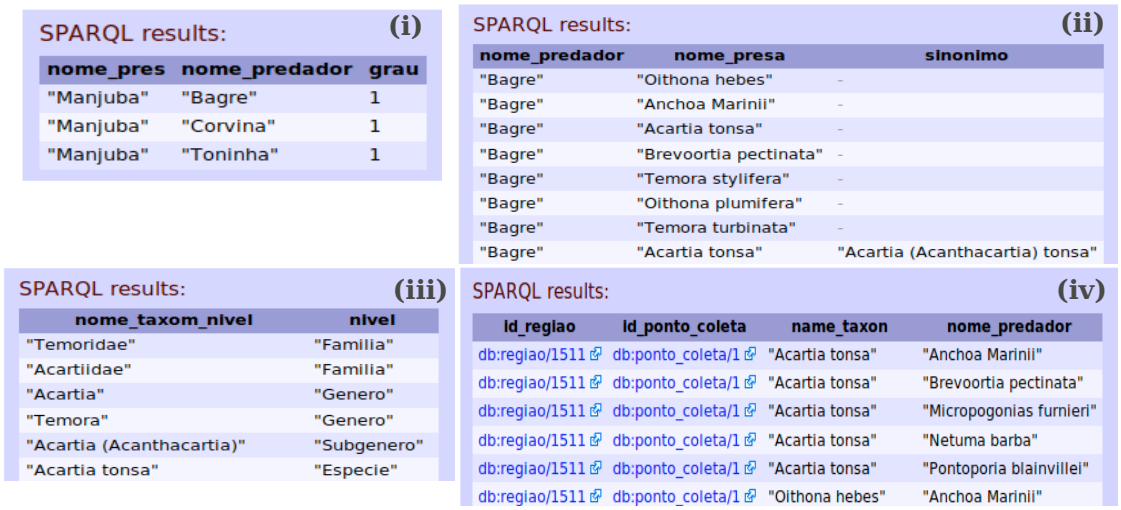


Figura 3. Resultados das consultas

Agradecimentos

Este trabalho é parcialmente financiado pelo CNPq através de bolsas PCI (Proc. 382.489/09-8), produtividade de pesquisa (Proc. 309502/2009-8), PCI (Proc. 383344/2010-7) e DTI (Proc. 381663/2010-8).

Referências

Campos, S.R., Martinhago A.Z., Massahud R.T., França A.M., Prieto L. E., Mendes J.D.C. (2009) Modelagem do Banco de Dados do Zoneamento Ecológico Econômico de Minas Gerais usando UML-GeoFrame. Anais XIV Simpósio Brasileiro de Sensoriamento Remoto, Natal, Brasil, 25-30 abril, INPE, p. 4943-4949.

Cavalcanti, M. J. (2005) Bancos de Dados sobre Biodiversidade na Amazônia: a Experiência do Projeto Biotupé. Biotupé: Meio Físico, Diversidade Biológica e Sociocultural do Baixo Rio Negro, Amazônia Central. Santos-silva, Aprile, Scudeller, Editora INPA, Manaus.

Cavalcanti, M.J. (1993) ECOLOG: sistema gerenciador de bancos de dados para levantamentos ecológicos de campo. I Simpósio de Aplicações da Informática em Biologia, Campinas p. 44-45.

Cavalcanti, M.J. (1996) Designing a graphical user interface for DELTA: some considerations. DATA Newsletter, 12: 2-4.

Cavalcanti, M.J. (1998) A Windows-based data capture program for ATF. Alice Software News , 6: 13.

Celko, J. SQL for Smarties, Advanced SQL Programming, Morgan Kaufmann, San Francisco, USA, 3 ed., 2005.

Crivelenti, R.C., Coelho R.M., Adami S.F., Oliveira, S.R.M. (2009) Mineração de dados para inferência de relações solo-paisagem em mapeamentos digitais de solo. Pesq. agropec. bras., Brasília, v.44, n.12, p.1707-1715.

Fontes, C.A. (2011) Explorando Inferência em um Sistema de Anotação Semântica, Dissertação de mestrado, Instituto Militar de Engenharia.

Macário, C.G.N., Medeiros C.B. (2009) A framework for semantic annotation of geospatial data for agriculture. International Journal of Metadata, Semantics and Ontologies v.4 n.1/2, p.118-132.

Macário, C.G.N., Sousa, S.R, Medeiros C.B. (2009) Annotating geospatial data based on its semantics. 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems.

Manzi, A. 2009. Gerenciamento de dados de Projetos Ecológicos de Longa Duração, Projeto de pesquisa, Edital MCT/CNPq Nº 59/2009 - Propostas para apoio a projetos no âmbito do PELD.

Navathe, S., Elmasri, R. (2005) Sistemas de Banco de Dados, Pearson Addison Wesley, São Paulo, 4 ed., 2005.

VTPortal: a Scientific Community Web Portal for Reusable VisTrails Workflows

Sérgio Manuel Serra da Cruz, Alexandre Ribeiro, Marta Mattoso

PESC – COPPE/Federal University of Rio de Janeiro (UFRJ)

{serra, alexandreribeiro, marta}@cos.ufrj.br;

Abstract. *Managing large scientific experiments is a complex research challenge due to the amount of scientific resources to be managed. Science portals are a way to simplify this task by aggregating data from different sources and by providing a set of pre-designed analyses. However, such portals are often built manually, and are not flexible enough to support sharing, reusing and executing workflows enacted by SWfMS like VisTrails. In this paper we describe VTPortal, a science portal that combines a set of tools and an infrastructure for providing a collaborative environment for scientists.*

1. Introduction

Large scale scientific experiments demand high performance computing (HPC) in environments such as clusters, grids, P2P or cloud environments, spanning organizational and spatiotemporal boundaries while managing huge amounts of data. There are several research and implementation initiatives in the use of scientific workflow management systems (SWfMS) for e-Science applications. SWfMS (Taylor *et al.,* 2007) are tools that assist the scientist in conducting studies to prove a hypothesis during a scientific process. Scientific workflows are a result of collaborative team efforts which require specialist expertise that is often complex and challenging to build (Goderis *et al*., 2005). The provenance, or the origin, or source of something (Moreau *et al*., 2007), can capture a wide range of information, including, for example, who or what generated the data, history of data stewardship, manner of manufacture, place and time of manufacture.

Currently, provenance support in the existing SWfMS is designed to capture the information related to isolated workflow´s run. However, the collaborative process of scientific experiments often involves multiple executions of workflows within a single experiment. For example, different or the same workflow definition is executed through a common infrastructure by changing input data or parameters. These set of executions requires the registering of annotations which are tightly connected with provenance data. Scientists do more than produce and consume data: they comment on it and refer to it, and to the results of queries upon it. The provenance and annotations are often critical to the understandability, reusability, and reproducibility of the scientific experiment (Mattoso *et al.,* 2010).

The goal of this work is to extend the current single-workflow and single-user targeted provenance approach to a number of workflow runs within a controlled environment such as a community portal for sharing data, programs and workflows. To achieve this goal we present *VTPortal*, a scientific community portal that allows users to add annotations, to share reusable workflows, to remotely submit previously defined workflows available on VisTrails, and to select a machine for this remote execution. *VTPortal* does not require modifications on VisTrails' code. To the best of our knowledge this is the first system that supports the collection

of annotations and combines the retrospective provenance generated during distinct executions of the workflows that belong to the same scientific experiment.

2. Related Work

The advances of IT technologies are encouraging people to form large-scale and multidisciplinary e-Science research projects to solve complex scientific problems in different fields of knowledge, such as: bioinformatics, engineering, astronomy. These projects demand intensive computation and data resources and the use of workflows; they are collaborative in nature and include multiple domain scientists with domain-specific expertise located at geographically distributed organizations. Collaborative scientific projects like: BioPauá, BioWEP , WHIP, Virolab, VL-e, CrowdLabs , myExperiment and CAMERA 2.0 have something in common, they use community portals and scientific workflows.

Community scientific portals provide access to advanced tools and databases that could be shared by a community of users via web, allowing them to interact with other colleagues, processes, documents and content in a personalized and role-based fashion through the web browser. Historically, community portals environments have been evolved into two groups. The first group is represented by workflow-oriented portal concept (like BioPauá, BioWep and WHIP) that enables the interoperability of various grids during the execution of workflows, but has no concerns about data provenance. BioWeP and WHIP projects aim to bring the desktop-based workflow and the Web closer and provide rich interfaces for Web applications. The second group is represented by collaborative-oriented portal concept (like VL-e, ViroLab and CAMERA 2.0), where scientists can publish, share and find other scientific workflows, such group takes provenance into account but with distinct approaches. For instance, in the VL-e project, the focus is on the execution of workflows developed in distinct SWfMS.

3. *VTPortal*

VTPortal is a novel tool and is different from the previous discussed ones. It aims at helping to run pre-existent workflows available on VisTrails system and storing provenance from a multiple runs of the workflow that composes one *in silico* experiment. *VTPortal* captures and maintains associations across runs and allows scientists to add annotations about each individual run, while taking advantage of existing provenance gathering mechanisms of VisTrails.

3.1 *VTPortal* Features

A Portal provides a layer of abstraction over a complex system (such as clusters, grids and clouds) to make its use streamlined while efficient. The core features of *VTPortal* are: (*i*) It is agnostic with respect to the application domain; (*ii*) it provides integration of provenance metadata from different executions of the workflows of a given scientific experiment; (*iii*) it takes into account the needs of scientists for accessing remote environments like the Web; (*iv*) offers a transparent and simple access to scientific workflows, despite their physical location. The system goes beyond simple Web accesses, it provides components that can be used to connect the VisTrails workflow engine with Web servers, allowing scientists to share workflows stored in a remote repository.

3.2 *VTPortal* Architecture

We have developed a multi-tier architecture which follows standard software development guidelines. Scientists interact with the system through a standard browser. The front-end tier is the topmost level of *VTPortal*, at the Web server implements the application rules and displays

the presentation, *i.e.,* it allows the configuration of scientific experiments, the registering of the reusable workflows and users, the upload of dataset and personal annotations.

The front-end offers *VTPortal* navigation features. The Web server hosts the core system, the system is composed by several Python and PHP modules. The main modules are the *WfParser*, the *WfPersistence* and the *WfMonitor*. The *WfParser* is XML parser that converts reusable workflows into a set of tuples at the database at the back-end tier. It is responsible to parse the workflow definition and the existing prospective provenance provided by VisTrails. The *WfPersistence* module encapsulates all data access to the relational database system. The *WfMonitor* encapsulate the calls from the Web server to the workflow enactor, *i.e.,* a SWfMS running at the logic-tier, the *WfMonitor* also displays workflow status information about completed and currently executing workflows at logic-tier. The monitor reads tracking information from the database and displays current workflow state at the user session. The logic-tier is pulled out from the front-end tier; it hosts the SWfMS Server, the SWfMS not only enact reusable workflows but also offer built-in provenance gathering facilities. The provenance metadata is stored at the back-end tier.

At the back-end tier we have the Database Server, it could be any relational database system, the database stores the prospective provenance parsed from the reusable workflows and the retrospective provenance collected from the experiment and their workflow invocations. This tier keeps data neutral and independent from application servers or workflow engine. Giving data its own tier also improves scalability and performance of *VTPortal*.

4 *VTPortal* in action

Scientists need to organize their workflows into a particular experiment. An example is a research project, which contains all the VisTrails workflows executions and the dataset they use together. This allows for defining different levels of visibility: Research groups can have discussions and upload reusable workflows and dataset that are only visible to the people involved in the experiment. The ability to selectively disclose information for people outside the group is extremely important for scientists, who may work for many years before deciding to release certain types of data. Another advantage is that a project can have its own dedicated servers (application and database). This creates the possibility of having specialized servers for different types of workflows. For example, let us suppose a neural network experiment (Figure 1), where a scientist developed a set of workflows, they can be uploaded to on a remote server machine and previously registered users on *VTPortal* can execute those workflows remotely.

Ideally, the reusable workflows that belongs to *VTPortal* would be executable anywhere. One of the key requirements to make this possible is to abstract away references of parameters, paths and input data as to allow other scientists to execute those workflows. In *VTPortal* there is a simple way of making this possible, the workflow can take advance of VisTrails aliases (defined at design-time). The aliases values are defined by the scientist at the run-time and incorporated by *WfParser* module the at the workflow´s run-time. Like aliases values, annotations (defined by scientists for each workflow run) are valuable sources of retrospective provenance metadata defined at run-time.

We believe that one of the most interesting applications of a system like *VTPortal* is the impact that it can have on the provenance of the scientific experiment. In *VTPortal*, we advocate a direct linkage to the provenance information of the workflows and the experiment. We provide mechanisms to store the provenance of the different executions of workflows of a scientific experiment in a single database. The provenance metadata (input dataset, parameters, workflow, host, user account and profile) are stored in the VisTrails database, so querying experiments provenance is simple. Figure 1 presents some *VTPortal* screenshots related to a

neural network clustering data experiment, where a scientist select a workflow (Kohonen workflow), previously deployed by the experiment administrator, he specifies the input dataset and the parameters from a list and, to prepare it, he need to press the "Prepare a workflow" button (Figure 1a). Thus, *VTPortal* merges variable data with the reusable workflow and submit it to VisTrails server. In the second screen (Figure 1b) the scientist can add annotations regarded to that workflow run, he can also choose the machine that hosts the SWfMS server and the version of the workflow that will be enacted. Finally, and the scientist can run the workflow by pressing the "Execute" button. All such data are stored at the provenance database and the result dataset stored at the server file system.

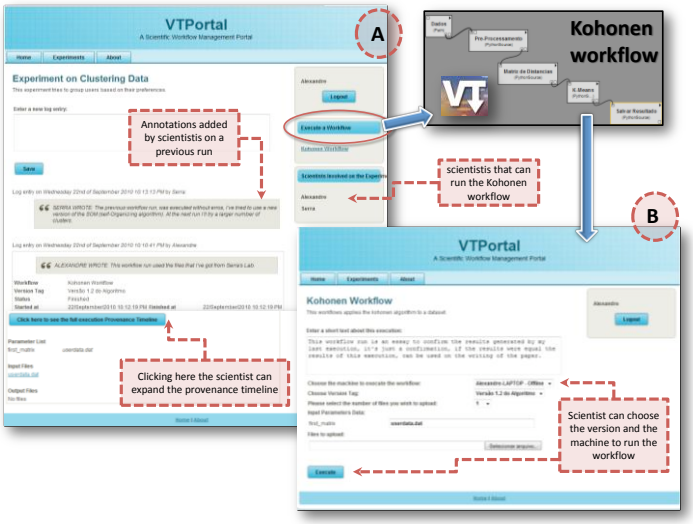


Figure 1. Configuring the workflows of the Clustering Data Experiment (1a). Setting annotation and workflows´ parameters (1b).

5. Conclusion and Future Work

Scientists involved in a scientific exploration are domain knowledge experts and should not spend their time in computational environment setups. In this paper we have introduced the *VTPortal*, a novel tool that delivers connectivity ("last mile") from remote experiments to scientists, offering a gateway that allows them to remotely browse, select, share, annotate, enact predefined reusable VisTrails workflows stored in remote machines and collect provenance from its executions. The system allows setting new parameters for pre-defined workflows at run time, querying experiments data and navigates on provenance timelines of one scientific experiment as a whole. The system was easily attached to VisTrails SWfMS and did not require modifications on it. The system suggests to the scientist the possibility to get a deeper comprehension of distinct executions of the workflows that describe the scientific protocol of an *in silico* experiment. For its most basic use, it does not require any installation of tools in the scientists´ machine, and we see this as an important advantage.

References

Goderis, A. et al., (2005) "Seven Bottlenecks to Workflow Reuse and Repurposing in The Semantic Web"In: ISWC 2005 pp. 323-337.

Mattoso, M., et al., (2010). "Towards supporting the life cycle of large scale scientific experiments" In: International Journal of Business Process Integration and Management.

Moreau, L. et al., (2007) "The Open Provenance Model", Electronics and Computer Science, University of Southampton,

Taylor, I. al., (2007) "Workflows for e-Science: Scientific Workflows for Grids". 1 ed. Springer.

Aplicação de inferência difusa em bioinformática para identificação de SNPs

Wagner Arbex[1], Fabrízzio Condé de Oliveira[2], Luís Alfredo Vidal de Carvalho[3]

[1]Empresa Brasileira de Pesquisa Agropecuária
Rua Eugênio do Nascimento, 610 - 36038-330 - Juiz de Fora - MG

[2]Universidade Salgado de Oliveira
Av. dos Andradas, 731 - 36036-000 - Juiz de Fora - MG

[3]Universidade Federal do Rio de Janeiro
Centro de Tecnologia - Bloco H-319 - 21945-970 - Rio de Janeiro - RJ

arbex@cnpgl.embrapa.br, fabrizzioconde@gmail.com, alfredo@ufrj.br

Abstract. *Research involving the discovery of single nucleotide polymorphisms (SNPs) requires bioinformatics tools to be applied to different cases, with the ability to analyze "reads" from different sources, levels of coverage and to establish reliable measures. These tools work with different methodologies on different attributes, however, it is expected similar results, even when dealing with a same data set, but it's not unusual to provide different results, which leads to uncertainty in decision making, when the results are discordant.*

Resumo. *A investigação de polimorfismos de base única necessita de ferramentas de bioinformática que devem ser aplicadas a diferentes casos, com capacidade para analisar seqüências de diferentes fontes, níveis de cobertura e que consigam medidas confiáveis. Essas ferramentas trabalham com diferentes metodologias, sobre diferentes atributos, contudo, espera-se que apresentem resultados similares, ao tratarem um mesmo conjunto de dados, mas, não é incomum fornecerem resultados diferentes, o que produz incerteza na tomada de decisão, quando os resultados são discordantes.*

1. Introdução

Polimorfismos de base única (*single nucleotide polymorphisms* - SNPs) são pares de bases em uma única posição no DNA genômico, que se apresentam com diferentes alternativas nas seqüências e podem ser encontrados no genoma de indivíduos isoladamente ou em grupos de indivíduos, em alguma população. A individualidade é conseqüência da expressão do código genético, ou seja, em sua essência, as seqüências de nucleotídeos formam as moléculas e seqüências de DNA, RNA e proteínas, que, por sua vez, interagem e formam as células, as quais também, interagem e formam os tecidos, os órgãos, até que, finalmente, formam os indivíduos. Essa é a importância dos SNPs, pois, em síntese, a alteração de um único nucleotídeo, uma única base, em uma dada seqüência, pode alterar a formação de proteínas e o conjunto dessas alterações pode provocar variações nas características dos indivíduos.

Esse texto apresenta um modelo matemático e computacional para tomada de decisão, desenvolvido e implementado com o fuzzyMorphic.pl [Arbex 2009], aplicado à investigação de SNPs em seqüências expressas de cDNA, que utiliza-se de lógica difusa para a implementação de um sistema de inferência, auxiliar à tomada de decisão, partindo de resultados prévios, obtidos por diferentes ferramentas de descoberta de SNPs e que apresentam resultados possivelmente conflitantes. O modelo é aplicado para auxiliar na tomada de decisão, nos casos em que as informações sejam divergentes e, também, na confirmação de informações coincidentes.

2. Inferência difusa como suporte à decisão

A subjetividade no raciocínio em geral, utilizada no cotidiano, sendo transmitida e perfeitamente compreendida entre interlocutores, é expressa em "termos e variáveis lingüísticas" [Zadeh 1973] e não é expressa sob a lógica clássica ou qualquer abordagem matemática tradicional. O uso de, p. ex., adjetivos comuns que representam imprecisão ou incerteza, tais como, *alto*, *baixo* ou, relações e agrupamentos, como, *conjunto das pessoas altas*, não podem ser expressos por essas abordagens, a menos que seja definido, com exatidão, o conceito ou o valor que determine a altura, a partir da qual, uma pessoa pode ser considera alta.

Os termos e variáveis lingüísticas aumentam a complexidade dos sistemas computacionais frente à capacidade trabalharem com números, valores exatos, discretos e, por vezes, excludentes, o que sugere a idéia de que, trabalhar com valores incertos, possibilita a modelagem de sistemas complexos, mesmo que se reduza a precisão do resultado, mas não retira a credibilidade. Se as incertezas, quando consideras isoladamente, são indesejáveis, quando associadas a outras características, em geral, permitem a redução da complexidade do sistema e aumentam a credibilidade dos resultados obtidos [Klir and Yuan 1995].

As abordagens clássicas são falhas para valores limítrofes e, portanto, resultados matemática e logicamente precisos, porém, questionáveis, podem ser encontrados. P. ex., o *Polyphred score* (*PPS*) estabelece seis classes com intervalos precisos (Tab. 1) [Nickerson et al. 2008] e, supondo que fossem determinados os *scores* 70 e 89 para dois pontos, então, para ambos, seria considerada a taxa de 35% de verdadeiros positivos na decisão desses pontos virem a ser SNPs (Classe 4).

Tabela 1. Classes definidas pelo PPS [Nickerson et al. 2008].

Classe	PPS	Taxa de verdadeiros positivos
1	99	97%
2	95 - 98	75%
3	90 - 94	62%
4	70 - 89	35%
5	50 - 69	11%
6	0 - 49	1%

Essa decisão, lógica e matematicamente precisa, pode ser questionada devido à subjetividade que a envolve, visto que, 70 e 89, se encontram nos limites da classe a qual pertencem e, portanto, muito próximos de diferentes interpretações. Todavia, as abordagens clássicas da lógica e da matemática não possuem as ferramentas necessárias para tratar valores limítrofes, imprecisão ou incerteza. Um valor limítrofe acarretará dúvidas na "decisão" de o ponto ser, ou não, considerado polimórfico, o que sugere um sistema de inferência difusa (SID) para o tratamento dessa incerteza.

O problema de valores limítrofes, em geral, não é tão simples quanto parece, do contrário, as abordagens clássicas poderiam facilmente resolvê-lo, mas, ao aproximar-se do raciocínio subjetivo para a interpretação e a extração de uma resposta, uma decisão, torna-se complexo e a aparente simplicidade é conferida pela modelagem por lógica difusa e seu embasamento na teoria dos conjuntos difusos. A subjetividade intrínseca ao raciocínio trata situações complexas, mediante imprecisão, incerteza ou aproximação e, então, são utilizados "operadores humanos", também de natureza imprecisa, que são expressos por termos ou variáveis lingüísticas, o que, em geral, não permite uma solução em termos exatos, mas, pode propor uma classificação, agrupamento ou agregação qualitativa em categorias ou possíveis conjuntos de soluções [de Almeida and Evsukoff 2005].

A teoria dos conjuntos difusos e a lógica difusa são adequadas para representar a informação imprecisa e caso seja possível organizar os operadores humanos em regras da forma *se ANTECEDENTE então CONSEQÜENTE*, o raciocínio subjetivo pode ser descrito em um algoritmo computacionalmente executável [Tanscheit 2007] capaz de classificar, de modo impreciso, as variáveis que participam dos termos antecedentes e conseqüentes das regras, em conceitos qualitativos, o que representa a idéia de variável lingüística [de Almeida and Evsukoff 2005]. Assim, como sistemas capazes de processar informações imprecisas e qualitativas, os modelos de inferência difusa são adequados à situação de tomada de decisão [de Almeida and Evsukoff 2005].

3. Descrição do modelo e do SID para identificação de SNPs

Em geral, as etapas de um SID são: a fuzzificação, que converte os dados "precisos" (*crisps*) de entrada em valores difusos; a inferência, propriamente dita; e a defuzzificação, que converte os resultados difusos em grandezas numéricas precisas. No modelo proposto, consideram-se como valores de entrada, as probabilidades, previamente determinadas, de o ponto vir a ser um SNP e o valor de qualidade do ponto na seqüencia consenso. Os *Casos 1* e *2* serão utilizados ao longo do texto para demonstrar o modelo, assumindo, para o *Caso 1*, 99% e 96%, quanto as probabilidades e 43 de qualidade e, para o *Caso 2*, os valores são, respectivamente, 94%, zero e 50.

3.1. Fuzzificação

Avalia-se um valor de entrada por sua "função de pertinência", o que determina um "grau de pertinência" (GP) do valor para a sua função e as funções de pertinência adotadas foram baseadas:

1. no *PPS* (Tab. 1), com a função de pertinência definida pela variável linguística *probabilidade*, com os termos (Exps. 1 e 2): *improvável* (P_{IM}), *pouco provável* (P_{PP}), *medianamente provável* (P_{mP}), *provável* (P_{PR}), *muito provável* (P_{MP}) e *altamente provável* (P_{AP});
2. na qualidade das bases do consenso – o *Phrap quality score* (PQS) – que varia entre 4 e 90 e sua função de pertinência (Exps. 3) define a variável linguística *qualidade*, nos termos: *ruim* (Q_R), *boa* (Q_B) e *ótima* (Q_O).

$$P_{IM}(x) = \begin{cases} 1 & x \leq 49 \\ \frac{59-x}{59-49} & 49 < x < 59 \\ 0 & x \geq 59 \end{cases} \quad P_{PP}(x) = \begin{cases} 0 & x \leq 25 \\ \frac{x-25}{50-25} & 25 < x < 50 \\ 1 & 50 \leq x \leq 69 \\ \frac{79-x}{79-69} & 69 < x < 79 \\ 0 & x \geq 79 \end{cases} \quad P_{mP}(x) = \begin{cases} 0 & x \leq 60 \\ \frac{x-60}{70-60} & 60 < x < 70 \\ 1 & 70 \leq x \leq 89 \\ \frac{91,5-x}{91,5-89} & 89 < x < 91,5 \\ 0 & x \geq 91,5 \end{cases} \quad (1)$$

$$P_{PR}(x) = \begin{cases} 0 & x \leq 80 \\ \frac{x-80}{90-80} & 80 < x < 90 \\ 1 & 90 \leq x \leq 94 \\ \frac{96-x}{96-94} & 94 < x < 96 \\ 0 & x \geq 96 \end{cases} \quad P_{MP}(x) = \begin{cases} 0 & x \leq 92,5 \\ \frac{x-92,5}{95-92,5} & 92,5 < x < 95 \\ 1 & 95 \leq x \leq 98 \\ \frac{99-x}{99-98} & 98 < x < 99 \\ 0 & x \geq 99 \end{cases} \quad P_{AP}(x) = \begin{cases} 0 & x \leq 96,5 \\ \frac{x-96,5}{99-96,5} & 96,5 < x < 99 \\ 1 & x \geq 99 \end{cases} \tag{2}$$

$$Q_R(x) = \begin{cases} 1 & x \leq 20 \\ \frac{30-x}{30-20} & 20 < x < 30 \\ 0 & x \geq 30 \end{cases} \quad Q_B(x) = \begin{cases} 0 & x \leq 15 \\ \frac{x-15}{30-15} & 15 < x < 30 \\ 1 & 30 \leq x \leq 40 \\ \frac{70-x}{70-40} & 40 < x < 70 \\ 0 & x \geq 70 \end{cases} \quad Q_O(x) = \begin{cases} 0 & x \leq 40 \\ \frac{x-40}{50-40} & 40 < x < 50 \\ 1 & x \geq 50 \end{cases} \tag{3}$$

Os resultados da fuzzificação para o *Caso 1*, $PPS_1 = 99$, $PPS_2 = 96$ e $PQS = 43$, em suas respectivas funções de pertinência, podem ser vistos nas Tabs. 2 e 3 e as Figs. 1 e 2 representam graficamente seus conjuntos difusos.

Tabela 2. GPs para a variável probabilidade, para o Caso 1.

	PPS_1	PPS_2
Improvável	0	0
Pouco provável	0	0
Medianamente provável	0	0
Provável	0	0
Muito Provável	0	1
Altamente provável	1	0

Tabela 3. GPs para a variável qualidade, para a Caso 1.

	PQS
Ruim	0
Boa	0,9
Ótimo	0,3

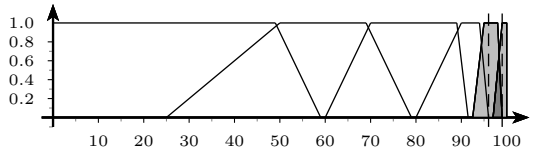


Figura 1. Fuzzificação para a variável probabilidade, no Caso 1.

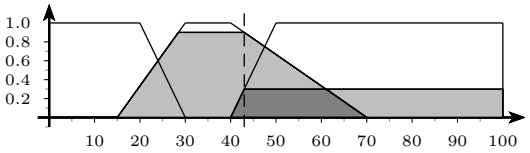


Figura 2. Fuzzificação para a variável qualidade, no Caso 1.

Para o *Caso 2*, o resultado da fuzzificação para $PPS_1 = 94$, $PPS_2 = 0$ e $PQS = 50$, pode ser visto nas Tabs. 4 e 5 com as representações nas Figs. 3 e 4.

Tabela 4. GPs para a variável probabilidade, para o Caso 2.

	PPS_1	PPS_2
Improvável	0	1
Pouco provável	0	0
Medianamente provável	0	0
Provável	1	0
Muito Provável	0,6	0
Altamente provável	0	0

Tabela 5. GPs para a variável qualidade, para o Caso 2.

	PQS
Ruim	0
Boa	0,67
Ótimo	1

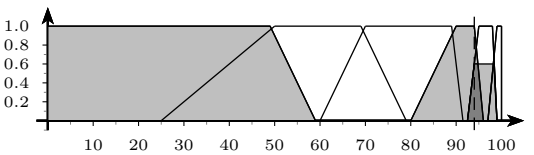


Figura 3. Fuzzificação para a variável probabilidade, no Caso 2.

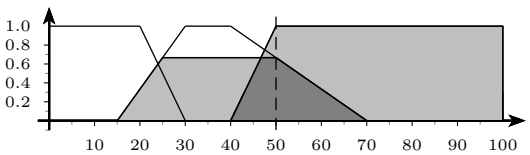


Figura 4. Fuzzificação para a variável qualidade, no Caso 2.

A *probabilidade*, para o *Caso 1*, é expressa pelos termos *muito provável* e *altamente provável*, e a *qualidade*, pelos termos *bom* e *ótimo* e, essas mesmas variáveis do *Caso 2*, pelos termos *improvável*, *provável*, *muito provável*, *bom* e *ótimo*.

3.2. Inferência

A inferência executa operações sobre os conjuntos difusos, com a combinação dos antecedentes das regras, a implicação e a aplicação do *modus ponens* generalizado, sendo, esse procedimento, feito em dois passos: a "agregação", que corresponde ao operador lógico E que executa a intersecção entre conjuntos e, portanto, determina o mínimo entre os valores disparados pelas regras, seguido da "composição".

Os modelos ("máquinas") de inferência adequados para esse SID, são os modelos de Mamdani ou de Larsen, visto que são sensíveis ao disparo de múltiplas regras sobre o conjunto de saída, quando, então, inicia-se o procedimento de defuzzificação, que começa com o segundo passo da inferência, a "composição", que é equivalente ao operador lógico OU e executa a união entre conjuntos, na qual o maior valor entre os mínimos resultantes da agregação é considerado para a defuzzificação.

Foram estabelecidas trinta e seis regras de inferência (Tab. 6), sendo que em metade dessas seus antecedentes são avaliados pelas variáveis *probabilidade* (PPS_1) e *qualidade* e, a outra metade, é avaliada pelas variáveis *probabilidade* (PPS_2) e *qualidade*. Essas regras, relacionam termos de entrada com a função de saída, expressa pelos termos *SNP descartado*, *SNP não confirmado* e *SNP confirmado*.

Tabela 6. Regras de inferência utilizadas no SID.

se	improvável	*e*	qualidade ruim	*então*	SNP descartado	(R_1)
	pouco provável		qualidade ruim		SNP descartado	(R_2)
	medianamente provável		qualidade ruim		SNP descartado	(R_3)
	provável		qualidade ruim		SNP descartado	(R_4)
	muito provável		qualidade ruim		SNP descartado	(R_5)
	altamente provável		qualidade ruim		SNP descartado	(R_6)
	improvável		qualidade boa		SNP descartado	(R_7)
	pouco provável		qualidade boa		SNP descartado	(R_8)
	medianamente provável		qualidade boa		SNP não confirmado	(R_9)
	provável		qualidade boa		SNP não confirmado	(R_{10})
	muito provável		qualidade boa		SNP confirmado	(R_{11})
	altamente provável		qualidade boa		SNP confirmado	(R_{12})
	improvável		qualidade ótima		SNP descartado	(R_{13})
	pouco provável		qualidade ótima		SNP descartado	(R_{14})
	medianamente provável		qualidade ótima		SNP não confirmado	(R_{15})
	provável		qualidade ótima		SNP não confirmado	(R_{16})
	muito provável		qualidade ótima		SNP confirmado	(R_{17})
	altamente provável		qualidade ótima		SNP confirmado	(R_{18})

No *Caso 1*, as funções de pertinência (Exps. 1, 2 e 3), resultam em $P_{MP} = 1$, para PPS_2, $P_{AP} = 1$, para PPS_1, $Q_B = 0,9$ e $Q_O = 0,3$ (Tabs. 2 e 3 e Figs. 1 e 2), então, a agregação é feita entre Q_B e Q_O, o que resulta no termo *ótima* para a variável *qualidade*. Os demais valores obtidos são iguais e, assim, não aplica-se a agregação, o que resulta em *muito provável* (PPS_2) e *altamente provável* (PPS_1), para *probabilidade*, que disparam as regras R_{17} e R_{18}.

Para o *Caso 2*, após a agregação, toma-se $P_{IM} = 1$ (PPS_2), $P_{MP} = 0,6$ (PPS_1) e $Q_B = 0,67$ que são levados à máquina de inferência, que dispara R_7 e R_{11}.

O modelo de inferência mapeia os antecedentes, resultantes da agregação, no termo conseqüente, que, para os modelos de Mamdani ou Larsen, representa uma função de saída em termos lingüísticos, exatamente como uma função de pertinência.

A função de saída que foi estabelecida, reduz as seis classes definidas para o *PPS* aos termos *SNP descartado* (SNP_D), *SNP não confirmado* (SNP_{NC}) e *SNP confirmado* (SNP_C), que, então, compõem a variável linguística *SNP* (Exps. 4):

$$SNP_D(x) = \begin{cases} 1 & x \leq 20 \\ \frac{30-x}{30-20} & 20 < x < 30 \\ 0 & x \geq 30 \end{cases} \quad SNP_{NC}(x) = \begin{cases} 0 & x \leq 15 \\ \frac{x-15}{30-15} & 15 < x < 30 \\ 1 & 30 \leq x \leq 40 \\ \frac{70-x}{70-40} & 40 < x < 70 \\ 0 & x \geq 70 \end{cases} \quad SNP_C(x) = \begin{cases} 0 & x \leq 40 \\ \frac{x-40}{50-40} & 40 < x < 50 \\ 1 & x \geq 50 \end{cases} \tag{4}$$

As regras R_{17} e R_{18}, disparadas no *Caso 1*, são processadas como:

1. R_{17} tem como antecedentes o valor *muito provável*, com $GP = 1$, e o valor *ótima*, com $GP = 0,3$; assim, a aplicação da regra mapeia o conseqüente *SNP confirmado*, com $GP = 1$ e $GP = 0,3$, isto é $SNP_C = 1$ e $SNP_C = 0,3$;
2. R_{18} tem como antecedentes o valor *altamente provável*, com $GP = 1$, e o valor *ótima*, com $GP = 0,3$; então, da mesma forma, mapeia o conseqüente *SNP confirmado*, com $GP = 1$ e $GP = 0,3$, isto é $SNP_C = 1$ e $SNP_C = 0,3$.

Com a aplicação das duas regras, cujos resultados foram coincidentes, apenas o termo *SNP confirmado* foi mapeado e o procedimento de composição deve ser tomado somente sobre esse termo. A composição busca o máximo entre os GPs de cada termo, no caso, somente sobre o termo *SNP confirmado*, fazendo $SNP_C = 1$.

Para o *Caso 2*, são disparadas as regras R_7 e R_{11}, que avaliam os valores antecedentes $P_{IM} = 1$ e $Q_O = 0,67$, para R_7, e $P_{MP} = 0,6$ e $Q_O = 0,67$, para R_{11}. A regra R_7 mapeia na função de saída o valor *SNP descartado*, com $GP = 1$ e $GP = 0,67$, enquanto a regra R_{11} mapeia na função de saída o valor *SNP não confirmado*, com $GP = 0,67$ e $GP = 0,6$. O termo *SNP confirmado* não foi mapeado, logo o procedimento de composição aplicado aos demais termos resulta em *SNP descartado*, com $GP = 1$ ($SNP_D = 1$), e *SNP não confirmado*, com $GP = 0,67$ ($SNP_{NC} = 0,67$).

As Figs. 5 e 6 representam, respectivamente, a aplicação das regras de inferência sobre a função de saída (Exps. 4) para os *Casos 1* e *2*.

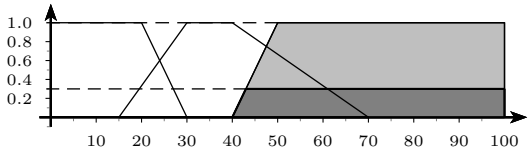


Figura 5. Aplicação das regras de inferência para o Caso 1.

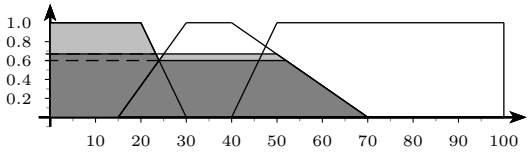


Figura 6. Aplicação das regras de inferência para o Caso 2.

3.3. Defuzzificação

A defuzzificação executa a composição, que determina os valores que representam cada um dos conjuntos mapeados na função de saída, e, a partir desses, calcula um valor preciso (VP), obtido com a aplicação do método de defuzzificação.

Para o modelo proposto, o método de defuzzificação deve considerar múltiplos disparos, pois o valor da qualidade da base no consenso é utilizada como um "valorizador" dos valores de probabilidade confrontados (PPS_1 e PPS_2). Assim, havendo

disparos múltiplos, esses devem ser avaliados, pois, servem à idéia de valorizar os conjuntos difusos estabelecidos na função de saída. Para esse fim, deve ser utilizado o método centro de máximo (*center of maximum* - COM) e, a partir dos modelos de inferência, aplica-se o método de defuzzificação adequado ao problema. Como o fuzzyMorphic.pl permite a inferência pelos modelos de Mamdani e Larsen, ambos podem ser aplicados e, juntamente com os valores tomados da composição, definem os valores para a defuzzificação.

O COM (Exp. 5), trata-se de uma média ponderada, onde o numerador é o somatório dos valores da composição (h_i), isto é, a altura do conjuntos de saída, multiplicados pelos valores no universo de discurso (u_i), encontrados pelo modelo de inferência, do seu respectivo conjunto de saída, e o denominador é o somatório das alturas (h_i).

Para o *Caso 1*, o VP (Exp. 6) e sua representação (Fig. 7) são iguais para os modelos de Mamdani e Larsen, mas, para o *Caso 2*, como conseqüência desses modelos, a defuzzificação apresenta diferentes resultados (Exps. 7 e 8 e Figs. 8 e 9).

$$VP = \frac{\sum h_i \cdot u_i}{\sum h_i} \qquad (5)$$

$$VP_{C_1} = \frac{75 \cdot 1}{1} = 75 \qquad (6)$$

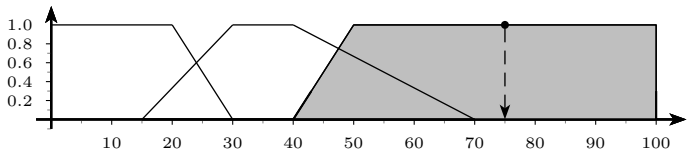


Figura 7. Aplicação do modelo de inferência, para o Caso 1.

$$VP_{C_2} = \frac{(10 \cdot 1) + (37,475 \cdot 0,67)}{1 + 0,67} = 21,02 \qquad (7)$$

$$VP_{C_2} = \frac{(10 \cdot 1) + (35 \cdot 0,67)}{1 + 0,67} = 20,03 \qquad (8)$$

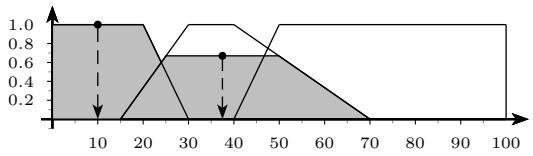


Figura 8. Aplicação do modelo de Mamdani para o Caso 2.

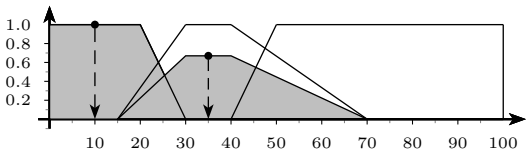


Figura 9. Aplicação do modelo de Larsen para o Caso 2.

3.4. Discussão sobre o modelo de inferência para identificação de SNPs

O *Caso 1* inicia com resultados prévios similares, 99% e 96% de probabilidades do ponto vir a ser um SNP, entretanto, o *Caso 2*, parte de resultados divergentes, 94% e zero. O SID incluiu um novo atributo, a qualidade da base no consenso, 43 e 50, para os *Casos 1* e *2*, respectivamente, ampliando as possibilidades de investigação e utilizando-se deste como um "valorizador" para a tomada da decisão. Assim, aos resultados prévios de o ponto vir a ser um SNP, acrescenta-se a qualidade do ponto, utilizando-os como as variáveis do modelo que permite a determinação de uma das três possibilidade excludentes: a confirmação do SNP, a eliminação dessa possibilidade ou, uma situação intermediária, sem elementos conclusivos para a confirmação ou o descarte dessa possibilidade.

A análise desses casos fornece elementos suficientes para apresentar o modelo, contudo, resultados efetivos são obtidos mediante a análise de conjuntos de dados, quando os valores inferidos a partir do modelo, podem, então, ser agrupados, determinando os conjuntos de pontos que melhor se ajustam às possibilidades investigadas. Estabelecer grupos é uma tarefa complexa, pois procura-se dizer como são e em quantas classes os dados se distribuem, sem o conhecimento a priori dos mesmos e, caso os valores se distribuam equitativamente no espaço, não caracterizando qualquer categoria, as classes podem não existir, uma vez que são definidas com base na semelhança entre os elementos, cabendo a verificação das possíveis classes para avaliar a existência de algum significado útil [de Carvalho 2005].

4. Conclusões

Critérios fixos e precisos de classificação, em geral, não são adequados, quando um estudo apresenta resultados próximos à divisão das classes, o que pode ser tratado por SIDs, que também são convenientes e possuem capacidade para tratar problemas que apresentam incerteza ou imprecisão para a tomada de decisão.

Ao adicionar um novo atributo aos resultados prévios, o modelo de inferência é capaz de decidir, de forma única, entre suas três possibilidades e, então, agrupá-las a partir de um algoritmo não-supervisionado e com estabelecimento dinâmico do número de grupos, esperando que o resultado desse agrupamento confirme o particionamento do conjunto em três grupos, não necessitando de limites fixos e precisos para a identificar possíveis SNPs.

Referências

Arbex, W. (2009). fuzzyMorphic.pl. 1 CD. Perl. Ambiente UNIX-like com GUI e interpretador Perl 5.0 ou posterior.

de Almeida, P. E. M. and Evsukoff, A. G. (2005). *Sistemas fuzzy*, pages 169–202. Manole, Barueri.

de Carvalho, L. A. V. (2005). *Datamining: a mineração de dados no marketing, medicina, economia, engenharia e administração*. Ciência Moderna, Rio de Janeiro.

Klir, G. J. and Yuan, B. (1995). *Fuzzy sets and fuzzy logic: theory and applications*. Prentice Hall, Upper Saddle River.

Nickerson, D. A., Taylor, S. L., Kolker, N., Sloan, J., Bhangale, T., Stephens, M., and Robertson, I. (2008). *Polyphred users manual*. University of Washington, Seattle. Version 6.15 Beta.

Tanscheit, R. (2007). *Sistemas fuzzy*, pages 229–264. Thomson Learning, São Paulo.

Zadeh, L. A. (1973). Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-3:28–44.

Composicionalidade e Reuso em Workflows Científicos com Propriedades Não-Funcionais

Vívian Medeiros, Antônio Tadeu Azevedo Gomes

Laboratório Nacional de Computação Científica (LNCC) – Petrópolis, RJ – Brasil
{vivian, atagomes}@lncc.br

Abstract. *In this paper we propose and illustrate the use of connectors as first-class constructs in workflow description languages. The aim is to demonstrate that it is possible to improve the reuse and compositionality of configurations involving non-functional properties related with the communication between workflow tasks.*

Resumo. *Neste artigo propomos e ilustramos o uso de conectores como construções de primeira classe em linguagens de descrição de workflows. O objetivo é demonstrar ser possível melhorar o reuso e a composicionalidade das configurações dos mecanismos que lidam com propriedades não-funcionais ligadas à comunicação entre tarefas de um workflow.*

1. Introdução

O aprimoramento da qualidade na execução de workflows tem sido objeto de vários trabalhos relacionados a sistemas gerenciadores de workflows científicos [Ludäscher et al., 2006; Fahringer et al., 2007; Zhao et al., 2008]. Em geral, tais trabalhos advogam a introdução, nesses sistemas, de mecanismos que lidam com propriedades não-funcionais de interesse típico para workflows científicos, como tolerância a falhas, rastreamento da proveniência de dados, desempenho e interoperabilidade.

A hipótese que pretendemos confirmar neste trabalho é que os sistemas gerenciadores existentes, como apresentado na Seção 2, são limitados no que se refere à expressividade das linguagens de descrição de workflows por eles oferecidas. Em particular, a configuração dos mecanismos responsáveis pelo tratamento das propriedades não-funcionais de interesse em um workflow científico ou inexiste nesses sistemas (que tratam essas propriedades de forma transparente ao usuário), ou é concentrado somente na descrição das tarefas (componentes computacionais) do workflow.[1] A transparência "não-seletiva" ao usuário torna mais simples a especificação de um workflow, porém oferece menor flexibilidade na configuração desses mecanismos. O foco da configuração desses mecanismos em tarefas, por sua vez, compromete o reuso e a composicionalidade dessas configurações, pois dificulta o encapsulamento das estratégias de comunicação e coordenação empregadas *entre* tarefas, e é sobre essas estratégias que vários desses mecanismos tipicamente atuam.

Nesse contexto, propomos o uso de *conectores* como construções de primeira classe em linguagens de descrição de workflows. O conceito de conector é bastante explorado na área de arquitetura de software para modelar tanto tipos quanto instâncias de interações entre componentes computacionais e regras que governam essas interações [Clements et al., 2007]. Na Seção 3, apresentamos uma linguagem de descrição de wokflows com suporte à especificação

[1] Tomamos como condição para nossa hipótese que um workflow em si também é visto pelo sistema gerenciador como uma tarefa composta por outras tarefas aninhadas.

de conectores com o objetivo de demonstrar que é possível melhorar o reuso e a composicionalidade das configurações dos mecanismos que lidam com propriedades não-funcionais ligadas à comunicação entre tarefas de um workflow científico. Na Seção 4 discutimos o estado atual de nosso trabalho e as perspectivas de desenvolvimento futuro no mesmo.

2. Trabalhos Relacionados

Vários trabalhos abordam o tratamento de propriedades não-funcionais em workflows científicos na literatura. Yu e Buyya [2006] expõem uma coletânea ampla – embora não exaustiva – de sistemas gerenciadores de workflows científicos. Porém, por questões de espaço, apresentamos somente alguns daqueles que trazem claramente algum suporte à configuração de mecanismos de tratamento dessas propriedades através de linguagens de descrição de workflows.

Os sistemas Gridbus [Yu e Buyya, 2004], QWE [Brandic et al., 2006] e ASKALON [Fahringer et al., 2007] definem linguagens de descrição de workflows com construções para a especificação de parâmetros de desempenho e de tolerância a falhas. Os sistemas Triana [Taylor et al., 2004], Kepler [Ludäscher et al., 2006] e Taverna [Oinn et al., 2006] oferecem suporte à configuração de mecanismos de rastreamento de proveniência de dados e de tolerância a falhas. Zhao et al. [2008] integram o sistema Swift [Zhao et al., 2007] com mecanismos para escalonamento eficiente de tarefas e rastreamento de proveniência de dados.

Em linhas gerais, percebe-se que os trabalhos apresentados oferecem algum suporte à composicionalidade de configuração dos mecanismos de tratamento de propriedades não-funcionais. Contudo, esse suporte é provido de forma *ad hoc* e sem a preocupação com o desenvolvimento *para* reuso de configurações. Nesse sentido, é importante destacar alguns trabalhos que ressaltam a importância de se oferecer primitivas para reuso de (partes de) workflows em sistemas gerenciadores de workflows [Qin et al., 2008; Gil et al., 2009]. Porém, nesses trabalhos o suporte para reuso é associado unicamente à configuração das tarefas, não havendo suporte para reuso de configuração dos mecanismos de tratamento de propriedades não-funcionais nas interações entre tarefas.

O uso de conectores e sua associação a propriedades não-funcionais em workflows organizacionais já foi explorado na literatura. Por exemplo, Han e Shim [2000] utilizam conectores para compartilhamento de dados entre tarefas, bem como para o registro de eventos e a geração de estatísticas relacionadas a esses dados. Contudo, conectores não são tratados como entidades de primeira classe, limitando a possibilidade de desenvolvimento para reuso.

3. Conectores em Workflows Científicos

A abordagem apresentada neste trabalho introduz *conectores* como construções de primeira classe na linguagem de descrição de workflows baseada em XML oferecida pelo framework CSBase [de Lima et al., 2006].[2] Nesta proposta, tarefas comunicam-se entre si obrigatoriamente por meio de conectores. Tarefas e conectores possuem interfaces denominadas, respectivamente, de *portas* e *papéis*. A composição de um workflow envolve a ligação de portas de tarefas a papéis de conectores.

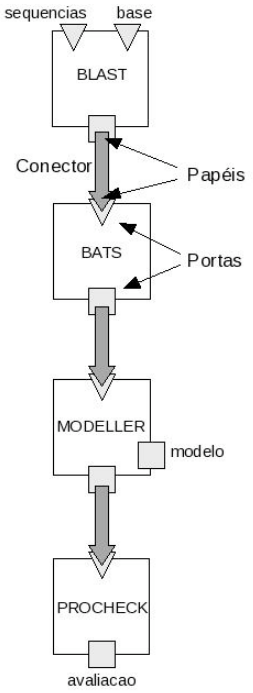
A linguagem considera a existência de dois tipos de usuários no processo de especificação de workflows científicos: *projetistas* e *cientistas*.

O *projetista* define tipos de tarefas e de conectores (desenvolvimento *para* reuso). Tipos de tarefas são usados para associar o conceito abstrato de tarefa no workflow com sua

[2]Esse framework oferece também uma ferramenta gráfica que auxilia na edição de descrições de workflows.

implementação concreta (um programa ou ferramenta, por exemplo). Tipos de conectores são usados para associar o conceito abstrato de conector no workflow com os mecanismos concretos que implementam as estratégias de comunicação e coordenação entre tarefas (p.ex. um *pipe* de caracteres, o transporte de um arquivo ou a automação de um fluxo). Por questões de espaço, não expomos neste artigo como tais tipos podem ser definidos em nossa linguagem.

O *cientista* descreve workflows a partir de relações entre instâncias de tipos de tarefas e conectores (desenvolvimento *com* reuso). Como exemplo, a Figura 1 ilustra uma simplificação do workflow MHOLine [Cavalcanti et al., 2003] descrita através do uso de nossa linguagem.




```
1   <fluxo nome='mholine'>
2     <tarefa id='0'><tipo nome='BLAST'/>
3       <porta id='0'><valor sequencias='in.fasta'/></porta>
4       <porta id='1'><valor base='pdb/'/></porta>
5       <porta id='2' nome='OUTPUT'/>
6     </tarefa>
7     <tarefa id='1'><tipo nome='BATS'/>
8       <porta id='0' nome='INPUT'/>
9       <porta id='1' nome='OUTPUT'/>
10    </tarefa>
11    <tarefa id='2'><tipo nome='MODELLER'/>
12      <porta id='0'/>
13      <porta id='1'/>
14      <porta id='2'><valor modelo='model.pdb'/></porta>
15    </tarefa>
16    <tarefa id='3'><tipo nome='PROCHECK'/>
17      <porta id='0'/>
18      <porta id='1'><valor avaliacao='eval.zip'/></porta>
19    </tarefa>
20    <conector id='0'><tipo nome='remoto'/>
21      <papel id='0'/>
22      <papel id='1'/>
23    </conector>
24    <conector id='1'><tipo nome='local'/><tipo nome='proveniencia'/>
25      <papel id='0'/>
26      <papel id='1'/>
27    </conector>
28    <conector id='2'><tipo nome='local'/>
29      <papel id='0'/>
30      <papel id='1'/>
31    </conector>
32    <ligacao>
33      <tarefa id='0' id_porta='2'/><conector id='0' id_papel='0'/>
34    </ligacao>
35    <ligacao>
36      <conector id='0' id_papel='1'/><tarefa id='1' id_porta='0'/>
37    </ligacao>
38    ... <!-- outras ligações -->
39  </fluxo>
```

Figura 1. Exemplo de workflow com conectores.

A partir da Figura 1 podemos observar duas características de nossa abordagem. Em primeiro lugar, a configuração do conector 1 (linhas 24-27) compõe dois tipos de mecanismos distintos: de distribuição de tarefas (tipo `local`) e de rastreamento de proveniência (tipo `proveniencia`). A composição entre esses tipos de mecanismos é independente da definição das tarefas. Em segundo lugar, os tipos de conectores `local` e `remoto` (linhas 20, 24 e 28) definem estratégias de distribuição de tarefas nos recursos computacionais disponíveis para a execução do workflow. A informação provida por esses dois tipos é útil aos mecanismos de escalonamento de tarefas usados pelo sistema gerenciador de workflows. Nesse caso, a identificação da *afinidade* de uma tarefa com um recurso [Brandic et al., 2006] está relacionada às características das tarefas, mas o desejo do usuário que as mesmas executem ou não em um mesmo recurso não pode ser declarado, de forma reusável e sem quebrar a composicionalidade do workflow, nem nas propriedades das tarefas nem nas do workflow como um todo.

4. Estado atual e Perspectivas

Neste trabalho apresentamos a hipótese de que os sistemas gerenciadores de workflows científicos existentes têm expressividade limitada no que se refere à configuração e composição

dos mecanismos responsáveis pelo tratamento das propriedades não-funcionais de interesse nesses workflows. Nesse contexto, propusemos e ilustramos o uso de conectores em linguagens de descrição de workflows, objetivando demonstrar que é possível melhorar o reuso e a composicionalidade das configurações desses mecanismos quando as propriedades não-funcionais correspondentes estão ligadas à comunicação entre as tarefas de um workflow.

Este trabalho, ainda em estágio preliminar, será aprofundado em diferentes frentes. Em primeiro lugar, precisamos definir mais claramente quais mecanismos devem poder ser definidos e configurados por meio de conectores. Em segundo lugar, precisamos testar nossa abordagem com outros workflows, identificando novas oportunidades e desafios no uso de conectores. Por fim, pretendemos explorar o conceito de *estilos arquiteturais* [Clements et al., 2007] em complemento a conectores para melhorar o nível de reuso de especificações de workflows.

Referências

Brandic, I., Pllana, S., e Benkner, S. (2006). An approach for the high-level specification of QoS-aware grid workflows considering location affinity. *Scientific Programming*, 14:231–250.

Cavalcanti, M. C., Baião, F., Rössle, S. C., Bisch, P. M., Targino, R., Pires, P. F., Campos, M. L., e Mattoso, M. (2003). Structural genomic workflows supported by web services. In *Proceedings of the 14th International Workshop on Database and Expert Systems Applications*, pages 45 – 49.

Clements, P., Bachmann, F., Bass, L., Garlan, D., Ivers, J., Little, R., Nord, R., e Stafford, J. (2007). *Documenting Software Architectures: Views and Beyond*. Addison Wesley.

de Lima, M. J., Ururahy, C., Lucia de Moura, A., Melcop, T., Cassino, C., dos Santos, M. N., Silvestre, B., Reis, V., e Cerqueira, R. (2006). CSBase: A framework for building customized grid environments. In *Proc. 15th IEEE WETICE*, pages 187–194, Washington, DC, USA. IEEE ComSoc.

Fahringer, T., Prodan, R., Duan, R., Hofer, J., Nadeem, F., Nerieri, F., Podlipnig, S., Qin, J., Siddiqui, M., Truong, H.-L., Villazon, A., e Wieczorek, M. (2007). ASKALON: A development and grid computing environment for scientific workflows. In *Workflows for e-Science*, pages 450–471. Springer.

Gil, Y., Groth, P., Ratnakar, V., e Fritz, C. (2009). Expressive reusable workflow templates. In *Proc. 5th IEEE E-SCIENCE*, pages 344–351, Washington, DC, USA. IEEE ComSoc.

Han, D. e Shim, J. (2000). Connector oriented workflow system for the support of structured ad hoc workflow. In *Proceedings of the 33rd Hawaii International Conference on System Sciences*, volume 6, page 6029, Los Alamitos, CA, USA. IEEE Computer Society.

Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., Lee, E. A., Tao, J., e Zhao, Y. (2006). Scientific workflow management and the Kepler system. *CCPE*, 18(10):1039–1065.

Oinn, T., Greenwood, M., Addis, M., Alpdemir, N., Ferris, J., Glover, K., Goble, C., Goderis, A., Hull, D., Marvin, D., Li, P., Lord, P., Pocock, M. R., Senger, M., Stevens, R., Wipat, A., e Wroe, C. (2006). Taverna: lessons in creating a workflow environment for the life sciences. *CCPE*, 18(10):1067–1100.

Qin, J., Fahringer, T., e Berger, M. (2008). Towards workflow sharing and reuse in the ASKALON grid environment. In *Proc. CGW*, pages 111–119, Cracow, Poland.

Taylor, I., Shields, M., e Wang, I. (2004). Resource management for the Triana peer-to-peer services. In *Grid resource management*, pages 451–462. Kluwer, Norwell, MA, USA.

Yu, J. e Buyya, R. (2004). A novel architecture for realizing grid workflow using tuple spaces. In *Proc. 5th IEEE/ACM GRID*, pages 119–128, Washington, DC, USA. IEEE ComSoc.

Yu, J. e Buyya, R. (2006). A taxonomy of workflow management systems for grid computing. *Journal of Grid Computing*, 3:171–200.

Zhao, Y., Hategan, M., Clifford, B., e Foster, I. (2007). Swift: Fast, reliable, loosely coupled parallel computation. In *Proc. IEEE SCW*, pages 199–206.

Zhao, Y., Raicu, I., Foster, I., Hategan, M., Nefedova, V., e Wilde, M. (2008). Realizing fast, scalable and reliable scientific computations in grid environments. In *Grid Computing Research Progress*, pages 1–40. Nova Science, Commack, NY, USA.

Human Transcriptional Regulation Interaction database (HTRIdb): Atualização 2010.

Luiz A. Bovolenta[1], **Marcio L. Acencio**[1], **Ney Lemke**[1]

[1]Instituto de Biociências de Botucatu – Universidade Estadual Paulista (UNESP)
CEP 18618-970 – Botucatu – SP – Brasil

{labovolenta,lemke,mlacencio}@ibb.unesp.br

Abstract. *In this paper we described the Human Transcription Regulation Interaction database (HTRIdb), an open-source database of transcription factor (TF)-regulated gene (RG) interactions with Pubmed reference to the experimental evidence of regulation that has been created in response to need of a user-friendly open-access repository of TF-regulated gene interactions. Currently, most TF-RG data are either dispersed in literature or found in paid subscription databases or open-access, non-user friendly databases. The HTRIdb allows users to easily retrieve and download TF-RG interaction data in spreadsheet or text format as well as to graphically visualize the interactions.*

Resumo. *Neste trabalho, nós descrevemos o desenvolvimento de um banco de dados de interações transcricionais de seres humanos experimentalmente verificados, denominado de Human Transcription Regulation Interaction database (HTRIdb), para suprir a necessidade de uma base de dados de acesso livre e mais completa que centralize os dados que atualmente se encontram dispersos na literatura biomédica ou em bancos de dados pagos ou de uso complexo. O HTRIdb permite aos usuários recuperarem os dados de uma interação regulatória específica de interesse ou todas as interações regulatórias presentes no banco de dados, fazerem download dos dados em diferentes formatos de arquivo e visualizarem graficamente as interações regulatórias e com outras proteínas.*

1. Introdução

O entendimento da regulação gênica exige um conhecimento preciso e abrangente das interações de regulação transcricional. A fim de responder à necessidade crescente de repositórios integrados de fatores de transcrição (FT) e seus respectivos genes regulados, vários bancos de dados de fatores de transcrição de organismos específicos foram desenvolvidos, tais como o *"C. elegans differential gene expression database"(EDGEdb)* [Barrasa et al. 2007], o *"Yeast Search for Transcriptional Regulators And Consensus Tracking"(YEASTRACT)* [Teixeira et al. 2006], o *"TRANSFAC® database"* [Matys et al. 2006] e o *"Transcriptional Regulatory Element Database"(TRED)* [Jiang et al. 2007]. Estes dois últimos bancos de dados contêm dados de interações de regulação transcricional de seres humanos, mas ambos possuem limitações: *TRANSFAC®* [Matys et al. 2006] é o principal banco de dados de FTs, mas o acesso é restrito com apenas uma parte do seu conteúdo disponível na versão pública; *TRED* [Jiang et al. 2007] é um banco de dados de acesso livre, mas contém um número limitado de FTs humanos (113 FTs de um total de ≈1500 FTs conhecidos). Para superar essas

limitações, foi desenvolvido o *Human Transcriptional Regulation Interactions database (HTRIdb)*, um banco de dados relacional de interações de FTs com seus genes regulados com referência para a prova experimental da regulação no *Pubmed* baseado no Sistema de Gerenciamento de Banco de Dados PostgreSQL. Até agora, *HTRIdb* foi populado com ≈100 FTs que regulam ≈2000 genes. O *HTRIdb* pode ser acessado através de uma interface web *(http://tinyurl.com/lbbc-htridb)* onde os usuários são capazes de recuperar os dados de uma interação regulatória específica de interesse ou todas as interações regulatórias presentes no banco de dados, Além disso, usuários podem fazer *download* e recuperar os dados em formato de planilha eletrônica ou em formato de texto. O *HTRIdb* também disponibiliza uma ferramenta capaz de apresentar de forma gráfica as interações regulatórias.

2. Metodologia

2.1. Banco de Dados

A primeira fase do desenvolvimento foi a modelagem de dados, ou seja, a transformação do problema do mundo real em uma estrutura abstrata capaz de demonstrar a estrutura que irá armazenar os dados. Essa fase é realizada através de conceitos descritos no Modelo Entidade Relacionamento (MER) criado por Peter Chen [Chen 1976]. Esta estrutura abstrata pode ser representada graficamente utilizando figuras geométricas, onde cada figura representa de maneira distinta os conceitos descritos no MER, conhecido como Diagrama Entidade Relacionamento (DER).

A segunda fase foi a criação do modelo físico e a codificação da estrutura do *HTRIdb* em SQL, que foi realizada com o auxílio de uma ferramenta case, *Case Studio 2*, responsável por auxiliar na elaboração do modelo e gerar os *scripts* SQL automaticamente que foram implantados no Sistema de Gerenciamento de Banco de Dados Objeto-Relacional (SGBDOR) PostgreSQL.

2.2. Interface

As páginas da interface com conteúdo estático foram desenvolvidas em HTML, e as páginas com conteúdo dinâmico foram desenvolvidas através dos recursos da linguagem Java (JavaBeans, JSP e Servlets). A programação das páginas com conteúdo dinâmico seguiu o modelo *MVC* (*Model View Controller*). Neste modelo, as páginas JSP são responsáveis pelas entradas de requisições e da visualização e formatação das respostas dos processamentos. Os Servlets são responsáveis pelo controle dos processos e o retorno das respostas para as páginas JSP. O modelo dos dados requisitados, por sua vez, são de responsabilidade do JavaBeans. Com este modelo, a lógica de programação e a visualização das informações ficam separadas.

Foram utilizadas ferramentas de tratamento de imagem para a criação das imagens presentes em cada uma das páginas da interface do banco de dados e recursos de CSS para a estruturação do *layout* da interface gráfica.

Para possibilitar a visualização gráfica das interações entre os FTs e seus genes alvos e outras proteínas, foi incorporada uma ferramenta denominada de Cytoscape Web, desenvolvida através de tecnologias Flash e JavaScript pelo laboratório de Gary Bader e Quaid Morris na Universidade de Toronto.

2.3. Coleta e inserção dos dados

Os dados das interações entre os FTs e seus genes alvos foram coletados manualmente a partir da literatura biomédica. Os dados coletados foram os seguintes:

- Nomes oficiais dos genes que codificam os FTs e dos genes regulados por esses fatores: *Human Genome Organization Gene Nomenclature Committe (HGNC)*.
- GeneIDs dos genes que codificam os fatores de transcrição e dos genes regulados por esses fatores: *National Center for Biotechnology Information (NCBI)*;
- Interações regulatórias entre FTs e seus genes alvos: literatura biomédica no *Pubmed*;
- Dados das proteínas que interagem fisicamente entre si [Costa et al. 2010].

3. Resultados

O resultado foi o desenvolvimento de um banco de dados de fatores de transcrição humanos disponibilizado gratuitamente por meio de uma interface web e desenvolvida com base na tecnologia Java.

A estrutura do banco de dados foi o primeiro resultado obtido durante o desenvolvimento. Este é demonstrado através do DER na Figura 1.

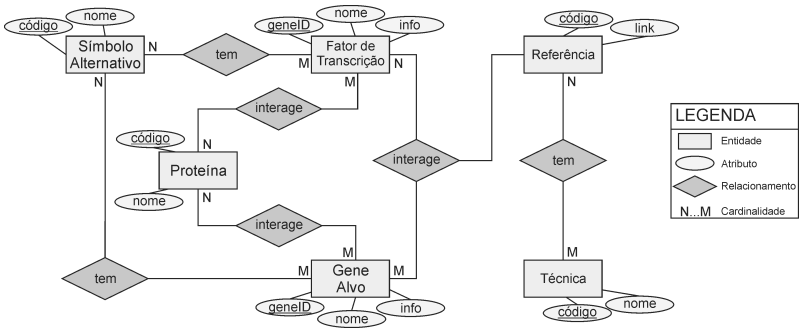
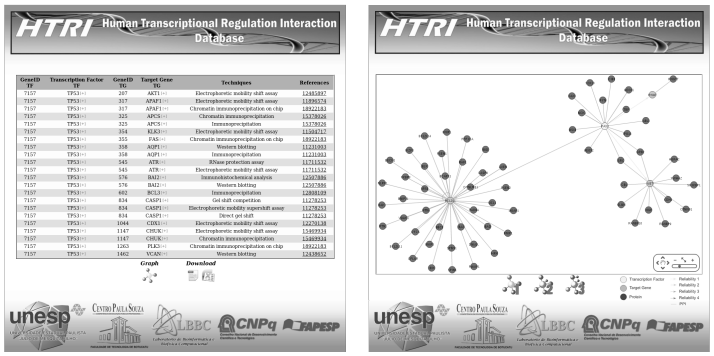


Figura 1. Diagrama Entidade Relacionamento do HTRIdb.

Outro resultado é a interface web, que possui recurso de busca que disponibiliza as seguintes informações relacionadas às interações entre os FTs e seus genes alvos em forma de uma tabela: os números identificadores e os símbolos oficiais dos genes que codificam os FTs e de seus genes alvos, os links direcionados para as páginas do *EntrezGene* contendo informações completas sobre os genes que codificam os FTs e seus genes alvos, as técnicas experimentais utilizadas para detectar as interações e os links direcionados para as referências no *Pubmed* que provam essas interações (Figura 2(a)).

O *HTRIdb* também oferece aos usuários recursos de exportação parcial ou integral dos dados presentes no banco de dados e a visualização gráfica das interações entre os FTs e seus genes alvos e com outras proteínas (Figura 2(b)).

Esse interface gráfica também disponibiliza uma página que permite especialistas a contribuir com novas descobertas relacionadas às interações regulatórias entre FTs e seus genes alvos.



(a) Página dos resultados de pesquisas realizadas.

(b) Página de visualização gráfica das interações.

Figura 2. Interface gráfica do HTRIdb

4. Conclusão

O banco de dados e os meios de busca do sistema compõem uma ferramenta capaz de auxiliar na gestão de informação para pesquisadores biomédicos e outros pesquisadores da área, permitindo com que estes pesquisadores encontrem informações sobre as interações entre FTs e genes regulados em humanos de maneira mais eficiente. Possibilita, também, um maior entendimento sobre a rede de interações entre FTs e seus genes alvos que controlam todos os processos celulares, promovendo novas descobertas que podem resultar na criação de novos medicamentos capazes de melhorar tratamentos ou, até mesmo, possibilitar a cura de doenças relacionadas aos processos anormais de transcrição gênica.

Referências

Barrasa, M. I., Vaglio, P., Cavasino, F., Jacotot, L., and Walhout, A. (2007). Edgedb: a transcription factor-dna interaction database for the analysis of c. elegans differential gene expression. *BMC Genomics*, 8(1):21.

Chen, P. P.-S. (1976). The entity-relationship model toward a unified view of data. *ACM Trans. Database Syst.*, 1:9–36.

Costa, P. R., Acencio, M. L., and Lemke, N. (2010). A machine learning approach for genome-wide prediction of morbid and druggable human genes based on systems-level data. *BMC Genomics*, 11:S9.

Jiang, C., Xuan, Z., Zhao, F., and Zhang, M. Q. (2007). Tred: a transcriptional regulatory element database, new entries and other development. *Nucleic Acids Res*, 35(Database issue):D137–40.

Matys, V., Kel-Margoulis, O. V., Fricke, E., Liebich, I., Land, S., Barre-Dirrie, A., Reuter, I., Chekmenev, D., Krull, M., Hornischer, K., Voss, N., Stegmaier, P., Lewicki-Potapov, B., Saxel, H., Kel, A. E., and Wingender, E. (2006). Transfac and its module transcompel: transcriptional gene regulation in eukaryotes. *Nucleic Acids Res*, 34(Database issue):D108–10.

Teixeira, M. C., Monteiro, P., Jain, P., Tenreiro, S., Fernandes, A. R., Mira, N. P., Alenquer, M., Freitas, A. T., Oliveira, A. L., and Sa-Correia, I. (2006). The YEASTRACT database: a tool for the analysis of transcription regulatory associations in Saccharomyces cerevisiae. *NUCLEIC ACIDS RESEARCH*, 34(Sp. Iss. SI):D446–D451.

SBCUP - III Simpósio Brasileiro de Computação Ubíqua e Pervasiva

PREFÁCIO

Desde o início desta década, tem-se intensificado as pesquisas em tecnologias relativas à proposta de Mark Weiser, 1991, chamada de Computação Ubíqua (*Ubiquitous Computing*) ou Pervasiva (*Pervasive Computing*), que propõe uma mudança do paradigma centrado em atividades do usuário final, resgatando um dos objetivos primordiais da computação em sua criação. Por prever uma mudança de rumo e uma inserção onipresente da computação no cotidiano, a computação ubíqua e pervasiva traz muitos desafios.

O contínuo progresso tecnológico dos dispositivos computacionais, relativos à capacidade de processamento e comunicação, indica que haverá uma total integração da computação nas atividades humanas. Em poucos anos, microprocessadores se tornarão pequenos e baratos o suficiente para serem embutidos em quase tudo – como em dispositivos digitais, carros, eletroeletrônicos, brinquedos, ferramentas, pequenos objetos e roupas. Todos esses artefatos podem estar interconectados por uma rede sem fio, trocando informações e provendo serviços de forma invisível e ciente ao contexto dos seres humanos ao seu redor. De fato, espera-se uma revolução onde uma escala de bilhões desses dispositivos, acoplando o mundo físico ao das informações digitais, possam tornar objetos, ambientes e serviços mais inteligentes e úteis aos seres humanos.

Desta forma, o SBCUP constitui um fórum dedicado à apresentação e discussão dos projetos em desenvolvimento na área de Computação Ubíqua e Pervasiva. O principal objetivo é congregar a comunidade científica que desenvolve trabalhos em temas relacionados e estimular a discussão em torno dos desafios que a área traz para o ambiente computacional do futuro.

O interesse pelo tema se comprova na presente edição do SBCUP. Em sua terceira edição, o evento recebeu 37 artigos científicos completos de alta qualidade, dos quais foram selecionados 13 trabalhos para publicação, caracterizando um taxa de aceitação de 35 %. Aliado a um comitê de programa de elevada representatividade da comunidade científica nacional, o evento se mostra em fase de consolidação e caminha para se tornar um dos mais respeitados do cenário nacional.

COMISSÕES

Coordenação Geral

André Luiz Lins de Aquino, IC-UFAL (Coordenador Geral)
Carlos Maurício Seródio Figueiredo, FUCAPI-UFAM (Coordenador de Comitê de Programa)
Cristiano André da Costa, Unisinos (Coordenador de Comitê de Programa)
Marcio Eduardo Kreutz, DIMAp-UFRN (Coordenador Local)

Comitê de Programa

Adenauer Yamin - UCPEL/UFPEL
Antonio Alfredo Ferreira Loureiro - UFMG
Bruno Schulze - LNCC
Carlos Ferraz - UFPE
Carlos Kamienski - UFABC
Carlos Maurício Seródio Figueiredo - FUCAPI
Claudio Geyer - UFRGS
Clever Farias - FFCLRP/USP
Cristiano André da Costa - Unisinos
Débora Barbosa - Learningware Tecnologia Educacional
Eduardo Nakamura - UFAM/FUCAPI
Fabio Costa - UFG
Fernando Dotti - PUCRS
Francisco José Silva - UFMA
Gerson Geraldo H. Cavalheiro - UFPEL
Giovani Rubert Librelotto - UFSM
Guilherme Travassos - COPPE/UFRJ
Iara Augustin - UFSM
Jorge Barbosa - Unisinos
Jorge Silva - Universidade de Coimbra (Portugal)
Lisandro Zambenedetti Granville - UFRGS
Luis Ferreira Pires - University of Twente (Países Baixos)
Marcelo Pias - University of Cambridge (Reino Unido)
Mario Dantas - UFSC
Markus Endler - PUC-RJ
Patricia Dockhorn Costa - UFES
Regina Araujo - UFSCar
Ricardo Neisse - Fraunhofer IESE
Ricardo Rabelo Oliveira - UFOP
Sérgio Kofuji - USP
Wanderley Lopes de Souza - UFSCar

ARTIGOS

Descoberta Semântica de Recursos na Computação Ubíqua

Renato M. Dilli[1], Sérgio L. Rodrigues[1], Nelsi Warken[2], Luthiano R. Venecian[3], João L. B. Lopes[1,5], Adenauer C. Yamin[3,4], Claudio F. R. Geyer[5]

[1]Instituto Federal Sul-rio-grandense (IFSul)

[2]Empresa Brasileira de Pesquisa Agropecuária (Embrapa)

[3]Universidade Católica de Pelotas (UCPel)

[4]Universidade Federal de Pelotas (UFPel)

[5]Universidade Federal do Rio Grande do Sul (UFRGS)

{dilli,sergio,jlopes_cavg}@ifsul.edu.br, nelsi.warken@cpact.embrapa.br

{venecian,adenauer}@ucpel.tche.br, geyer@inf.ufrgs.br

Abstract. *In ubiquitous environments, the resources must be shared so that they can be accessed anywhere and anytime. In this sense, the process of resource discovery plays an important role. This paper presents the EXEHDA-SD (EXEHDA-Semantic Discovery), a mechanism for resource discovery in ubiquitous computing. This mechanism combines in its architecture technologies for semantic processing of resources requests. This way, we intend to increase the expressiveness of the resource representation and query. The mechanism also provides the dynamics with which resources enter and leave the environment and include aspects such as scalability and user preferences.*

Resumo. *Em ambientes ubíquos, os recursos devem estar compartilhados para que possam ser acessados de qualquer lugar e a qualquer momento. Nesse sentido, o processo de descoberta de recursos assume um importante papel. Este artigo apresenta o EXEHDA-SD (EXEHDA-Semantic Discovery), um mecanismo para descoberta de recursos na computação ubíqua, que agrega em sua arquitetura tecnologias para o processamento semântico de requisições por recursos. Com isso, busca-se aumentar a expressividade na representação e consulta de recursos. O mecanismo também prevê a dinamicidade com que os recursos entram e saem do ambiente e contempla aspectos como escalabilidade e preferências do usuário.*

1. Introdução

Os ambientes ubíquos são compostos por uma grande quantidade de recursos heterogêneos, dispersos e de disponibilidade variável [Zhu et al. 2005].

Um mecanismo para descoberta de recursos deve considerar a dinamicidade com que os recursos entram e saem do ambiente computacional. Estes mecanismos devem gerenciar a relação entre os consumidores e provedores de recursos, permitindo a localização dos recursos disponíveis, com pouca ou nenhuma intervenção do usuário [Robinson and Indulska 2007].

O emprego de tecnologias semânticas eleva a expressividade na representação e consulta dos recursos, bem como permite a definição de regras para consistência e extensão das consultas. O uso de ontologias na representação dos recursos torna a con-

sulta mais precisa, evitando ambiguidade de conceitos e possibilitando a realização de inferência sobre o modelo ontológico [Soldatos et al. 2007].

Considerando este cenário, propomos o EXEHDA-*Semantic Discovery* (EXEHDA-SD), um mecanismo para descoberta de recursos com suporte semântico que visa qualificar a descoberta de recursos do *middleware* EXEHDA [Lopes et al. 2007] no ambiente ubíquo gerenciado pelo mesmo. Este ambiente é formado por células compostas por: (i) um EXEHDAbase, responsável por todos os serviços básicos da célula; (ii) EXEHDAnodos, são os equipamentos de processamento disponíveis no ambiente; (iii) EXEHDAmob-nodos, são os nós do sistema com elevada portabilidade.

O modelo proposto também busca suprir as demandas inerentes aos ambientes ubíquos, dentre as quais destacam-se: escalabilidade, preferências do usuário, notificação do cliente quando o recurso desejado estiver disponível, controle de acesso a recursos de acordo com o perfil do usuário e controle da disponibilidade de recursos através de intervalos de tempo [Costa et al. 2008].

Esse trabalho desenvolve-se no âmbito do grupo de pesquisa G3PD do PPGInf/UCPel, tendo sido um *position paper* com o andamento do mesmo apresentado no SBCUP 2010. O presente artigo traduz os resultados completos da pesquisa e está organizado nas seguintes seções: a seção 2 apresenta o modelo arquitetural baseado em processamento semântico; a seção 3 destaca um estudo de caso desenvolvido na Empresa Brasileira de Pesquisa Agropecuária (Embrapa); a seção 4 sistematiza e compara os trabalhos relacionados e a seção 5 apresenta as considerações finais.

2. Modelo Arquitetural

A arquitetura de software do EXEHDA-SD, apresentada na Figura 1, está organizada em três componentes distintos: Componente Cliente (CC), Componente Recurso (CR) e Componente Diretório (CD).

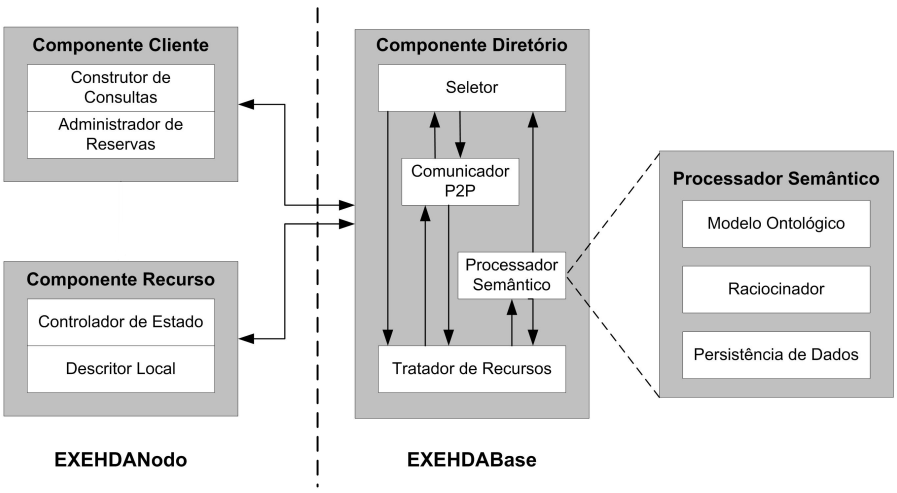


Figura 1. Modelo Arquitetural

2.1. Componente Cliente

O CC é responsável pela especificação da consulta aos recursos desejados, definindo os parâmetros a serem empregados no processamento da mesma. Dependendo da natureza

da consulta podem ser utilizadas as preferências do cliente, caracterizando assim, um mecanismo organizado em dois níveis. Os clientes são os componentes de software das aplicações em execução.

O CC é composto pelos seguintes módulos: (i) Construtor de Consultas, responsável por processar os requisitos para descoberta de recursos da aplicação, e gerar o arquivo XML com a especificação da correspondente pesquisa, bem como receber as respostas das consultas realizadas; (ii) Administrador de Reservas, responsável por notificar o cliente quando o diretório verificar que o recurso desejado tornou-se disponível.

2.2. Componente Recurso

O CR é responsável por notificar o estado atual recurso. Isso é feito através de troca de mensagens com o CD, considerando o intervalo de tempo especificado para o recurso (*lease*). Outra função do CR é anunciar os recursos dos nodos a medida que estes se deslocam no ambiente ubíquo. Quando um nodo ingressa em uma nova célula, o CR envia o arquivo OWL [McGuinness and van Harmelen 2009] com a descrição dos seus recursos para o CD da célula que está sendo visitada. O CD, por sua vez, adiciona as informações descritas em OWL na ontologia da célula que está sendo visitada. O CR é composto pelos módulos Controlador de Estado e Descritor local.

O Controlador de Estado tem por finalidade anunciar a disponibilidade do recurso no ambiente ubíquo. O anúncio é feito enviando uma mensagem ao Tratador de Recursos do Diretório. A técnica utilizada pelo mecanismo para controlar o estado dos recursos é *soft state*. Como o ambiente ubíquo pode conter uma grande quantidade de recursos, a melhor alternativa é os recursos se anunciarem para o CD. As mensagens de anúncio ocorrem após o CD confirmar o registro do recurso no diretório da célula ([Dilli 2010]).

O Descritor Local tem por objetivo armazenar e manter atualizado o arquivo OWL com as descrições do recurso localmente. Este arquivo é gerado pela interface de cadastro de recursos, localizada no diretório, no momento em que o recurso é instanciado na ontologia, ou são alterados atributos. Este módulo recebe um novo arquivo de descrições em OWL sempre que o recurso sofrer alterações no descritor, localizado no diretório. A cada ativação do recurso na célula, o Descritor Local envia o arquivo de descrições em OWL para o Tratador de Recursos.

2.3. Componente Diretório

O CD está localizado no EXEHDABase de cada célula do ambiente ubíquo. Esse componente é formado por quatro módulos descritos a seguir.

2.3.1. Tratador de Recursos

Quando um CC solicita que seja notificado sobre a disponibilidade de algum recurso é realizada uma consulta sobre todos os recursos “Ativos” no ambiente ubíquo. Caso a consulta não retorne resultados que atendam à requisição, a consulta será refeita fazendo uso dos recursos “Inativos”, mas existentes na ontologia. A consulta retornando recursos que satisfaçam a requisição, o Tratador de Recursos notifica o CC quando o CR do recurso desejado renovar o *lease*, alterando o estado do recurso para “Ativo”. No estado

"Ativo" o recurso estará disponível para ser alocado pelo *Resource Broker* do *middleware* EXEHDA.

O Controle de Recursos Ativos (CRA) é um repositório, mantido pelo Tratador de Recursos, onde são armazenados o identificador do nodo (Nodo_ID) e o intervalo de tempo gerenciado pelo Tratador de Recursos (*lease*). O CRA gerencia os recursos que entram e saem do ambiente celular. Quando um recurso entra no ambiente o CR envia uma mensagem para o Tratador de Recursos e este adiciona o Nodo_ID onde está localizado o recurso e o *lease* padrão definido pelo mecanismo. O CR precisa renovar o *lease* do recurso localizado no CD periodicamente, caso contrário o Tratador de Recursos irá remover o recurso do CRA.

Quando uma pesquisa é recebida pelo Tratador de Recursos, e no seu perfil está declarada a solicitação de notificação de recursos disponíveis, este módulo armazena no repositório Controle de Notificação de Recursos Disponíveis (CNR) (i) o Nodo_ID do recurso desejado, mas que não está disponível no momento; (ii) o ID do Cliente que solicitou a pesquisa; (iii) a consulta; (iv) a data e a hora que foi feita a pesquisa; (v) a data e a hora que irá expirar a espera pelo recurso desejado.

Quando o CR anuncia um recurso ao Tratador de Recursos, este verifica se o recurso está no CRA e renova seu *lease* ou adiciona o recurso com o *lease* padrão caso não esteja presente do CRA. Após, o Tratador de Recursos consulta o CNR para verificar se o recurso não está sendo esperado por algum cliente. Caso positivo, o cliente será notificado que o recurso está disponível.

Este módulo também tem por objetivo controlar o agendamento da disponibilidade do recurso, especificando, por exemplo, dias da semana e horários em que o recurso está disponível no ambiente.

2.3.2. Processador Semântico

O Processador Semântico emprega a API Jena [Dickinson 2009], sendo formado pelos seguintes componentes: (i) Modelo Ontológico (vide Figura 2), responsável pela manutenção da ontologia do mecanismo de descoberta. Esta ontologia é desenvolvida com a linguagem OWL. (ii) Raciocinador, responsável por aplicar regras e processar as consultas realizadas em SPARQL [Prud'hommeaux and Seaborne 2008]. O raciocinador procura por recursos idênticos e semelhantes ao solicitado; (iii) Persistência de Dados, responsável pelo armazenamento das triplas da ontologia no banco de dados PostgreSQL.

O modelo ontológico utilizado pelo Processador Semântico está fundamentado em duas ontologias: (i) OntUbi: desenvolvida pelo grupo G3PD da Universidade Católica de Pelotas para ser utilizada pelos mecanismos de consciência e adaptação de contexto e descoberta de recursos; (ii) OntSD: desenvolvida para atender exclusivamente o EXEHDA-SD. Esta ontologia integra-se à OntUbi através de vários relacionamentos entre classes.

O Processador Semântico é responsável pela instanciação de novos recursos e processamento de regras de inferência, possibilitando ao mecanismo de descoberta o reconhecimento de conceitos implícitos na ontologia. Para processar as regras é executado um código em Java que adiciona os conceitos deduzidos pelas regras na ontologia e realiza a consulta através da linguagem SPARQL, considerando o novo conceito. Através

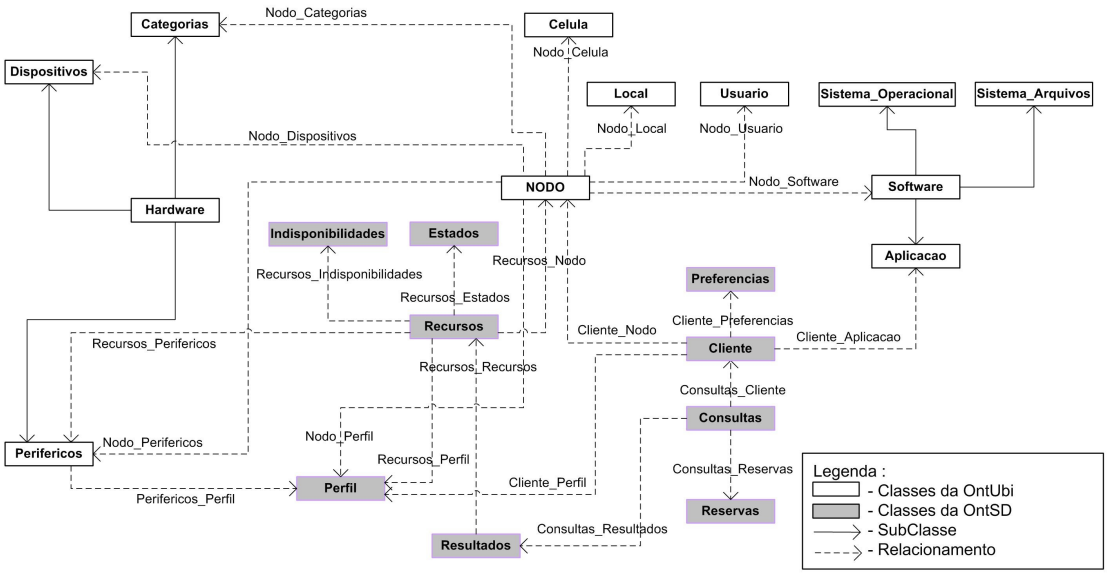


Figura 2. Modelo Ontológico

do suporte à persistência de dados da API Jena, as triplas da ontologia são armazenadas, tornando possível executar consultas SPARQL diretamente no banco de dados.

2.3.3. Comunicador P2P

O Comunicador P2P é o módulo responsável pela comunicação com as células vizinhas. A comunicação entre as células é realizada através de mecanismo P2P entre os EXEHDABase de cada célula. Utiliza-se uma variação do P2P puro, chamada *super peer* ([Dilli 2010]). Neste modelo a comunicação cliente/servidor ocorre apenas entre os diretórios localizados nos EXEHDABase de cada célula. Os CC e CR acessam apenas o diretório da célula local. O Comunicador P2P é responsável por repassar a pesquisa para as células vizinhas de acordo com o número de saltos definidos pelo CC.

O fluxo de informação entre os Comunicadores P2P está organizado em duas fases: (i) Configuração Celular: instanciação e presença de recursos no ambiente celular; (ii) Pesquisa por Recursos: nesta fase o fluxo de informação acontece a partir da consulta de recursos entre células, originada por um cliente. Essas fases estão organizadas em 11 etapas, conforme os itens a seguir. A Configuração Celular corresponde às etapas 1 a 3 e a Pesquisa por Recursos diz respeito às etapas 4 a 11.

1. Administrador do ambiente instancia um novo recurso no modelo ontológico e o Processador Semântico envia o arquivo no formato OWL com as descrições do recurso cadastrado ao Tratador de Recursos;
2. Tratador de Recursos envia o arquivo OWL para o módulo Descritor Local do CR que contém os recursos instanciados no CD;
3. Controlador de estado do CR envia uma mensagem ao Tratador de Recursos para anunciar sua presença no ambiente, tornado-se disponível;
4. Módulo Construtor de Consultas gera um arquivo XML contendo as características do recurso desejado pelo cliente e envia para o Tratador de Recursos, no CD

de sua célula;
5. Tratador de Recursos realiza a leitura do arquivo XML e identifica que o escopo de pesquisa envolve várias células e repassa este XML para o Comunicador P2P. O Tratador de recursos interpreta a consulta especificada em XML, gerando uma nova consulta na linguagem SPARQL e repassa ao Processador Semântico juntamente com as regras de inferência especificadas no arquivo XML, se existirem;
6. Processador Semântico processa a consulta e as regras de inferência no modelo ontológico da célula local e repassa os resultados da consulta para o Seletor. O Comunicador P2P envia o arquivo XML para células vizinhas utilizando tecnologia *super peer*. Também é verificado se o escopo de pesquisa envolve células estáticas definidas pelo administrador do ambiente;
7. Comunicador P2P de uma célula recebe o arquivo XML e a mensagem do P2P cliente contendo o ID da Base que originou a consulta. O arquivo XML é repassado para o Tratador de Recursos;
8. Tratador de Recursos interpreta o arquivo XML e gera a consulta na linguagem SPARQL e repassa ao Processador Semântico com regras de inferência, se especificadas;
9. Processador Semântico processa a consulta no modelo ontológico e repassa os resultados ao Seletor;
10. Seletor aplica as preferências do usuário, se especificadas, e entrega os resultados da célula ao Comunicador P2P;
11. Comunicador P2P decrementa a profundidade de pesquisa informada na mensagem recebida pelo P2P origem e repassa os resultados da pesquisa ao Comunicador P2P que solicitou a pesquisa.

2.3.4. Seletor

Quando uma pesquisa retorna mais de um recurso, o módulo Seletor faz a classificação e ordena os recursos, posicionando os que melhor satisfazem a requisição no topo da lista. O Seletor é responsável por receber e organizar as respostas das consultas realizadas em outras células, recebidas pelo módulo P2P. O processamento do resultado da consulta envolvendo várias células coloca no topo da lista os recursos que estiverem: (i) localizados na própria célula que originou a consulta; (ii) nas células vizinhas estáticas; (iii) em células com menor profundidade de pesquisa.

As preferências do cliente definidas no arquivo XML do Componente Cliente são consideradas pelo Seletor na composição do resultado da pesquisa.

3. Estudo de Caso

As funcionalidades do EXEHDA-SD tem sido avaliadas através de estudos de caso desenvolvidos na Embrapa Clima Temperado em Pelotas, RS. O estudo de caso apresentado nesse artigo explora os aspectos de escalabilidade, preferências do cliente, perfil de acesso e expressividade na representação e consulta por recursos através do Processador Semântico. Para a execução do cenário proposto a estrutura física da Embrapa é organizada em três células, conforme Figura 3, as quais são instanciadas no modelo ontológico, através do Tratador de Recursos.

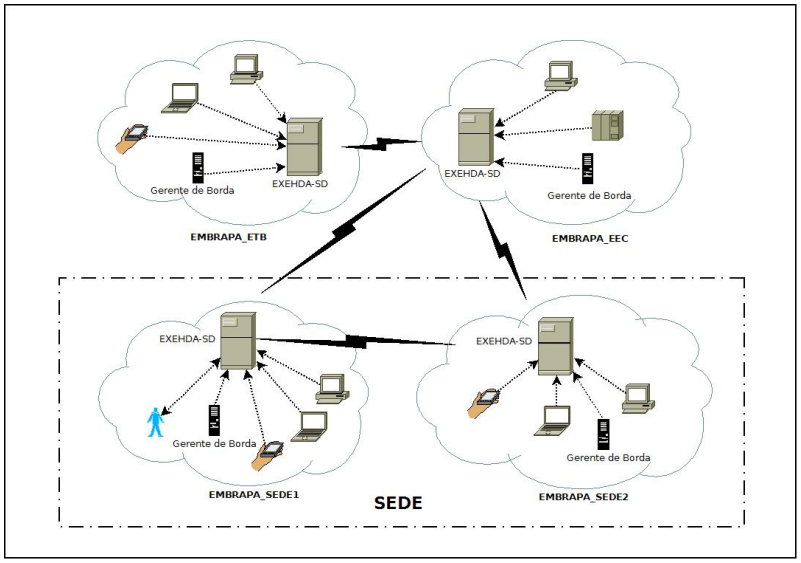


Figura 3. Organização Celular

No cenário proposto são gerenciados os recursos envolvidos nas casas de vegetação, as quais constituem ambientes monitorados e controlados para realização de experimentos pelos pesquisadores. O espaço disponível para utilização pelo pesquisador dentro da casa de vegetação é chamado de *slot*. Cada casa de vegetação possui um número limitado de *slots* que são considerados recursos a serem gerenciados pelo EXEHDA-SD. Os recursos gerenciados são instanciados no modelo ontológico através do administrador do ambiente com a interface disponibilizada pelo Tratador de Recursos.

Nesse cenário, um pesquisador localizado na célula EMBRAPA_SEDE1 executa uma aplicação, localizada em seu nodo, que tem por objetivo encontrar *slots* livres, em casas de vegetação, que possuam sensor e atuador de umidade. A regra de inferência construída pelo cliente cria um novo conceito no modelo ontológico relacionado a canteiros de vegetação que possuam pelo menos um atuador e um sensor de umidade. A regra é instanciada no modelo ontológico, conforme a Figura 4. A aplicação permite selecionar o perfil do cliente, a célula em que deseja pesquisar, o conceito deduzido (criado através de regras de inferência), a situação do *slot* no canteiro de vegetação (livre ou ocupado), local do canteiro, nome do canteiro, tipo de atuador e tipo de sensor.

Para processar a solicitação do pesquisador, o Componente Cliente gera o XML com as especificações de recursos necessárias para realizar a consulta e envia ao Tratador de Recursos do Componente Diretório da célula (informações adicionais sobre a forma de especificação dos recursos são encontradas em [Dilli 2010]). Por sua vez, o Processador Semântico traduz e processa a consulta de XML para SPARQL, envia para o Seletor aplicar as preferências do usuário. Por fim, o Tratador de Recursos envia no formato XML o resultado ao CC.

A Figura 5 mostra o resultado obtido pelo cliente. Os slots 3, 6 e 45 possuem sensores e atuadores de umidade. Os dois primeiros estão localizados na mesma célula (celula_10) em que o cliente estava no momento da solicitação da consulta. O slot 45, localizado na "celula_6", é exibido em função da profundidade de pesquisa definida pelo cliente (vide Figura 4), a qual altera o escopo da consulta para utilizar células vizinhas

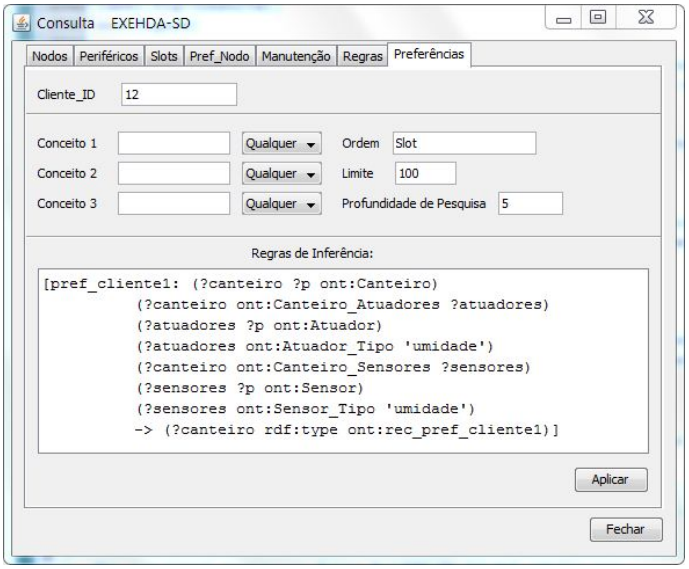


Figura 4. Preferências do Cliente

estáticas e dinâmicas. Como está definido o valor "5" na profundidade de pesquisa, os Comunicadores P2P replicam a consulta aos Comunicadores P2P de células vizinhas, até o quinto nível.

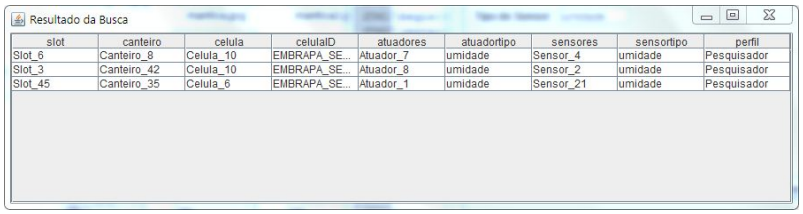


Figura 5. Resultado da Consulta

O EXEHDA-SD possibilita localizar recursos dispersos entre as células do ambiente ubíquo de forma escalável. Desta maneira o ambiente pode expandir-se de forma dinâmica, enquanto as aplicações executam, sem comprometer aspectos funcionais do mecanismo de descoberta.

4. Trabalhos Relacionados

[Allemand 2006] utiliza ontologias descritas em OWL para representação dos recursos e a linguagem RDQL (*RDF Data Query Language*) para consultar recursos instanciados no modelo ontológico. O projeto foi concebido para uso em ambientes de grade computacional. Esse projeto não considera as preferências do usuário, ao contrário do EXEHDA-SD. O EXEHDA-SD também diferencia-se pelo uso da linguagem de consulta SPARQL, considerada uma evolução da RDQL, sendo recomendada pelo W3C como padrão para realização de consultas em ontologias descritas com OWL.

DReggie [Chakraborty et al. 2001] consiste em um modelo para atuar com aplicações de comércio eletrônico em dispositivos móveis. Para isso é utilizada a linguagem DAML na descrição de recursos e um raciocinador Prolog para realizar *matching*

na base de conhecimento. O EXEHDA-SD utiliza raciocinadores disponíveis na API Jena e foi modelado para atuar tanto com dispositivos fixos como móveis, espalhados pelo ambiente ubíquo.

OMM [Tangmunarunkit et al. 2003] é um serviço de *matchmaking* em ambiente de grade. Este projeto utiliza tecnologias de Web Semântica para o *matching* de recursos no ambiente distribuído. As consultas são realizadas através da linguagem TRIPLE sobre ontologias descritas em RDF *Schema*. O EXEHDA-SD utiliza OWL para representação das ontologias. Essa linguagem é recomendada pela W3C e se traduz em uma evolução de outras linguagens, como RDF *Schema* e DAML. De modo diferente deste projeto e dos já referidos, o EXEHDA-SD tem uma maior abrangência de descoberta, com um escopo que envolve ambientes ubíquos e não somente ambientes de grade ou *mobile commerce*.

PerDiS [Schaeffer 2005] foi modelado para descoberta de recursos em ambientes ubíquos. Assim como o EXEHDA-SD, utiliza redes *super-peer* para localização de recursos entre células. Um diferencial entre o EXEHDA-SD e o PerDiS está na representação e consulta por recursos. No EXEHDA-SD isso é feito através de processamento semântico, empregando ontologia descrita com OWL e consulta com SPARQL. Já no PerDiS os recursos são representados em XML e a consulta é realizada pela comparação de atributos e valores, utilizando XML. Assim, ao contrário do EXEHDA-SD, o PerDiS permite apenas *matching* sintático e não possui motores de inferência para dedução de recursos.

A Tabela 1 apresenta uma comparação entre os trabalhos relacionados, incluindo o EXEHDA-SD. As seguintes características são consideradas na comparação: (i) a linguagem utilizada para descrição e consulta por recursos; (ii) a abrangência de descoberta do mecanismo; (iii) o emprego de algum tipo de preferência do usuário para a realização da consulta ou processamento de resultados.

Tabela 1. Comparação entre os Trabalhos Relacionados e o EXEHDA-SD

Modelos	Descrição	Consulta	Abrangência	Preferências
Allemand	OWL	RDQL	Grade	-
DReggie	DAML	Prolog	*m-commerce*	Sim
OMM	RDF *Schema*	TRIPLE	Grade	Sim
PerDiS	XML	XML	Ubíquo	Sim
EXEHDA-SD	OWL	SPARQL	Ubíquo	Sim

Além das diferenças destacadas nos parágrafos anteriores, o EXEHDA-SD contempla as seguintes características que não foram identificadas nos trabalhos relacionados: (i) perfil definido nos recursos e clientes para realizar o *matching*; (ii) notificação ao cliente quando o recurso desejado tornar-se disponível no ambiente ubíquo; (iii) agendamento de indisponibilidades para definir períodos de tempo em que o recurso não estará acessível no ambiente.

5. Considerações Finais

O principal diferencial do EXEHDA-SD em relação aos trabalhos relacionados diz respeito a maior expressividade na representação e consulta de recursos. Isso é obtido através da concepção de um modelo arquitetural baseado em ontologias, que provê processamento semântico das requisições por recursos.

Na continuidade da pesquisa do EXEHDA-SD os seguintes aspectos poderão ser explorados em trabalhos futuros: (i) contemplar a interoperabilidade de ontologias, possibilitando que cada célula possua um modelo ontológico diferenciado, mas equivalente; (ii) agregar novos algoritmos no processo de *ranking* utilizado pelo módulo Seletor; (iii) revisar e expandir constantemente o modelo ontológico.

Referências

Allemand, J. N. C. (2006). *Serviço Baseado em Semântica para Descoberta de Recursos em Grade Computacional*. Dissertação (mestrado em ciência da computação), UnB, Brasília, DF.

Chakraborty, D., Perich, F., Avancha, S., and Joshi, A. (2001). Dreggie: Semantic service discovery for m-commerce applications. In *Workshop on Reliable and Secure Applications in Mobile Environment, 20th Symposiom on Reliable Distributed Systems*.

Costa, C. A., Yamin, A. C., and Geyer, C. F. R. (2008). Toward a general software infrastructure for ubiquitous computing. *IEEE Pervasive Computing*, 7(1):64–73.

Dickinson, I. (2009). Jena ontology api. Disponível em: <http://jena.sourceforge.net/ontology/>. Acesso em abril de 2011.

Dilli, R. M. (2010). *Uma Proposta para Descoberta de Recursos na Computação Ubíqua com Suporte Semântico*. Dissertação (mestrado em ciência da computação), UCPEL, Pelotas, RS.

Lopes, J. L. B., Pilla, M. L., and Yamin, A. C. (2007). Exehda: a middleware for complex, heterogeneous and distributed applications. *Iberian-American Conference on Technology Innovation and Strategic Areas*.

McGuinness, D. L. and van Harmelen, F. (2009). Owl web ontology language overview. Disponível em: <http://www.w3.org/TR/owl-features/>. Acesso em abril de 2011.

Prud'hommeaux, E. and Seaborne, A. (2008). Sparql - a query language for rdf. Disponível em: <http://www.w3.org/TR/rdf-sparql-query/>. Acesso em abril de 2011.

Robinson, R. and Indulska, J. (2007). *Resource Discovery in Pervasive Computing Environments*. Handbook on Mobile Ad Hoc and Pervasive Communications. American Scientific Publishers.

Schaeffer, A. E. (2005). *PerDiS: um Serviço para Descoberta de Recursos no ISAM Pervasive Environment*. Dissertação (mestrado em ciência da computação), UFRGS, Porto Alegre, RS.

Soldatos, J., Stamatis, K., Azodolmolky, S., Pandis, I., and Polymenakos, L. (2007). Semantic web technologies for ubiquitous computing resource management in smart spaces. *Int. J. Web Eng. Technol.*, 3(4):353–373.

Tangmunarunkit, H., Decker, S., and Kesselman, C. (2003). Ontology-based resource matching in the grid - the grid meets the semantic web. In Fensel, D., Sycara, K. P., and Mylopoulos, J., editors, *International Semantic Web Conference*, volume 2870 of *Lecture Notes in Computer Science*, pages 706–721. Springer.

Zhu, F., Mutka, M., and Ni, L. (2005). Service discovery in pervasive computing environments. *IEEE Pervasive Computing*, 4(4):81–90.

Um *Framework* para o Gerenciamento de Aplicações Direcionadas à Medicina Ubíqua*

Sérgio L. Rodrigues[1], Renato M. Dilli[1], Nelsi Warken[2], Luthiano R. Venecian[3], João L. B. Lopes[1,6], Iara Augustin[5], Adenauer C. Yamin[3,4], Claudio F. R. Geyer[6]

[1]Instituto Federal Sul-rio-grandense (IFSul)
[2]Empresa Brasileira de Pesquisa Agropecuária (Embrapa)
[3]Universidade Católica de Pelotas (UCPel)
[4]Universidade Federal de Pelotas (UFPel)
[5]Universidade Federal de Santa Maria (UFSM)
[6]Universidade Federal do Rio Grande do Sul (UFRGS)

{sergio, dilli, jlopes_cavg}@ifsul.edu.br, nelsi@cpact.embrapa.br
august@inf.ufsm.br, {venecian, adenauer}@ucpel.tche.br, geyer@inf.ufrgs.br

Abstract. *Ubiquitous medicine aims to provide healthcare services at any time, without restriction of location, enhancing the mobility of health professionals. Considering this premise, in this paper we propose a framework that integrates sensors and computing devices (mobile or fixed) in a ubiquitous environment, providing context-awareness service. This framework, called uMED, and their target applications include the monitoring of patient's vital signs and the possibility of health professionals remotely control medical equipments. To evaluate the uMED, we present a case study related to clinical monitoring of patients.*

Resumo. *A medicina ubíqua objetiva prover serviços de saúde a qualquer momento, sem restrição de localização, melhorando a mobilidade dos profissionais de saúde. Considerando essa premissa, propomos nesse artigo uma infraestrutura que integra sensores e dispositivos computacionais (móveis ou fixos) em um ambiente ubíquo, provendo serviços de sensibilidade ao contexto. Esse framework, denominado uMED, e suas respectivas aplicações alvo contemplam o monitoramento de sinais vitais de pacientes e a possibilidade dos profissionais de saúde controlarem remotamente equipamentos médicos. Para avaliar o uMED, apresentamos um estudo de caso direcionado ao acompanhamento clínico de pacientes.*

1. Introdução

A computação ubíqua (ubicomp) prevê acesso a sistemas de informação, independente de tempo, localização e equipamento. Essa premissa vem sendo materializada pela disseminação de dispositivos portáteis, com acesso a redes sem-fio. Nesse sentido, o uso da ubicomp vem sendo potencializado em diversas áreas do mundo real, ampliando a usabilidade das aplicações tradicionais. Na área médica, diversas atividades podem ser atendidas por aplicações ubíquas, como telemonitoramento, diagnósticos remotos, segunda opinião, juntas médicas, *Home Care*, sistemas de prontuário eletrônico, etc. Esse

*trabalho parcialmente financiado pelo projeto Pertmed-Finep/MCT

cenário caracteriza uma integração entre medicina e ubicomp, e vem sendo denominado de medicina ubíqua [Diniz 2009].

Uma das motivações centrais para medicina ubíqua é a necessidade de ambientes computacionais que ofereçam suporte à mobilidade dos profissionais de saúde. A mobilidade dos médicos, especialmente, é algo inerente à própria profissão. Além desse caráter nômade do médico, a atividade médica é bastante fragmentada, estando sujeita a interrupções durante sua execução, uma vez que médicos passam pouco tempo em cada local ou atividade. Dessa forma, mecanismos que facilitem a continuidade das atividades durante seus constantes deslocamentos tendem a melhorar a produtividade dos mesmos [Tentori and Favela 2008].

A construção de aplicações ubíquas é uma tarefa complexa com custo elevado para desenvolvimento, gerenciamento e manutenção. Uma série de funcionalidades devem ser providas, abrangendo desde a aquisição de informações de contexto, a partir de um conjunto de fontes heterogêneas e distribuídas, até a representação dessas informações, armazenamento, processamento e raciocínio para seu uso em tomadas de decisão. Para reduzir essa complexidade, tem-se observado o emprego de infra-estruturas de suporte às aplicações, que passam a ser responsáveis pelo provimento destas funcionalidades [Costa et al. 2010].

Considerando esse cenário, apresentamos nesse artigo o *framework* uMED (*ubiquitous MEDicine*)[Rodrigues 2011] que objetiva suprir as funcionalidade de gerenciamento de aplicações direcionadas à medicina ubíqua. O uMED está inserido nos esforços de pesquisa do projeto PERTMED [PERTMED 2007] e desenvolve-se no âmbito do grupo de pesquisa G3PD do PPGInf/UCPel.

O artigo está organizado da seguinte forma: a seção 2 apresenta a arquitetura de software prevista para o uMED. A seção 4 destaca um estudo de caso. A seção 3 sintetiza os trabalhos relacionados e os compara com o uMED. Por fim, a seção 5 apresenta as considerações finais.

2. Arquitetura de Software

O *framework* uMED tem como premissa tratar as informações adquiridas por sensores, notificar as partes interessadas conforme as regras de tratamento de contexto utilizadas pelas diferentes aplicações, e possibilitar que os sensores e/ou os atuadores possam ser ativados, desativados, configurados e consultados a qualquer hora de forma ubíqua, além de permitir alterações das regras de tratamento de contexto durante a execução das aplicações.

O *framework* uMED utiliza o ambiente ubíquo provido pelo *middleware* EXEHDA. E explora as funcionalidades do subsistema de reconhecimento e adaptação ao contexto do mesmo [Venecian et al. 2010].

A arquitetura de software prevista para o uMED é apresentada na Figura 1. Essa arquitetura é composta por cinco módulos: Gerente de Atuação, Gerente de Aplicações, Gerente de Borda, Gerente de Comunicação e Servidor de Contexto. Os serviços da arquitetura são autônomos e cooperantes e possibilitam a coleta de dados, assim como o controle da ativação, desativação e configuração de atuadores. A arquitetura conta com diversos repositórios de dados, os quais são atualizados e/ou consultados durante a operação

da mesma.

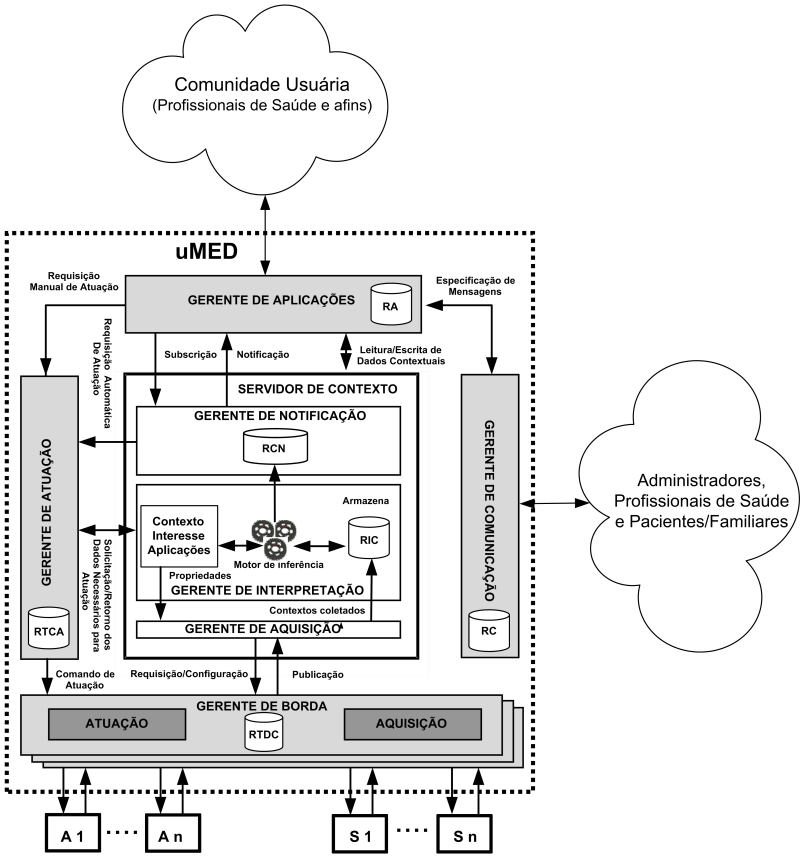


Figura 1. Arquitetura de Software do uMED

A arquitetura de software é alimentada por contextos de interesse das diversas aplicações, nos quais estão definidas as informações contextuais a serem adquiridas pelos sensores e processadas (traduzidas e deduzidas) pela arquitetura, bem como os comandos empregados no controle dos atuadores. Essas especificações são relacionadas por aplicação, componente e adaptador, considerando os seguintes parâmetros e regras:

- instanciação dos atuadores e sensores que participam das aplicações;
- parâmetros operacionais para controle da ativação, da desativação, bem como da configuração dos sensores e/ou atuadores;
- especificação do número máximo de registros pelos sensores a serem armazenados no repositório contextual. Ultrapassando este número, os valores mais antigos coletados pelos sensores são apagados;
- regras de deduções sobre os dados adquiridos pelos sensores;
- regras de deduções sobre os dados utilizados para controle dos atuadores (dados de ativação).

2.1. Gerente de Atuação

O Gerente de Atuação é responsável pelo controle da ativação, desativação e configuração dos atuadores empregados pelas aplicações gerenciadas pelo *framework*. Esse gerente trabalha com duas abordagens distintas: Atuação Manual e Atuação Automática.

Compreende-se como uma atuação manual aquela que é disparada pelo usuário da aplicação a qualquer instante, por exemplo, um profissional de saúde deseja ativar um dispositivo qualquer ou alterar algum parâmetro operacional do mesmo. Os dispositivos podem ser de gerência ambiental, tais como: aquecedores, umidificadores, alarmes, luzes de sinalização, bem como relacionados a demandas médicas: aparelho de oxigênio, bomba de infusão, ventilador pulmonar, dentre outros.

A atuação automática é disparada através de notificações produzidas pelo Servidor de Contexto, ou seja, o processamento de um determinado contexto de interesse pode implicar a ativação e/ou desativação e/ou configuração de um dispositivo. Os parâmetros operacionais necessários para atuação são definidos no próprio Servidor de Contexto.

Para tratamento da perda de conexão entre o Gerente de Atuação e o Gerente de Borda, existe um mecanismo de bufferização dos comandos emitidos (RTCA - Repositório Temporário de Comandos de Atuação - Vide Figura 1), o qual se vale de um parâmetro do tipo *time-to-live* para avaliar quais dos comandos ainda tem sentido serem tratados pelo Gerente de Borda.

2.2. Gerente de Aplicações

O Gerente de Aplicações é responsável por disponibilizar ao usuário final as aplicações que estão sendo gerenciadas pelo *framework* uMED. O Gerente de Aplicações trabalha com duas abordagens distintas: (i) componentes de software das aplicações assíncronos ao contexto; (ii) componentes de software das aplicações síncronos ao contexto. Ambas abordagens possuem características complementares na definição das aplicações ubíquas do uMED.

Os componentes de software das aplicações síncronos ao contexto são destinados ao monitoramento de sinais vitais de pacientes ou do ambiente relacionado aos mesmos, emitindo de forma automatizada, diferentes níveis de alertas, em função dos dados coletados, para os profissionais de saúde. Por sua vez, os componentes assíncronos ao contexto viabilizam as seguintes funcionalidades: (i) manipulação instantânea de sensores e atuadores; (ii) atuação no ambiente ubíquo através de alterações nas configurações operacionais dos atuadores; (iii) concessão ao usuário da aplicação da possibilidade de criar relatórios personalizados.

2.3. Gerente de Borda

O Gerente de Borda é responsável pelo primeiro processamento dos dados capturados pelos sensores, bem como o tratamento final das informações utilizadas para controle dos atuadores. Os dados adquiridos pelos sensores passam por várias etapas, nas quais são feitos tratamentos em níveis crescentes de abstração, culminando com a persistência dos mesmos no RTDC (Repositório Temporário de Dados Coletados - Vide Figura 1). O dado persistido é publicado no Servidor de Contexto para ser analisado [Santos et al. 2011].

A arquitetura contempla que os contextos possam ser capturados por diferentes Gerentes de Borda, cada um deles com um conjunto particular de sensores e/ou atuadores, com regimes operacionais individualizados. Esses Gerentes de Borda podem se valer da Internet e operar em diferentes instituições.

O controle dos atuadores coordenado pelo Gerente de Atuação, emprega os parâmetros operacionais especificados nos contextos de interesse das aplicações.

No Gerente de Borda, sensores e/ou atuadores podem ser implementados por meio de arquivos de configuração independentes. Ficando facilitadas desta forma as operações de adição e/ou substituição de sensores e atuadores.

2.4. Gerente de Comunicação

O Gerente de Comunicação centraliza a relação do uMED com os diferentes segmentos de usuários através de interfaces pró-ativas. Seu objetivo é notificar informações dos pacientes que estão sob monitoramento, através de mecanismos de uso rotineiro pelos envolvidos.

Com este serviço é possível notificar os profissionais de saúde, pacientes e seus familiares através de três tipos de mensagens: (i) Mensagens instantâneas através da rede celular, SMS - *Short Message Service* (on-line); (ii) Mensagens através do Google Talk (on-line); (iii) Mensagens por e-mail (off-line).

Com intuito de caracterizar para o profissional de saúde responsável o estado das mensagens enviadas, o Gerente de Comunicação contempla um repositório (RC - Repositório de Comunicações - Vide Figura 1), no qual é registrado o sucesso ou não do envio dos três tipos de mensagem.

2.5. Servidor de Contexto

O Servidor de Contexto é responsável pelo processamento das informações de contexto, empregando suporte semântico na execução destas funcionalidades [Venecian et al. 2010].

No uMED, de forma específica, o Servidor de Contexto realiza as seguintes tarefas: (i) receber informações de contexto do Gerente de Borda; (ii) processar as regras de deduções; (iii) notificar ao Gerente de Aplicações conforme o contexto de interesse das aplicações; (iv) interagir com Gerente de Atuação.

No Servidor de Contexto existe a possibilidade das regras de tratamento contextual serem modificadas durante a execução. Dentre outros aspectos este perfil operacional garante uma operação ininterrupta, o que se mostra oportuno, considerando que um mesmo Servidor de Contexto pode atender simultaneamente a várias aplicações.

3. Estudo de Caso

O estudo de caso empregado para explorar o suporte oferecido pelo uMED consiste em uma aplicação direcionada à área de medicina ubíqua denominada ACP (Acompanhamento Clínico de Pacientes), cujas funcionalidades são típicas desta área. Para ter liberdade na manipulação dos dados durante os testes é utilizada uma base de dados sensoriados criada pelo G3PD especificamente para este estudo de caso.

A premissa buscada é qualificar o acompanhamento clínico dos pacientes, que não estejam internados em UTI (Unidade de Tratamento Intensivo), com o intuito de aumentar o período de tempo que o paciente permaneça sob cuidado e evitar deslocamentos desnecessários pelos profissionais de saúde para saber o estado em que se encontra um paciente. Nesta perspectiva, os objetivos contemplados na aplicação ACP são:

- exibir dados de pacientes adquiridos dinamicamente por mecanismo de coleta de sinais vitais;
- emitir, de forma automatizada, diferentes níveis de alertas, em função dos dados coletados, para os profissionais de saúde;
- integrar o serviço de alertas da aplicação a rede aberta de comunicação *Google Talk*, SMS e e-mail;
- prover possibilidade de uso, tanto a partir de dispositivo móveis, como de mesa;
- permitir acesso ubíquo ao histórico dos dados coletados dos pacientes pelos profissionais de saúde;
- permitir acesso ubíquo ao histórico dos dados de atuação nos pacientes pelos profissionais de saúde;
- atuar remotamente em dispositivos médicos que estejam associados ao paciente.

A forma como está organizada a aplicação ACP permite sua integração aos demais sistemas de informações hospitalares. As informações cadastrais são de quatro naturezas: (i) dados cadastrais de pacientes; (ii) dados dos profissionais de saúde (médicos e enfermeiros); (iii) dados aquisitados dinamicamente pela rede de sensores de sinais vitais de pacientes; (iv) dados de ativação de atuadores.

Os profissionais de saúde, se desejarem, podem disparar o Módulo Aquisição do Gerente de Borda e capturar em determinado instante os dados correspondentes aos sinais vitais dos pacientes, bem como ativar o Módulo Atuação do Gerente de Borda e atuar remotamente em dispositivos associados ao tratamento do paciente. Este procedimento manual não interfere nas rotinas automáticas de captura de dados.

Os sinais capturados para este estudo de caso são frequência cardíaca, pressão arterial e temperatura corporal. Em função dos valores dos sinais coletados são produzidos diferentes níveis de alertas aos profissionais de saúde. Dependendo do nível de alerta é disponibilizada a opção de envio de mensagens aos profissionais de saúde ou, ainda, no caso de alerta máximo, o envio incondicional de mensagens.

Na aplicação ACP é possível monitorar de forma pró-ativa os pacientes, observando os últimos alertas disparados (vide Figura 2). Os níveis de alerta são classificados pela cor do alerta, da seguinte forma: (i) Alerta Verde: sinais vitais normais. Todos os sinais monitorados se encontram na faixa aceitável; (ii) Alerta Amarelo: sinal vital irregular. Um dos sinais monitorados se encontra em uma faixa irregular; (iii) Alerta Vermelho: sinais vitais críticos. Dois ou mais sinais monitorados se encontram em uma faixa irregular, este alerta é considerado o alerta máximo.

Os sensores associados ao paciente podem ser configurados através de uma interface disponibilizada ao profissionais de saúde. Através da mesma são selecionados o tipo de sensor e determinados seus parâmetros operacionais.

Com as informações armazenadas dos sensores e atuadores é possível a geração de relatórios de histórico clínico, proporcionando estudos mais aprofundados dos casos clínicos dos pacientes (vide Figura 3).

Na aplicação ACP é possível configurar o envio de notificações aos profissionais de saúde para cada nível de alerta. São explorados explorados os três tipos de notificações existentes, sendo que no alerta vermelho é obrigatória a escolha de pelo menos um tipo.



Figura 2. Monitoramento Pró-ativo de Pacientes

4. Trabalhos Relacionados

No levantamento do estado-da-arte foram considerados projetos que constituem uma parcela representativa do que vêm sendo desenvolvido nos últimos anos para atender as demandas da medicina ubíqua, e cuja arquitetura de software contempla o emprego de um middleware. Os projetos ABC [Pedersen et al. 2006], Awareness [Wegdam 2005], UbiDoctor [Diniz 2009] e ClinicSpace [da Silva 2010] são sintetizados na tabela 1. A revisão sistematizada destes projetos, foi central na definição das principais características previstas na modelagem do uMED.

Tabela 1. Síntese dos Trabalhos Relacionados

Características	ABC	Awarenes	UbiDoctor	ClinicSpace
Mobilidade Remota		X	X	X
Adaptação de Conteúdo	X		X	X
Sensibilidade ao Contexto	X	X	X	X
Processamento Semântico				X
Intervenção no Ambiente Ubíquo				X
Notificação Automática	X	X		X
Descoberta de Recursos	X	X		
Personalização de Regras				X

A concepção do uMED prevê uma arquitetura que contempla serviços cooperantes para atender as tarefas de monitoramento de dados contextuais, de atuação no meio e emissão de alertas aos profissionais de saúde. Esse modelo arquitetural não é encontrado nos projetos pesquisados, os quais, ao contrário do uMED, empregam serviços específicos no atendimento das diferentes funcionalidades.



Figura 3. Relatório de Histórico Clínico

Assim como o uMED, os demais projetos estudados contemplam uma arquitetura para suporte à sensibilidade ao contexto, modelada para ser expansível, tanto no que diz respeito a captura de dados do ambiente ubíquo, bem como quanto aos possíveis consumidores de contextos de interesse.

Nos projetos pesquisados, ao contrário do uMED, não existem mecanismos para criação de relatórios personalizados pelos usuários, possibilitando a geração de dados para estudo dos casos clínicos.

Ao contrário do uMED, os trabalhos relacionados não facultam a criação de estratégias que permitam a inserção ou remoção de sensores e/ou atuadores de forma que sejam abstraídas as especificidades dos mesmos.

O uMED disponibiliza um mecanismo de tolerância a falhas de comunicação nos gerentes de borda, o qual persiste os dados adquiridos pelos sensores, garantindo sua disponibilização ao servidor de contexto quando forem restabelecidas as comunicações. Somente o projeto Awareness possui um mecanismo similar de tolerância a falhas nos sensores.

O uMED disponibiliza uma interface para os profissionais de saúde especificarem o recebimento de alertas através de mensagens, esta interface permite ao profissional de saúde determinar as suas preferências quanto aos tipos de mensagens. Apenas o projeto ClinicSpace viabiliza funcionalidade análoga ao envio de alertas através de mensagens por e-mail.

Todos os projetos relacionados e o uMED possuem funcionalidades que permitem ao usuário realizar leitura instantânea de sensores e especificar os parâmetros operacionais para publicação dos sensores pertencentes as aplicações.

O uMED permite a especificação de características funcionais que possibilitam ao usuário realizar alterações de regras em tempo de execução, manipular de forma instantânea os atuadores, promover alteração nos parâmetros de configuração dos atuadores remotamente e especificar os parâmetros operacionais para configuração dos atuadores pertencentes as aplicações. Estas características estão presentes somente no projeto ClinicSpace.

5. Considerações Finais

Com o intuito de colaborar para superar alguns dos desafios identificados para a área de medicina ubíqua, propomos um *framework* para gerenciamento de aplicações ubíquas sensíveis ao contexto direcionadas a esta área. Consideramos que o emprego do *framework* uMED em um sistema de apoio a vida pode possibilitar uma diminuição no nomadismo e fragmentação de atividades exercidas pelos profissionais de saúde.

A arquitetura de software proposta contempla o monitoramento dos sinais vitais do paciente e/ou condições do seu ambiente, tendo a possibilidade de gerar alertas conforme regras definidas pelos próprios profissionais de saúde, bem como produzir relatórios personalizáveis, tanto com informações gerenciais como destinadas a estudo de casos clínicos. As regras de monitoramento podem ser alteradas em tempo de execução. Entende-se por tempo de execução a flexibilidade da arquitetura de software em reconfigurar dinamicamente as regras para manipulação e processamento das informações contextuais.

O uMED ainda possibilita a atuação no ambiente. Com base nos dados obtidos através do monitoramento, o profissional de saúde pode decidir por promover o acionamento, a configuração ou o desligamento de algum equipamento ou dispositivo.

Resumidamente, entendemos que a contribuição central do trabalho consiste na concepção de um *framework* direcionado a aplicações da medicina ubíqua, que trata de modo sinérgico quatro frentes: (i) a aquisição de dados monitorados; (ii) a atuação no ambiente ubíquo; (iii) a gerência dos diferentes níveis de alerta, conforme processamento das regras contextuais; (iv) a geração de dados para estudo dos casos clínicos.

Na continuidade da pesquisa do uMED destacam-se os seguintes aspectos: (i) ampliar o estudo de padrões de sensores e atuadores no desenvolvimento de aplicações direcionados a área médica; (ii) utilizar um sistema especialista no processamento de regras contextuais no Servidor de Contexto; (iii) reavaliar os protocolos e procedimentos empregados na troca de informações entre os Gerentes.

Referências

Costa, C. A., Silva, L. C. d., Barbosa, J. L. V., Yamin, A. C., and Geyer, C. F. R. (2010). A primer of ubiquitous computing challenges and trends. In Neto, F. M. M. and Neto, P. F. R., editors, *Designing Solutions-Based Ubiquitous and Pervasive Computing: New Issues and Trends*, volume 1, chapter 15, pages 282–303. IGI Global Publishing, Hershey.

da Silva, F. L. (2010). ClinicSpace: modelagem de uma ferramenta-piloto para definição de tarefas clínicas em um ambiente de computação pervasiva baseado em tarefas e direcionado ao usuário-final. Dissertação de mestrado em ciência da computação, PPGI/UFSM, Santa Maria, RS.

Diniz, J. (2009). *UbiDoctor: Arquitetura de Serviços para Gerenciamento de Sessão e Adptação de Conteúdo em Ambientes de Medicina Ubíqua*. Tese (doutorado em ciência da computação), Universidade Federal de Pernambuco, UFPE, Recife, PE.

Pedersen, J. B., Mogensen, M., and Bardram, J. E. (2006). The abc adaptive fusion architecture. *Proceedings of the 4th international workshop on Middleware for Pervasive and Ad-Hoc Computing (MPAC 2006)*, 182.

PERTMED (2007). Sistema de telemedicina móvel. Disponível em: <http://olaria.ucpel.tche.br/pertmed/>. Acesso em: março de 2011.

Rodrigues, S. L. (2011). umed: Uma arquitetura para desenvolvimento de software direcionada à medicina ubíqua. Dissertação de mestrado em ciência da computação, PPGINF/CPOLI/UCPEL, Pelotas, RS.

Santos, I., Rodrigues, S., Venecian, L., Lopes, J., and Yamin, A. (2011). Exehda-gb: Um mecanismo para captura e modificação do estado de contextos na computação ubíqua. In *Anais da 11ª Escola Regional de Alto Desempenho*, Porto Alegre, RS.

Tentori, M. and Favela, J. (2008). Activity-aware computing for healthcare. *IEEE Pervasive Computing*, 7(2):51–57.

Venecian, L. R., Lopes, J. L. B., Augustin, I., Yamin, A. C., and Geyer, C. (2010). Um mecanismo para sensibilidade ao contexto com suporte semântico na computação ubíqua. In *XXXVI Latin American Informatics Conference - CLEI*, Asuncion, Paraguay.

Wegdam, M. (2005). Awareness: A project on context aware mobile networks and services medical systems international. *In the Proceedings of the 14th Mobile and Wireless Communications Summit*, pages 19–23.

Uma Proposta de Modelo de Pagamento Móvel para o Comércio Ubíquo

Alex Roehrs[1], Cristiano André da Costa[1], Jorge Luis Victória Barbosa[1]

[1]Programa de Pós-Graduação em Computação Aplicada (PIPCA)
Universidade do Vale do Rio dos Sinos (UNISINOS)
Av. Unisinos, 950 - B. Cristo Rei / CEP 93.022-000 - São Leopoldo – RS - Brazil

alexroehrs@gmail.com, {cac, jbarbosa}@unisinos.br

Abstract. *This paper aims at presenting an architectural model proposal for a novel Mobile Payment System, called 4iPay. This work considers the following premises: independence of device, location, carrier and cardholder to meet the needs of executing payment transactions in ubiquitous commerce. The paper describes the implementation, resources, security requirements, reliability, and some existing architectural models for electronic commerce in ubiquitous environments. Our proposal considers the convergence of various concepts, including ubiquity, unity, universality, and unison.*

Resumo*. Este trabalho tem por objetivo apresentar uma proposta diferenciada de modelo de arquitetura para um sistema de pagamento móvel (Mobile Payment System), denominado 4iPay. O trabalho considera as premissas de independência de dispositivo, de localização, de operadora de telefonia e de administradora para suprir as necessidades de efetivação de transações de pagamento em comércio ubíquo (Ubiquitous Commerce). O trabalho descreve a implementação, os recursos, requisitos de segurança, confiabilidade e a proposta de modelo de arquitetura para comércio eletrônico em ambientes ubíquos. A proposta considera a convergência de vários conceitos, incluindo ubiquidade, unicidade, universalidade e unissonância.*

1. Introdução

Com o avanço da tecnologia da informação, o comércio também está evoluindo do atual *eletronic-commerce* (e-commerce) para o *ubiquitous-commerce* (u-commerce). Por definição, num ambiente ubíquo podem ser encontrados diversos tipos de dispositivos, dentre eles aparelho celular, televisor, *personal digital assistant* (PDA), e outros aparelhos que estejam ao alcance do usuário [Qiang et al 2008]. Assim, u-commerce é uma solução universal para todo tipo de necessidade de comércio que possa surgir [Xin 2009].

De acordo com [Xin 2009], quatro características definem u-commerce:

1. Ubiquidade: usuários acessam a partir de qualquer lugar a qualquer momento;
2. Unicidade: usuários identificados de forma unívoca;
3. Universalidade: dispositivos universalmente utilizáveis e multifuncionais;
4. Unissonância: dados são integrados e consistentes em diferentes aplicativos.

Para prestar o serviço de cartão de crédito, um comerciante deve possuir conta em banco, alugar um terminal POS (*Point of Sale*), prever uma taxa de carregamento e outra taxa de manutenção mensal do POS. Com o pagamento via celular, o lojista precisa usar apenas um aparelho e uma conexão móvel de dados. Já com o micro-pagamento, é possível realizar transações diretamente entre comprador e vendedor. Os cartões de crédito não se destinam às relações interpessoais, que são feitas com dinheiro ou cheque. Com o uso de aparelhos celulares este tipo de transação se torna uma opção viável. Além disso, com o serviço móvel de pagamento, o usuário não necessita pagar taxas adicionais, já que o aparelho celular pode ser o seu próprio [Tiwari and Buse 2007].

Desta forma, a motivação para este trabalho consiste em propor um modelo de arquitetura de pagamento móvel com características de fato ubíquas, ou seja, compreendendo as quatro principais premissas de um sistema ubíquo, composto pelo atendimento a ubiquidade, unicidade, universalidade e unissonância [Watson et al 2002], com vistas a suportar pagamentos móveis em ambientes de comércio ubíquo. Embora o tradicional comércio eletrônico (e-commerce) seja um modelo popular de transações de pagamentos, ainda existem muitos aspectos que necessitam ser melhorados, sendo o pagamento móvel um dos gargalos para a adoção e desenvolvimento do comércio ubíquo [Qiang et al 2008].

O presente trabalho, denominado 4iPay, consiste em uma proposta de arquitetura de sistema de pagamento móvel para ambientes de comércio ubíquo e tem por objetivo propor uma solução diferenciada de sistema de pagamento móvel. Como escopo do trabalho serão detalhados aspectos e propostas, levando em consideração, nas transações comerciais eletrônicas de transferência de fundos, questões como independência de dispositivo, de operadora, de administradora, de posição física e em relação à segurança e confiabilidade da aplicação. Neste interim, é importante destacar que, o modelo proposto visa não apenas atender ao pagamento móvel via aparelho celular, que é fundamental para o u-commerce, mas também reunir diferentes tecnologias num barramento de serviços para incorporar outros tipos de comércio eletrônico que possam envolver qualquer tipo de dispositivo com capacidade computacional e compor assim as quatro características do comércio ubíquo.

O artigo está organizado em 6 seções. A seção 1 constitui a Introdução. A seção 2 descreve os trabalhos relacionados. O modelo proposto é detalhado na seção 3, incluindo a proposta de arquitetura. Na seção 4 são descritos os cenários de utilização. A seção 5 apresenta as avaliações realizadas e possíveis trabalhos futuros. Por fim, a seção 6 apresenta as conclusões.

2. Trabalhos Relacionados

Para o desenvolvimento desta proposta de modelo de sistema de pagamento móvel, foram analisados cinco sistemas que possuem características similares e compatíveis com as funcionalidades inerentes ao comércio ubíquo.

O primeiro sistema analisado tem o nome de UbiPay. Este sistema que tem por principal objetivo minimizar os custos em pagamentos móveis [Lehdonvirta et al 2009]. Consiste em um sistema de pagamento móvel que tem a intenção de reduzir o custo das transações, oferecendo uma gama de modos de interação e escolhendo o de mínimo

valor. O objetivo é fazer com que pagar, seja como a respiração: algo que fazemos inconscientemente e sentimos apenas quando exercemos de forma além do habitual.

A seguir foi analisado o sistema denominado mFerio. Este sistema foi concebido e avaliado tendo como escopo a utilização da tecnologia NFC (*Near Field Communication*). O sistema mFerio é baseado em solicitação de pagamento móvel p2p, ou também denominado p-commerce. Em suma, consiste no comércio realizado por proximidade dos dispositivos envolvidos na transação de transferência de fundos. Se destina a substituir as transações baseadas em dinheiro [Balan et al, 2009].

Outro sistema analisado possui o nome U-PR. Este sistema tem por objetivo principal propor um regime de pagamento para comércio ubíquo promovendo a uniformidade, segurança e privacidade nas transações eletrônicas com dispositivos móveis. Tem como elemento fundamental da sua arquitetura e modelo de negócio, o suporte e proteção a privacidade [Lee et al 2006].

O próximo sistema analisado tem o nome de SmartRestaurant. Este sistema é um modelo de aplicação que permite aos clientes utilizarem dispositivos móveis para encomendar e pagar refeições a partir de um restaurante nas proximidades do campus da Universidade de Oulu, Finlândia. Além disso, o sistema é capaz de colaborar com a administração do restaurante, na medida que permite ajustar as vendas, verificar a capacidade de produção e ter conhecimento prévio de encomendas futuras [Lukkari et al, 2004].

Outro modelo analisado é denominado MUCS (*Model for Ubiquitous Commerce Support*). Este sistema consiste em um modelo de comércio ubíquo que tem por objetivo disponibilizar recursos da computação ubíqua para potencializar oportunidades de negócio aos usuários nos papeis de clientes ou fornecedores. É um sistema que utiliza perfis de usuários e informações de localização como guia para identificar oportunidades de negócios em ambientes ubíquos. O sistema realiza este processo através de um motor de análise que acompanha a mobilidade do usuário e, baseado no seu perfil e na sua posição física, infere oportunidades contextualizadas [Rosa et al 2009].

Por fim, completando o conjunto de sistemas relacionados, foi analisado o sistema denominado fairCASH que, segundo [Kreft, 2010], deve ser visto como um conjunto de tecnologias, coletivamente servindo ao apoio do ciclo de vida de objetos financeiros (tokens) anônimos e intransferíveis, denominados como eCoins, incluindo todas as fases: criação, manipulação, distribuição e compensação para evitar cópias ilegais. É uma solução centrada em serviços ICT (*Internet Communication Technology*) que tem por objetivo operar um conjunto de ações, procedimentos, políticas, propriedades e ferramentas numa arquitetura distribuída de eWallets (*digital wallet*, ou seja, carteira digital), utilizadas para realizar os pagamentos. Em outras palavras fairCASH fornece as principais tecnologias necessárias para construir um sistema de dinheiro digital (*Digital Cash*), o mais próximo possível ao seu homólogo físico bem conhecido.

3. O Modelo 4iPay

Nesta seção é apresentada a descrição do sistema proposto. O nome 4iPay dado ao modelo de sistema tem o objetivo de ilustrar seu acrônimo obtido da seguinte forma:

- *4* - significando suporte às quatro premissas da computação ubíqua: ubiquidade, unicidade, universalidade e unissonância, e uma paronomásia com a preposição da língua inglesa "for", em português "para";
- *i* – significando a independência de dispositivo, de operadora, de administradora e de posição física;
- *Pay* – significando o objetivo principal do modelo que é o suporte ao pagamento móvel de transações eletrônicas de fundos (TEF), do inglês *eletronic funds transfer* (EFT) [Lassila et al 2010];

A figura 1 mostra uma visão do diagrama de componentes do modelo 4iPay, onde consta a camada representada pelos dispositivos possíveis de utilização, a camada de entrada na aplicação, formada por um gateway ou webservices de recebimentos das requisições, um barramento de serviços, responsável pelo armazenamento e distribuição das requisições, e os diversos *slots* de serviços possíveis de serem invocados.

Além disso, o modelo proposto prevê um *site* de gerenciamento e monitoramento das transações. A infraestrutura do sistema 4iPay é baseada na arquitetura ESB (*Enterprise Service Bus*) [Chappell 2004], com o objetivo de fornecer uma categoria de serviços independentes e um barramento de serviços complexos baseados no processamento de mensagens de transações eletrônicas de fundos.

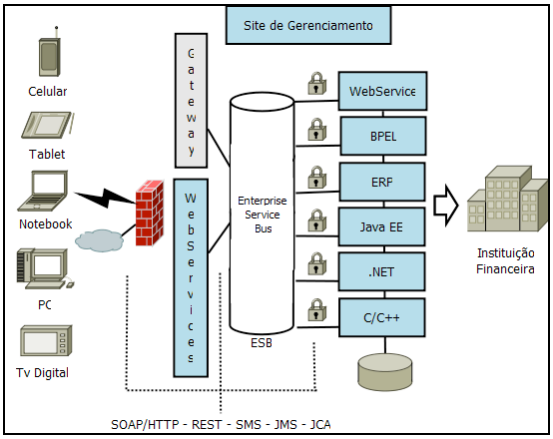


Figura 1. Diagrama de Componentes do 4iPay

Conforme pode ser observado na Figura 1, o modelo 4iPay está dividido em camadas verticais, onde, da esquerda para a direita, na primeira camada, se pode observar a multiplicidade de dispositivos suportados. A segunda camada tem por objetivo demonstrar o suporte a múltiplos protocolos de recepção e resposta aos dispositivos integrados, como SOAP (*Simple Object Access Protocol*), REST (*Representational State Transfer*), SMS (*Simple Message System*), JMS (*Java Message Service*) e JCA (*Java Connector Architecture*). A terceira camada diz respeito ao barramento ESB, responsável pela integração e gerenciamento de todo o ambiente. A quarta camada tem por objetivo demonstrar as interconexões com as instituições financeiras, através da construção e invocação de regras de negócio de forma transparente, visto que o ESB pode fazer uso de diferentes plataformas de desenvolvimento e linguagens [Chappell 2004], tornando transparente ao sistema a interligação com diferentes sistemas das administradoras.

A infraestrutura do sistema 4iPay é baseada na arquitetura ESB, com o objetivo de fornecer uma categoria de serviços independentes de plataforma de invocação, fornecendo uma base de serviços complexos baseados no processamento de mensagens de transações eletrônicas de fundos. Este sistema tem o objetivo também de fornecer uma camada de abstração da implementação do sistema, de forma a permitir uma integração das mensagens transparentes segundo a arquitetura, fornecendo as características necessárias a uma arquitetura orientada a serviços (SOA – *Service Oriented Architecture*) [Papazoglou and Georgakopoulos, 2003]. Outro objetivo desta arquitetura é possibilitar padrões flexíveis no transporte e interpretação das mensagens trocadas com os dispositivos, suportando diferentes padrões e aparelhos, removendo assim o acoplamento entre o serviço invocado e o protocolo de transporte.

3.1 Arquitetura Proposta

A figura 2 apresenta o detalhamento da arquitetura do 4iPay. Nesta figura são apresentados os componentes da arquitetura proposta, a qual é formada por um barramento de serviços na camada servidora, subdivida em núcleo central, responsável pelos conversores, parseadores, gerenciamento de controle e gerenciador de exceções ou falhas, adaptadores para suporte a diferentes protocolos e um framework de suporte ao reaproveitamento dos diversos componentes da arquitetura.

O framework da aplicação promove a distribuição das camadas de visão, controle, negócio e persistência, colaborando para a reutilização dos componentes construídos. Os adaptadores se fazem necessários em função da proposição de independência de plataforma, bem como suporte a diferentes protocolos de comunicação, tanto a nível de interface com a camada cliente, quanto interface com as operadoras e administradoras financeiras.

Também constam da infraestrutura proposta um repositório de serviços, um site de gerenciamento e uma camada de componentes de segurança, utilizados por toda a aplicação. A descrição da utilização dos componentes de suporte ao barramento de serviços é detalhada no item a seguir.

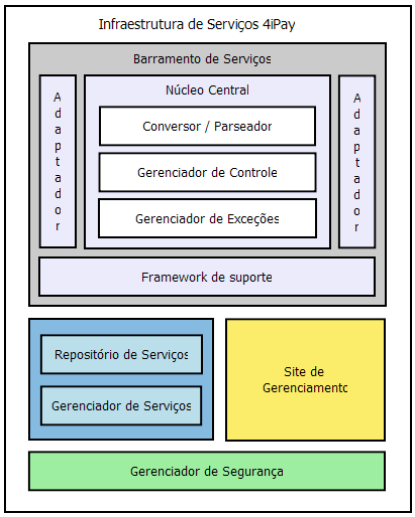


Figura 2. Diagrama da Infraestrutura de Serviços

3.1.1. Utilização de ESB. A proposta de incorporar um barramento de serviços (*Enterprise Service Bus*) advém da utilização desta arquitetura de forma a abstrair tanto o formato dos dados quanto dos protocolos que estarão envolvidos, facilitando a integração entre sistemas heterogêneos, já que impõem um contrato específico para a troca de mensagem entre os sistemas participantes. Com o ESB o sistema está apto a utilizar componentes como [Chappell 2004]:

- **Invocação**: Componente responsável por prover suporte a protocolos de maneira síncrona e assíncrona;
- **Roteamento:** Componente responsável por enviar as informações para determinados destinos, de forma estática ou dinâmica;
- **Transformação**: Componente responsável por prover a transformação de protocolos, promovendo a interdependência de integração;
- **Mensageria**: Componente responsável por prover o tratamento, processamento e reenvio de mensagens. Este item tem especial importância nos casos em que se faz necessário a resubmissão da transação, caso por exemplo de algum erro ou serviço que ao ter sido invocado estivesse fora do ar. Assim, a transação não é descartada e pode ser reaproveitada;
- **Orquestração**: Componentes relacionados a processos complexos e BPMN/BPEL integrados a invocação de *WebSevices*.

4. Cenários

Nesta seção são apresentados alguns cenários de casos de teste possíveis para avaliar a utilização sistema, os quais compõem os principais aspectos que o sistema tem por objetivo atender. Na seção seguinte descrevemos a execução e avaliação do cenário 2.

4.1 Cenário 1 – Pagamento Local

Cenário que pretende demonstrar a utilização do modelo em uma transação de pagamento local, mas entre um dispositivo móvel e uma máquina fixa, com um pagamento por aproximação. Este é o cenário onde o usuário possui um aparelho com suporte a tecnologia de aproximação (NFC) e deseja realizar o pagamento em algum terminal que tenha suporte a este tipo de tecnologia. Um exemplo de uso deste caso é o pagamento de ticket de metro, ou então a compra de um refrigerante em máquinas que suportam esta tecnologia.

4.2 Cenário 2 – Pagamento entre Dispositivos

Cenário que pretende demonstrar a utilização do modelo em uma transação de pagamento local, todavia realizando a transferência entre dispositivos móveis. Este cenário é composto de um cenário onde o comprador e o vendedor estão próximos utilizando cada um seu aparelho. Nesta situação o sistema tem a possibilidade de efetivar uma transação sem passar por um servidor no momento exato que está ocorrendo a transação. Neste caso os meios de conexão podem ser a tecnologia de aproximação (NFC), *Bluetooth* ou mesmo uma rede sem fio (*Wireless*) se for o caso. Esta situação pode ocorrer por basicamente dois motivos: quando não há conexão

remota dos aparelhos com qualquer rede de dados ou quando não há necessidade de efetivar a transação naquele exato instante.

Um exemplo desta situação é o pagamento de uma corrida de táxi no subsolo de um shopping center. Provavelmente o sinal da rede de dados da operadora estará indisponível ou mesmo que esteja disponível o taxista pode optar por efetivar todas as transações do dia em uma transação única ao final do dia. Neste caso o sistema deverá prever a efetivação em lote de transações.

4.3 Cenário 3 – Pagamento Remoto

Cenário que pretende demonstrar a utilização do modelo em um pagamento remoto, ou seja, com o comprador e o vendedor estando distantes um do outro. Este cenário faz uso de conexões de longa distância, tais como a rede de dados da operadora de telefonia, uma rede sem fio ou mesmo mensagens SMS. Nesta situação o comprador e o vendedor estão distantes. Exemplos destes cenário são *deliverys*, como o pagamento de uma pizza, onde o cliente está em casa e precisa efetivar o pagamento antes da entrega. Neste caso, o vendedor irá requisitar um pagamento, o qual será efetivado remotamente com a autorização do comprador. Outra situação que poderá acontecer neste cenário é autorização por terceiro.

Um exemplo deste caso é o filho que se dirige a uma loja de bairro para comprar um refrigerante, mas não tem créditos para efetivar o pagamento com o seu aparelho. Assim, o vendedor remete a requisição para o número da mãe que já está aguardando para efetivar a compra pelo aparelho dela.

5. Avaliação

Para a avaliação deste trabalho foi utilizado o cenário dois, no qual as transações ocorrem entre dispositivos, seja por conexão *wireless*, *bluetooth* ou NFC. Como estudo de caso utilizou-se a implementação base desenvolvida para um outro projeto em desenvolvimento pelo nosso grupo, denominado Mingle [Costa 2007].

O Mingle tem por objetivo gerenciar a interação ubíqua numa rede social dinâmica, com o mínimo de infraestrutura, formada pelas pessoas que estão fisicamente presentes em determinados locais, sendo então o modelo 4iPay aplicado sobre esta plataforma. O principal resultado consiste no desenvolvimento de um sistema de interação ubíqua, funcional, e que possa ser aplicado a diversos cenários reais, tanto com exploração comercial como puramente social [Costa 2007]. O modelo 4iPay, neste caso, atua como um módulo de serviço comercializável dentro da infraestrutura disponibilizada pela rede Mingle. A figura 3 demonstra os meios de transferência dos dados de pagamento que o sistema 4iPay suporta, onde se pode observar, na ordem, conexão wireless, rede de dados da operadora, bluetooth, NFC e via SMS. Nesta figura também é possível observar que apenas a conexão NFC está ligada.

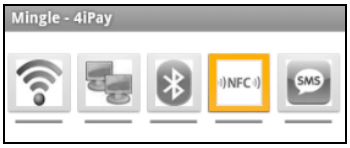


Figura 3. Possibilidades de transmissão

Para realizar a avaliação foi utilizada a plataforma de desenvolvimento Google Android, na versão 2.3.3, codinome Gingerbread, que possui suporte a tecnologia NFC [Android 2011]. A seguir são apresentados os passos de como ocorre um pagamento utilizando o modelo 4iPay sobre a plataforma Mingle como estudo de caso. A figura 4 demonstra o cenário dois, onde é realizado uma transferência eletrônica de fundos entre dois dispositivos que estão próximos, utilizando para tanto a tecnologia NFC para um pagamento local e de forma automática.



(a) (b) (c)

Figura 4. Interfaces de um pagamento entre dispositivos

A Figura 4 apresenta três telas. A primeira tela (a) mostra um cenário onde o usuário pode optar entre realizar um pagamento, requerer o pagamento de outro usuário – um exemplo é o caso de um vendedor requerendo um pagamento do cliente -, consultar pagamentos efetuados e configurar a aplicação com preferências particulares. Na segunda tela (b) , são exibidas as opções relacionadas com a determinação da forma de pagamento, caso o usuário tenha escolhido a primeira opção na tela anterior. A terceira tela (c) demonstra a realização da transferência entre dispositivos que estão próximos, configurando um pagamento do tipo local. Para tanto, o sistema tem a capacidade de detectar que apenas o suporte a NFC está ligado na aplicação, como se observa nos ícones superiores de cada tela. Desta forma, o sistema consegue realizar o pagamento com pouca e transparente interação com o usuário.

5.1 Comparação com outros sistemas

Em comparação com o sistema proposto, o sistema UbiPay [Lehdonvirta, 2009] tem um foco mais concentrado no micro-pagamento, utilizando apenas a tecnologia NFC para realizar transferências eletrônicas de fundos de diversas formas, mas somente entre dispositivos que utilizam a mesma tecnologia.

Assim, como o sistema anterior, o sistema mFerio se concentra exclusivamente na tecnologia NFC para realizar os pagamentos, mas com a diferença de inovar utilizando a tecnologia de biometria para autenticar e autorizar as transações.

O terceiro sistema analisado, U-PR também utiliza apenas a tecnologia p2p-commerce, ou seja, o comércio ubíquo por aproximação de dispositivos [Xin 2009], mas com um foco em segurança e privacidade, propondo um sistema com diversos controles de segurança, mas de forma transparente para os participantes. Já o sistema SmartRestaturant consiste em um modelo que baseia-se somente na tecnologia de

WebService, não permitindo o pagamento entre dispositivos, mas que pode ser acessado por *smartphones*, *tablets* ou qualquer computador com acesso a Internet. Além disso, este sistema possui o diferencial de permitir que os usuários façam encomendas, já que se trata de um sistema para restaurantes, estendendo assim a funcionalidade de pagamento.

O sistema MUCS [Rosa, 2009] é um modelo que se concentra nas recomendações de oportunidades de negócios em comércio ubíquo, não se detendo na efetivação destas transações, mas utilizando o contexto do usuário para realizar as recomendações, o que também é uma funcionalidade adotada pelo modelo proposto, na medida em que se propõe a realizar o pagamento com alguma característica diferenciada conforme o contexto dos participantes da transação.

Em relação ao modelo fairCash [Kreft, 2010], o modelo se diferencia em contemplar, além de muitas características já existentes em outros modelos, transações efetuadas com sensibilidade ao contexto, possibilidade de utilizar diversos níveis de segurança, inclusive com agregação, como biometria e localização; utilização de qualquer tecnologia de comunicação móvel disponível, independência de dispositivo e de localização dos participantes das transações.

6. Conclusão

O pagamento móvel possui um grande potencial. Para ter sucesso, o sistema de pagamento móvel deverá ser um modelo de ganha-ganha, com conveniência no micro-pagamento e segurança no macro-pagamento. O modelo de pagamento móvel do futuro deverá ser de qualquer tipo, realizado por qualquer um, com qualquer aparelho, em qualquer lugar, de qualquer valor e para qualquer tipo de negócio ou transação.

É neste âmbito que o modelo 4iPay se insere, propondo uma solução aberta e independente de dispositivo, localização, operadora e instituição financeira para a efetivação de transações eletrônicas de fundos. Para tanto, o sistema procura demonstrar cenários de utilização do modelo de forma atender tanto local quanto remotamente transações com diferentes dispositivos, deixando transparente ao usuário questões como o tipo de conexão que será utilizada.

Como trabalhos futuros, pretende-se adicionar suporte a diferentes plataformas de desenvolvimento, bem como portar a aplicação cliente para novos e diferentes dispositivos móveis. Outro foco importante do trabalho são as melhorias de implementação em termos de segurança da informação e otimização do desempenho da aplicação, visto que são premissas fundamentais em um sistema de pagamento móvel. Também pretende-se pesquisar e avaliar os outros cenários de utilização propostos, empregando para tanto diversos dispositivos móveis e protocolos de comunicação diferentes daqueles citados neste trabalho.

Referências Bibliográficas

Android (2011). Near Field Communication, API. Acessado em 10 de Março, 2011 em: http://developer.android.com/guide/topics/nfc/index.html

Balan, R. K., Ramasubbu, N., Prakobphol, K., Christin, N., Hong, J. (2009) mFerio: The Design and Evaluation of a Peer-to-Peer Mobile Payment System. MobiSys'09, June 22–25, 2009, Kraków, Poland.

Costa, C. A. (2007) Projeto de Pesquisa Mingle: Gerenciamento de interação ubíqua em uma rede social espontânea. Programa Interdisciplinar de Pós-Graduação em Computação Aplicada – PIPCA. Unisinos. São Leopoldo, RS, Brazil.

Chappell, David A. (2004) Enterprise Service Bus – Theory In Practice. O'Reilly Media, Inc. First Edition. Gravenstein Highway North, Sebastopol, CA 95472, USA.

Kreft, H. (2010). fairCASH based on Loss resistant Teleportation. Kiel, Germany.

Lassila, D., Lapine, K., Andress, R., McCullough, B., Pilecki, P., Weisblat, H. (2010) Electronic Funds Transfer. Banking Law – http://litigation-essentials.lexisnexis.com

Lee, K. J., Ju, J., Jeong, J. (2006) A Payment & Receipt Business Model in U-Commerce Environment. Kyung Hee University of Business Administration. ICEC'06, August 14–16, 2006, Fredericton, Canada. ACM.

Lehdonvirta, V., Soma, H., Hitoshi, I., Yamabe, T., Kimura, H., Nakajima, T. (2009) UbiPay: Minimizing Transaction Costs with Smart Mobile Payments. ACM. Mobility 2009, Sep 2-4, Nice, France.

Lukkari, J., Korhonen, J., Ojala, T. (2004) SmartRestaurant – Mobile Payments in Context-Aware Environment. ICEC'04, Sixth International Conference on Electronic Commerce.

Papazoglou M. P.; Georgakopoulos, D. (2003) Introduction: Service-oriented computing. Communications of the ACM, New York, v.46, n.10, p. 24-28.

Qiang, Yan, and Tingjie, Lu. (2008) A Study of Mobile Payment Mode in U-commerce. Economics and Management School. Beijing University of Posts and Telecommunications. Beijing, China.

Rosa, J., Franco, L., Barbosa, J. (2009) MUCS: a Model for Ubiquitous Commerce Support. Mobile Computing Lab (MOBILAB) - Programa Interdisciplinar de Pós-Graduação em Computação. X Salão de Iniciação Científica – PUCRS

Tiwari, R. and Buse, S. (2007) The Mobile Commerce Prospects: A strategic analysis of opportunities in the banking sector. Hamburg: Hamburg University Press.

Watson, R., Pitt, L., Berthon, P., Zinkhan, G. (2002). U-commerce: expanding the universe of marketing. (Vol. 30(4)). Journal of the Academy of Marketing Science.

Xin, C. (2009) U-Commerce New Technologies and Issues. Wuhan University Guangdong University of Foreign Studies. China. 2nd International Conference on Power Electronics and Intelligent Transportation System.

Análise do Impacto Combinado de Algoritmos de Localização e Controle de Densidade na Cobertura-3

André Campos[1], **Efren Souza**[2], **Eduardo Nakamura**[2]

[1]Departamento de Ciência da Computação – Universidade Federal do Amazonas

[2]Fundação Centro Análise, Pesquisa e Inovação Tecnológica - FUCAPI

andre.campos@acm.org, efren@dcc.ufam.br, nakamura@pq.cnpq.br

Abstract. *Target tracking is an important application of wireless sensor networks (WSN). It is also an application tightly coupled with localization. Since adding a GPS to every sensor in the network is unfeasible, it becomes necessary to use localization algorithms in order for the network to have the target tracking capabilities. In addition, WSNs are often deployed in hostile environments; recharging the sensors' batteries become either impossible or too costly. Therefore, it is desirable to use a density control algorithm to increase the network lifetime. In this article, we measure the impact of different localization techniques on the density control, considering a target tracking application; such application needs at least three nodes to detect a target (3-coverage). Through simulations, we evaluate the performance of three localization systems (ideal, RPE and DPE) and three density control algorithms (GAF, OGDC and A3). Results indicate the DPE-OGDC combination is the most effective on maintaining 3-coverage.*

Resumo. *Rastreamento de alvos é uma importante aplicação de redes de sensores sem fio (RSSF). Por se tratar de uma aplicação fortemente relacionada à localização, a inviabilidade de equipar todos os sensores com GPS torna necessário o uso de algoritmos de localização. Além disto, as RSSFs podem, em muitos casos, ser instaladas em ambiente hostil, onde é inviável recarregar as baterias dos sensores. Sendo assim, algoritmos de controle de densidade são utilizados para prolongar o tempo de vida da rede. Neste trabalho quantificamos o impacto de diferentes técnicas de localização sobre o controle de densidade, considerando uma aplicação de rastreamento que necessita de pelo menos três nós detectando o alvo (cobertura-3). Através de simulações, avaliamos o uso de três mecanismos de localização (ideal, RPE e DPE) e de três algoritmos de controle de densidade (GAF, OGDC e A3). Resultados indicam que a combinação DPE-OGDC é a que melhor mantém a cobertura-3.*

1. Introdução

Uma rede de sensores sem fio (RSSF) consiste em um conjunto de sensores distribuídos sobre determinada área de interesse, capaz de coletar informações do ambiente. Muitas destas informações só se tornam relevantes quando a localização dos eventos com elas relacionados é conhecida [Albowicz et al. 2001]. Desta forma, a localização correta dos sensores é indispensável para a análise destes eventos. A posição dos nós, na maioria dos casos, não é conhecida de antemão. Apenas alguns nós são capazes de autolocalizar-se

(através de GPS, por exemplo); estes nós são conhecidos como *beacons*. Os algoritmos de localização, com base na posição dos *beacons*, fazem com que os demais nós, conhecidos como nós livres, também sejam capazes de autolocalizar-se.

Os sensores são dispositivos com recursos limitados. Recarregar a bateria de um sensor é, na maioria dos casos, inviável [Tian and Georganas 2002]. O tempo de vida útil da rede está diretamente relacionado à quantidade de energia restante em cada sensor. Naturalmente, é desejável maximizar o tempo de vida da rede. Para tanto, é necessário empregar algoritmos de controle de densidade. De forma simplificada, estes algoritmos consistem em manter o maior número possível de sensores desligados durante o máximo possível de tempo, mas mantendo a funcionalidade da rede.

Os problemas de localização e controle de densidade são fundamentais para aplicações de rastreamento de alvos. Este tipo de aplicação é importantíssima na área de redes de sensores sem fio. Por exemplo, no Projeto SAUIM, temos que rastrear esta espécie de macacos (em extinção). O rastreamento contínuo e de longo prazo permite um estudo mais detalhado da espécie, colaborando para sua preservação. Consideramos que é necessário que 3 sensores detectem a presença de um alvo simultaneamente (cobertura-3) para que o rastreamento seja possível. Sendo assim, há um importante *trade-off* a ser avaliado: tempo de vida versus rastreabilidade.

É importante sabermos qual combinação de algoritmos de localização e controle de densidade é a mais eficiente na manutenção da cobertura-3. Esta é justamente a principal contribuição deste trabalho; avaliamos como uma RSSF mantém a cobertura-3 quando algoritmos de localização e controle de densidade são utilizados simultaneamente. Até onde sabemos, não há, na literatura, um estudo que combine estes três problemas num mesmo cenário.

O restante deste trabalho está organizado da seguinte forma. Na Seção 1.1, apresentamos a fundamentação teórica dos algoritmos e técnicas relacionados à localização e controle de densidade. A Seção 1.2 traz uma série de trabalhos relacionados. A Seção 2 discute a metodologia dos experimentos e avaliação de seus resultados. Por fim, a Seção 3 traz nossas conclusões e aponta para direções futuras.

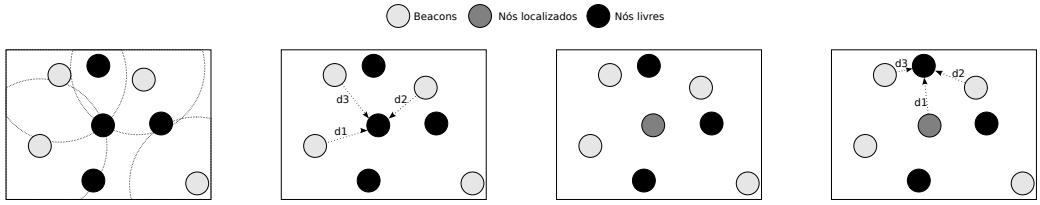
1.1. Fundamentação Teórica

1.1.1. Localização

O processo de localização dos nós da rede pode ser dividido em três etapas [Boukerche et al. 2007]: estimativa de distância, computação de posição e algoritmos de localização. A estimativa de distância pode ser feita por *Received Signal Strength Indicator* (RSSI) ou alguma variação de *Time of Arrival* (ToA) [Boukerche et al. 2007]. A computação da posição pode ser feita por métodos como a trilateração ou a multilateração. Por fim, um algoritmo de localização é empregado para propagar as informações de localização dos *beacons* para os nós livres. Nosso trabalho avalia o *Recursive Position Estimation* (RPE) [Albowicz et al. 2001] e o *Directed Position Estimation* (DPE) [Oliveira et al. 2009b].

O RPE é divido em quatro estágios, conforme a Figura 1. No primeiro, o nó livre estabelece o seus nós de referência. Após isto, o nó estima a distância para seus pontos de referência utilizando RSSI ou ToA, por exemplo. Em seguida, o nó estima sua própria

posição utilizando multilateração. Por fim, caso o nó possua certa confiança de que sua posição é correta, ele passa a se comportar como um *beacon* e ajuda outros nós livres no processo de localização.

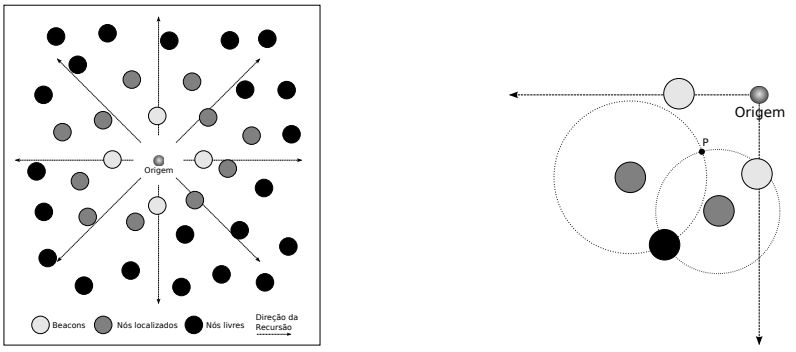


(a) Seleção das Referências (b) Estimativa de Distância (c) Cálculo de Posição (d) Novo *Beacon* Criado

Figura 1. Estágios do RPE.

O RPE faz com que o número de referências na rede cresça rapidamente; isto é sua principal vantagem. Entretanto, há um efeito colateral: os erros de localização também são propagados de maneira rápida [Oliveira et al. 2005]. Experimentos mostram que o RPE necessita que pelo menos 5% dos nós sejam *beacons*; o algoritmo é capaz de localizar 90% dos nós com erro de até 3% da distância entre os nós [Albowicz et al. 2001].

O DPE é uma derivação do RPE. A diferença é que o DPE necessita apenas de 4 *beacons*, dispostos em uma estrutura de cruz. O uso dessa estrutura garante que a recursão tenha apenas uma origem, conforme a Figura 2(a); isto implica que a direção de propagação da recursão seja conhecida (do centro da recursão para as extremidades da área sensoriada). A estrutura de cruz garante que a direção de propagação da recursão seja



(a) Escolha dos *Beacons* (b) Estimativa de Distância

Figura 2. Funcionamento do DPE.

do centro da cruz para as bordas da área de sensoreamento. Isto faz com que apenas dois sensores de referência sejam necessários para que um terceiro sensor calcule sua posição; basta que o sensor escolha o ponto de interseção dos dois discos de sensoreamento mais distante da origem da recursão, conforme mostrado na Figura 2(b).

1.1.2. Controle de Densidade

Controle de densidade consiste em manter a maior quantidade possível de sensores desligados de modo a maximizar o tempo de vida da rede, mas mantendo sua fun-

cionalidade. Este trabalho avalia três algoritmos, que representam classes diferentes de soluções para o controle de densidade: *Geography-informed Energy Conservation for Ad Hoc Routing* (GAF) [Xu et al. 2001] (baseado em *grid*), o *A Tree Algorithm* (A3) [Wightman and Labrador 2008] (independente de localização) e o *Optimal Geographical Density Control* (OGDC) [Zhang and Hou 2005] (baseado em otimização).

O GAF consiste em montar uma grade virtual sobre a área de sensoriamento. Uma vez que um nó livre passa a conhecer sua posição, ele é capaz de determinar em qual célula da grade ele se encontra. Nós que pertencem à mesma célula são considerados equivalentes. O tamanho de cada célula é dimensionado de modo que todos os nós de duas células adjacentes consigam se comunicar. Em outras palavras, o alcance do rádio dos sensores é maior do que a grandeza R da Figura 3. Xu, Heidemann e Estrin [Xu et al. 2001] mostram que o GAF consegue diminuir entre 40% e 60% o consumo de energia da rede em comparação à uma rede sem controle de densidade; além disso, mostram que o tempo de vida da rede aumenta de acordo com a densidade dos nós.

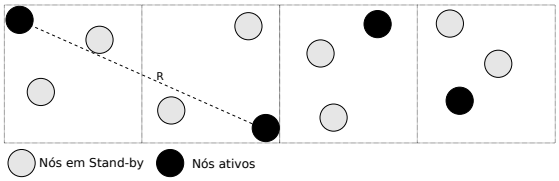
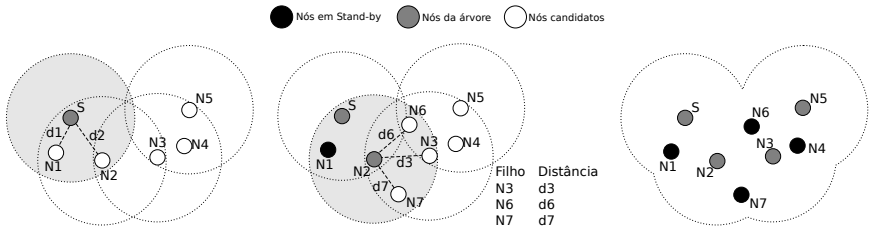


Figura 3. Definição da Célula do GAF.

O A3 é uma solução aproximada do problema de *Minimal Connected Dominating Set (CDS)*, que é NP-Difícil. Este algoritmo não necessita saber as posições de todos os nós da rede, ele utiliza apenas as distâncias estimadas por RSSI como base da criação da árvore CDS subótima. Ao iniciar o algoritmo, um sensor escolhe o seu vizinho mais distante para fazer parte da topologia; os demais são desativados. Em seguida, este vizinho seleciona o seu vizinho mais distante, como mostrado na Figura 4. Isto se repete até que todos os nós estejam ligados ou desativados. Wightman e Labrador [Wightman and Labrador 2008] mostram que o A3 precisa de apenas 6% de nós ativos quando há uma alta densidade de nós da rede. Este número sobe para 41% em densidades baixas. Além disso, o número de mensagens enviadas por nó é linear.



(a) Nós 1 e 2 são candidatos (b) Nó 2 repete o processo (c) Árvore CDS criada

Figura 4. Funcionamento do A3.

A ideia por trás do OGDC é de que minimizar a área coberta por mais de um sensor implica na utilização ótima de energia. Assim como o GAF, o OGDC assume que todos os nós conhecem a sua posição de antemão. O algoritmo baseia-se na comprovação

matemática de que a mínima interseção dos perímetros de sensoriamento de 3 nós é obtida quando as localizações destes 3 sensores formam um triângulo equilátero de lado $\sqrt{3}r$, onde r é o raio de sensoriamento, conforme mostrado na Figura 5. Isto implica que a rede deve ser densa o suficiente para que haja um sensor suficientemente perto de qualquer ponto. Zhang e Hou [Zhang and Hou 2005] mostram que o OGDC consegue utilizar menos da metade dos nós utilizados por uma variação do GAF, mantendo a mesma cobertura da área de sensoriamento.

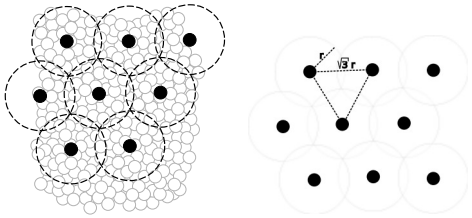


Figura 5. Funcionamento do OGDC.

1.2. Trabalhos Relacionados

O problema de localização já foi amplamente estudado na área de RSSF. Niculescu e Nath [Niculescu and Nath 2001] propõem o *Ad Hoc Positioning System* (APS). No APS, os nós livres utilizam o número de *hops* até os *beacons* para estimar suas distâncias até eles; esta técnica é chamada de DV-Hop. O algoritmo tem a vantagem de demandar um número baixo de *beacons* e densidade de sensores baixa. Entretanto, sua aplicação na prática é questionável, já que apresenta altos erros de localização e alto custo de comunicação e consumo de energia [Oliveira et al. 2009a]. Oliveira [Oliveira et al. 2009a] propõe um algoritmo chamado DV-Loc. Este algoritmo, assim como o APS, utiliza DV-Hop para estimar as distâncias entre nós livres e *beacons*. A diferença é que o escopo dos inundamentos do DV-Hop é limitado à células de um diagrama de Voronoi. Com isso, o algoritmo apresenta melhor escalabilidade do que o APS e localização mais precisa.

Para solucionar o problema de controle de densidade, Ye et al. propõem o *Probing Environment and Adaptive Sleeping* (PEAS) [Ye et al. 2003]. Este algoritmo consiste em acordar cada nó de tempos em tempos de modo a determinar se seu vizinho necessita ser substituído, sua limitação é a premissa de que "a densidade dos nós pode ser muitas vezes, ou até ordens de magnitude, maior do que o mínimo requerido para funcionamento normal" [Ye et al. 2003]. Cerpa e Estrin [Cerpa and Estrin 2004] propõem o *Adaptive Self-Configuring sEnsor Networks Topologies* (ASCENT), onde cada nó decide se vai participar ou não da rede de acordo com o seu nível de conectividade com a rede. Shang et al. introduzem o *Non-Overlapping Density Control* (NODC) [Shang and Shi 2004], que é uma versão modificada do OGDC (discutido na Seção 1.1.2).

Mohaptra e Gui [Gui and Mohapatra 2004] analisam o impacto de diferentes algoritmos de controle de densidade no rastreamento de alvos; o foco de seu estudo é manter a área de cobertura original, maximizando o tempo de vida da rede. Já Souza et al. [Souza et al. 2009] analisam a acurácia tanto do Filtro de Partículas quanto do Filtro de Kalman na presença de erros de localização introduzidos pelo RPE e pelo DPE. Oliveira et al. investiga o efeito dos erros de localização em algoritmos geográficos, como o roteamento e o controle de densidade [Oliveira et al. 2005].

2. Avaliação

2.1. Adaptações nos Algoritmos de Controle de Densidade

Zhang e Hou [Zhang and Hou 2005] sugerem que o OGDC necessita de ajustes para a cobertura-3, além de indicarem este problema como uma direção futura de seus estudos. Os ajustes sugeridos fizeram com que o algoritmo se comportasse como esperado.

O GAF utiliza uma grade virtual com células retangulares de lado $w = R/\sqrt{5}$, onde R é o raio de comunicação. Para garantir a cobertura-3, podemos redefinir o lado para $w' = R_s/\sqrt{5}$, onde R_s é o raio de **sensoriamento**. Intuitivamente podemos argumentar que isto leva, na verdade, à cobertura-5 na célula (i, j), já que um evento ali seria captado pelos sensores nas células (i, j), $(i - 1, j)$, $(i + 1, j)$, $(i, j - 1)$, $(i, j + 1)$. No entanto, nossos experimentos mostram que $w = R_s/\sqrt{5}$ nos dá cobertura-4; a diferença se dá por causa das células nas bordas. Através da variação de w, observamos o seguinte: quando $w = 0.884R_s$, o número médio de sensores que cobrem cada ponto da rede é 3. Nossos experimentos indicam que o referido valor de w levou, de fato, à um aumento da cobertura média em comparação à $w = R_s/\sqrt{5}$.

O A3 desliga tantos nós que inviabiliza a trilateração de um alvo. Entretanto, utilizamos suas ideias fundamentais e elaboramos uma variação do algoritmo. Ao invés de tentarmos encontrar a árvore cuja topologia consiste dos vizinhos com maior distância entre si, montamos uma árvore onde os nós da topologia são aqueles mais próximos da borda do disco de sensoreamento dos nós que iniciam o algoritmo.

2.2. Metodologia

Os experimentos foram feitos no simulador Sinalgo [Distributed Computing Group 2010]. Nós são distribuídos pela área de interesse de forma aleatória uniforme; após isto, um algoritmo de localização e um de controle de densidade são executados em sequencia. Os nós que estão localizados e ligados após a execução destes algoritmos são capazes de identificar a presença do alvo, desde que esteja à uma distância de no máximo R_s.

O tempo no Sinalgo é discretizado e baseia-se em *rounds*. A duração de uma instância da simulação vai até 100.000 *rounds* A simulação também pode ser terminada caso a cobertura-3 de determinado ponto da área de sensoriamento seja inferior à 60% por um tempo superior à duração de um *round* de controle de densidade.

A imprecisão da estimativa entre dois sensores A e B é simulada através de um ruído gaussiano com média $0,1d$ e variância $0,01d$, onde d é a distância cartesiana entre A e B. O modelo de energia utilizado é semelhante àquele utilizado por Xu e Estrin [Xu et al. 2001]: se a quantidade de energia utilizada para manter um nó ativo por um *round* é de $1,0J$, então o nó necessita de $1,6J$ para transmitir uma mensagem e de $0,0025J$ para permanecer em *stand-by*. Os nós recebem energia o suficiente para permanecerem ativos por 25.000 *rounds* sem enviar ou receber mensagens. O modelo de mobilidade utilizado para simular o movimento do alvo é o Correlated Random Walk (CRW) [Gillis 1955]; isto gera o tráfego na rede. Os parâmetros deste modelo de mobilidade foram configurado para que ele simule o movimento de um animal com velocidade média de aproximadamente $3,9m/s$.

A área da rede é fixada em $70 \times 70m^2$. Cada experimento consiste em utilizar 256, 512 ou 1024 nós com todas as combinações possíveis dos algoritmos de localização

(localização exata, RPE e DPE) e de controle de densidade (OGDC, GAF e A3). Para medir a cobertura-3, nós criamos uma grade virtual com 100 pontos dentro da área de sensoriamento. Cada ponto desta grade deve estar dentro do raio de sensoriamento de pelo menos 3 sensores. Após cada *round* da simulação, a porcentagem da área com cobertura-3 é registrada. Como prova de conceito, as tabelas de roteamento são montadas periodicamente com o AODV [Perkins et al. 2003]. O raio de comunicação é fixo em $15m$ e o raio de sensoriamento é fixo em $7,5m$. O número de instâncias de cada experimento é 32. Desta forma, temos um total de 864 simulações.

2.3. Resultados

Tanto o OGDC quanto o A3 utilizam o conceito de *rounds* (diferente de um *round* da simulação). No contexto dos algoritmos de controle de densidade, um *round* significa o período de tempo entre duas execuções do algoritmo de controle de densidade. Por isso, podemos observar uma queda periódica na cobertura-3 para estes dois algoritmos. Em outras palavras, todos os nós são desativados para a reconfiguração da rede e nenhum deles pode detectar o alvo. Já no GAF, um nó manda mensagens periódicas para determinar se sua célula possui um sensor ligado. Caso não haja, o nó assume o papel de sensor daquela célula. O resultado disto é que a cobertura-3 do GAF não apresenta descontinuidades tão perceptíveis quanto aquelas do OGDC e do A3 (Figura 6).

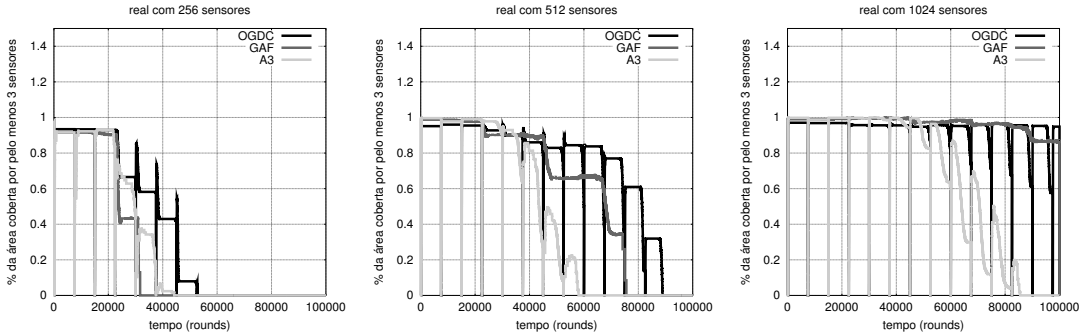


Figura 6. Determinando o Algoritmo de Controle de Densidade para Cobertura-3.

A Figura 6 mostra ainda como a cobertura-3 se comporta no tempo com os diferentes algoritmos de controle de densidade com localização exata; o mesmo comportamento é observado quando os algoritmos de localização (RPE e DPE) são introduzidos. O A3 não apresenta performance satisfatória em comparação aos demais algoritmos de controle de densidade avaliados. Sua única vantagem é que a cobertura-3 não é influenciada pela escolha do algoritmo de localização, como ocorre com o GAF e com o OGDC. O OGDC é claramente superior com $n = 256$ e $n = 512$; não fica claro o que é mais apropriado para $n = 1024$. Neste caso, a Figura 7 poderia levar a acreditar que o GAF é melhor que o OGDC. Esta aparente melhora acontece porque a simulação acaba antes de podermos observar uma degradação maior na cobertura-3 do GAF e o GAF não apresenta as descontinuidades do OGDC. Entretanto, se o tempo máximo de simulação fosse estendido para 200.000 *rounds*, por exemplo, observaríamos a superioridade do OGDC, como acontece com $n = 512$ e $n = 256$.

Uma vez que foi estabelecido que o OGDC é, dentre os algoritmos avaliados, aquele mais apropriado para a manutenção da cobertura-3, é necessário determinar com

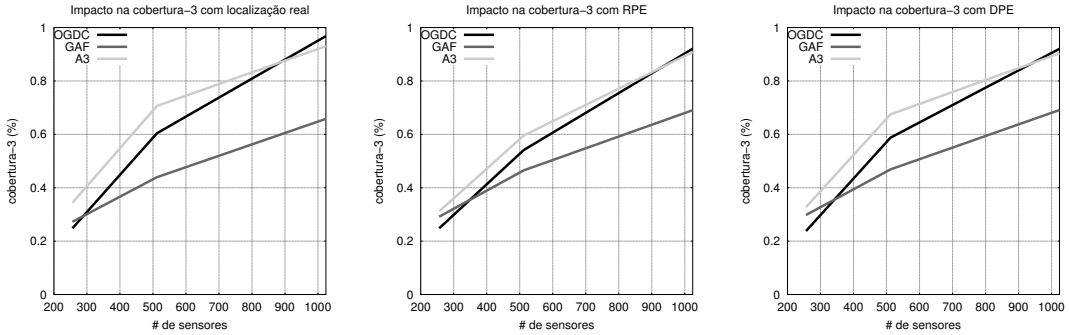


Figura 7. Análise Combinada de Localização e Controle de Densidade na Cobertura-3

que algoritmo de localização o OGDC funciona melhor. A Figura 8 traz uma comparação do OGDC com os dois algoritmos de localização e a localização exata (*baseline*). Podemos observar, na Figura 8, que o OGDC mantém a cobertura-3 por mais tempo quando utilizado com localização exata, DPE e RPE, respectivamente. Na prática, o DPE é a melhor solução, já que a localização real é inviável.

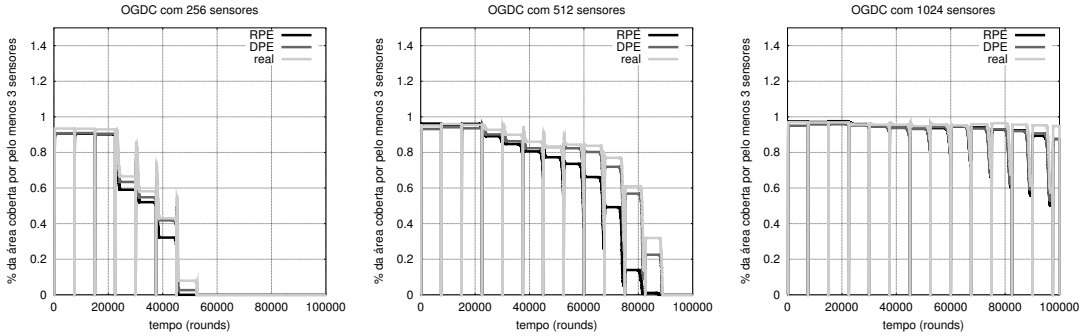


Figura 8. Determinando o Algoritmo de Localização para Cobertura-3.

3. Considerações Finais

Neste artigo, avaliamos como combinações de mecanismos de localização e algoritmos de controle de densidade afetam a cobertura-3 de uma RSSF. Para esta análise, utilizamos a localização real, RPE e DPE para os mecanismos de localização, já que o RPE é a solução pioneira e o DPE é mais preciso e barato. Os algoritmos de controle de densidade utilizados foram o OGDC, GAF e A3.

O OGDC teve o melhor desempenho; sua superioridade em termos de cobertura-3 chegou a ser de 37% em redes de baixa densidade ($n = 256$). O GAF, por sua vez, é interessante por não apresentar as mesmas descontinuidades que o OGDC e o A3. Desta forma, o algoritmo pode rastrear um alvo por mais tempo do que o OGDC antes que os nós comecem a ficar sem energia. O A3, por sua vez, apesar de não sofrer influência dos algoritmos de localização, é bastante inferior aos demais na manutenção da cobertura-3.

Os trabalhos futuros avaliarão se uma alta cobertura-3 implica também em uma melhor qualidade de rastreamento (menor erro médio), ou se outras combinações de algoritmos podem levar à um rastreamento mais preciso, apesar da cobertura-3 reduzida.

4. Agradecimentos

Este trabalho foi parcialmente financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), sob o processo 55.4087/2006-5 (SAUIM).

Referências

Albowicz, J., Chen, A., and Zhang, L. (2001). Recursive position estimation in sensor networks. In *Proceedings of the 9th IEEE International Conference on Network Protocols*, pages 35–41, Riverside, USA.

Boukerche, A., Oliveira, H. A., Nakamura, E. F., , and Loureiro, A. A. (2007). Localization systems for wireless sensor networks. *IEEE Wireless Communications*, pages 6–12.

Cerpa, A. and Estrin, D. (2004). Ascent: Adaptive self-configuring sensor networks topologies. *IEEE Transactions on Mobile Computing*, pages 272–285.

Distributed Computing Group (2010). Sinalgo. `http://disco.ethz.ch/projects/sinalgo`.

Gillis, J. (1955). Correlated random walk. *Mathematical Proceedings of the Cambridge Philosophical Society*, pages 639–651.

Gui, C. and Mohapatra, P. (2004). Power conservation and quality of surveillance in target tracking sensor networks. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*, pages 129–143, New York, NY, USA.

Niculescu, D. and Nath, B. (2001). Ad hoc positioning system (aps). In *Proceedings of the IEEE Global Telecommunications Conference*, pages 2926–2931, San Antonio, TX, USA.

Oliveira, H. A., Nakamura, E. F., Loureiro, A. A., and Boukerche, A. (2005). Error analysis of localization systems in sensor networks. In *Proceedings of the 13th ACM International Symposium on Geographic Information Systems*, pages 71–78, Bremen, Germany.

Oliveira, H. A., Nakamura, E. F., Loureiro, A. A., and Boukerche, A. (2009a). Dv-loc: a scalable localization protocol using voronoi diagrams for wireless sensor networks. *IEEE Wireless Communications*, pages 50–55.

Oliveira, H. A. B. F., Boukerche, A., Nakamura, E. F., and Loureiro, A. A. F. (2009b). An efficient directed localization recursion protocol for wireless sensor networks. *IEEE Transactions on Computers*, pages 677–691.

Perkins, C., Belding-Royer, E., and Das, S. (2003). Ad hoc on-demand distance vector (aodv) routing.

Shang, Y. and Shi, H. (2004). A new density control algorithm for wireles sensor networks. In *Annual IEEE Conference on Local Computer Networks*, pages 577–578, Los Alamitos, CA, USA.

Souza, E. L., Nakamura, E. F., and de Oliveira, H. A. (2009). On the performance of target tracking algorithms using actual localization systems for wireless sensor networks. In *Proceedings of the 12th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 418–423, New York, NY, USA.

Tian, D. and Georganas, N. D. (2002). A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pages 32–41, New York, NY, USA.

Wightman, P. M. and Labrador, M. A. (2008). A3: A topology construction algorithm for wireless sensor networks. In *Proceedings of the IEEE Global Telecommunications Conference*, pages 1–6, New Orleans, USA.

Xu, Y., Heidermann, J., and Estrin, D. (2001). Geography-informed energy conservation for ad hoc routing. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, pages 70–84, Rome, Italy.

Ye, F., Zhong, G., Cheng, J., Lu, S., and Zhang, L. (2003). Peas: A robust energy conserving protocol for long-lived sensor networks. *International Conference on Distributed Computing Systems*, page 28.

Zhang, H. and Hou, J. C. (2005). Maintaining Sensing Coverage and Connectivity in Large Sensor Networks. *Wireless Ad Hoc & Sensor Networks*, pages 89–123.

Classificação de Anuros Baseado em Vocalizações para Monitoramento Ambiental Pervasivo

Juan Gabriel Colonna[1], **Eduardo Freire Nakamura**[1,2], **Eulanda Miranda dos Santos**[1]

[1]Departamento de Ciência da Computação (DCC)
Universidade Federal do Amazonas (UFAM)

[2]Núcleo de Estudos e Pesquisas em Computação (NEPComp)
Fundação Centro de Análise, Pesquisa e Inovação Tecnológica (FUCAPI)

{juancolonna, nakamura, emsantos}@dcc.ufam.edu.br

Abstract. *In this work, we developed an automatic system for classifying anura (frogs and toads) based on their vocalizations. Each bioacustic signal has been segmented on the first stage into smaller units called "syllables", followed by a preprocessing step, composed by a pre-emphasis filter and a Hamming window that prepares the signal for feature extraction. We used Mel-fourier Cepstral Coefficients (MFCCs) to represent the acoustic signals and two classifiers were evaluated: kNN and SVM. In our experiments, we have achieved a classification rate of 98.97%, which shows that the MFCC, usually used in speech recognition, can be used for the recognizing anura species as well. The anuran recognition rate was improved in 16.09%, using SVM and MFCCs, compared with results found in literature.*

Resumo. *Neste trabalho desenvolvemos um sistema automático de classificação de anuros (rãs e sapos) baseado em suas vocalizações (coachar). Cada sinal bioacústico foi primeiramente segmentado em unidades menores chamadas "sílabas", posteriormente uma etapa de pré-processamento, composta por um filtro de pre-ênfase e uma janela de Hamming, prepara o sinal para a extração das características. Foram usados os Coeficientes Mel (MFCCs) para representar os sinais bioacústicos e dois classificadores foram avaliados: kNN e SVM. Em nossos experimentos, alcançamos uma taxa de acerto de 98,97%, que demonstra que MFCC, geralmente utilizada em reconhecimento de fala, pode ser usada para o problema de reconhecimento de anuros. A taxa de reconhecimento de anuros foi melhorada em 16,09%, usando SVM e os MFCCs, comparada com os resultados encontrados na literatura.*

1. Introdução

Os anfíbios, mais precisamente os anuros, relacionam-se intimamente com o ecossistema, sendo afetados de forma direta pelas mudanças ambientais [Collins and Storfer, 2003]. Segundo Carey et al. [2001] existe uma relação direta entre as mudanças climáticas e a mortandade em populações de anfíbios, de maneira que estes podem ser utilizados como indicadores de estresse ecológico precoce. Assim, um sistema de classificação de espécies

e estimação de população de anuros representa uma ferramenta de grande importância para a área de Ecologia e Biologia.

Geralmente, a classificação de anuros usando as vocalizações é realizada manualmente mediante o uso de um analisador de espectro e um computador [Riede, 1993]. O resultado da classificação depende da experiência da pessoa que realiza a tarefa, tornando-se lenta e sujeita a erros. A combinação do sistema de classificação automático com uma Rede de Sensores Sem Fio (RSSF) [Akyildiz et al., 2002], auxilia esta tarefa.

Resultados prévios [Taylor et al., 1996; Huang et al., 2009], usando técnicas de aprendizagem de máquina [Marsland, 2009], mostram que é possível reconhecer anuros mediante a aplicação das técnicas: C4.5 [Quinlan, 1993], *k-Nearest Neighbor* (kNN) [Cover and Hart, 1967] e *Support Vector Machine* (SVM) [Boser et al., 1992].

Neste trabalho, são usados os *Mel-Fourier Cepstral Coeficient* (MFCCs) [Davis and Mermelstein, 1980], geralmente usados em reconhecimento de fala, para representar os sinais bioacústicos emitidos por anuros. Estes coeficientes são comparados com as características implementadas por Huang et al. [2009], e com os classificadores kNN e SVM para melhorar a taxa de classificação (Seção 4).

Usando os MFCCs, é possível reduzir a quantidade de informação necessária, em comparação à transmissão das amostras completas de um áudio, para classificar os sinais bioacústicos. Esta vantagem permite integrar os MFCCs junto com uma RSSF para realizar a classificação só transmitindo o valor dos coeficientes pela rede, aumentado a vida útil da RSSF, poupando energia e memória. A coleta dos áudios e a classificação pode ser feita como é mostrado na figura 1.

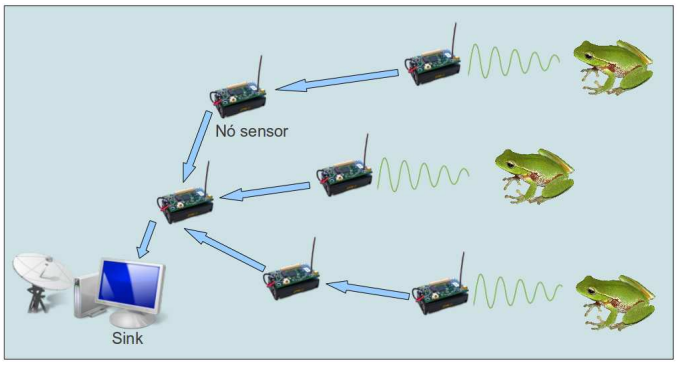


Figura 1. Coleta de áudios e classificação usando uma RSSF.

A principal contribuição deste trabalho é um método de classificação para anuros que apresenta uma melhora de 16,09% na taxa de reconhecimento de anuros, em relação aos métodos da literatura. A taxa de acerto foi comparada com o resultado obtido pela aplicação da técnica de Huang et al. [2009] com o mesmo classificador. Portanto, o foco do trabalho está na eficácia do método proposto, em relação à tarefa de classificação. Assim, a rede de sensores foi simplificada para apenas coletar o áudio dos animais.

O restante deste trabalho está organizado como segue. A seção 2 apresenta uma síntese dos trabalhos relacionados. Na seção 3 é explicado o método de reconhecimento proposto. Na seção 4 encontram-se os experimentos e resultados obtidos. Finalmente, na seção 5 são analisados os resultados e discutidas extensões futuras deste trabalho.

2. Trabalhos relacionados e fundamentos

Os anuros produzem vocalizações intensas e agudas, as quais possuem características de excitação na freqüência mais concentradas, com uma maior quantidade de harmônicos. Existe similaridade entre o modelo de geração da fala humana, descrita por Deller et al. [1993], e o modelo de geração da vocalização de um anuro, ambos formados por cavidades ressonantes. Além disso, [Bee and Micheyl, 2008] destacam as similaridades entre a forma de percepção dos sons emitidos por animais e por humanos.

A primeira proposta para monitoramento de anuros usando uma RSSF foi apresentada por Hu et al. [2005], cujo interesse principal era identificar o anuro *Cane-Toad*, conhecido como *Bufo marinus*, no Parque Nacional Kakadu, Austrália. A técnica de classificação usada foi desenvolvida por Taylor et al. [1996]. A característica usada pelo classificador foi o nível de energia dos pixels do espectrograma. As informações de cada pixel e seus vizinhos eram classificadas por uma árvore de decisão C4.5. O classificador mostrou um desempenho de 50% de classificação correta em condições onde ocorre ruído proveniente de outras espécies. Neste trabalho o áudio capturado pelos nós sensores da rede era transmitido até o nó central, que era o encarregado de efetuar a classificação. O trabalho permitiu concluir que 90% da área do parque Kakadu é habitada pelo *Bufo marinus*, permitindo identificar as zonas de concentração de suas populações.

Uma abordagem diferente de reconhecimento de anuros mediante a aplicação das técnicas kNN e SVM foi proposta por Huang et al. [2009]. Neste trabalho foram utilizadas cinco espécies de anuros diferentes. Cada sinal bioacústico foi segmentado em unidades menores chamadas sílabas (seção 4.1), e de cada sílaba três características foram extraídas: *Spectral Centroid* (S), *Signal Bandwidth* (B) e *Zero-crossing Rate* (R). O *Spectral Centroid* - S é considerado o ponto central de espectro de freqüências e se relaciona com o tom da vocalização. Sua definição se assemelha ao do centro de massa, representando a distribuição de potência sobre a freqüência,e definido como:

$$S = \frac{\sum_{n=0}^{M} n|X_n|^2}{\sum_{n=0}^{M} |X_n|^2}, \tag{1}$$

onde X_n é a DFT (*Transformada Discreta de Fourier*) da sílaba, M é a metade positiva do espectro de X_n e n é a freqüência.

A largura de banda (*Signal Bandwidth* - B) inclui a ocupação da freqüência fundamental e de seus principais harmônicos. Esta é calculada como a média dos pontos da DFT de cada sílaba, definida como:

$$B = \sqrt{\frac{\sum_{n=0}^{M}(n-S)^2|X_n|}{\sum_{n=0}^{M} |X_n|^2}}, \tag{2}$$

onde X_n é a DFT da sílaba, M é metade positiva do espectro de X_n e S é o *Spectral Centroid*.

A taxa de cruzamento por zero (*Zero-crossing Rate* - R) indica quantas vezes o sinal passou de um valor positivo para um negativo, ou vice-versa. Esta característica

proporciona uma estimação do comprimento da sílaba e definida como:

$$T = \frac{1}{2}\sum_{n=0}^{M-1} |tsgn(x_n) - tsgn(x_{n+1})|, \tag{3}$$

$$tsgn(x_n) = \begin{cases} +1 & x_n \geq \eta \\ -1 & x_n \leq -\eta \end{cases}, \tag{4}$$

onde x_n é o valor de amplitude da amostra e η é o limiar de amplitude.

Os coeficientes Mel MFCCs foram usados anteriormente por Clemins [2005] para classificar elefantes e baleias. O objetivo de seu trabalho era desenvolver um sistema de reconhecimento animal, que permita identificar a maior quantidade de animais possível.

3. Método proposto

A arquitetura do sistema proposto para a identificação de anuros é composta por três blocos principais: pré-processamento, extração das características e classificação, onde cada bloco contém blocos menores que realizam tarefas específicas. A figura 2(a) mostra como está formado o sistema.

O primeiro bloco do sistema, que realiza o pré-processamento, é formado por três blocos menores: segmentação, pre-ênfase e janelamento, figura 2(b). Esta etapa é a encarregada de acondicionar o sinal para a posterior extração de características (figura 2(c)).

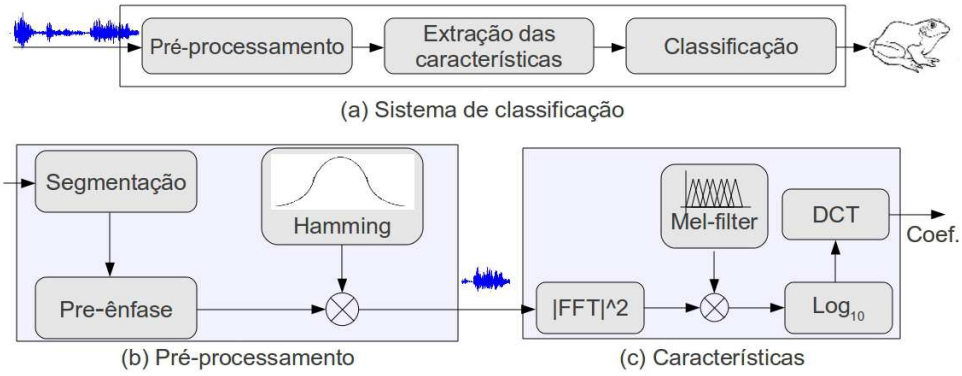


Figura 2. Sistema de classificação.

Começando pela segmentação, cada sinal é dividido em unidades menores chamadas sílabas (seção 4.1). As sílabas, dependendo da espécie, possuem freqüências fundamentais (*pitch*) superiores a 1,0 kHz. Em algumas espécies, como o *Bufo ornatus*, as freqüências começam em 2,5 kHz.

Dada a característica de geração, as freqüências elevadas tendem a possuir magnitudes menores, distorção ou saturação. Para compensar este efeito, um filtro de pre-ênfase amplifica as altas freqüências. Para realizar a pre-ênfase, o sinal digitalizado $s_{[n]}$, é modificado mediante a aplicação do filtro:

$$s_{[n]} = s_{[n]} - as_{[n-1]}, \tag{5}$$

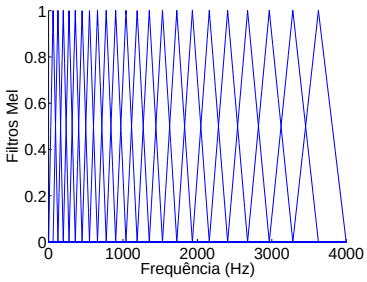
onde normalmente em sistemas de comunicação $a = 0,9$.

Cada sílaba pode ser considerada um sinal aperiódico. O algoritmo de cálculo da transformada de Fourier (FFT) produz uma aproximação do espectro com um número finito de harmônicos. Para diminuir o efeito desta aproximação, as sílabas são multiplicadas por uma janela tipo *Hamming*, onde as amplitudes dos componentes são alteradas pela aplicação da seguinte equação:

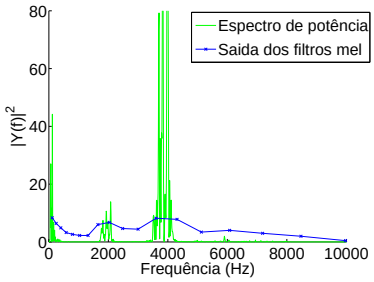
$$w_{[n]} = 0,54 - 0,46 \cos\left(2\pi \frac{s_{[n]}}{N}\right), 0 \leq s_{[n]} \leq N, \tag{6}$$

onde N é o comprimento da janela e $s_{[n]}$ são as amostras do sinal.

Após o pré-processamento, cada sinal ingressa no bloco de extração de características, figura 2(c). A característica usada para representar os sinais são os coeficientes MFCCs [Davis and Mermelstein, 1980]. A idéia básica desta técnica é fazer uma análise espectral baseada num banco de filtros triangulares espaçados logaritmicamente na freqüência, figura 3(a).



(a) Banco de filtros Mel.



(b) Espectro de potência antes e depois do banco de filtros Mel do *Hylaedactylus*.

Figura 3. Banco de filtros Mel e espectro de freqüências.

Esta característica representa melhor os sinais da fala que as características que usam uma escala de freqüência linear, servindo para sinais periódicos e aperiódicos. O uso destes coeficientes, permite reduzir significativamente a quantidade de informação necessária para descrever o sinal, sem uma perda de informação relevante [Cai et al., 2007]. A figura 3(b) mostra o espectro de freqüências de uma sílaba da espécie *Hylaedactylus* e a redução de informação após aplicar o banco de filtros Mel. O espaçamento entre filtros é definido na escala Mel por:

$$f_{mel} = 1127 * \ln\left(1 + \frac{f_{Hz}}{700}\right). \tag{7}$$

A aplicação dos filtros sobre o espectro do sinal produz um valor $M_{[r]}$ para cada filtro. Após obter os valores $M_{[r]}$, é feito o logaritmo e aplicada a transformada discreta do cosseno (DCT), para obter os coeficientes Mel $mfcc_{[m]}$ [Rabiner and Schafer, 2007]:

$$mfcc_{[m]} = \frac{1}{R} \sum_{r=1}^{R} \log\left(M_{[r]}\right) \times \cos\left(\frac{2\pi}{R}\left(r + \frac{1}{2}\right) m\right), \tag{8}$$

onde m é o número de coeficiente, R é a quantidade de filtros.

O bloco de classificação, figura 4, identifica a espécie de anuro baseando-se nos valores dos coeficientes Mel. Os métodos implementados dentro deste bloco são: *k-Nearest Neighbors* (kNN) e *Support Vector Machine* (SVM) [Marsland, 2009]. Posteriormente na seção 4.2 são comparados os resultados dos classificadores.

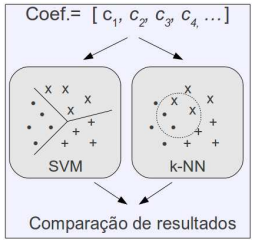


Figura 4. Bloco de classificação.

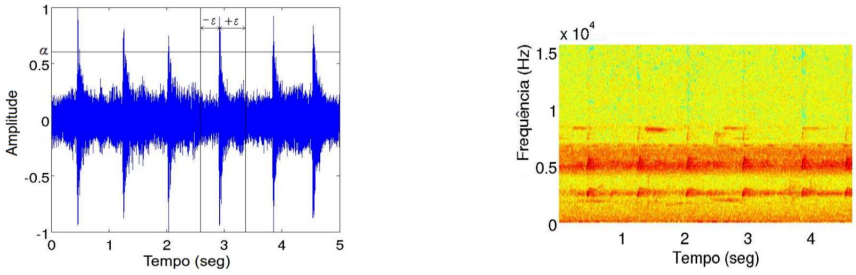
4. Avaliação

A avaliação do sistema foi realizada sobre uma base de dados com quatro espécies: *Hylaedactylus*, *Rhinella granulosa*, *Adenomera andreae* e *Ameerega trivittata*. A métrica de avaliação usada é a taxa de classificação correta Tc, definida como a razão entre a quantidade de mostras classificadas corretamente e o total de amostras.

4.1. Metodologia

As amostras de áudio foram coletadas nas regiões próximas a Manaus, durante um intervalo de dois meses e complementados com áudios de Haddad [2005]; Márquez et al. [2002]. Cada vocalização foi armazenada em formato *wav* com uma freqüência de amostragem máxima de 44,1 kHz e 16 bits por amostra, o que permite analisar sinais bioacústicos de até 22,05 kHz.

Um exemplo típico de vocalização do anuro *Adenomera andreae* é mostrado na figura 5. Existem dois métodos para segmentar o sinal em sílabas: (1) o método descrito por Huang et al. [2009] e (2) o desenvolvido por Harma [2003]. Neste trabalho é usado primeiro método, pois proporciona um comprimento uniforme para todas as sílabas.



(a) Sinal bioacústico (b) Espectrograma

Figura 5. Vocalização da espécie *Adenomera andreae*.

A segmentação em sílabas é realizada pelo seguinte algoritmo simplificado:

1. Buscar o valor de máximo de amplitude do sinal $S_{(t)}$;
2. Se $S_{(t)}$ for menor que o limiar α, ir ao passo 5, senão, ir ao passo 3;
3. Extrair ε milissegundos a direita e a esquerda do valor máximo, $S_{(t-\varepsilon)} \leq$ sílaba $\leq S_{(t+\varepsilon)}$, formando a sílaba;
4. Extrair as características da sílaba e apagar os valores do sinal original S no intervalo $t \pm \varepsilon$, voltar ao passo 1;
5. Terminar a segmentação.

A variável ϵ, que determina o comprimento da sílaba, é definida com o valor de 200 ms, o que garante a inclusão da sílaba inteira, das cinco espécies, e mais uma margem. No mesmo processo de segmentação, a variável α, do limiar de amplitude, foi definida com os valores seguintes: 40% ($\alpha = 0,4$), 50% ($\alpha = 0,5$) e 60% ($\alpha = 0,6$) da amplitude máxima. Os valores de α maiores que 60% não apresentaram melhoras significativas na taxa de classificação. Com valores menores que 40% foram extraídas sílabas com uma relação sinal-ruído baixa.

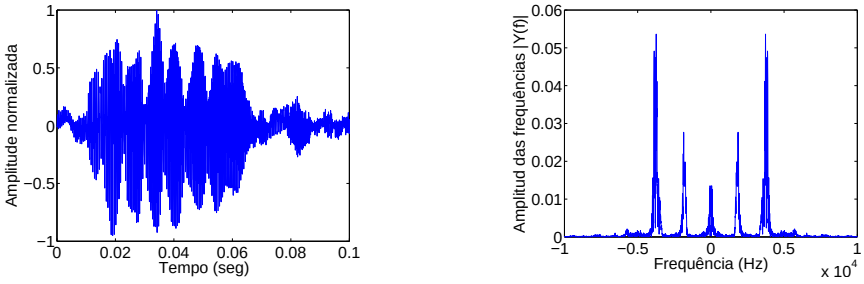
Cada sílaba passa pela etapa de pré-processamento e extração das características para formar a base de treinamento e teste dos classificadores. Do processo de segmentação são obtidas três bases, uma para cada valor da variável α. A tabela 1 mostra a quantidade de indivíduos de cada espécie e a quantidade total de sílabas usadas como base de treino e teste para os classificadores.

Espécie	Indivíduos	Sílabas		
		$\alpha = 0,4$	$\alpha = 0,5$	$\alpha = 0,6$
Rhinella granulosa	8	1147	1103	964
Hylaedactylus	3	3354	3172	2898
Adenomera andreae	7	348	241	202
Ameerega trivittata	5	614	545	328
Total	23	5463	5061	4392

Tabela 1. Quantidade de sílabas nas bases usadas nos experimentos.

Cada base é formada pelos vetores de características. Neste caso, cada vetor é composto por quinze valores de características. A primeira característica usada possui doze coeficientes Mel (MFCCs), que são obtidos pela aplicação de vinte e dois filtros triangulares, distribuídos logaritmicamente entre 0 e 22,05 kHz. As últimas três características são: *Spectral Centroid* (S), *Signal Bandwidth* (B) e *Zero-crossing Rate* (R), usadas por Huang et al. [2009] descritas na seção 2.

O espectro de freqüências de cada sílaba é obtido através da FFT com 2048 pontos. A figura 6 exibe o espectro de freqüências de uma sílaba da espécie *Hylaedactylus*.



(a) Forma de onda da sílaba do *Hylaedactylus*

(b) Espectro do *Hylaedactylus*

Figura 6. Forma de onda e espectro da vocalização do anuro *Hylaedactylus*.

Para a implementação dos classificadores foi usado o software Weka [Hall et al., 2009]. Estes classificadores foram configurados da forma seguinte: kNN com valores $k = 5$ e $k = 10$ e SVM com Kernel polinomial de grau um e dois ($p = 1$ e $p = 2$). Os

experimentos foram realizados usando validação cruzada com dez conjuntos (*folds = 10*). Os valores definidos para os parâmetros foram calculados empiricamente para maximizar a taxa de classificação.

4.2. Resultados

Os resultados da classificação realizada com kNN, onde *k* é igual a cinco e dez, são apresentados na tabela 2. A tabela mostra a relação entre a taxa de classificação e a variável de amplitude da segmentação em sílabas α. A coluna esquerda, indica a combinação de característica testada, e no centro, o valor indica a taxa de classificação correta. Nas tabelas a seguir, S significa *Spectral Centroid*, B, *Signal Bandwidth*, R, *Zero-crossing Rate*, e RSB as três características em conjunto (método de Huang et al. [2009]).

Características	kNN $k = 5$			kNN $k = 10$		
	$\alpha = 0,4$	$\alpha = 0,5$	$\alpha = 0,6$	$\alpha = 0,4$	$\alpha = 0,5$	$\alpha = 0,6$
MFCCs	98,90%	98,97%	98,97%	98,90%	98,85%	98,90%
RSB	90,40%	91,70%	93,76%	90,75%	91,87%	93,28%
R e MFCCs	98,95%	98,99%	98,99%	99,13%	98,97%	99,02%
S e MFCCs	99,17%	99,26%	99,18%	99,08%	99,15%	99,22%
B e MFCCs	99,19%	99,22%	99,15%	99,17%	99,15%	99,22%
RSB e MFCCs	99,28%	99,24%	99,27%	**99,30%**	99,26%	99,24%

Tabela 2. Taxa de classificação de kNN em relação a α.

A tabela 2 mostra que com o classificador kNN a maior taxa de classificação correta foi obtida usando $k = 5$, $\alpha = 0,6$ e a combinação de RSB. A maior diferença entre a característica MFCCs e RSB foi 8,5% para $k = 5$ e 8,15% para $k = 10$, ambos na base com $\alpha = 0,4$. Este último resultado indica que os MFCCs possuem um desempenho melhor que os RSB em sinais com ruído. A característica combinada com os MFCCs que produziu maior impacto no resultado foi *S*.

O mesmo procedimento de combinação de características descrito anteriormente foi utilizado com o classificador SVM. A tabela 3 mostra os resultados das classificações com SVM em relação à variável α, com Kernel polinomial de grau um e dois.

Características	*SVM p=1*			*SVM p=2*		
	$\alpha = 0,4$	$\alpha = 0,5$	$\alpha = 0,6$	$\alpha = 0,4$	$\alpha = 0,5$	$\alpha = 0,6$
MFCCs	97,10%	97,35%	97,60%	97,74%	97,98%	98,26%
RSB	81,54%	86,36%	88,02%	82,53%	85,55%	87,24%
R e MFCCs	97,36%	97,72%	97,65%	97,98%	98,24%	98,31%
S e MFCCs	97,36%	97,47%	97,79%	98,09%	98,39%	98,58%
B e MFCCs	97,08%	97,47%	97,54%	97,98%	98,39%	98,13%
RSB e MFCCs	97,45%	97,86%	97,90%	98,62%	**98,69%**	98,56%

Tabela 3. Taxa de classificação de SVM em relação a α.

Neste caso , usando o classificador SVM e os MFCCs, o melhor resultado foi uma taxa de classificação correta de 98,26% com $p = 2$, $\alpha = 0,5$. O resultado total maior foi 98,69% com $p = 2$, $\alpha = 0,5$ e a combinação de todas as características. Este resultado superou em 0,71% o resultado obtido com os MFCCs e 13,14% com RSB, na mesma base. Neste caso, pode ser observado que a combinação dos MFCCs e os

RSB, não apresentou uma diferença substancial no resultado. Novamente a característica combinada que produz o maior impacto foi *S*.

Para mostrar os erros de classificação, montamos uma matriz de confusão que indica quais espécies e quantas sílabas de cada uma foram confundidas. Cada linha é a espécie que originou a sílaba e a coluna é a espécie com a que foi confundida. A tabela 4 apresenta as matrizes para os casos kNN ($k = 5$) e SVM ($p = 2$) que apresentaram taxa de classificação melhor usando os MFCCs e $\alpha = 0,6$. Pode-se observar que as espécies mais confundidas pelo classificador foram: *Rhinella granulosa* com *Hylaedactylus* e *Adenomera andre* com *Ameerega trivittata*. Isto ocorre pela proximidade entre as características sonoras entre estas espécies.

Espécie	kNN, $k = 10$				SVM, $p = 2$			
	a	b	c	d	a	b	c	d
a = Adenomera andre	179	20	0	3	166	34	0	2
b = Ameerega trivittata	5	320	1	2	15	308	1	4
c = Hylaedactylus	0	2	952	10	0	5	949	10
d = Rhinella granulosa	0	1	1	2896	1	1	3	2893

Tabela 4. Matriz de confusão para kNN e SVM (melhor caso usando MFCCs).

5. Comentários finais

Neste trabalho, propusemos e avaliamos um sistema de classificação automática de anuros, usando MFCCs para a representação dos sinais bioacústicos emitidos por eles. O uso desta característica foi comparado com características desenvolvidas anteriormente no trabalho realizado por Huang et al. [2009]. Com um resultado de 98,97% de acerto, podemos concluir que os MFCCs, usados para representar sinais de fala, podem ser adaptados ao problema de reconhecimento de anuros. Observamos ainda que a combinação de todas as características não produz uma diferença substancial na taxa de acerto. Os classificadores kNN e SVM apresentaram bom desempenho. A melhor combinação foi: o classificador kNN com todas as características. O aumento do limiar α produz um aumento na taxa de classificação usando os RSB, indicando uma dependência desta característica com o nível de ruído, não sendo assim com os MFCCs. A taxa de reconhecimento de anuros foi melhorada em 16,09% usando SVM comparada o estado-da-arte (método de Huang et al. [2009]).

Futuramente, mais espécies serão agregadas com objetivo de determinar a capacidade de diferenciação dos MFCCs. O classificador será implementado em uma RSSF para facilitar o monitoramento das variações nas populações de anuros. Também consideramos a implementação de uma etapa de filtragem de ruídos no método, visando melhorar o resultado da classificação em cenários de floresta.

Agradecimentos

Agradecemos ao Professor Marcelo Gordo (UFAM) pela colaboração na coleta e análise dos áudios; agradecemos também aos projetos: ANURA (PRONEX e LACCIR) financiado pela FAPEAM, CNPq e Microsoft; e SAUIM (CNPq, processo 55.4087/2006-5). Agradecemos também à CAPES pelo supporte através de bolsa de mestrado.

Referências

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422.

Bee, M. A. and Micheyl, C. (2008). The cocktail party problem: what is it? How can it be solved? And why should animal behaviorists study it? *Journal of comparative psychology*, 122(3):235–51.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory.*, pages 144–152, New York, USA. ACM.

Cai, J., Ee, D., Pham, B., Roe, P., and Zhang, J. (2007). Sensor Network for the Monitoring of Ecosystem: Bird Species Recognition. In *3rd International Conference on Intelligent Sensors, Sensor Networks and Information*, pages 293–298. IEEE.

Carey, C., Heyer, W. R., Wilkinson, J., a. Alford, R., Arntzen, J. W., Halliday, T., Hungerford, L., Lips, K. R., Middleton, E. M., a. Orchard, S., and a. Stanley Rand (2001). Amphibian Declines and Environmental Change: Use of Remote-Sensing Data to Identify Environmental Correlates. *Conservation Biology*, 15(4):903–913.

Clemins, P. J. (2005). *Automatic classification of animal vocalizations*. PhD thesis, Marquette University, Wisconsin.

Collins, J. P. and Storfer, A. (2003). Global amphibian declines: sorting the hypotheses. *Diversity and Distributions*, 9:89–98.

Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *Transactions on Information Theory, IEEE*, 13(1):21–27.

Davis, S. and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366.

Deller, J., Hansen, J., and Proakis, J. (1993). *Discrete-time processing of speech signals*, volume 1. IEEE.

Haddad, C. (2005). Guia Sonoro dos Anfíbios Anuros da Mata Atlântica.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA Data Mining Software: An Update.

Harma, A. (2003). Automatic identification of bird species based on sinusoidal modeling of syllables. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 545–548. IEEE.

Hu, W., Tran, V. N., Bulusu, N., Chou, C. T., Jha, S., and Taylor, A. (2005). The design and evaluation of a hybrid sensor network for cane-toad monitoring. In *Fourth International Symposium on Information Processing in Sensor Networks.*, pages 503–508.

Huang, C.-J., Yang, Y.-J., Yang, D.-X., and Chen, Y.-J. (2009). Frog classification using machine learning techniques. *Expert Systems with Applications*, 36(2):3737–3743.

Márquez, R., Riva, I., Matheu, B., and Matheu, E. (2002). Sounds of Frogs and Toads of Bolivia.

Marsland, S. (2009). *Machine Learning: an algorithmic perspective.*, volume 1. CRC Press, Palmerston North, New Zealand.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Rabiner, L. R. and Schafer, R. W. (2007). Introduction to Digital Speech Processing. *Foundations and Trends in Signal Processing*, 1:1–194.

Riede, K. (1993). Monitoring biodiversity: analysis of Amazonian rainforest sounds. *Ambio*, 22(8):546–548.

Taylor, A., Watson, G., Grigg, G., and McCallum, H. (1996). Monitoring Frog Communities: An Application of Machine Learning. *Proceedings of the 8th Innovative Applications of Artificial Conference*.

Avaliação de Desempenho dos Algoritmos Criptográficos Skipjack e RC5 para Redes de Sensores sem Fio

Tiago M. Cavalcante[1,3]*, Fernando P. Garcia[2,3,4], Danielo G. Gomes[1,3], Rossana M. C. Andrade[1,2,3]†

Universidade Federal do Ceará (UFC)
[1]Programa de Pós-Graduação em Engenharia de Teleinformática (PPGETI)
[2]Mestrado e Doutorado em Ciência da Computação (MDCC)
[3]Grupo de Redes de Computadores, Engenharia de Software e Sistemas (GREat)
[4]Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE)
Fortaleza,CE, Brasil

{tiagocavalcante,fernandogarcia,dgomes,rossana}@great.ufc.br

Resumo. *O aumento de aplicações em Redes de Sensores sem Fio (RSSF) faz com que a preocupação com a segurança da informação nesse tipo de redes torne-se mais relevante. Levando em consideração que a seleção do algoritmo de criptografia apropriado é essencial para prover a segurança dos dados em RSSF, a nossa proposta é avaliar o desempenho dos algoritmos Skipjack e RC5, dois algoritmos muito utilizados em RSSF, com o objetivo de orientar um projeto de RSSF quanto à escolha do algoritmo criptográfico mais adequado à sua aplicação. Experimentos reais foram realizados na plataforma MicaZ com o auxílio de um osciloscópio digital e os resultados das medições indicam que o algoritmo RC5 requer menos ROM (7%), tempo (8%) e energia (8%) na encriptação. Em contrapartida, o Skipjack requer menos memória RAM (9%), tempo (5%) e energia (5%) na decriptação.*

Abstract*. The increase in the number of applications for wireless sensor networks (WSN) makes more relevant the concern about information security in such networks. Taking in consideration that the selection of the appropriate encryption algorithm is essential for providing data security in WSN, the goal of this paper is to evaluate the performance of RC5 and Skipjack algorithms aiming to direct the design of WSN in the choice of cryptographic algorithm. Experiments were carried out in a real platform MicaZ, with the help of a digital oscilloscope. The measurement results indicate that the RC5 algorithm requires less ROM (7%), time (8%) and energy (8%) for encryption. In contrast, the Skipjack requires less RAM (9%), time (5%) and energy (5%) in the decryption.*

* Bolsista de mestrado da CAPES
† Bolsista de produtividade DT 2 do CNPq

1. Introdução

Os avanços recentes da microeletrônica proporcionaram o desenvolvimento de microsensores que, acoplados a dispositivos com capacidade de processamento e recursos de computação limitados e providos de comunicação sem fio, formam um nó sensor [Akyildiz et al. 2002]. Dentre as inúmeras aplicações das Redes de Sensores Sem Fio (RSSF), destacam-se o monitoramento ambiental, a aquisição de dados na indústria e aplicações de tempo real [Borges Neto et al. 2010].

Se por um lado, o aumento das aplicações de RSSF em diversas áreas enseja a preocupação com a segurança dos dados coletados, por outro, a segurança é um aspecto que vem sendo considerado um desafio em RSSF devido suas limitações de memória, processamento e energia [Kaps, Gaubatz and Sunar 2007]. Além disso, o custo da segurança para o projeto de RSSF ainda se apresenta como uma variável desconhecida [Lee, Kapitanova and Son 2010].

Como os algoritmos de criptografia estão entre os principais elementos utilizados para prover a segurança dos dados em um processo de comunicação, a seleção do algoritmo apropriado é essencial para prover a segurança dos dados em RSSF [Simplício Jr e Barreto 2010]. Portanto, a avaliação de algoritmos de criptografia é de fundamental importância para orientar o projeto de aplicações de RSSF quanto à escolha do algoritmo criptográfico mais adequado à sua aplicação.

Diante desse contexto, nossa proposta é avaliar o desempenho, por meio de experimentos reais, dos algoritmos Skipjack [Nist 1998] e RC5 [Rivest 1997], dois algoritmos muito utilizados em RSSF, com o objetivo de orientar um projeto de RSSF. As métricas consideradas na avaliação de cada algoritmo são a quantidade de memória requerida (ROM e RAM), o tempo de execução e o consumo de energia nas fases de encriptação e decriptação dos dados.

2. Algoritmos de Criptografia

Nesta seção são descritas as principais características dos algoritmos de criptografia avaliados neste trabalho, quais são Skipjack e RC5. A escolha desses algoritmos é justificada pelo fato de que ambos foram desenvolvidos para dispositivos com capacidade limitada de processamento e memória, além de fazer parte da camada de segurança TinySec [Karlof, Sastry and Wagner 2004], utilizada em aplicações de RSSF, sendo que o Skipjack também é a cifra padrão do protocolo de segurança para RSSF MiniSec [Luk et al. 2007].

2.1 Skipjack

O Skipjack é um algoritmo de cifra de bloco desenvolvido na década de 80 pelo U.S *National Security Agency* (NSA), tornando-se público em 1998. Ele foi concebido especialmente para dispositivos com capacidade de processamento limitada, utilizando apenas operações de bit para a criptografia e decriptografia das mensagens. O Skipjack possui as seguintes características principais: tamanho do bloco de 64 bits, tamanho da chave criptográfica de 80 bits e 32 *rounds* [Nist 1998].

Em 2002, foi apresentada a primeira criptoanálise completa dos 32 *rounds* do Skipjack, utilizando *slide attack* [Phan 2002]. Assim, observa-se a vulnerabilidade do

algoritmo Skipjack, sendo a limitação da chave um dos fatores principais que tornam o algoritmo inseguro.

2.2. RC5

O RC5 é um algoritmo de cifra de bloco desenvolvido em 1994 por Ronald Rivest. Ele foi concebido para, entre outros objetivos, requerer pouca quantidade de memória e possuir alto nível de segurança [Rivest 1997]. O RC5 possui as seguintes características principais: tamanho do bloco de 32, 64 ou 128 bits, tamanho da chave criptográfica variável de 0 a 2040 bits (0 a 255 bytes) e número de *rounds* variável de 0 a 255 [Rivest 1997].

O nível de segurança do algoritmo RC5 é estabelecido pela escolha de seus parâmetros. Rivest (1997) definiu a configuração RC5 64/12/16 (tamanho do bloco de 64 bits, chave de 16 bytes e 12 *rounds*) como padrão. Porém, para ser considerado seguro contra ataques de segurança, o número de *rounds* deve ser maior do que 16 [Potlapally et al. 2005]. Assim, a configuração RC5 64/18/16 foi escolhida para a realização dos experimentos neste trabalho.

3. Metodologia

Nesta seção são apresentadas a aplicação desenvolvida e as métricas escolhidas para a avaliação do desempenho dos algoritmos estudados neste trabalho, além da descrição dos experimentos realizados.

3.1. Aplicação Utilizada na Avaliação

Para avaliar o desempenho dos algoritmos Skipjack e RC5 foi necessário o desenvolvimento de uma aplicação[‡] (chamada de Criptotest) utilizando a linguagem NesC para o sistema operacional TinyOS, versão 2.1.

Algoritmo 1 CriptoTest

Entrada: *textoClaro*, o bloco a ser criptografado; *chave*, a chave criptográfica da cifra de bloco;
Saída: *textoCifrado*, o bloco criptografado; Alteração lógica do bit 4 da porta E do microcontrolador.

```
1:  //A cifra de bloco é definida em tempo de compilação
2:  Altere o estado do bit 4 da porta E para 0
3:  se segurança então
4:     inicializaChave(chave)
5:     altere o estado do bit 4 da porta E para 1
6:     textoCifrado ← criptografaBloco(textoClaro, chave)
7:     altere o estado do bit 4 da porta E para 0
8:     textoClaro ← decriptografaBloco(textoCifrado, chave)
9:  fimse
10: escreva(textoClaro, textoCifrado)
```

[‡] Códigos da aplicação disponíveis em: http://www.great.ufc.br/malveira/.

A aplicação desenvolvida para a avaliação está descrita no Algoritmo 1. Esse algoritmo fornece condições para a medição do tempo de execução da operação de encriptação (linhas 5 e 7) pelo processo que será descrito mais adiante. A instrução entre as linhas 5 e 7 do algoritmo determinam a operação a qual seu tempo será medido. Assim, com uma pequena alteração no Algoritmo 1 (posição das linhas 5 e 7) é possível avaliar as operações de inicialização da chave e de decriptação, de forma análoga à encriptação.

Ao criptografar ou decriptografar uma mensagem, a aplicação envia para um microcomputador, via USB, o conteúdo da mensagem, permitindo a depuração durante o desenvolvimento do código. O algoritmo de criptografia a ser utilizado na aplicação pode ser selecionado em tempo de compilação, podendo ser escolhido o RC5, o Skipjack ou nenhum algoritmo de criptografia (sem segurança).

Os algoritmos de criptografia contidos na TinySec estão disponíveis apenas para a versão do TinyOS 1.x. Dessa forma foi necessário portar os algoritmos de criptografia para a versão atual do TinyOS, a 2.X. Esse processo exigiu alteração do código fonte original e mudanças na configuração do TinyOS, por meio da inclusão da interface BlockCipher que é implementada pelos componentes RC5 e Skipjack da TinySec.

3.2. Métricas Utilizadas na Avaliação

Foram definidas as seguintes métricas para a avaliação do desempenho dos algoritmos Skipjack e RC5: quantidade de memória ROM requerida, quantidade de memória RAM requerida, tempo de execução e consumo de energia. As métricas foram escolhidas levando-se em consideração as principais limitações de recursos computacionais em RSSF.

3.3. Descrição dos Experimentos

Todos os experimentos foram realizados em um nó sensor MicaZ, uma plataforma muito utilizada em aplicações de RSSF, sendo bastante popular [Liu and Ning 2008]. Ela foi desenvolvida pela Crossbow Technology e possui como principais características: 4KB de memória RAM, 128KB de memória ROM, conversores analógico-digitais de 10 bits e transceptor de radio frequência CC2420 [Crossbow 2006].

A quantidade de memória RAM e ROM requerida por cada algoritmo foi obtida a partir do próprio processo de compilação do código fonte no TinyOS. O resultado do processo de compilação com a indicação da quantidade de memória requerida é ilustrado na captura de tela da Figura 1.



Figura 1. Ilustração do processo de compilação, destacando a quantidade de memória ROM e RAM requerida pela aplicação.

As medições do tempo de execução foram realizadas com o auxílio de um osciloscópio digital Tektronix TDS 210. O pino E4 do microcontrolador ATMega128L da plataforma MicaZ foi conectado a um canal do osciloscópio. Com a utilização da interface GeneralIO do TinyOS, o nível lógico desse pino foi alterado de 0 para 1 antes e de 1 para 0 após as operações de criptografia, decriptografia e inicialização da chave. O tempo no qual o pino permanece com nível lógico 1 representa o tempo de execução de cada operação que pode ser medido no osciloscópio. O procedimento utilizado para nas medições do tempo de execução é mostrado na Figura 3 e uma ilustração real do uso do osciloscópio para efetuar as medições é mostrada na Figura 4. O consumo de energia foi obtido a partir das seguintes equações:

$$potência = tensãoBateria * correnteOperação \quad (1)$$

$$energiaConsumida = potência * tempoExecução \quad (2)$$

Os valores de tensão da bateria e de corrente de operação foram obtidos no documento de especificação da plataforma MicaZ (*datasheet*), que são 3V e 8mA, respectivamente, e o tempo de execução foi medido com um osciloscópio, conforme descrito anteriormente. Assim, a potência foi calculada, utilizando-se a Equação 1 da seguinte forma*:*

$$potência = 3*8*10^{-3} = 24\ mW.$$

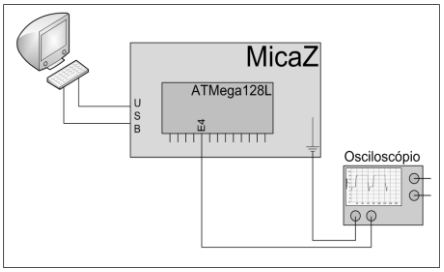


Figura 3. Esquema simplificado de medição do tempo de execução.

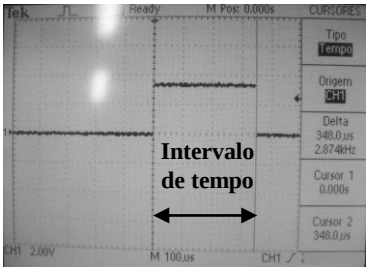


Figura 4. Imagem ilustrativa da medição do tempo de execução no osciloscópio.

A partir da potência calculada, a quantidade de energia requerida por cada algoritmo é obtida de acordo com a Equação 2, sendo que o tempo de execução é o intervalo de tempo medido com o osciloscópio.

$$energiaConsumida = 24*10^{-3} * tempoExecução$$

Com essa estratégia de medição, obtém-se a energia gasta por cada algoritmo de forma aproximada, uma vez que não são utilizados os valores reais de tensão e corrente. Porém, Lee, Kapitanova e Son (2010) verificaram em seus trabalhos que a diferença entre os valores reais e teóricos não é significativa, não afetando a comparação do desempenho dos algoritmos estudados neste trabalho.

4. Resultados

Nesta seção, os resultados obtidos na avaliação de cada métrica são apresentados e discutidos.

4.1. Quantidade de Memória Requerida

As quantidades de memória ROM e RAM requeridas por cada algoritmo são apresentadas nas Figuras 5 e 6, respectivamente. Pode-se observar na Figura 5 que o Skipjack requer mais memória ROM do que o RC5, aumentando a quantidade em quase 20% quando comparado com a aplicação sem segurança. O Skipjack requer cerca de 7% mais memória ROM que o RC5. A quantidade de memória ROM requerida pela aplicação desenvolvida, utilizando o Skipjack, representa 16% do total disponível na MicaZ (128KB), enquanto a quantidade, utilizando o RC5, representa apenas 15%. Por outro lado, com relação à memória RAM, o RC5 requer cerca de 9% a mais que a quantidade requerida pelo Skipjack. A quantidade requerida pelo RC5 representa quase 15% a mais que a quantidade requerida pela aplicação sem segurança. A quantidade de memória RAM requerida pela aplicação desenvolvida, utilizando o RC5, representa 25% do total disponível na MicaZ, enquanto a quantidade, utilizando o Skipjack, representa apenas 23%.

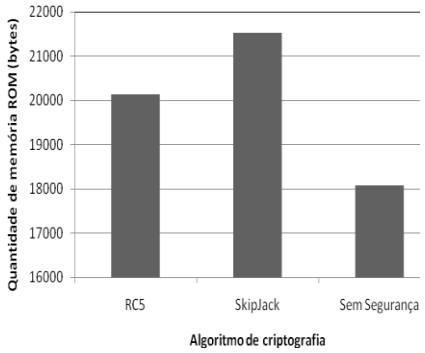


Figura 5. Quantidade de memória ROM requerida.

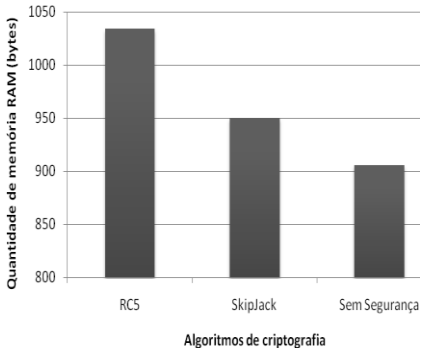


Figura 6. Quantidade de memória RAM requerida

4.2. Tempo de Execução

O tempo de execução de cada algoritmo na criptografia e decriptografia de um bloco de 64 bits é mostrado na Figura 7. Os números representam a média dos valores obtidos em 10 medições. A Tabela 1 apresenta os valores medidos com intervalo de confiança de 95%. Como o número de experimentos é menor do que 30 e o desvio padrão populacional é desconhecido, a tabela t-Student foi utilizada para o cálculo do intervalo [Feldman and Valdez-Flores 1995]. Devido à pequena magnitude do intervalo de confiança, preferimos apresentá-lo apenas na Tabela 1 para facilitar a visualização dos resultados.

Na Figura 7 é possível perceber que a operação de *Key Setup* (inicialização e expansão da chave criptográfica) do RC5 requereu significativamente mais tempo que do Skipjack – cerca de 3,5 vezes mais tempo. Porém essa fase não é decisiva, pois ela é realizada apenas uma vez durante a inicialização da aplicação. A operação de encriptação com o algoritmo RC5 demorou 0,32 ms, sendo cerca de 8% mais rápida do que com o Skipjack. Por outro lado a operação de decriptação demorou 0,34 ms, sendo cerca de 5% mais lenta do que com o Skipjack. Considerando a soma do tempo de execução da operação de encriptação com a decriptação, o RC5 apresentou tempo de execução 1% menor que o Skipjack.

4.3. Consumo de Energia

O consumo de energia de cada algoritmo é mostrado na Figura 8. Conforme mencionado anteriormente, o consumo de energia foi obtido a partir do tempo de execução dos algoritmos e através da aplicação da Equação 2, uma vez que foram utilizados valores teóricos para o cálculo da potência. A média dos valores de energia calculados a partir das 10 medições de tempo de processamento com intervalo de confiança de 95% é mostrada na Tabela 2. Da mesma forma que o tempo de execução, a tabela *t-Student* foi utilizada para o cálculo do intervalo de confiança e preferimos apresentá-lo apenas na Tabela 2 para facilitar a visualização dos resultados.

Na Figura 8 é possível verificar que, na etapa de *Key Setup*, o RC5 consome significativamente mais energia que o Skipjack – cerca de 3,6 vezes. Na operação de encriptação, o RC5 apresentou um consumo de energia de 7,7 µJ, sendo cerca de 8% menor do que o Skipjack. Porém, na operação de decriptação, o RC5 apresentou um consumo de 8,8 µJ, sendo cerca de 5% maior do que o Skipjack. Considerando a soma da energia gasta na operação de encriptação com a decriptação, o RC5 apresentou uma quantidade 1% menor que o Skipjack.

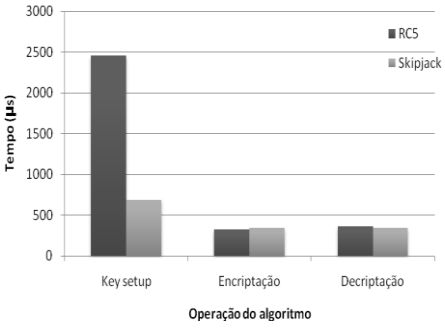


Figura 7. Tempo de execução.

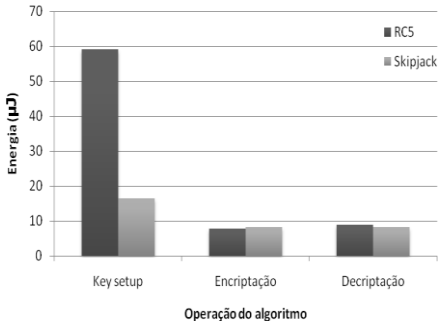


Figura 8. Consumo de energia.

Tabela 1. Tempo de execução em µs com intervalo de confiança de 95%.

Algoritmo	*Key setup (µs)*	*Encriptação (µs)*	*Decriptação (µs)*
Skipjack	683,1 ± 0,8297	347,4 ± 0,5272	348,6 ± 0,5272
RC5	2460,7 ± 0,6207	321,0 ± 0,7108	368,0 ± 0,5026

Tabela 2. Quantidade de energia requerida em µJ com intervalo de confiança de 95%.

Algoritmo	*Key setup (µJ)*	*Encriptação (µJ)*	*Decriptação (µJ)*
Skipjack	16,394 ± 0,0199	8,337 ± 0,0126	8,366 ± 0,0126
RC5	59,056 ± 0,0148	7,704 ± 0,0170	8,832 ± 0,0120

4.4. Discussão dos resultados

Com base nos resultados apresentados neste trabalho, pode-se deduzir o nível de adequabilidade dos algoritmos Skipjack e RC5 às aplicações de RSSF. Primeiramente, a partir do exposto na descrição dos dois algoritmos na Seção 3, pode-se afirmar que o RC5 é mais seguro do que o Skipjack, sob certas configurações de tamanho do bloco,

tamanho da chave e número de *rounds*. Dessa forma, em aplicações que exigem prioritariamente um alto nível de segurança, como aplicações militares e médicas, o algoritmo RC5 é o mais apropriado.

Para aplicações em que a prioridade é a eficiência de memória ROM, o algoritmo mais adequado é o RC5. Por outro lado, o Skipjack é o mais apropriado para aplicações em que a eficiência de memória RAM é mais importante. Para aplicações com restrições de tempo, como aplicações de tempo real em geral, o algoritmo RC5 é a melhor opção, uma vez que ele requer menos tempo de processamento do que o Skipjack. Finalmente, em aplicações como o monitoramento ambiental, nas quais a sobrevivência dos nós é o fator determinante, o RC5 é mais apropriado por requerer um menor consumo de energia.

5. Trabalhos Relacionados

Diversos trabalhos têm apresentado estudos sobre o impacto de algoritmos de criptografia em RSSF [Law, Doumen and Hartel 2004, Guimarães et al. 2005, Simplício Jr 2008, Koo et al. 2008, Jinwala, Patel e Dasgupta 2009, Casado and Tsigas 2009, Lee, Kapitanova and Son 2010], procurando avaliar o desempenho de alguns algoritmos e descobrir qual o custo de prover segurança em tais redes.

Law, Doumen and Hartel (2006) estudaram e avaliaram o desempenho dos algoritmos de criptografia RC5, RC6, Rijndael, MISTY1, KASUMI e Camelia em microcontroladores MSP430F149. Os autores utilizaram apenas duas métricas, quantidade de memória requerida e tempo de processamento, ao contrário desta proposta.

Guimarães et al (2005) avaliaram o desempenho dos algoritmos Skipjack, RC5, RC6, TEA e DES em RSSF, verificando o impacto em CPU e a quantidade de memória requerida por cada algoritmo. No entanto, os autores utilizaram nós sensores Mica2 para a realização dos experimentos, enquanto que na nossa proposta foi utilizada a plataforma MicaZ. Ressalta-se que, de acordo com [Lee, Kapitanova and Son 2010], os resultados para a Mica2 não podem ser usados como referência para a MicaZ.

Simplício Jr (2008) desenvolveu uma nova cifra de bloco para dispositivos com recursos limitados, Curupira, e comparou seu desempenho com os algoritmos Skipjack e AES implementados em plataformas de 8 e 32 bits.

Koo et al (2008) analisaram o desempenho do algoritmo de criptografia HIGHT e compararam com os algoritmos RC5 e Skipjack. Os autores utilizaram nós sensores Mica2 para a realização dos experimentos, contudo o consumo de energia foi avaliado apenas através de simulações. Da mesma forma, Jinwala, Patel e Dasgupta (2009) utilizaram simulações para verificar o impacto da família de algoritmos de criptografia TEA em RSSF e compararam com o desempenho do Skipjack quanto à quantidade de memória requerida, *throughput* e consumo de energia. Diferentemente desses trabalhos, nós realizamos experimentos reais para avaliar os algoritmos Skipjack e RC5.

Casado and Tsigas (2009) avaliaram os algoritmos AES, RC5, Skipjack, Triple-DES, Twofish e XTEA quanto à quantidade de memória requerida, tempo de execução e consumo de energia. Eles realizaram os experimentos na plataforma MSB-430, utilizando o sistema operacional Contiki.

Lee, Kapitanova and Son (2010) realizaram um trabalho denso sobre segurança em RSSF. Os autores avaliaram algoritmos de criptografia, modos de operação e algoritmos de autenticação em plataformas MicaZ e TelosB. Os algoritmos de criptografia analisados foram o Skipjack, RC5, AES e XXTEA. Porém, os autores não apresentaram intervalos de confiança dos dados nem a quantidade de experimentos realizados. Além disso, de acordo com seus resultados, o RC5 consome mais energia que o Skipjack, diferentemente do que nossos resultados indicam. Por fim, a diferença entre os resultados dos autores e os nossos, em relação à quantidade de memória RAM e ROM se deve a possível inclusão do modo de operação ou outras rotinas aos algoritmos originais.

6. Conclusão e Trabalhos Futuros

O trabalho apresentou uma avaliação de desempenho dos algoritmos de criptografia RC5 e Skipjack para RSSF. A quantidade de memória requerida, o tempo de execução e o consumo de energia foram as métricas analisadas para avaliar o desempenho de cada algoritmo.

Os experimentos foram realizados na plataforma MicaZ com o auxílio de um osciloscópio digital. Os resultados indicam que o RC5 requer menos memória ROM, menos tempo de execução e menos energia para a encriptação de dados do que o Skipjack. Em contrapartida, o Skipjack levou vantagem em relação à quantidade requerida de memória RAM, ao tempo de execução e ao consumo de energia durante a decriptação de dados.

Portanto, nosso objetivo foi alcançado, uma vez que a análise dos algoritmos e a descrição do processo detalhado de medição são capazes de orientar o projeto de RSSF quanto à escolha do algoritmo de criptografia mais adequado a sua aplicação, considerando-se os dois algoritmos avaliados. Como trabalhos futuros, pode-se citar a avaliação do impacto e verificação da viabilidade de outros algoritmos de cifra de bloco em RSSF.

Referências

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). Wireless sensor networks: A survey. *Computer networks*, 38: 393-422.

Borges Neto, J. B., Gomes, D. G., Ribeiro Neto, P. F., and Andrade, R. M. C. (2008). *In Proceedings of Euro American Conference on Telematics and Information Systems*.

Casado, L., and Tsigas, P. (2009). ContikiSec: a secure network layer for wireless sensor networks under the contiki operating systems. *In Proceedings of Nordsec*, pp. 133-147.

Crossbow Technology. (2006). MPR-MIB user manual. http://www.cs.ucsb.edu/~nchohan/docs/moteManual.pdf, December.

Feldman, R. M., and Valdez-Flores, C. (1995). Applied probability and stochastic processes. New York, Springer.

Guimarães, G., Souto, E., Kelner, J., and Sadok, D. (2005). Evaluation of security mechanisms in wireless sensor networks. *In Proceedings of 10th Simpósio Brasileiro*

de em Segurança da Informação e Sistemas Computacionais (SBSEG), pages 152-165.

Jinwala, D., Patel, D., and Dasgupta, K. S. (2009). Investigating and analyzing the light-weight ciphers for wireless sensor networks", *INFOCOMP Journal of Computer Science,* Vol 8, Issue 2, pp 39-50.

Karlof, C., Sastry, N., and Wagner, D. (2004). TinySec: A link layer security architecture for wireless sensor networks. *In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys),* pages 162-175.

Kaps, J. P., Gaubatz, G., Sunar, B. (2007). Cryptography on a speck of dust. *Computer*, 40: pp. 38-44.

Koo, W. K., Lee, H., Kim, Y. H., and Lee, D. H. (2008). Implementation and analysis of new lightweight cryptographic algorithm suitable for wireless sensor networks. *In Proceedings Of International Conference On Information Security And Assurance* 2008, Busan, Korea, IEEE Computer Society, pages 73-76.

Law, Y. W., Doumen, J., and Hartel, P. (2006). Survey and benchmark of block ciphers for wireless sensor networks. *ACM Transactions on Sensor Networks TOSN*, vol 2, issues 1.

Lee, J., Kapitanova, K., and Son, S. H. (2010). The price of security in wireless sensor networks. *Computer network*, 54: 2967-2978.

Liu, A., and Ning, P. (2008). TinyECC: A cofigurable library for elliptic curve cryptography in wireless sensor networks. *In Proceedings of 7th International Conference on Information Processing in Sensor Networks (IPSN)*, pages 245-256.

Luk, M., Mezzour, G., Perrig, A., Gligor, V. D. (2007). MiniSec: a secure sensor network communication architecture". *In Proceedings of IEEE International Conference on Information Processing in Sensor Networks (IPSN)*.

Potlapally, N. R., Ravi, S., Raghunathan, A., Jha, N K. A study of the energy consumption characteristics of cryptographic algorithms and security protocols, *IEEE TMC 2005*, pp. 128–143.

Nist, National Institute of Standards and Technology. (1998). Skipjack and kea algorithm specifications. http://cryptome.org/jya/skipjack-spec.htm, December.

Phan, R. C. (2002). Cryptanalysis of full Skipjack block cipher. *Electronics Letters*, 38, 2: 69-71.

Rivest, R. L. The RC5 encryption algorithm. (1994). *In Proceedings of the Second International Workshop on Fast Software Encryption (FSE), pages* 86–96.

Simplício Jr, M. A., Barreto, P. S. L. (2010). Algoritmos criptográficos para redes de sensores. *In Proceedings of 10th Simpósio Brasileiro de em Segurança da Informação e Sistemas Computacionais (SBSEG)*, pages 523-530.

Elegendo Nós Líderes em Redes Tolerantes a Interrupções e Atrasos de Comunicação

Vinícius F. S. Mota[1], **Daniel F. Macedo**[1], **José Marcos Silva Nogueira**[1]

[1]Departamento de Ciência da Computação
Universidade Federal de Minas Gerais (UFMG) – Belo Horizonte – MG – Brazil

{vfsmota,damacedo,jmarcos}@dcc.ufmg.br

Abstract. *Nowadays, mobile communication devices such as smartphone and tablets are more and more cheaper.In many ad hoc applications, reliability in data delivery is still a challenge due to intermittent communication. Delay/Disruption Tolerant Networks (DTN) are a suitable approach to these scenarios. This papper discusses the impact of clusterhead election policies in DTN routing, changing the clusterhead election algorithm in Hierarquichal Group Routing Protocol (HIGROP) [Mota et al. 2009]. The results show that electing clusterheads with higher degrees improves the delivery rates latency times in comparison with the Lowest ID policy, without affecting the election overhead.*

Resumo. *Atualmente, os dispositivos móveis tais como smartphones e tablets estão cada vez mais acessíveis. Nas aplicações de redes ad hoc, prover confiabilidade na entrega de dados ainda é um desafio devido à intermitência de comunicação. As redes tolerantes a interrupções e atrasos (Delay/Disruption Tolerant Network - DTN) são uma abordagem adequada a estes cenários. Este trabalho investiga o impacto de políticas para eleição de nós líderes para no roteamento em DTNs. Para tal, modificamos as políticas de eleição de líderes do algoritmo Hierarquichal Group Routing Protocol (HIGROP) [Mota et al. 2009]. Os resultados mostram que escolher nós com maior vizinhança como líderes melhoram a taxa de entrega e a latência quando comparado à política de menor identificador, sem aumentar o overhead.*

1. Introdução

Conforme o uso de dispositivos móveis de comunicação se tornam cada vez mais comuns cresce demanda por aplicações *ad hoc*. Em certos cenários tais como redes de emergência, comunicação entre grupos de soldados, etc. as aplicações que utilizam os protocolos de roteamento tradicionais não podem garantir a entrega de dados. Os protocolos de roteamento MANET necessitam estabelecer um caminho fim-a-fim para a comunicação entre os nós, como por exemplo os protocolos AODV e DSR [Jain et al. 2004]. Características como baixa densidade de nós, altas taxas de erros de comunicação, alta latência, limitações de banda e tempo de vida dos nós criam cenários desafiadores de rede, que na literatura são chamados de *Challenged Networks* [Fall 2003].

As redes tolerantes a interrupções e atrasos (*Disruption-Delay Tolerant Networking* - DTN) surgiram para possibilitar a comunicação em *Challenged Networks* [Fall 2003]. Para isso, foi definida uma nova camada (denominada *bundle*) capaz de suportar interrupções de comunicação ao utilizar armazenamento temporário de mensagens.

Em [Mota et al. 2009] é proposto um protocolo de roteamento hierárquico baseado em grupos denominado Algoritmo de Roteamento em Grupos hierárquicos (ARGH), o qual foi rebatizado para *Hierarquical Group Routing Protocol* (HIGROP) em [Mota 2009]. No HIGROP cada nó constrói uma tabela com informações dos nós que têm contatos mais frequentes. Em seguida, escolhe-se um líder a partir da tabela de vizinhos. A política de roteamento segue um princípio de hierarquia: a mensagem é transmitida ao nó líder de um grupo e este a encaminha para outro nó.

Neste trabalho investigamos algoritmos de agrupamento e eleição de nós líderes para o HIGROP em cenários em que os nós se movem em grupos, tais como grupos de resgate, campos de batalha, *campi* universitários, etc. Diferentemente das MANETs, na qual o objetivo de um nó líder geralmente é fazer broadcasting eficiente, em DTNs o objetivo de um nó líder é receber a mensagem e repassá-la a outro nó até que esta atinja o destino. Neste artigo modificamos as políticas de eleição de líder do HIGROP e analisamos o impacto da escolha de nós líderes no desempenho do protocolo.

O restante deste trabalho está organizado como se segue: Na Seção 2 apresentamos uma revisão da literatura dos algoritmos de roteamento em DTNs e dos protocolos de agrupamento. Na Seção 3 é detalhado o funcionamento do HIGROP, assim como as políticas utilizadas para eleição de nós líderes e para o repasse de mensagens. Uma análise do comportamento dos grupos de nós e da eficiência do protocolo é apresentada na Seção 4. Por fim, concluímos e apresentamos os trabalhos futuros na Seção 5.

2. Trabalhos Relacionados

2.1. Roteamento em DTN

Os protocolos de roteamento em DTNs diferem no conhecimento que os nós têm sobre a rede. Os protocolos determinísticos inferem a conectividade entre os nós baseado em informações como o tempo médio entre encontros sucessivos de dois nós, estimativas do congestionamento dos nós ou os seus padrões de mobilidade [Handorean et al. 2004, Jain et al. 2004, Liu and Wu 2007, Merugu et al. 2004]. Os protocolos estocásticos, por outro lado, não possuem nenhum conhecimento sobre o estado da rede. Os algoritmos estocásticos são aplicáveis quando a rede tem um comportamento aleatório e pouco pode ser inferido sobre posições futuras dos nós. Esses protocolos variam desde o repasse da mensagem para todos os nós que se conseguir estabelecer contatos até a decisões baseadas no histórico, padrões de mobilidade ou outras informações [Vahdat and Becker 2000, Lindgren et al. 2003, Grossglauser and Tse 2002, Spyropoulos et al. 2005, Mota 2009, Whitbeck and Conan 2010]. Em geral, nas redes sem fio não é possível prever os padrões de movimentação dos nós.

Em [Vahdat and Becker 2000] foi proposto um protocolo de roteamento estocástico Epidêmico para DTNs, no qual o nó origem difunde a mensagem para todos os seus vizinhos e cada um destes por sua vez repassa a mensagem para seus vizinhos. Desta forma, a mensagem é rapidamente distribuída em todos os nós alcançáveis. Quando a largura de banda e o espaço de armazenamento persistente das mensagens (*buffer*) são infinitos, o protocolo Epidêmico é ótimo em relação à taxa de entrega e ao atraso. Devido a isso, o protocolo Epidêmico é comumente utilizado como base de comparação com outros protocolos de roteamento.

Um protocolo de roteamento probabilístico chamado *Probabilist Routing Protocol using History of Encounters and Transitivity* (Prophet) é proposto em [Lindgren et al. 2003]. O Prophet estima uma métrica denominada "previsora de entrega"

$P_{(A,B)}$ sempre que um nó *A* estabelece uma conexão com um nó *B*. As mensagens são repassadas somente para nós com maior previsão de entrega ao destino.

No Prophet, o cálculo da "previsora de entrega" $P_{(A,B)}$ possui três fases e P_{init}, θ e γ são parâmetros configuráveis do algoritmo. Na primeira fase, quando o nó *A* encontra o nó *B*, $P_{(A,B)}$ é atualizado como mostrado na equação 1. Na fase 2, os nós trocam informações sobre outros nós que já conhecem. Com essa propriedade de transitividade o nó *A* atualiza a previsibilidade para um nó *C* que *B* já conhecia. A equação 2 mostra essa transitividade. Na terceira fase, a cada período *k* a previsora de entrega para os nós conhecidos é atualizada (eq. 3).

$$P_{(A,B)} = P_{(A,B)old} + (1 - P_{(A,B)old}) \times P_{init} \quad (1)$$

$$P_{(A,C)} = P_{(A,C)old} + (1 - P_{(A,C)old}) \times P_{A,B} \times P_{B,C} \times \theta \quad (2)$$

$$P_{(A,B)} = P_{(A,B)old} \times \gamma^k \quad (3)$$

Visando diminuir o *overhead* de comunicação, foi proposto em [Mota et al. 2009] o protocolo *Hierarquichal Group Routing Protocol* (HIGROP), que utiliza o agrupamento de nós para fazer o roteamento eficiente das mensagens. Uma mensagem só é encaminhada ao líder do grupo, e este nó a encaminha ao seu destino ou a um outro grupo.

2.2. Protocolos de Agrupamento

Diversos algoritmos de agrupamento e eleição de líderes em redes móveis *ad hoc* foram propostos na literatura [Chinara and Rath 2009]. O algoritmo *Lowest-ID* (LID) [Ephremides et al. 1987], elege como líder o nó com menor identificador. No *Highest Degree* (HD) [Parekh 1994], o nó com maior grau de conectividade se declara líder, enquanto no *Low-Energy Adaptive Clustering Hierarchy* (LEACH) cada nó possui uma probabilidade fixa de se assumir líder, e o agrupamento é refeito periodicamente [Heinzelman et al. 2000].

Existem ainda os protocolos de controle de topologia e roteamento hierárquico que não são protocolos de agrupamento, porém possuem efeitos similares ao agrupamento na rede. No SPAN, um nó decide se vai dormir ou se mantém ativo baseado na conectividade de seus vizinhos a até dois saltos [Chen et al. 2002]. Contudo, o SPAN foi proposto para redes de sensores sem fios estáticas. O protocolo de agrupamento *On-Demand Group Mobility-Based Clustering* (ODGMBC) é utilizado juntamente a protocolos de roteamento reativos [Cramer et al. 2004]. Ele utiliza um campo reservado da camada MAC para fazer o controle da topologia da rede. Em [Whitbeck and Conan 2010] é proposto o HYMAD, um protocolo híbrido para MANETs e DTNs que agrupa os nós baseado em um diâmetro máximo pré-estabelecido, repassando as mensagens somente entre grupos distintos ou aplicando um algoritmo de vetor de distância para roteamento intra-grupos.

3. Hierarchical Group Routing Protocol

O HIGROP constrói uma hierarquia virtual para definir as políticas de encaminhamento de mensagens e evitar assim a disseminação da mensagem por todos os nós da rede. Para isso, cada nó mantém uma tabela com informações sobre os seus contatos com os demais nós na rede. Em cada contato entre um par de nós, *enlace(A,B)*, é atualizado o tempo de contato cumulativo (TCC), que indica o tempo que dois nós estão mantendo contato, e o tempo em que ocorreu essa última conexão (UC). Cada nó HIGROP possui um identificador único (ID) e um papel (P). O HIGROP é composto por dois módulos: Identificação de vizinhança e o encaminhamento e recebimento de mensagens.

As Seções 3.1, 3.2 e 3.3 detalham os módulos de Identificação de Vizinhança, as políticas para eleição de líder e o Repasse e recebimento de mensagens, respectivamente.

3.1. Criação do Agrupamento

O HIGROP possui um Serviço de Identificação de vizinhança (SIV) que mantém uma tabela com as informações dos nós conhecidos. Essas informações são uma quádrupla composta por:

- Identificador do nó (ID), por exemplo, o endereço MAC.
- Papel do nó (P), indica qual o papel do nó em relação ao grupo.
- Última conexão (UC), indica o instante em que recebeu o último *BEACON* do nó.
- Tempo de conexão cumulativo (TCC), tempo total que o nó está na tabela de vizinhos.

Quando dois nós se encontram, cada nó verifica se a tabela de vizinhos já contém o outro nó; caso não contenha, adiciona na tabela o ID e o papel do nó, marca o tempo em que ocorreu essa conexão (UC) e inicializa o tempo de contato cumulativo (TCC = 0). Com um nó já existente na tabela, o SIV atualiza o tempo da UC e calcula o TCC conforme mostrado na Eq. 4, que define há quanto tempo existe o *enlace(A,B)*, sendo T_{atual} o instante em que ocorreu o contato. Enquanto a conexão estiver ativa, a cada período ω o valor de TCC é atualizado.

$$TCC = TCC + T_{atual} - UC \tag{4}$$

O papel inicial de cada nó é *ISOLADO* e dessa forma todos são potenciais líderes. Em um primeiro contato, cada nó adiciona o endereço do outro em sua a tabela e marca o tempo (*t*) em que ocorreu a conexão ($UC = t$). A cada nova conexão ou a cada *BEACON* recebido de uma conexão ativa com o mesmo nó, é executado o cálculo da diferença entre o novo tempo (T_{atual}) e o tempo marcado anteriormente. Um nó é considerado *vizinho* se satisfazer as Equações 5 e 6.

$$TCC_{enlace(A,B)} \geq \beta \tag{5}$$

$$(T_{atual} - UC)_{enlace(A,B)} \leq \alpha \tag{6}$$

As constantes α e β são parâmetros do HIGROP, sendo α o tempo máximo que define um nó como vizinho sem que estejam conectados (evitando que pequenas desconexões descaracterizem o nó como um vizinho). A constante β indica o tempo mínimo que um nó tem que manter a conexão para ser considerado um *vizinho*.

3.2. Eleição de Líder

Neste trabalho implementamos duas políticas de eleição de líder para o HIGROP: Uma utilizando o algoritmo de menor identificador, denominado HIGROP-LID, e outra utilizando o algoritmo de maior grau para se declarar líder, batizado de HIGROP-HD. É importante ressaltar que esses algoritmos somente influenciam na escolha do nó líder. O restante do funcionamento do algoritmo é idêntico para ambas versões do HIGROP.

Algoritmo de Menor ID - HIGROP-LID: Os nós aguardam um período de *warm up* para que todos os vizinhos possam ser reconhecidos. Após esse período, a partir da lista

de vizinhos construída, o nó com o menor identificador assume a liderança e altera seu papel para *LIDER*. O nó líder envia uma mensagem aos seus vizinhos informando sobre sua liderança. Cada nó informado sobre o líder passa para o papel *MEMBRO* e informa aos seus vizinhos (que estão no papel *ISOLADO*) sobre o líder; estes também passam para o papel de *MEMBRO*. Periodicamente é enviado um *BEACON* com a operação *difusao(LIDER,m)*, sendo LIDER o papel do nó *m*. Um nó deixa de ser LIDER caso fique isolado (sem nenhum vizinho), evitando mudanças constantes de lideres e consequentemente trocas desnecessárias de mensagens.

Algoritmo de Maior Grau - HIGROP-HD: Após um período ω, cada nó informa para os nós que já foram considerados *vizinhos* e já estão com o papel de MEMBRO o seu grau, que é calculado com base no número de enlaces ativos no instante. Ao final dessa iteração os nós vizinhos sabem qual o nó com maior grau entre eles. Aquele que tiver o maior grau se declara LIDER e informa essa situação aos seus vizinhos. Periodicamente, a cada período ω é feito o re-cálculo do grau de cada nó. Diferentemente do algoritmo de líder com menor ID, se neste re-cálculo um nó que era LIDER tiver grau menor, o outro nó passa a ser LIDER e o antigo LIDER assume o papel de MEMBRO.

Em [Chiang et al. 1997] é apresentada uma prova que o algoritmo de eleição de líder Lowest ID (LID) é mais estável em MANETs do que os algoritmos baseados no *Highest Degree*. Os resultados apresentados na seção 4 mostram essa estabilidade do LID. Porém, devido ao grau dos nós responsáveis por receber a mensagem no *Highest Degree* ser maior que os líderes no LID, estes têm mais chances de encaminhar a mensagem ao destino. Contudo, a eleição de líder utilizando o *Highest Degree* ocasiona um aumento no overhead de comunicação.

3.3. Políticas de Roteamento

No HIGROP um nó MEMBRO somente repassa a mensagem ao destino ou ao LIDER, enquanto o nó LIDER distribui as mensagens entre grupos.

Se uma mensagem *M* deve ser enviada entre nós que fazem parte do mesmo grupo, então o repasse de mensagem é feito diretamente para o destino. Se a mensagem deve ser enviada a outro grupo, ela é repassada para o LIDER, e esse envia a mensagem para um nó que não faça parte do seu grupo, repassando a um nó que não seja conhecido em sua lista de *vizinhos*. O nó do grupo que recebeu a mensagem repete essas operações até que a mensagem seja entregue.

4. Simulação e Análise

Nesta seção foram avaliadas as variações do HIGROP utilizando o simulador ONE (*Opportunistic Network Environment*) [Keränen et al. 2009]. O ONE simula um modelo de comunicação tolerante a interrupções, onde os nós seguem o paradigma *armazenar-segurar-repassar* mensagens (*store-carry-foward*), podendo mantê-las em um *buffer* caso o nó não tenha conexão direta com o destino. Todos os resultados apresentados possuem 95% de intervalo de confiança. Cada teste foi executado quinze vezes, alterando a semente geradora do padrão de mobilidade a cada vez.

O modelo de mobilidade *Reference Point Group* (RPGM) foi utilizado em todos os cenários. O RPGM representa o movimento aleatório de grupos de nós [Camp et al. 2002], no qual os movimentos dos nós de um grupo são baseados no caminho percorrido pelo centro lógico do grupo. A aplicação simulada consiste em um cenário com uma área de 3000x3000m. O tempo simulado foi de 6000s, cada nó possuía

um buffer de 500M e movia-se com velocidade entre 0-10m/s normalmente distribuídos. As mensagens foram geradas com distribuição uniforme entre 20 e 35s e com tamanho entre 10kB e 500kB, que representa troca de arquivos de texto e imagens de baixa resolução.

No PROPHET, foram utilizados os parâmetros $P_{init} = 0.75$, $\theta = 0.25$, $\gamma = 0.98$ e k = 30s como em [Lindgren et al. 2003].

Avaliamos o tamanho médio dos grupos criados, quantidade média de nós líderes e o desempenho do protocolo em relação à taxa de entrega, *overhead* relativo de comunicação e latência comparando-o com o protocolo epidêmico. O *overhead* relativo informa quantas mensagens tiveram que ser transmitidas na rede para cada mensagem entregue corretamente ao destino.

A sub-seção 4.1 verifica o comportamento dos nós ao variar o raio de alcance dos mesmos. Na Subseção 4.2 variamos a quantidade de nós e comparamos o desempenho do HIGROP com o protocolo epidêmico e o PROPHET.

4.1. Estabilidade dos Grupos

No primeiro teste, verificamos o impacto do parâmetro β no número médio de vizinhos. Os parâmetros ω e α foram fixados em 60s, ou seja, uma reavaliação era feita a cada 60s e um par de nós somente era desconsiderado vizinho se ficassem mais de 60s desconectados. O raio de comunicação dos nós foi fixado em 250m em uma rede com 50 nós.

A Figura 1 apresenta a variação no número de vizinhos. O HIGROP-HD e o HIGROP-LID possuem políticas idênticas para tornar um nó vizinho. Observa-se que aumentar o tempo necessário para dois nós deixarem de serem vizinhos (β) faz com que o número médio de nós considerados *vizinhos* seja maior ao longo do tempo, mas para todos os valores a curva possui o mesmo comportamento ao longo do tempo. Testes preliminares mostraram que o número de líderes gerados é idêntico para toda configuração do parâmetro β. No segundo teste, o alcance foi variado entre 50m e 250m, o parâmetro β foi fixado em 30s por ter gerado o menor número médio de vizinhos no primeiro teste. O HIGROP-LID e o HIGROP-HD foram comparados quanto ao número de nós LIDERES, média de *vizinhos* e média de contatos em relação ao tempo simulado.

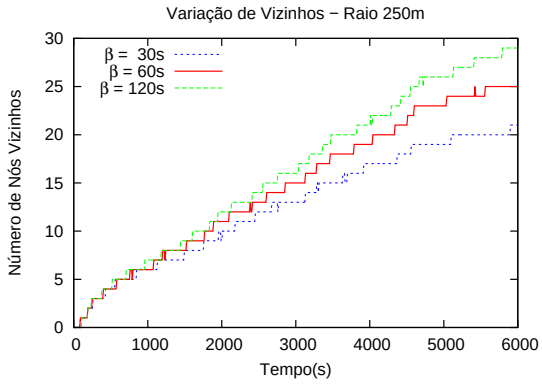
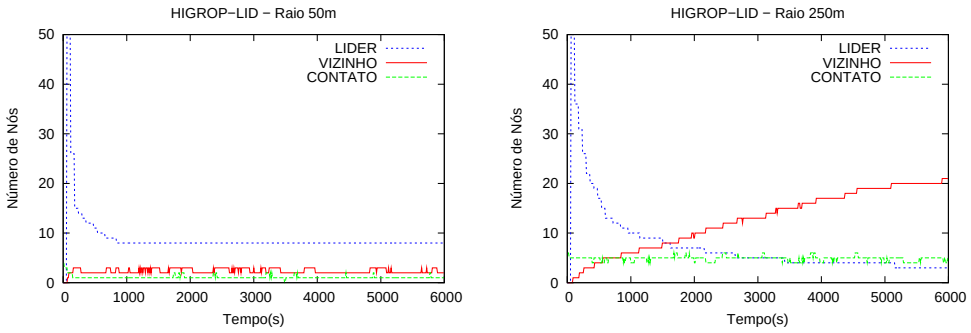


Figura 1. Número médio de vizinhos

As Figuras 2(a) e 2(b) mostram as mudanças de vizinhança no HIGROP-LID. Como esperado, o número de nós que se tornam lideres no começo é alto, contudo conforme

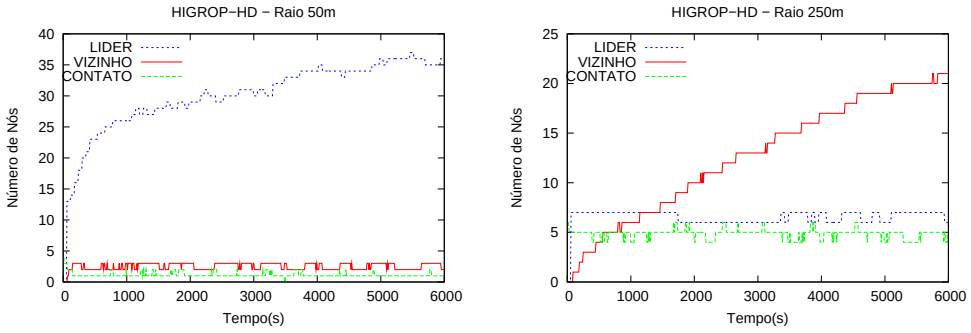
os nós vão se agrupando o número de nós LIDERES diminui ao longo do tempo, estabilizando com aproximadamente 1000s. Quando o raio de comunicação dos nós é de 50m, Figura 2(a), observa-se que o número de *vizinhos* é próximo ao de contatos. Quando o raio é aumentado para 250m observamos que mesmo com um valor β baixo (30s), o número de *vizinhos* é acumulativo, o que pode ser explicado devido aos contatos frequentes entre os nós participantes da rede, assim o número de LIDERES diminui.

As Figuras 3(a) e 3(b) apresentam os resultados para o HIGROP-HD. Diferentemente do HIGROP-LID, com o raio de comunicação de 50m, o número de líderes é baixo inicialmente. Caso um nó LIDER se torne *vizinho* de outro nó LIDER com maior grau, esse passa a ser um MEMBRO do grupo, ocasionando variações no número de LIDERES ao longo do tempo. Ao aumentar o raio para 250m, observa-se que o número de nós líderes permanece praticamente estável, enquanto o número médio de nós *vizinhos* torna-se acumulativo.



(a) HIGROP-LID - Raio de 50m (b) HIGROP-LID - Raio de 250m

Figura 2. HIGROP-LID: Número de nós líderes, vizinhos e contatos - 50 nós



(a) HIGROP-HD - Raio de 50m (b) HIGROP-HD - Raio de 250m

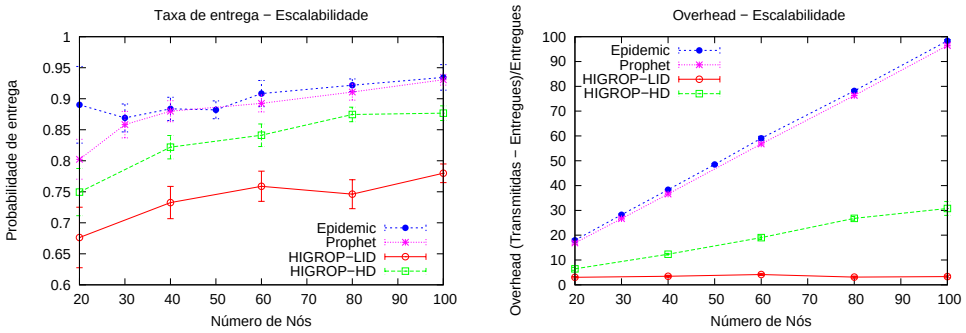
Figura 3. HIGROP-HD: Número de nós líderes, vizinhos e contatos - 50 nós

No HIGROP um nó somente deixa o papel de LIDER caso o mesmo não esteja carregando mensagens destinadas a outros nós. Isso evita que as mensagens tenham que ser todas transferidas para o novo nó líder por causa da política de roteamento. Porém, a alternância de nós LIDERES é eficiente evitaria o esgotamento de recursos de um único nó [Heinzelman et al. 2000].

4.2. Escalabilidade

Para avaliar a escalabilidade, o número de nós foi variado entre 20 e 100 e as duas versões do HIGROP foram comparadas ao protocolo Epidêmico e ao PROPHET. No HIGROP, baseado em testes anteriores, os parâmetros β, ω e α foram fixados em 30s,60s e 60s respectivamente.

As figuras 4(a), 4(b) e 5 mostram a taxa de entrega, o *overhead* relativo e a latência respectivamente. Observamos que as taxas de entrega possuem comportamento quase constante em relação ao número de nós para os quatro algoritmos avaliados. Quando há *buffer* de armazenamento alto, o Epidêmico é ótimo em relação à taxa de entrega, tendo nesse cenário uma taxa média de entrega de 90% das mensagens. Nesse cenário o PROPHET se comportou muito semelhante ao Epidêmico, mantendo a taxa de entrega e overhead comparáveis ao Epidêmico, porém com um atraso inferior. Em cenários com baixa densidade



(a) Taxa de Entrega (b) Overhead

Figura 4. Escalabilidade

de nós (20 nós) tanto o HIGROP-LID quanto o HIGROP-HD obtiveram desempenho bem abaixo do epidêmico e do Prophet, com 68% e 75% das mensagens entregues respectivamente. Com o aumento da densidade de nós na rede ambos os protocolos mostram uma melhoria de 10% na taxa de entrega. Em especial, o HIGROP-HD atinge 87% da taxa de entrega no cenário com 100 nós. Contudo, isso acarreta em um aumento do *overhead* em relação ao HIGROP-LID. Quando comparado ao Epidêmico, que tem *overhead* linear de acordo com o número de nós, o HIGROP-HD mantém *overhead* três vezes menor na rede com 100 nós participantes. Percebe-se que o overhead gerado pelo HIGROP-HD é similar ao número de líderes criados na rede. Como o HIGROP-HD seleciona os nós líderes

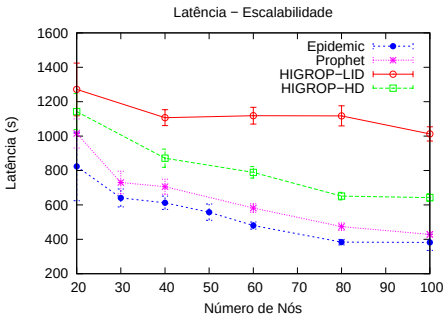


Figura 5. Latência

como aqueles com maior número de contatos, ocorre um repasse de mensagem mais efi-

ciente tanto intra-grupos como intergrupos. Isso pode ser verificado pela figura 5 que apresenta a latência nos três protocolos. A latência do protocolo Epidêmico é um limite inferior para a comparação entre os protocolos de roteamento DTN. Em todos os protocolos, a latência diminui de acordo com o número de nós. A latência no HIGROP-HD no melhor caso (rede com 100 nós) é 63% inferior à do HIGROP-LID. Como o número de líderes gerados pelo HIGROP-HD é maior, isso acarreta em mais repasses de mensagens inter-grupos, o que aumenta o *overhead*, mas diminui a latência.

5. Conclusões

Neste trabalho estudamos o impacto de uma eleição de líder baseado no grau do nó em relação à escolha baseada no menor identificador. Foram desenvolvidas duas versões do HIGROP, que é um protocolo hierárquico que cria uma vizinhança "virtual" entre os nós. O HIGROP-LID, em que o nó com menor identificador dentro de um grupo assume a liderança e o HIGROP-HD, no qual o nó com maior número de contatos é escolhido líder.

O HIGROP-HD, quando comparado ao HIGROP-LID permite mais mudanças de nós líderes, e fazendo com que a rede tenha um número maior de líderes. Isso permite que o repasse de mensagem seja maior, aumentando a probabilidade de entrega da mensagem. Devido à escolha de líderes que possuem maior número de contatos, o HIGROP-HD mostrou-se mais eficiente na entrega de mensagens, porém adicionando *overhead* de comunicação na rede. Ao custo desse *overhead*, foi possível diminuir a latência do protocolo para qualquer número de nós. Apesar do *overhead* no HIGROP-HD ser até três vezes inferior ao apresentado pelo protocolo epidêmico, a latência no HIGROP-HD é 70% maior do que a latência no epidêmico. Diminuir a latência sem impactar o *overhead* de comunicação ainda é um desafio em redes DTN.

Como trabalhos futuros pretende-se adicionar a possibilidade de utilizar o roteamento tradicional para os momentos em que a rede for conexa entre a origem e o destino da mensagem e utilizar modelos de mobilidade baseados em *traces* reais.

Referências

Camp, T., Boleng, J., and Davies, V. (2002). A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing*, 2(5):483–502.

Chen, B., Jamieson, K., Balakrishnan, H., and Morris, R. (2002). Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *Wireless Networks*, 8(5):481–494.

Chiang, C., Wu, H., Liu, W., and Gerla, M. (1997). Routing in clustered multihop, mobile wireless networks with fading channel. In *proceedings of IEEE SICON*, volume 97, pages 197–211. Citeseer.

Chinara, S. and Rath, S. (2009). A Survey on One-Hop Clustering Algorithms in Mobile Ad Hoc Networks. *Journal of Network and Systems Management*, 17(1):183–207.

Cramer, C., Stanze, O., Weniger, K., and Zitterbart, M. (2004). Demand-driven clustering in manets. In *International Conference on Wireless Networks*, pages 81–87.

Ephremides, A., Wieselthier, J., and Baker, D. (1987). A design concept for reliable mobile radio networks with frequency hopping signaling. *Proceedings of the IEEE*, 75(1):56–73.

Fall, K. (2003). A delay-tolerant network architecture for challenged internets. *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 27–34.

Grossglauser, M. and Tse, D. (2002). Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transactions on Networking (TON)*, 10(4):477–486.

Handorean, R., Gill, C., and Roman, G. (2004). Accommodating Transient Connectivity in Ad Hoc and Mobile Settings. *Proceedings Pervasive Computing: Sec. International Conference*, pages 18–23.

Heinzelman, W. R., Chandrakasan, A., and Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Hawaii International Conference on System Sciences*.

Jain, S., Fall, K. R., and Patra, R. K. (2004). Routing in a delay tolerant network. *ACM SIGCOMM Computer Communication Review*, pages 145–158.

Keränen, A., Ott, J., and Kärkkäinen, T. (2009). The one simulator for dtn protocol evaluation. In *SIMUTools '09: Proceeding of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA. ACM.

Lindgren, A., Doria, A., and Schelen, O. (2003). Probabilistic routing in intermittently connected networks. *Mobile Computing and Communications Review*, 7(3):19–20.

Liu, C. and Wu, J. (2007). Scalable routing in delay tolerant networks. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, pages 51–60. ACM Press New York, NY, USA.

Merugu, S., Ammar, M., and Zegura, E. (2004). Routing in space and time in networks with predictable mobility. Technical report, Georgia Institute of Technology.Technical Report GIT-CC-04-7.

Mota, V. (2009). Um protocolo de roteamento tolerante a interrupções de comunicação para redes sem fios móveis em cenários de emergência. Master's thesis, Universidade Federal de Minas Gerais.

Mota, V. F. S., Silva, T., and Nogueira, J. (2009). Introduzindo Tolerância a Interrupções em Redes Ad Hoc Móveis para Cenários de Emergência. *27o Simpósio Brasileiro de Redes de Computadores*, pages 671–684.

Parekh, A. (1994). Selecting routers in ad-hoc wireless networks. In *Proceedings SBT/IEEE Intl Telecommunications Symposium*, pages 420–424.

Spyropoulos, T., Psounis, K., and Raghavendra, C. (2005). Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In *ACM SIGCOMM workshop on Delay-tolerant networking*, pages 252–259. ACM.

Vahdat, A. and Becker, D. (2000). Epidemic routing for partially connected ad hoc networks. *Tec. Report, Duke University*.

Whitbeck, J. and Conan, V. (2010). HYMAD: Hybrid DTN-MANET routing for dense and highly dynamic wireless networks. *Computer Communications*, 33(13):1483–1492.

Um método para otimização de aplicações de Redes de Sensores através de análises do código executável

Jônatas C. Oliveira [1], Carlos M. S. Figueiredo[2], Eduardo F. Nakamura[2]

[1]Departamento de Ciência da Computação - Universidade Federal do Amazonas (UFAM)
Av. Gen. Rodrigo Octávio Jordão Ramos, 3000, Coroado I, 69077-000 Manaus, AM.

[2]Fundação Centro de Análise, Pesquisa e Inovação Tecnológica - FUCAPI
Av. Danilo de Matos Areosa, 381, Distrito Industrial - 69075-351 Manaus, AM.

jco@dcc.ufam.edu.br,{eduardo.nakamura,mauricio.figueiredo}@fucapi.br

Abstract. *In Wireless Sensor Networks, the concern of developing efficient applications is a constant need due to its limited resources. Especially, frameworks of development, operating systems and compilers are concern in generating efficients executables codes. However, optimization opportunities are still possible to find through analysis of the application executable code. We present a method that, through the analysis and simulation of the executable code of the sensor node, we can find optimization opportunities, and new considerations of application efficiency are observed. We utilized the method on the TinyOS's Antithief Application, reducing 6,2% the microprocessor time in active state.*

Resumo. *Em Redes de Sensores, a preocupação com o desenvolvimento de aplicações mais eficientes é uma necessidade constante devido a limitação de recursos. Particularmente, ferramentas de desenvolvimento, SOs e compiladores se preocupam em gerar executáveis eficientes. No entanto, oportunidades de melhoria na aplicação ainda podem ser encontradas através de análises do seu código executável. Apresentamos um método que possibilita encontrar oportunidades de otimizações, além de que novas considerações da eficiência da aplicação sejam observadas. Aplicamos o método na aplicação Antithief do TinyOS, reduzindo em 6,2% o tempo ativo do processador.*

1. Introdução

Rede de Sensores Sem Fio (RSSF)[Akyildiz et al. 2002] tem obtido atenção dos pesquisadores devido ao seu potencial de ser empregado como infraestrutura para computação ubíqua. Devido a limitação de recursos do nó sensor, como energia, processamento e armazenamento, todo o projeto de uma RSSF visa usufruir o máximo da eficiência desses recursos. No entanto, as otimizações dos compiladores para códigos de RSSFs, assim como nos compiladores de uso comum, têm como principais parâmetros de otimização o processamento e o tamanho de código, enquanto a energia consumida pelo código não é levada em consideração, que, em RSSF, é o recurso mais escasso. Poucos trabalhos têm sido propostos na literatura que abordam esse problema nos compiladores de RSSF [Lane and Campbell 2006, Zhang et al. 2009].

Outro problema a ser destacado é que o Sistema Operacional (SO) em RSSFs mais utilizado, TinyOS [Levis et al. 2004], possui um conjunto de funções que foram escritas para uso geral, ou seja, para qualquer que seja o contexto da RSSF, não permitindo assim, a otimização dessas funções de acordo com a aplicação. Dessa forma, o trabalho visa demonstrar que códigos gerados para RSSFs muitas vezes ainda podem ser otimizados, através da utilização de análises do código executável. Ao final, pretende-se apresentar um método capaz de oferecer possíveis otimizações a partir dessas análises. Tais análises de código são: análise das interrupções do código, análise do consumo de energia, e o Grafo de Fluxo de Controle (GFC) do código. Elas são análises implementadas, em parte, pela ferramenta Avrora [Titzer et al. 2005], e posteriormente estendidas por nós, de acordo com o que era necessário de se obtê-las.

O artigo está organizado da seguinte forma: na seção 2 são descritos os trabalhos relacionados. Na seção 3, é apresentado o método proposto e a ferramenta utilizada pelo método. Na seção 4, aplica-se o método nas aplicações Blink e Antithief como estudos de caso, e são discutidos os resultados. E, por fim, na seção 5, conclui-se e cita-se os trabalhos futuros.

2. Trabalhos Relacionados

Os trabalhos relacionados se limitaram aos simuladores de RSSF que oferecem funcionalidades de análises de código.Simulação em RSSFs tem sido muito estudada nesta década, já que o funcionamento incorreto da rede ou de estimações são irreversíveis. A maioria dos simuladores em RSSF [Sundresh et al. 2004, S. Park and Srivastava 2000, Levis et al. 2003] foram destinados mais a analisar a rede em si, abstraindo o *hardware* e o código executável do nó sensor, inviabilizando o uso deles em nossa análise.

Os simuladores Avrora [Titzer et al. 2005] e Ipen [Joe et al. 2008], contudo, simulam em nível de instrução. O Ipen foi desenvolvido com o único objetivo de analisar o consumo de energia, enquanto o Avrora é uma ferramenta que fornece simulação e um conjunto de análises para códigos escritos para nós sensores Mica2 e MicaZ. Ele é flexível, oferecendo uma infraestrutura que facilita o acréscimo de novas funcionalidades. A finalidade de tais análises é facilitar o entendimento do projetista quanto a execução de sua aplicação. Porém, entendemos que esse conjunto de análises do Avrora somadas a outras análises por nós implementadas podem ser mais úteis que uma simples observação, permitindo localizar otimizações na aplicação. Por isso, escolhemos o Avrora, estendendo em cima dele nossas análises, para a aplicação do método proposto.

3. Método Proposto e Ferramenta Estendida

É apresentado um método que, a partir de análises do código executável do nó sensor, oferecidos por uma ferramenta estendida do Avrora, possibilita que se encontre otimizações na aplicação. O método é composto por uma análise crítica e um guia de uso. A primeira é a descrição de que maneira cada análise oferecida pela ferramenta poderia gerar alguma otimização na aplicação, enquanto a segunda é útil para que o usuário final da ferramenta possa aplicar o método através de uma sequência de

passos na tentativa de se obter alguma otimização da aplicação. A efetuação da otimização detectada é feita manualmente pelo usuário. É importante ressaltar que as análises são dependentes do cenário da simulação, cabendo ao usuário especificar o cenário de acordo com a rede real, para que as otimizações sejam efetivas. A apresentação é, então, dividida para cada análise de código da ferramenta. As análises de código da ferramenta são: grafo de fluxo de controle, análise do consumo de energia e análise das interrupções. Para uma melhor compreensão das análises, pode-se observar as figuras e tabelas de cada uma, nos estudos de caso.

3.1. Grafo de Fluxo de Controle

Um GFC é uma representação do fluxo de execução do código, que permite uma observação detalhada da execução do código do nó sensor através dos dados inseridos nele. O GFC é uma funcionalidade do Avrora, que foi expandida.

Os dados inseridos nos nós do GFC são: nome do bloco (endereço no código), quantidade de ciclos do processador, alteração do estado de algum componente do nó sensor e quantidade de vezes que o caminho foi percorrido. Os nós se classificam em: entrada de alguma interrupção, entrada de procedimento, nó simples que termina com retorno simples ou retorno de uma interrupção, e nó simples.

Uma funcionalidade estendida no GFC do Avrora é a marcação dos trechos de código mais executados durante o tempo de execução do código. Essa funcionalidade consiste na marcação, que se faz pela colorização, dos nós que são mais executados a fim de destacá-los visualmente. A percentagem dos nós mais executados que se queira marcar é passado como parâmetro no uso dessa funcionalidade da ferramenta.

Análise Crítica do GFC

O GFC do Avrora permite observar como o programa é estruturado e o mapeamento do código em alto nível para o código *assembly* realizado pelo compilador. Com a extensão, pode-se observar os comportamentos dos componentes do nó sensor, e da execução do código ao longo do tempo, pode-se realizar cálculos do consumo de processamento entre os caminhos de execução do código. Além disso, através da classificação dos nós do grafo, observa-se o que ocorre no tratamento de cada interrupção da aplicação e das chamadas de funções. A seguir, apresenta-se detalhadamente cada análise crítica que se pode fazer do GFC, e seus respectivos guias de uso.

A. Eliminação de Código

O nesC automaticamente detecta os nós do grafo que não são alcançáveis e os retira do código compilado, eliminando assim *código morto*. Porém, outra forma de eliminação de código é a detecção de componentes que foram adicionados pelo TinyOS e não serão úteis na aplicação. Essa detecção ocorre através da observação dos nomes dos nós de entrada de funções e saltos, pois a entrada da execução de algum componente do TinyOS sempre é um nó de entrada de uma função. Sendo assim, verifica-se os componentes inclusos pelo TinyOS e retiram-se aqueles desnecessários pela aplicação. Logo, necessita-se de um conhecimento dos componentes do TinyOS, assim como do funcionamento da aplicação.

Guia de Uso: (i) detectar funções no GFC que são inicializações de algum

componente; (ii) observar a aplicação em alto nível e verificar se a mesma necessita desses componentes adicionados pelo TinyOS; (iii) caso algum componente não seja necessário, descartar a inclusão do mesmo na aplicação, alterando as bibliotecas do SO.

B. Otimização do Consumo de Processamento

O último ponto destacado na eliminação de código consequentemente reduzirá o consumo de processamento da aplicação, mesmo que esta redução seja mínima. No entanto, o ponto de maior destaque na redução nos ciclos de processador está na funcionalidade de destaque dos nós do grafo mais percorridos, pois, possibilita que otimizações no código sejam feitas apenas para os locais do código mais percorridos.

Guia de Uso: (i) gerar GFCs para diferentes percentagens dos trechos de código mais executados, com o objetivo de definir qual a melhor percentagem de código a ser otimizada, de acordo com alguma métrica estabelecida pelo usuário; (ii) selecionar um conjunto de técnicas de otimização de código, cabendo ao usuário definir quais as mais apropriadas para a aplicação em desenvolvimento; (iii) efetuar as técnicas nas funções que possuem trechos de código mais executados.

C. Gerenciamento dos Componentes do Nó Sensor

O GFC gerado permite observar o comportamento dos componentes ao longo da execução do código. Aprimora-se esse estudo com a informação do custo de ciclos de processamento em cada bloco, pois, permite o cálculo do limite inferior e superior de tempo que um componente será ativado ou desativado. Assim, espera-se que o estudo do comportamento dos componentes contribua para o desenvolvimento de gerenciadores de componentes mais eficientes [Paleologo et al. 1998, Klues et al. 2007].

Guia de Uso: (i) encontrar no GFC os nós que contém alterações dos estados dos componentes; (ii) para cada nó do GFC que altera o estado de um determinado componente, fazer cálculos para se encontrar a distância mínima e máxima em ciclos de processamento até outro nó do GFC que poderia alterar o estado do mesmo componente; (iii) contar quantas vezes cada estado do componente é alterado ao longo da execução; (iv) utilizar tais dados estatísticos como auxílio na construção de gerenciadores de componentes do nó sensor.

3.2. Análise do Consumo de Energia

A análise do consumo de energia observa o custo de energia de cada componente do nó sensor durante o tempo de execução da aplicação. E mais especificamente, para cada componente, ela fornece a quantidade de tempo gasto e a energia consumidade de cada estado que foi transitado durante a simulação. A funcionalidade foi implementada pelo próprio Avrora.

Análise Crítica do Consumo de Energia

A vantagem é prever o consumo de energia de cada componente do nó sensor e do tempo de permanência em cada um de seus estados, assim como o tempo de vida da rede também. O principal objetivo dessa análise é servir como uma métrica de qualidade do gerenciamento de energia da aplicação, e também para usá-la para

comparações em casos de otimização.

Guia de Uso: (i) verificar quais os estados de ativação alcançados para cada componente durante a simulação e o tempo em cada estado, oferecendo um auxílio ao *profiling* da aplicação. O *profiling* é uma modelagem do comportamento da aplicação através de uma análisa dinâmica da mesma, útil para que otimizações sejam aplicadas de acordo com esse comportamento.

3.3. Análise das Interrupções

Análise das interrupções observa as interrupções de *hardware*. Ela é bastante útil para relatar o comportamento da aplicação, principalmente em Sistemas Operacionais orientados a eventos como o TinyOS. A funcionalidade é disponibilizada pelo Avrora, sem a necessidade da extensão da mesma. Ela é uma análise dinâmica da aplicação, onde se obtém uma tabela ao final da simulação com todas as interrupções que ocorreram, o intervalo de tempo entre uma mesma interrupção e o período de latência de cada. A tabela 1 é um exemplo de algumas das interrupções que ocorreram na aplicação Antithief.

Tabela 1. Exemplo de Tabela de Interrupção

Num	Nome	Chamadas	Separação	Latência	Wakeup
10	TIMER2 COMP	0	-	-	-
11	TIMER2 OVF	0	-	-	-
12	TIMER1 CAPT	8178	14583	161	4
13	TIMER1 COMPA	4104	29223	12.4	4
14	TIMER1 COMPB	0	-	-	-
15	TIMER1 OVF	263	389074	4.456	4

Análise Crítica das Interrupções

A tabela de interrupções permite observar quais interrupções não ocorreram durante a simulação. Dessa forma, mesmo que a tabela de interrupções não seja essencial para saber quais interrupções não possuem tratamento, ela facilita a observação em comparação ao GFC, pois, este último pode ser muito grande. A verificação de quais tratamentos de interrupções podem ser ativados durante a execução do código é necessário, pois, alguns componentes são ativados apenas nesses tratamento de interrupções.

A informação entre os intervalos das interrupções são úteis apenas para observar se alguma interrupção ocorre muito frequentemente, podendo levar a degradação do sistema. O período de latência é importante para ser acrescentado nos cálculos entre tempo máximo e mínimo de intervalo entre a alteração de estado de algum componente, pois a latência soma-se com os valores dos ciclos do processamento no nó de cada entrada no tratamento de interrupção.

Guia de Uso: (i) verificar quais interrupções ocorreram durante a simulação e, através disso, ter indícios da execução dos componentes do nó sensor. Um exemplo seria verificar quantas interrupções do rádio ocorreram durante a execução; (ii) utilizar os intervalos das interrupções para auxiliar a construção do *profiling* da

aplicação, sabendo quais componentes serão muito acionados; (iii) verificar se os intervalos das interrupções são o mínimo suficiente para não degradar o sistema; (iv) utilizar os períodos de latência como auxílio em um cálculo mais preciso dos ciclos de processamento máximo e mínimo para alteração do estado de um componente do nó sensor.

4. Estudos de Caso

4.1. Estudos de Caso - Blink

Aplicamos o método proposto na aplicação Blink, do TinyOS 2.1, usada pra testar os temporizadores e os *Leds* do nó sensor. Por sua simplicidade facilitar a observação do método, optou-se por apresentar primeiramente este estudo de caso. Simulamos quatro horas de sua execução, pois, como a aplicação só usa os temporizadores, executando sempre as mesmas operações em um ciclo curto de tempo, quatro horas é suficiente para que as execuções de inicialização não exerçam influência no *profiling* da aplicação.

As otimizações aplicadas foram duas. A primeira foi a detecção do componente CC1000, que é o rádio, que não seria útil nessa aplicação, por isso o removemos manualmente no TinyOS, e simulamos novamente. A segunda otimização foi do processador. Verificou-se pela análise de energia que somente dois estados eram transitados no processador, e que no GFC, eles tinham a mesma quantidade de vezes transitadas. Dessa forma, verificamos que sempre que a função MCU-Sleep do TinyOS executava, ela transitava para o mesmo estado. Então, a otimização consistiu em retirar as condições do código para o cálculo de qual seria o estado a ser transitado.

Tabela 2. Aplicação do Blink antes e depois do método ser aplicado.

-	Energia - MCU	Redução	Tamanho do Código	Redução
Antes do Método	0,3583 Joules	-	2198 Bytes	-
Ponto 1 - CC1000	0,3581 Joules	0,005%	2052 Bytes	6,6%
Ponto 2 - McuSleep	0,3374 Joules	5,8%	2070 Bytes	5,8%
Aplicação de ambos	0,3370 Joules	5,8%	1924 Bytes	12,4%

Ao aplicá-las, obtivemos otimizações de consumo de energia e tamanho do código que são expressas na tabela 2. A segunda coluna mostra a quantidade de energia consumida em cada etapa do método, a terceira apresenta a percentagem reduzida em comparação a aplicação antes de qualquer otimização ser efetuada, a quarta coluna é o tamanho de código em cada etapa, enquanto a última coluna é a percentagem de redução do tamanho do código em comparação a aplicação antes das otimizações.

4.2. Estudos de Caso - AntiThief

O estudo de caso Blink foi útil como demonstração de uso da ferramenta. Este outro foi escolhido para que o método fosse realizado numa aplicação mais realística e típica de RSSF do TinyOS, denominada AntiThief, para a plataforma MicaZ. A

aplicação consiste de leituras periódicas do sensor de luz por cada sensor, e este mesmo verifica se a quantidade de luz alcançou o limite adotado e, caso não, alerta a existência de um ladrão ligando o *Led* Vermelho conjuntamente com um aviso de alerta ao nó *sorvedouro* através de uma comunicação com múltiplos saltos.

Simulou-se a aplicação na ferramenta Avrora estendida em três diferentes tempos: dois minutos, seis minutos e dez minutos para a plataforma MicaZ. A rede foi composta por três nós, sendo o nó 0, o nó sorvedouro. Os nós foram distribuídos em uma linha, sendo que cada nó só possui alcance aos nós mais próximo à esquerda e à direita. Escolheu-se o nó 1 para ser analisado pela ferramenta, pois, como ele é um nó intermediário, terá que receber e transmitir pacotes, executando obrigatoriamente os trechos de código que incluem tais funções. Os eventos, detecção de ladrão por parte de um nó sensor, foram programados para ocorrer sempre uma vez por minuto no nó 1, e uma vez a cada dois minutos no nó 2, em todos os três tempos de simulação. Os tempos escolhidos de simulação foram curtos, pois, uma simulação longa geraria muitas modificações dos estados dos componentes no GFC, dificultando a análise, visto que ela depende de figuras e é feita manualmente, incluindo os cálculos de ganhos em processamento e energia. Dessa forma, pode-se concluir que o cenário aqui apresentado limita a garantia dos resultados das otimizações somente para esse cenário, porém, não se anula que a abordagem proposta aqui pôde obter otimizações.

A. Grafo de Fluxo de Controle

Para cada um dos três tempos de simulação, três diferentes colorações foram aplicadas. Os percentuais de cinco, dez e vinte dos blocos mais executados do código foram coloridos de azul. Então, tem-se nove distintas simulações e GFCs, sendo que a diferença entre os que possuem a mesma quantidade de tempo de simulação é somente a coloração.

A tabela 3 mostra para cada tempo de simulação e para cada percentagem dos blocos mais executados (cinco, dez e vinte porcento), passada por parâmetro na funcionalidade do GFC, a percentagem do tempo de execução nesse percentual dos blocos mais executados sobre o tempo de execução total do código.

Tabela 3. Relação entre os blocos mais executados e tempo total de execução para o Antithief.

Tempo de Simulação(minutos)	5% dos Blocos	10% dos Blocos	20% dos Blocos
2 min	79,1%	89,5%	97,6%
6 min	79,2%	89,6%	97,6%
10 min	79,3%	89,6%	97,7%

A figura 1 mostra para o mesmo trecho do GFC, para duas colorações distintas, para o tempo de simulação de dez minutos, comprovando que há mais nós azuis quando é elevada a percentagem de coloração dos blocos mais executados. As três informações nos nós do grafo são: nome do bloco (endereço), número de ciclos de processamento do bloco e a quantidade de vezes que este foi executado, respectivamente.

Análise Crítica do GFC

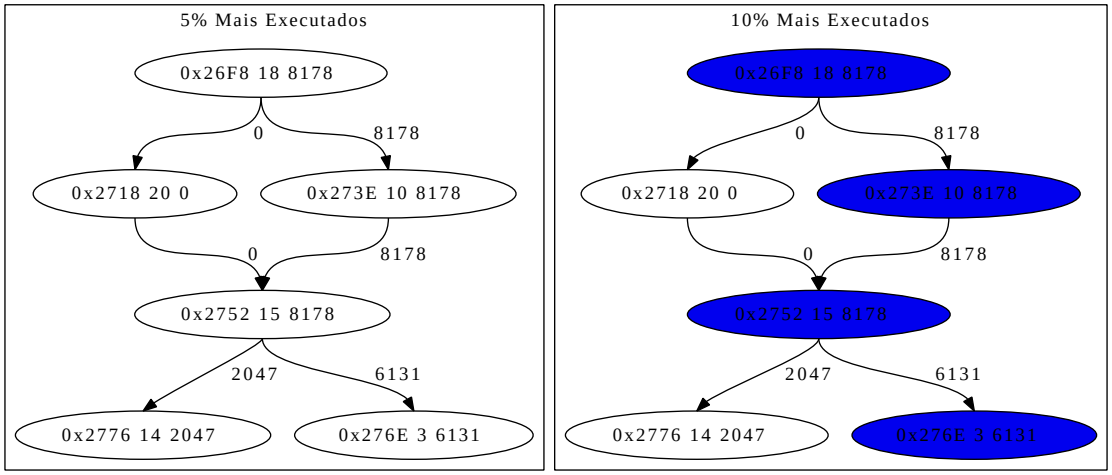


Figura 1. Grafos para um mesmo trecho de código da aplicação Antithief.

Assim como no caso do Blink, a etapa de análise crítica do GFC depende do conhecimento da estrutura do TinyOS, visto que as verificações de oportunidades de melhorias são aferidas de acordo com este conhecimento. Foi decidido aplicar a análise crítica somente em 10% dos nós mais executados do GFC, devido a análise crítica manual ser bastante trabalhosa, ao mesmo tempo em que o incremento da otimização para 20% difere em aproximadamente 8% a percentagem do tempo de execução da aplicação, segundo mostra a tabela 3. Também foi decidido somente o uso do tempo de execução de dez minutos, pois, apesar de os eventos ocorrerem sempre numa frequência constante nos nós 1 e 2, um maior tempo de simulação distribui de forma mais justa os tempos de execução pelos nós do GFC, visto os trechos do código de inicialização do nó sensor e da rede terão menor peso sobre o tempo total de execução.

Otimização do Consumo de Processamento

Nesse estudo, essa otimização teve por base as técnicas de otimização de desdobramento de *loops* e diretivas em linha. Tais técnicas diminuem o consumo de processamento, ao mesmo tempo em que incrementam o tamanho do código. Elas foram aplicadas nas funções que possuem blocos que estão inseridos nos 10% dos blocos de código mais executados durante a simulação. As otimizações foram realizadas em funções, porque não se poderia aplicar em apenas alguns blocos isolados das funções.

A tabela 4 apresenta algumas das funções do TinyOS no código executável que fazem parte dos blocos de código mais executados, e quais as otimizações aplicadas nelas (desdobramento de *loop*, diretivas em linha ou nenhuma). A terceira coluna apresenta o percentual da execução somente da função sobre a execução total de toda a aplicação, e a última coluna observa qual a redução de processamento na aplicação. A redução total foi de 6,28% do consumo de processamento efetuado pelas otimizações.

Tabela 4. Tabela das funções que contém blocos que estão entre os 10% dos blocos mais executados da aplicação Antithief.

Função	Otimizações	Fun./Apli.	Redução na Apli.
udivmodhi4	Loop/Em linha	20,60%	2,11%
udivmodsi4	Loop/Em linha	4,42%	0,77%
Atm128SpiP$SpiByte$write	Loop/Em linha	18,1%	2,28%
Atm128SpiP$sendNextPart	Em linha	20,6%	0,85%
Main	-	5,69%	-
SchedulerBasicP$TaskBasic$postTask	-	2,93%	-

B. Análise do Consumo de Energia

Através da quantidade total de ciclos reduzidos depois que todas as otimizações foram aplicadas no Antithief, conforme mostrado na tabela 5, calculou-se manualmente a energia consumida pelo processador após a otimização, retirando os ciclos do período ativo do processador e colocando-os no modo *idle*.

Tabela 5. Antithief - Reduções após efetuação das otimizações.

-	Redução após Otimizações
Estado Ativo da CPU	4102095 Ciclos (6,28%)
Energia CPU	0,00704949 Joules (1,14%)

4.3. Discussão dos Resultados

Embora a simulação do Antithief tenha ocorrido com apenas três nós integrando a rede, as funcionalidades de coleta e entrega, caracterizam a aplicação como típica de RSSF. Por ter um código bem maior que o Blink, o Antithief produziu um grafo muito mais extenso e complexo, de difícil detecção visual de otimizações, ao contrário do estudo de caso do Blink. Dessa forma, as otimizações se basearam nas técnicas de otimização de código utilizadas pelos compiladores, e foram aplicadas somente nos 10% dos blocos mais executados do grafo.

Observamos na tabela 5, que a redução do consumo de energia do processador foi somente de 1,14%, embora a redução do tempo de permanência no estado ativo ter sido de 6,28%. Isso ocorreu devido a aplicação ter sempre o rádio sempre em algum nível de atividade, impedindo assim, que o processador transite para um estado de menor ativação, pois, a possibilidade de uma interrupção do rádio necessita de um estado mínimo de ativação do processador. Em uma aplicação com coleta e entrega periódicas, onde o rádio é desligado regularmente, a energia do processador seria reduzida em maior escala.

5. Conclusão e Trabalhos Futuros

O desenvolvimento de *software* para RSSFs considera economia de recursos, seja pelas bibliotecas ou SO usado, ou seja, pelo compilador. Contudo, mostramos um método que após o código executável ser gerado, novas considerações da eficiência

da aplicação podem ser observadas e otimizações aplicadas através de análises do código, no TinyOS.

Aplicamos o método em dois estudos de casos, Blink e Antithief. Propõe-se como trabalhos futuros: a automatização do método, aplicar o método em um estudo de caso mais complexo, um estudo aprofudando das técnicas de otimização de código que diminuem o consumo de processamento enquanto aumentam o tamanho do código, e por fim, o uso do método para comparação dos SOs TinyOS e Contik.

Referências

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422.

Joe, H., Park, J., Lim, C., Woo, D., and Kim, H. (2008). Instruction-level power estimator for sensor networks. *ETRI Journal*, pages 47–58.

Klues, K., Handziski, V., Lu, C., Wolisz, A., Culler, D., Gay, D., and Levis, P. (2007). Integrating concurrency control and energy management in device drivers. In *SOSP '07: Proceedings of 21th ACM SIGOPS Symposium on Operating Systems Principles*, pages 251–264.

Lane, N. and Campbell, A. (2006). The influence of microprocessor instructions on the energy consumption of wireless sensor networks. In *EmNets 2006: The 3rd Workshop on Embedded Networked Sensors*.

Levis, P., Lee, N., Welsh, M., and Culler, D. (2003). Tossim: accurate and scalable simulation of entire tinyos applications. In *SenSys '03: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, pages 126–137.

Levis, P., Madden, S., Polastre, J., Szewczyk, R., Woo, A., Gay, D., Hill, J., Welsh, M., Brewer, E., and Culler, D. (2004). Tinyos: An operating system for sensor networks. In *Ambient Intelligence 2005*.

Paleologo, G. A., Benini, L., Bogliolo, A., and De Micheli, G. (1998). Policy optimization for dynamic power management. In *DAC '98: Proceedings of the 35th annual Design Automation Conference*, pages 182–187.

S. Park, A. S. and Srivastava, M. B. (2000). Sensorsim: a simulation framework for sensor networks. In *Proceedings of the 3rd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 104–111.

Sundresh, S., Kim, W., and Agha, G. (2004). Sens: A sensor, environment and network simulator. In *Proceedings of the 37th Annual Simulation Symposium*, pages 221–230.

Titzer, B., Lee, D. K., and Palsberg, J. (2005). Avrora: scalable sensor network simulation with precise timing. In *IPSN'05: The Fourth International Symposium on Information Processing in Sensor Networks*, pages 477–482.

Zhang, Z., Chan, W. K., Tse, T. H., Lu, H., and Mei, L. (2009). Resource prioritization of code optimization techniques for program synthesis of wireless sensor network applications. *Journal of Systems and Software*, 82(9):1376–1387.

Algoritmo de Navegação Robótica em Redes de Sensores sem Fio baseado no RSSI

Antonio R. de Carvalho-Jr[1], Afonso D. Ribas[1], Carlos M. S. Figueiredo[1]

[1]Núcleo de Pesquisa em Computação (NepComp)
Fundação Centro de Análise, Pesquisa e Inovação Tecnológica (FUCAPI)
Manaus – AM – Brasil

{antonio.ramos, afonso.degmar, mauricio.figueiredo}@fucapi.br

Abstract. *Most environmental monitoring applications require a wider observation or a simple interaction capability that Sensors Networks are unable to offer. Thus, many papers have addressed the interaction of these network with mobile robots. This article describes a navigation algorithm for a mobile robot in order to find a node sensor that sends a particular event. This solution is based on RSSI data obtained through the exchange of packet between a sensor node and the sink node, eliminating the requirement of a tracking system, which can be costly.*

Resumo. *Muitas aplicações de monitoramento ambiental exigem uma observação mais ampla ou uma capacidade de interação que simples Redes de Sensores são incapazes de oferecer. Nesse sentido, vários trabalhos da literatura têm abordado a interação dessas redes com robôs móveis. Este artigo descreve um algoritmo de navegação para um robô móvel de forma que o mesmo possa localizar um nó-sensor que emite um determinado evento. Esta solução baseia-se em dados de RSSI obtidos por meio da troca de pacotes entre um nó-sensor e o sink da rede, eliminando a exigência da existência de um sistema de localização, o que pode ser custoso.*

1. Introdução

A necessidade de monitoramento vem crescendo cada vez mais em nossa sociedade, seja em um ambiente doméstico, industrial ou ambiental. As pessoas necessitam obter informações do meio onde estão vivendo ou ambientes remotos que não são convencionais.

Particularmente, nas aplicações de monitoramento remoto, duas tecnologias bastante estudadas com grande potencial de aplicação prática são as redes de sensores sem fio, e.g. Loureiro et al. (2003), e os robôs móveis autônomos, e.g. Trincavelli et al. (2008). As RSSFs permitem a coleta de dados sensoriais a partir de pequenos dispositivos no ambiente. Tais dispositivos utilizam-se da comunicação sem fio para a coleta de dados e o desafio é a operação com economia de bateria. Já os robôs autônomos podem ter maior capacidade de monitoramento com sensores mais complexos, como câmeras, e o desafio é fazer com que o robô se locomova corretamente no ambiente.

Um problema explorado nas RSSFs é o da localização de nós, e.g. Boukerche et al (2007). Tal sistema de localização pode ser obtido de um sistema global de posicionamento ou a partir de algoritmos distribuídos. Seu objetivo é poder associar a

localização aos dados sendo monitorados, permitindo também a uma entidade externa interagir mais rapidamente com o local detectando algum evento de interesse. Por outro lado, localizar um nó sensor em uma rede de sensores sem fio não é uma tarefa trivial, pois a necessidade de equipamentos e/ou comunicação adicionais agregará aos nós sensores mais complexidade, consumo de energia e custo.

Neste artigo, apresentamos um algoritmo de navegação robótica para a localização de um determinado nó sensor gerando um evento de interesse a partir do RSSI (*Received Signal Strength Indicator),* indicador de nível de sinal recebido, como referência. Tal sistema não exige que a posição dos nós sensores seja previamente conhecida ou que haja a necessidade de um sistema de localização, simplificando a RSSF e possibilitando um incremento de observação e/ou interação a partir do robô. No entanto, há a necessidade de levar em consideração a real caracterização da propagação do sinal no ambiente em questão, o que também é mostrado e avaliado neste trabalho.

O conteúdo está organizado da seguinte forma: a seção 2 apresenta a fundamentação teórica e trabalhos relacionados; a seção 3 apresenta a solução proposta para o sistema de navegação; a seção 4 apresenta os testes e avaliações; por fim, a seção 5 apresenta as conclusões e trabalhos futuros.

2. Fundamentação Teórica e Trabalhos Relacionados

2.1. Redes de sensores sem fio

Segundo Loureiro et al (2003), redes de sensores sem fio (RSSFs) diferem de redes de computadores tradicionais em vários aspectos. Normalmente essas redes possuem um grande número de sensores distribuídos, têm restrições de energia, e devem possuir mecanismos para auto-configuração e adaptação devido a problemas como falhas de comunicação e perda de sensores. Ainda, uma RSSF tende a ser autômata e requer um alto grau de cooperação para executar as tarefas definidas para a rede. Isto significa que algoritmos distribuídos tradicionais devem ser revistos para esse tipo de ambiente antes de serem usados diretamente.

Sob a necessidade de operação em larga escala, em áreas remotas, de forma autônoma e por longos períodos de tempo, as RSSFs apresentam grandes desafios quanto ao consumo de energia e custo por dispositivo. Assim, observa-se que muitas aplicações têm empregado nós sensores simples, e.g. Arampatzis et al (2005), limitando a capacidade de observação dos mesmos.

2.2. Robótica Móvel

Segundo Marchi (2001), um robô móvel é um dispositivo mecânico montado sobre uma base não fixa, que age sob o controle de um sistema computacional, equipados com sensores e atuadores que o permitem interagir com o ambiente.

A navegação é um processo usado por um robô móvel para movimentar-se desde uma posição inicial a uma posição final tendo como base alguns pontos de referência , e.g. Ribeiro et al. (2002). A navegação robótica em um ambiente real é uma tarefa complexa, pois além do problema de locomoção com os obstáculos do mundo real, outros problemas são a determinação do alvo e o planejamento da trajetória. Segundo Ribeiro et al. (2002) as metodologias desenvolvidas para o planejamento de trajetória podem ser classificadas em dois grupos: métodos globais e métodos locais. Os métodos globais dão a habilidade ao robô para se localizar em termos absolutos (GPS – *Global*

Positioning System) ou em relação a um sistema de mapas, e mover-se para um ponto desejado. Os métodos locais dão a capacidade ao robô de determinar sua posição relativa a objetos próximos (estacionários ou não) e interagir com estes de forma correta.

2.3. Interação entre robôs e RSSFs

Robôs, em sua maioria, auxiliam os homens no monitoramento de área de difícil acesso. Dada a importância e utilização das RSSFs para o monitoramento também para este fim, cria-se um potencial muito grande na junção dessas duas áreas. Um robô pode incrementar a capacidade de monitoramento das RSSFs, bem como pode se utilizar delas para executar suas atividades.

Uma forma de interação entre RSSFs e robôs móveis consiste do auxílio da cobertura de áreas de uma RSSFs, o que é demonstrado em Batalin et al. (2002). Neste trabalho, um nó sensor em um robô móvel busca áreas ainda não cobertas pela rede e se posiciona para melhorar ou amplificar a área de monitoramento daquele ambiente.

Outro trabalho aborda o uso de RSSFs e robôs em aplicações de resgate em incêndios, e.g. Kantor et al. (2003). Sensores coletando dados de temperatura podem fornecer um gradiente de navegação aos robôs com o objetivo de auxiliar seres humanos a encontrarem uma saída segura.

Bulusu et al. (2002) e Corke et al. (2005) são trabalhos que visam a navegação robótica assistida por uma RSSF, onde o objetivo dos robôs é atingir um determinado nó da rede. Ambos dependem de um sistema de localização na RSSF. Em Bulusu et al. (2002), os nós têm suas localizações previamente definidas e o robô usa um sistema GPS para atingir o ponto determinado. Já Corke et al. (2005) assume que os nós sensores são dotados de GPS, onde o robô pode pode seguir o caminho sugerido pela rede e interagir com a mesma para futuros eventos.

A solução aqui proposta encontra-se exatamente nesse último contexto, onde robôs devem atingir os nós emitindo eventos. No entanto, este se diferencia dos demais ao desconsiderar totalmente o conhecimento prévio da localização dos nós ou da execução de um sistema de localização. O mesmo se baseia somente na troca de dados entre nós, uma atividade fundamental das RSSFs.

3. Sistema de navegação baseado em RSSI

3.1. Visão geral

Em uma RSSF, nós sensores coletam dados em os transmitem sem fio em um esquema *multihop* a um destino, comumente chamado nó *sink* ou *gateway*. Modelos de propagação nos mostram que há uma relação da distância entre transmissor e o receptor e o nível de sinal recebido por este último, e.g. Rappaport (2001). Portanto, um robô pode usar esse princípio para orientar sua navegação em direção a um nó transmitindo um sinal de interesse pela rede, numa técnica conhecida como *sniffing*. O princípio básico de navegação é explorar uma área de interesse medindo a intensidade do sinal recebido e caminhar na direção onde o sinal recebido é mais forte até um certo limite onde considera-se que o alvo foi atingido.

No entanto, experimentos práticos, como em Figueiredo et al. (2009), têm mostrado que a característica de propagação do sinal torna a estimativa da distância pelo RSSI muito imprecisa e extremamente dependente do ambiente. Tal fato exige que o

algoritmo proposto leve em consideração tais particularidades, o que é apresentado na subseção a seguir.

3.2. Caracterização do RSSI

Antes de apresentar o algoritmo de navegação foram realizados alguns experimentos para a identificação do padrão de propagação do sinal de um sensor em alguns ambientes. O ambiente escolhido foi o de floresta tropical, visando que o sistema em questão possa contribuir para o monitoramento ambiental da amazônia.

Foram realizados experimentos utilizando sensores Iris da Crossbow em duas situações comuns (ver figura 1): em mata fechada, com nós no solo, e trilha, com nós a 1,20 m do solo. Para caracterização da variação do RSSI, posicionou-se 2 nós, um transmissor e um receptor, variando-se a distância entre eles. Para cada distância foram feitas 33 etapas de medição sendo que cada etapa consistia do envio de 20 pacotes com dados de temperatura, pressão, umidade e RSSI para o receptor. Em seguida foi obtida a média do RSSI para cada distância, bem como o seu desvio padrão, como pode ser observado nos Gráficos 1 e 2.



Figura 1 – Experimentos em trilha e mata fechada

Observando os dados do Gráfico 1 e 2, podemos perceber que existe uma área de confiança na qual conseguimos identificar se estamos nos aproximando ou não do nó sensor de destino. É o caso de distâncias até 10 m na trilha e até 3.8 m em mata fechada. Porém, existe uma área na qual o RSSI permanece praticamente constante, em valores menores que -91 dB, onde há recepção por uma longa distância mas não há variação do RSSI. Isso deve-se ao fato particular do nó sensor possuir uma limitação da percepção do RSSI e de que há constantes retransmissões na zona de sinal fraco, fazendo a recepção se prolongar a distância maiores mesmo em baixos níveis de sinal.

Esse resultado ocasiona uma área de incerteza na estimativa da distância que representa 97.2% da área da circunferência no experimento na trilha, e 92.7% da área da circunferência no experimento da mata fechada. Tal característica deve ser levada em consideração na solução de navegação e dá uma dimensão do real problema a ser enfrentado na prática.

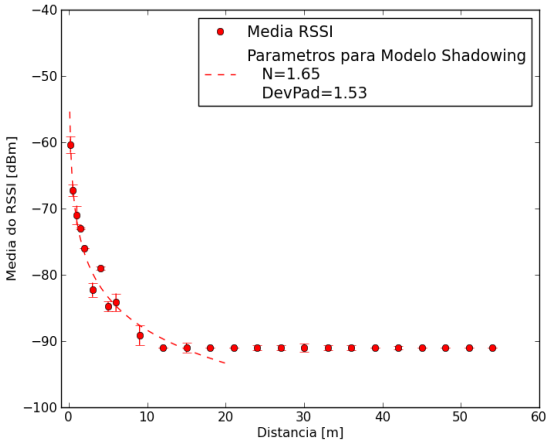


Gráfico 1 – Distância versus RSSI para o experimento da Trilha

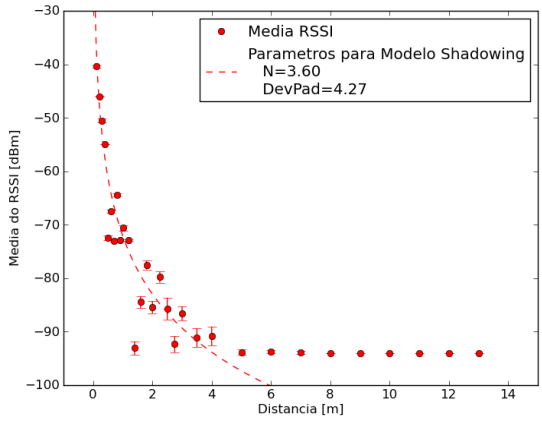


Gráfico 2 – Distância versus RSSI para o experimento da Mata Fechada

3.3. Algoritmo de Navegação

Com as características observadas no experimento notamos que há duas áreas que devem ser consideradas para a navegação até um nó sensor. Uma área de incerteza na qual um robô, tendo como informação apenas o RSSI, não saberá se está aproximando-se ou afastando-se de seu destino. A outra área é a que o robô consegue identificar se está se distanciando ou aproximando do nó (Figura 2a).

O algoritmo que descreve a solução de navegação é apresentado na Figura 2. O mesmo baseia-se na recepção de um pacote de dados de interesse onde extrai-se o RSSI. Com base no RSSI, verifica-se se o mesmo está na área de incerteza (RSSI < -91 dB) ou não.

Caso esteja na área de incerteza, o robô irá navegar em uma única direção até que perca o sinal ou chegue na área de confiança (Figura 2b). Se sair da área de sinal, o robô marca aquele ponto como um ponto inicial na “borda” da área coberta pelo sinal. Após isso, ele realizará uma série de passos (Figura 2c) para encontrar um segundo ponto de borda. Basicamente, são realizadas sucessivas conversões de 90 graus à esquerda, que independente do ângulo de saída fará com que o robô volte para a área de cobertura do sinal. Com dois pontos de borda, o robô poderá calcular uma secante na circunferência. De acordo com as propriedades da secante, se uma reta **s**, secante a uma

circunferência de centro **O**, intercepta a circunferência em dois pontos distintos **A** e **B** e se **M** é o ponto médio da corda **AB**, então o segmento de reta **OM** é perpendicular à reta secante **s**. Dessa forma, após encontrarmos os dois pontos na circunferência, basta o robô navegar perpendicularmente a partir do ponto médio do segmento de reta entre os pontos, que o mesmo irá em direção ao centro da área do sinal, ou seja, para o transmissor do sinal. Ao caminhar na direção do centro, o robô encontrará a área de confiança do RSSI.

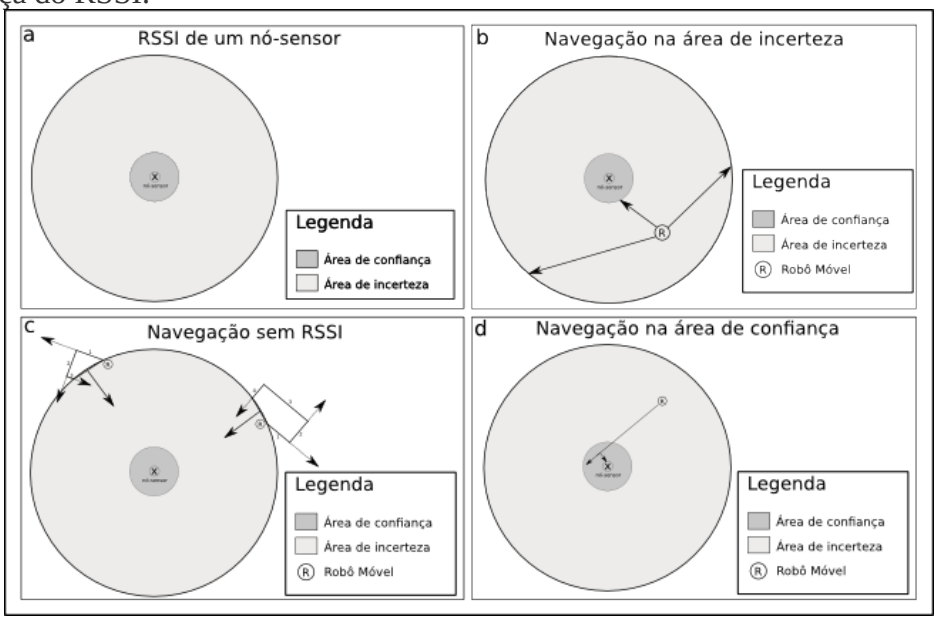


Figura 2 – a. Representação gráfica da propagação de sinal no experimento; b. Partida do robô em uma direção aleatória; c. Navegação do robô quando perde o sinal; d. Navegação do robô em direção ao centro ou quando encontra-se na área de confiança.

Uma vez estrando na área de confiança, ou já partindo desta área, o robô permanece andando na mesma direção (ângulo) enquanto o sinal do RSSI estiver aumentando. Ao detectar uma diminuição do sinal, o robô rotaciona 90º e continua navegando. Caso o sinal volte a diminuir, seguimos em direção contrária (giro de 180 graus) e caso ocorra novamente, voltamos na direção contrária repetindo os passos acima até estarmos em uma distância (ou por vezes em uma determinada faixa de RSSI estipulada) de confiança para afirmarmos que o robô móvel chegou ou está próximo ao seu destino.

```
funcao expiraTemporizador( )
 regiao = SEM_COMUNICACAO
 escolha( passos )
  caso 0:
   salvarPosicaoRelativaInicial( roboMovel.X(), roboMovel.Y() );
   avanca( DIST_SEM_COM )
  caso 1:
   girar_avancar( 90, DIST_SEM_COM )
  caso 2:
   girar_avancar( 90, 2*DIST_SEM_COM )
  caso 3:
   girar_avancar( 90, DIST_SEM_COM )
 fim_escolha
 passos = passos + 1
fim_funcao
```

```
funcao receberPacote(...)
 se ( (mediaRssiAtual = -91 e mediaRssiAnterior = -91) então
  escolha( regiao )
   caso CEGA:
    avancar( DIST_CEGA )
    passos=0
   caso PROXIMA:
    girar_avancar( 180, DIST_PROX )
   caso SEM_COMUNICACAO:
    salvarPosicaoRelativaFinal( roboMovel.X(), roboMovel.Y() )
    ponto_medio = calcularPontoMedio( pegarPosRelInicial,
                        pegarPosRelFinal )
    direcao = calcularDirecao( pegarPosoRelInicial,
                        pegarPosRelFinal )
    irPara( ponto_medio )
    avancarDirecao(DIST_CEGO, direcao)
  fim_escolha
  regiao = CEGA
 senao
   se ( (mediaRssiAtual - mediaRssiAnterior) < LIMIAR_RSSI) então
    girar_avancar( 90, DIST_PROX )
   senao
    avancar(DIST_PROX)
   fim_se
   regiao = próxima
 fim_se
 reinicia_temporizador(INTERVALO)
fim_funcao
```

Figura 3 – Pseudo-algoritmo para Navegação Robótica baseada no RSSI.

4. Testes e Avaliação

Após a análise dos ambientes nos quais iremos aplicar nosso algoritmo de navegação, realizamos alguns experimentos com o algoritmo descrito na Figura 3 para os dois ambientes (trilha e mata fechada) utilizando o NS-2 (Network Simulator) [Fall 2010].

Para simularmos o ambiente real do experimento, definimos os parâmetros para o modelo *Shadowing* (Gráfico 1 e Gráfico 2) para que pudéssemos obter o mesmo comportamento do RSSI obtido nos experimentos. Além dos ajustes, adaptamos o código do NS-2 de modo que tenhamos o mesmo comportamentos desejado para realizarmos a navegação robótica segundo o algoritmo proposto. As simulações utilizaram um nó fixo e um nó móvel (que foi utilizado como o robô em nosso experimento).

Utilizamos como métrica de avaliação a variação da distância percorrida pelo nó móvel até chegar em seu destino (d_f) pela distância inicial entre o nó fixo e o nó móvel (d_i).

Para cada distância (d_i) foram feitas 40 simulações, sendo cada uma com orientação inicial diferentes e, sem seguida, calculada a distância percorrida pelo nó móvel (d_f). Em seguida, calculamos a média dos d_f para cada d_i.

4.1. Resultados

Como já foi dito, utilizamos na simulação o modelo de propagação observado nos experimentos. Desta forma, possíveis interferências no sinal são levadas em consideração, fazendo com que o alcance máximo não seja determinado por uma circunferência perfeita. No entanto, o que foi observado é que para as simulações realizadas, essa interferência não modificou substancialmente os resultado, pois o robô móvel sempre encontra a área de confiança.

Os Gráficos 3 e 4 apresentam os resultados obtidos com as simulações feitas para cada ambiente no NS-2. Em todos as simulações, o robô móvel conseguiu chegar ao seu destino final.

Além disso, pode-se observar que nos pontos em que o robô parte da área de confiança, em média, ele sempre percorre uma distância relativamente curta em relação a média da distância percorrida quando o mesmo parte da área de incerteza.

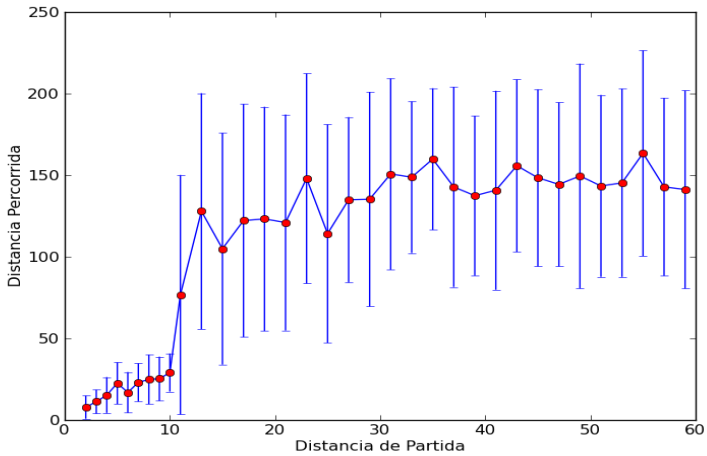


Gráfico 3 – Distância percorrida versus Distância de partida para o modelo da Trilha.

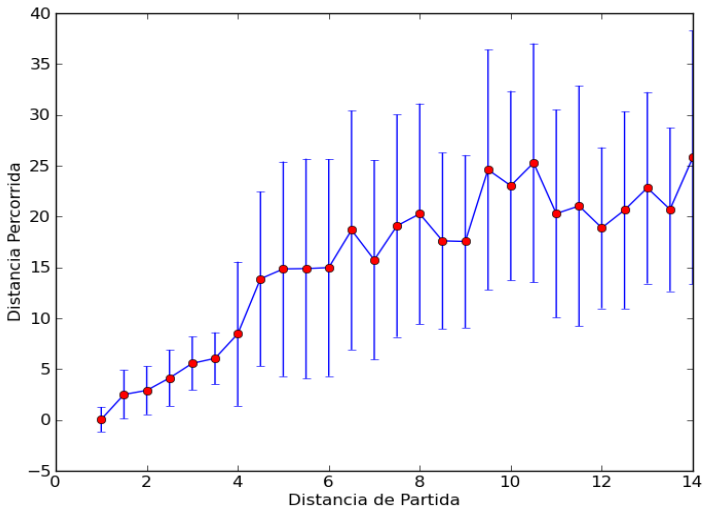


Gráfico 4 – Distância percorrida versus Distância de partida para o modelo da Mata Fechada.

5. Conclusão e Trabalhos Futuros

Este trabalho abordou o problema da navegação robótica assistida por Redes de Sensores sem Fio. Vislumbrando um cenário de monitoramento complexo, o sistema permite que robôs autônomos possam navegar até aos nós gerando eventos de interesse baseados apenas no RSSI do sinal transmitido por esses. A solução desenvolvida mostrou a real necessidade de conhecer as características da propagação do sinal no ambiente de interesse, a serem levadas em consideração pelo robô. A grande vantagem da solução é permitir a navegação em um ambiente sem conhecimento prévio e sem um sistema de localização em operação, o que simplifica os requisitos do sistema como um todo.

As simulações realizadas mostram um alto impacto das zonas de incerteza da cobertura do sinal transmitido, obrigando um robô autônomo a percorrer distâncias maiores. No entanto, mostrou-se que mesmo sob tais condições a tarefa pode ser bem sucedida.

Como trabalhos futuros, pretendemos avaliar a navegação robótica através de vários nós roteadores, percorrendo o caminho sink-transmissor em vários saltos. Ainda, pretende-se avaliar o emprego de antenas direcionais, que tendem a permitir uma diminuição das manobras exploratórias em áreas de incerteza.

Referências

Arampatzis, T. et al., 2005. A survey of applications of wireless sensors and wireless sensor networks," in Proc. of the Mediterranean Control Conference (Med05), Limassol Cyprus, 2005.

Boukerche, A. ; Oliveira, H. A. B. F. ; Nakamura, E. F. ; Loureiro, A. A. F.. Localization Systems for Wireless Sensor Networks. IEEE Wireless Communications, v. 14, p. 6-12, 2007.

Bulusu N., Heidemann J., Estrin D. "Adaptive beacom sensor network with fading and mobility". In Proceedings of IEEE PIMRC. Portugal, Lisboa. 2002.

Corke Peter, Peterson Ron, RUS Daniela. "Network Robots: Flying robot navigation using a sensor net". Robot Research, Springfield. April 2003.

Fall K., Varadhan K., "The ns Manual", Maio 2010.

Figueiredo, C. M. S. ; Nakamura, E. F. ; Ribas, A. D. ; Souza, T. R. B. ; Barreto, R. S. . Assessing the Communication Performance of Wireless Sensor Networks in Rainforests. In: IFIP Wireless Days 2009, 2009, Paris. Proceedings of the 2nd IFIP Wireless Days, 2009.

Kantor, G., Singh, S., Peterson, R., Rus, D., Das, A., Kumar, V., Pereira, G., Spletzer, J., "Distributed search and rescue with robot and sensor teams". The 4th International Conference on Field and Service Robotics, July 2003.

Loureiro, A.; Nogueira, J. M.; Ruiz, L. B. R.; Mini, R. ª F.; Nakamura, E. F.; Figueiredo, C. M. S., "Redes de sensores sem fio". 21º Simpósio Brasileiro de Redes de Computadores, 2003 (Minicurso).

Marchi, J. "Navegação de robôs móveis autônomos: Estudo e implementação de abordagens". Universidade Federal de Santa Catarina. Março de 2002.

Rappaport T. S., Wireless Communications: Principles and Practice, 2nd ed., Prentice Hall, 2001, pp. 123.

Ribeiro, M. I., Lima, P. “Motion Planning – Mobile robotics course”. Instituto Superior Técnico, Lisboa, Portugal. 2002.

Trincavelli, M., Reggente, M., Coradeschi, S., Loutf, A., Ishida, H., Lilienthal, A. J. “Towards environmental monitoring with mobile robots”. 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems Acropolis Convention Center. Nice, France, Sept, 22-26, 2008.

Supporting Context Events in an Interactive TV Platform*

Izon Thomaz Mielke[1], Patrícia Dockhorn Costa[1], Igor Magri Vale[1]

[1]Departamento de Informática – Universidade Federal do Espírito Santo (UFES)
Av. Fernando Ferrari 514 – Vitória – ES – Brasil

{izomtm, igormagrivale}@gmail.com, pdcosta@inf.ufes.br

Abstract. *This paper aims at investigating functionality of the middleware Ginga that can be leveraged to build context-aware interactive TV applications. We identify the elements of the middleware that can effectively be used to realize each of the particular aspects of context-awareness, namely capturing, processing and reacting upon context changes. We present a prototype application in order to illustrate how these elements have been identified and how they can be used. We also define a programming framework that facilitates capturing and distributing context events in this interactive TV platform.*

1. Introduction

Context-aware applications use and manipulate context information to detect the situations of their users and adapt application behavior accordingly. Since context-awareness allows applications and services to become tailored to the user's current situation and needs, several areas of research consider context-awareness as an important tool to enhance user experience [Dey 2001].

In the interactive television domain, in particular, context information has been used in various types of applications [da Silva et al. 2009][Thawani et al. 2004], ranging from TV content adaptation (based on user's preferences) to home banking applications (selectively displaying certain private information based on presence). In a digital television application, for example, context information can be used to detect presence of children in the living room, which can be used to change the behavior of the TV to only display adequate ("child safe") content.

However, the development of context-aware applications is a challenging task. Context-aware applications, in general, should be capable of: (i) sensing context from the environment, (ii) observing, collecting and composing context information from various sources (often sensors), (iii) autonomously detecting relevant changes in the context, and (iv) reacting to these changes, by either adapting their behavior or by invoking (composition of) services.

Since context-awareness is a relatively novel technology in the scope of the Brazilian Interactive TV Platform (SBTVD), it is still not clear whether the support offered by the platform is adequate for the development of context-aware applications. In this sense, this work investigates the functionality of the platform that can be leveraged

* This work is partially sponsored by the project HR-TVD # 225/2008 funded by CAPES. Igor M. Vale and Izon T. Mielke are supported by CAPES and FAPES, respectively.

to build context-aware interactive TV applications. We identify in this paper the elements of the platform that can effectively be used to realize each of the particular aspects of context-awareness, namely *capturing*, *processing* and *reacting* upon context changes. In particular, in order to support *capturing context events*, we have leveraged the general event framework provided by the SBTVD with a programming framework that facilitates capturing and distributing *context events*. We further present a prototype application in order to illustrate the development of a context-aware application using the framework we have proposed.

The paper is further structured as follows: Section 2 introduces a context-aware application scenario; Section 3 presents the Brazilian Digital TV System (SBTVD); Section 4 discusses how context can be captured and handled in a SBTVD application; Section 5 proposes a programming framework that facilitates context-aware development in the platform; Section 6 presents the context-aware application prototype implementation; Section 7 discusses this work on the light of related work and, finally, Section 8 presents some concluding remarks.

2. Context-Aware Application Scenario

We have used throughout the paper a running example in order to illustrate how context-aware requirements can be fulfilled by elements of the Ginga platform. We have chosen a representative application scenario that covers the most important requirements of a context-aware application, namely capturing, processing and reacting upon context information changes. The scenario storyline is as follows:

In exhibition events, it is common to find TV sets showing demonstrative videos of some product. These types of demonstrations can present a problem: the lack of audience. This occurs mainly because the TV only shows a video in loop, which rarely sparks curiosity to get people's attention to stop and watch the video about the product. In a tentative to solve this problem, we propose a context-aware application that utilizes information about audience (obtained by means of a sensor that detects presence) in order to control the video contents exhibited on the TV screen. While there is no sufficient public watching the video, an initial short clip is shown in order to spark public's attention and to attract other participants:

"Certain things cannot be done alone. Watching this video is one of them! Invite your friends and find out why."

As soon as the required audience is achieved, the product's presentation is initiated with the guarantee that a minimum number of persons is watching the video.

3. The Brazilian Digital TV System (SBTVD)

The Brazilian Digital TV System (SBTVD) implements a middleware platform, coined Ginga [Telemídia 2007], which supports the development and execution of interactive applications in the SBTVD environment. Interactive applications can be developed upon the Ginga middleware using a declarative language, called NCL, or using Java. Supporting Java application development in Ginga is optional for portable devices. Given the mobile nature of context-aware applications, we have chosen to concentrate our efforts on investigating the particularities of the declarative environment of the middleware (NCL in combination with NCLua scripts).

NCL (Nested Context Language) is a hypermedia authoring language that was originally developed to describe multimedia applications with space-time synchronization between media objects (e.g., video, audio, images, etc.). As such, it does not have built-in concepts for context-handling and reactivity. In order to implement these concepts in NCL, we have identified a number of NCL document elements, which together with certain NCLua scripts provide the required support. Next sections discuss each of these elements.

4. Handling Context Events in the Ginga Platform

In order to obtain context information from a sensor, an NCL application should be able to access a remote device through a network. For that, the Ginga middleware offers a mechanism called Interactivity Channel. The exchange of data can be performed on the Interactivity Channel by means of NCLua scripts, which are part of the NCL application. These scripts are responsible for managing connections, exchanging and processing data and keeping the NCL document up-to-date with respect to context information received from remote devices.

The communication among NCLua scripts and other components is realized by means of events. The NCLua mechanism for event dissemination can be compared to the publish/subscribe pattern [Eugster et al. 2003] in which there is a mediator managing the exchange of events between producers and consumers. This type of event architecture allows space-time decoupling among system parts.

In order to realize exchange of events, NCLua scripts use a module (mediator) called *event module* [ABNT 2007]. Main functions of this module are generating an event, by means of a *post* method, and receiving an event, by means of registering event handlers using the *register* method. NCLua events are categorized by classes [Sant'Anna et al. 2008], each responsible for specific functions, such as: class "ncl", responsible for communicating with the NCL document, and the class "tcp", responsible for communicating with the Interactivity Channel.

Since we would like to gather context from a remote device and we also would like to keep the NCL document up-to-date with respect to context information, we are particularly interested in event classes "tcp" and "ncl", respectively. Following sections elaborate on how to use the standard event API to implement a simple NCL context-aware application.

4.1. Communicating with External Devices

As mentioned in previous sections, the communication between NCLua scripts and the Interactivity Channel is realized by means of events (class "tcp"). As in TCP Sockets, the exchange of data occurs after a connection has been established between a client and a server.

In order to request for a connection, an NCLua script should send out an event of type "connect", informing the server (host) and the port with which to connect to. As an answer to this event, the event API generates an event of type "connect" containing the identification of the connection, or an error event, indicating the reason why this connection could not be established.

After the connection has been established, data can be exchanged by means of events of type "data". In order to send out data events, the client should inform the identification of the connection. Similarly, events received contain the identification of the connection. An event of type "disconnect" indicates that the connection has been closed.

Figure 1 depicts an NCLua code snippet that shows how an NCL application should communicate with an external sensor (e.g., the Audience Sensor) through the Interactivity Channel, using the standard API. In this code snippet, a "connection" object is used to manage the connection with the Audience Sensor. An event handler object is created and registered with the event module. This handler object is responsible for answering to connection events, as well as data events.

As can be seen in this code snippet, using the standard API for handling context events (from external sources) can get quite cumbersome, since it is necessary to identify the class of the event, the type of the event, to which connection that event belongs to, etc. This is reflected by the long and confusing conditional ("if") statements. In this particular simple example, we are manipulating a single type of context event. When various context events should be supported (more realistic scenario), the number of conditional statements increases proportionally to the number of context types.

```
local connection = nil
function handler (evt)
 if evt.class == 'tcp' and evt.type == 'connect' and
  evt.host = host and evt.port  = port and connection == nil then
    connection = evt.connection
    if connection then
       event.post  {class  =  'tcp',  type  =  'data',  connection  =
connection, value = 'request_audience'}
    end
 end
 if evt.class == 'tcp' and evt.type == 'data' and
  evt.connection == connection then
    verify_audience(event.value)
 end
end
event.register(handler)
event.post {class = 'tcp', type  = 'connect', host = host, port  =
port}
```

Figure 1. NCLua code for communicating with an external device using the standard API.

4.2. Context Representation in NCL

Besides capturing context from remote sensors, NCLua scripts must be able to inform the NCL document that new data has arrived. The communication between NCLua scripts and the NCL documents, in which these scripts are embedded, is realized by means of Property Anchors. Property Anchors can be defined as shared attributes between the NCL and NCLua environments and can be used to represent context information in an NCL document.

In order to change Property Anchor values, an NCLua script should send out an event of class "ncl" (of type "attribution") informing which property to be changed and the new value to be assigned. The assignment of Property Anchor values is a two phase activity: first, a start event should be generated in order to initiate the assignment; and then, a stop event should be generated, which ends the assignment.

Figure 2 depicts a code snippet showing how to assign a Property Anchor value, using the standard API. The function verify-audience is invoked every time audience events arrive from the Audience Sensor. If audience is sufficient (defined as "value>10"), the respective Property Anchor in the NCL document should be informed. Again, the developer must manipulate events and their attributes explicitly and can get cumbersome when various types of context events are supported.

```
function verify_audience(value)
 if tonumber(value) > 10 then
  evt = {class = 'ncl', type= 'attribution', property = 'audience',
action = 'start', value = 'sufficient'}
  event.post(evt)
  evt.action = 'stop'
  event.post(evt)
 end
end
```

Figure 2. NCLua code for controlling context variables.

5. Programming Framework

Handling events for both, controlling connections and managing the mechanisms to keep Property Anchors up-to-date, are basic requirements for most NCL context-aware applications. We have defined and implemented a programming framework that can be used to ease the development process by solving many of the issues related to event handling in the Ginga middleware.

As part of the framework, we have developed the following NCLua libraries: TCPEventHandler and Properties, both available online at [http://code.google.com/p/itveventframework/]. The TCPEventHandler library provides reusable code for facilitating the communication between NCL applications and external devices, while the Properties library provides code to help managing Property Anchor values in NCLua scripts. Given the asynchronous nature of context-aware applications, we have based the development of the framework using classical design patterns for handling asynchronous events, as discussed in the next sections.

5.1. TCPEventHandler: Communicating with External Devices

The TCPEventHandler library adopts concepts of object-oriented design in order to manage connections and their data, since the standard API does not offer this facility. The TCPEventHandler library encapsulates functions to handle connections, errors and data exchange so that the developer does not need to worry about details of the standard mechanism.

In order to open a connection with a remote device through the Interactivity Channel, an NCL application developer using the TCPEventHandler library should make

use of a Connection object, which is offered by the library. As opposed to explicitly sending/receiving events, this object abstracts from the concepts of NCLua events, by defining methods to open/close a connection and exchange data. The Connection object uses a specific event handler object (Handler) to react upon notifications of connection success, disconnection, data exchange and communication errors.

The design of this library was based on the Reactor Pattern (or Dispatcher) [Schmidt 1994]. Internally, the library manages all current connections by means of an event handler method registered with the NCLua event channel, which handles all events disseminated in the Interactivity Channel (of class "tcp"). Upon receiving an event, the library handler method identifies the Connection object to which this event belongs to and invokes (on the event handler specific of that connection), the method responsible for handling that event.

In order to create a connection to a remote device using the TCPEventHandler library, the NCL application should instantiate an object of type Connection in an NCLua script informing, as arguments, the host address and port of that remote device, and an event handler specific for managing the events of that connection.

5.2. Properties: Communicating with the NCL Document

There is no direct correspondence between Property Anchors of an NCL document and variables defined in NCLua scripts. We have implemented the Properties library, which is capable of managing context variables by means of methods *set*, *get*, *register* and *unregister*. For example, when an NCLua script needs to inform the NCL document that a particular context value needs to be refreshed, the script invokes the set method passing as arguments the name of the Property Anchor and new value. Similarly, to get a property value from the NCL document to the script, the Properties library offers the get method.

6. Prototype Implementation

We have implemented a prototype for the scenario proposed in Section 2 using the programming framework presented in Section 5. Prototype code and execution examples are available at [http://code.google.com/p/itveventframework/].

In the implementation, context information about audience is offered by a remote sensor (camera), which should be able to detect presence of persons watching TV. Audience information is constantly monitored to control the exhibition of the video contents: if the audience is sufficient, the TV should display a Product Video; otherwise, the TV should display a Home Video. To implement face recognition, we have used an OpenCV library, which provides an algorithm for object detection [Viola and Jones 2001].

Figure 3 depicts a high-level architecture of the prototype. The NCL Application communicates with the Audience Sensor by means of the TCPEventHandler library. Internally, the NCL Application is composed by an NCL document (declarative part) and NCLua Scripts. The communication between those two is realized by means of the Properties library.

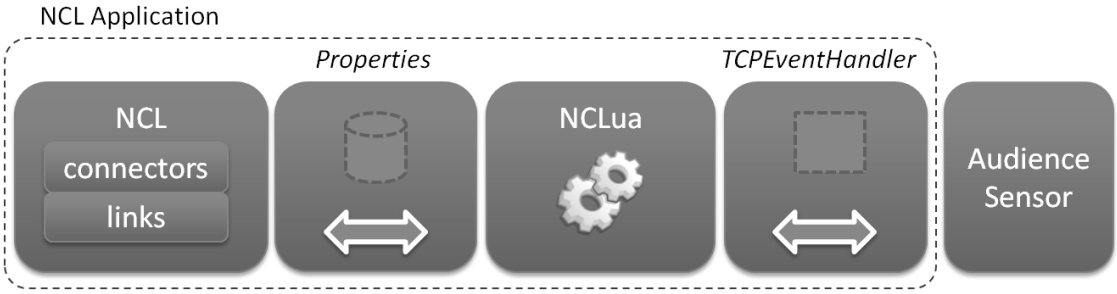


Figure 3. High-level architecture of the prototype.

Figure 4 depicts the content node (media element) that represents the NCLua script ("monitor.lua") in the NCL document, which defines a single Property Anchor (called "audience").

```
<media id="monitor" src="monitor.lua"/>
   <property name="audience"/>
</media>
```

Figure 4. Representation of the NCLua script in the NCL document.

Figure 5 shows the "monitor.lua" script, which loads the "TCPEventHandler" and "Properties" libraries, builds an event handler object (handler) and associates this object to a connection object (Connection), which represents the connection with the Audience Sensor.

The event handler object handles events by means of methods of this object. For example, when the TCPEventHandler library receives a data event from the event channel, it first verifies to each connection this event belongs to and then dispatches the event to proper consumers by invoking the *handle_receive* method of that particular handler. In our example, the *handle_receive* method invokes the *verify_audience* method, which verifies whether the audience has achieved a certain value. The method *verify_audience* should be invoked every time audience information arrives from the Audience Sensor.

Comparing Figures 1 and 5 (part 2), we can see that the code produced using the library (Figure 5) does not need to be overloaded with conditional statements, since most of these statements are shifted to the library. Therefore, the code is cleaner, easier for the object-oriented developer to write and understand, and it is also easier to extend with other types of context events.

When the audience information is sufficient, the *verify_audience* method invokes the set method (of Properties library) in order to indicate to the NCL document that the "audience" Property Anchor value should be now "sufficient'.

Again, comparing Figures 2 and 5 (part 1), we can see that great part of the code is shifted to the library, which makes the code cleaner and easier to understand.

```
require 'TCPEventHandler'
require 'Properties'

function verify_audience(value)                          (1)
    if tonumber(value) > 10 then
        Properties.set('audience', 'sufficient')
    end
end

handler = {}                                             (2)
function handler.handle_connect (self)
    self.connection.send('request_audience')
end
function handler.handle_receive (self, data)
    verify_audience(tonumber(data))
end
audience_connection =
  TCPEventHandler.Connection(host, port, handler)
audience_connection.connect()
```

Figure 5. NCLua code of prototype using the libraries.

Our context-aware application exhibits the following behavior: if audience is sufficient, the TV should start showing a Product Video, otherwise, a Home Video should be displayed in a loop until audience is sufficient. This reactive behavior has been implemented using “connectors” and “links”, both elements of the NCL language. Figure 6 depicts the connector element, which defines that at the end of the Property Anchor assignment (onEndAttribution), if the value assigned to this anchor is “sufficient”, a content video should be started (start). A link element links the behavior defined in the connector to the Property Anchor "audience" and the Product Video.

```
<causalConnector id="onAudienceTest">
 <compoundCondition operator="and">
  <simpleCondition role="onEndAttribution"/>
   <assessmentStatement comparator="eq">
    <attributeAssessment     role="compare"     eventType="attribution"
attributeType="nodeProperty"/>
    <valueAssessment value="sufficient"/>
   </assessmentStatement>
 </compoundCondition>
 <simpleAction role="start" max="unbounded"/>
</causalConnector>
```

Figure 6. Connector implementing the application’s reactive behavior.

6.1. Considerations

The development of NCL context-aware application prototypes has allowed us to identify particularities of the middleware Ginga that can be used to implement the most important requirements of context-aware applications (capturing, processing and reacting upon context changes). For example, we have identified the following:

- Capturing context from remote sensors can be realized by means of the Interactivity Channel, which allows an NCL application to communicate with remote devices over a network. In order to facilitate context event handling and distribution, we have proposed the TCPEventHandler Properties libraries;

- Processing context may be partially performed by the NCL document (by means of Connectors) and partially by NCLua scripts. We have also shown that context information can be represented by means of Property Anchors in NCL;
- Reacting upon context changes can be realized by means of the connector element, which allows the specification of causality rules that related NCLua scripts, Property Anchors, videos and invocations to external services.

7. Related Work

Research on context-awareness in the scope of interactive digital TV is usually associated with specific types of applications (and context information), such as in [da Silva et al. 2009], [Leite et al. 2007] e [Thawani et al. 2004]. The PersonalTVware [da Silva et al. 2009], for instance, proposes a context-aware personalized recommendation architecture for the interactive TV environment. This architecture supports specific types of context, such as user profile, and history of watched TV programs. In [Leite et al. 2007] it is proposed an architecture to determine the level of adequacy of TV contents according to the user's context (mostly computational context only). For example, it is possible to assess whether a certain content is adequate to the format of the device's screen. The approach presented in [Thawani et al. 2004] proposes an architecture, which selects and inserts, at real time, context-aware commercials to the broadcast flow transmitted to the TV. For that, user's profile and history are considered as context information.

As can be seen, most related work use context information in the scope of a single domain (e.g., personalization of content using user's profile and history). None of the approaches present proposals that facilitate capturing context from remote sensors and reacting upon context changes. Our work focuses on proposing mechanisms to support capturing and handling distributed context information, independent of the type of context being supported.

8. Concluding Remarks

Context-awareness is a relatively novel technology in the scope of the Brazilian Interactive TV Platform. Therefore, there is still no adequate methodology for developing context-aware applications, which incorporates the most important requirements for manipulating context information (capturing, processing and reacting upon context changes).

We have presented in this paper the design and implementation of a context-aware application prototype, which allowed us to identify existing functions of Ginga-NCL that can be used to facilitate the development of context-aware applications. In contrast to other approaches, our work does not intend to extend the Ginga middleware, which is already defined as part of the Brazilian DTV standard.

During the development process, we have also identified that the mechanisms for context event handling in the Ginga middleware is cumbersome. In order to facilitate handling context events, we have implemented a programming framework, which hides event handling complexity easing, therefore, context-aware development process.

As part of future work, we intend to incorporate programming facilities into a Context Management Infrastructure to support acquisition and manipulation of context

information from various sources of heterogeneous nature. We plan to use the programming framework proposed in this paper as bases for the development of this infrastructure, since they are generic to support various types of context and, therefore, facilitate the development of distributed context-aware applications in various application domains (e.g., telemedicine, personalization, gaming, etc.).

References

ABNT. (2007). ABNT NBR 15606-2:2007: Digital Terrestrial Television Standard 06: Data Codification and Transmission Specifications for Digital Broadcasting, Part 2 – GINGA-NCL: XML Application Language for Application Coding.

da Silva, F. S., Alves, L. G. P. and Bressan G. (2009). PersonalTVware: A Proposal of Architecture to Support the Context-aware Personalized Recommendation of TV Programs. *European Interactive TV Conference (EuroITV 2009)*, Leuven.

Dey, A. K. (2001). Understanding and Using Context. *Personal and Ubiquitous Computing*, pages 4-7.

Eugster, P. T., Felber, P. A., Guerraoui, R. and Kermarrec, A. (2003). The many faces of publish/subscribe. *ACM Computing Surveys (CSUR)*, 35, 2, pages 114-131.

Leite, L. E. C., Lima, O., Filho, G. L. S., Meira, S. R. L. and Tedesco, P. C. A. R. (2007). Uma Arquitetura de Serviço para Avaliação de Contextos em Redes de TV Digital. *In Anais do 25º Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos.* Belém, PA, pages 1015-1028.

Sant'Anna, F., Cerqueira, R. and Soares, L. F. G. (2008). NCLua – Objetos Imperativos Lua na Linguagem Declarativa NCL. Simpósio Brasileiro de Sistemas Multimídia e Hipermídia (Webmedia 2008), Vila Velha. In *Anais do XIV Simpósio Brasileiro de Sistemas Multimídia e Hipermídia*, v.1, pages 67–74.

Schmidt, D. C. (1994). Reactor: An Object Behavioral Pattern for Concurrent Event Demultiplexing and Dispatching. In *Proceedings of the First Pattern Languages of Programs conference in Monticello*, Illinois.

Souza Filho G. L., Leite, L. E. C. and Batista, C. E. C. F. (2007). Ginga-J: the procedural middleware for the Brazilian digital TV system. *Journal of the Brazilian Computer Society*. vol.12(4), pages 47-56.

Telemídia, L. (2007). *Ginga Digital TV Middleware Specification* Source: http://www.ginga.org.br.

Thawani, A., Gopalan, S. and Sridhar, V. (2004). Context Aware Personalized Ad Insertion in an Interactive TV Environment. In *Proceedings of Workshop on Personalization in Future TV*.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *In Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 511–518.

Um modelo para o gerenciamento de ontologias em ambientes ubíquos

Ana Helena O. R. Castillo, Fabrício N. Buzeto, Carla D. Castanho, Ricardo P. Jacobi

[1]Departamento de Ciência da Computação – Universidade de Brasília (UnB)
Caixa Postal 4466 – 70910-900 – Distrito Federal – DF – Brazil

anaozaki@unb.br,fabricio@intacto.com.br,{carlacastanho,rjacobi}@cic.unb.br

Abstract. *Besides providing interoperability, context aware middlewares must interact with the environment in order to collect relevant information for context determination. The aim of this work is use ontologies to model context informations, allowing, at the same time, that developers build applications capable of structurally and semantically modify the underlying context ontology by making use of an API (Application Programming Interface). Even more, it also includes the support for the task of consistency verification in the uOS middleware before each change, to guarantee that the inclusion of a new application will not interfere the others.*

Resumo. *Além de prover interoperabilidade, os middlewares sensíveis ao contexto devem interagir com o ambiente de modo a coletar informações relevantes para a determinação do contexto. A proposta desse trabalho é utilizar ontologias para modelagem de informações de contexto, permitindo, ao mesmo tempo, que desenvolvedores construam aplicações capazes de modificar estruturalmente e semanticamente essas ontologias de contexto através de uma API (Application Programming Interface). Além disso, a proposta inclui o suporte do middleware uOS na tarefa de verificar a consistência antes de cada mudança, para garantir que a inclusão de uma nova aplicação no middleware não afete as demais.*

1. Introdução

Computação ubíqua e computação pervasiva são termos muitas vezes utilizados de forma intercambiável [Saha and Mukherjee 2003]. A computação pervasiva tem como objetivo capturar o contexto do ambiente e construir modelos computacionais que alteram seu comportamento de acordo com contexto, ajustando-se dinamicamente às necessidades do usuário e sem a interferência deste. Já a computação ubíqua combina a computação pervasiva com os avanços da computação móvel para prover ao usuário um contexto computacional global e integrado [Zhao and Wang 2011]. Outro conceito importante é o de sensibilidade ao contexto (*Context Awareness*), que é definido como a habilidade de um dispositivo móvel ser sensível ao que está em volta do usuário, ao estado do ambiente físico [Muthukrishnan et al. 2005]. Um sistema é sensível ao contexto se utiliza contexto para prover informações e serviços relevantes ao usuário [Dey 2001]. Nessa definição a "relevância" depende das tarefas dos usuários e é descoberta utilizando informações de contexto.

Um dos principais objetivos de um *middleware* em ambientes distribuídos é prover uma camada interoperável [1], capaz de encapsular a heterogeneidade dos dispositivos. Os *middlewares* sensíveis ao contexto devem, além de prover interoperabilidade, interagir com o ambiente de modo a coletar informações relevantes para a determinação do contexto. Por exemplo: localização dos usuários e dispositivos; estado dos dispositivos; preferências dos usuários e características dos dispositivos. Essas informações precisam ser processadas para identificação do contexto e sua evolução, propagando o contexto para o nível de serviço.

Este trabalho aborda a utilização de ontologias para modelagem e gerenciamento das informações de contexto em ambientes ubíquos. Devido à diversidade de domínios nos quais as aplicações podem estar inseridas, propõe-se que ontologias de contexto não sejam modeladas de forma estática. Assim faz-se necessário o desenvolvimento de mecanismos que suportem a evolução de ontologias de acordo com a dinamicidade do ambiente ubíquo. Para atingir esses objetivos, este trabalho estrutura uma solução baseada em três componentes básicos: (i) um gerenciador de informações de contexto; (ii) um gerenciador de alterações das ontologias e (iii) a definição de uma interface para permitir que aplicações criem e alterem ontologias de contexto. O modelo proposto agrega sensibilidade ao contexto ao *uOS* [Buzeto et al. 2010], *middleware* voltado para a adaptabilidade de serviços em ambientes inteligentes.

O restante deste trabalho é organizado da seguinte forma. A Seção 2 introduz conceitos básicos relativos a ontologias. Trabalhos correlatos são apresentados na Seção 3. Na Seção 4 descreve-se brevemente o *middleware uOS*. O modelo proposto é discutido na Seção 5 e algumas considerações finais são apresentadas na Seção 6.

2. Ontologias

O termo "ontologia" tem origem na Filosofia e está ligado ao estudo da natureza da existência. Nas Ciências da Computação e da Informação, esse termo refere-se à modelagem do conhecimento acerca de algum domínio, seja real ou virtual [Liu and Zsu 2009]. Este trabalho adota a seguinte definição de ontologia:

Definição: *Uma **ontologia** é uma especificação formal e explícita de uma conceitualização compartilhada acerca um domínio de interesse* [Gruber et al. 1993].

Entende-se o termo conceitualização como uma visão de algo que existe e que deseja-se representar. Quando existe um consenso acerca de determinada conceitualização, diz-se que ela é compartilhada. Também cabe ressaltar que o propósito de uma ontologia não é o de conceituar tudo o que existe, mas sim informações consideradas relevantes no contexto de um problema específico. Desse modo, uma ontologia pode ser vista como uma forma de representar o conhecimento referente a um domínio. Basicamente as ontologias descrevem:

- conceitos relevantes a um domínio;
- relacionamentos entre os conceitos;

[1] Uma camada interoperável é uma camada em que os dispositivos conseguem comunicar-se de forma transparente, sejam semelhantes ou não.

- restrições sobre os elementos da ontologia;
- instâncias que correspondem a indivíduos em um domínio.

As ontologias podem abrigar qualquer tipo de conhecimento acerca de um domínio. Em um cenário ubíquo, dispositivos entram e saem de maneira dinâmica, conforme seus usuários deslocam-se de um ambiente para outro. Assim, dispositivos não expressamente projetados para trabalharem juntos precisam ser interoperáveis. Conforme relatado em [Heflin 2004], a interoperabilidade pode ser alcançada ao explicitar o conhecimento sobre as características, meios de acesso, e outras informações sobre os dispositivos em uma ontologia.

O uso de ontologias também permite o compartilhamento e o reúso do conhecimento em um ambiente aberto e distribuído [Peters and Shrobe 2003]. Ao descrever o conhecimento acerca de um domínio em uma ontologia, obtém-se uma representação semântica das informações, as quais podem usadas para direcionar o comportamento de aplicações sensíveis ao contexto.

2.1. Evolução da Ontologia

Na medida em que novos conceitos, com suas relações e propriedades, são incorporados, alterados ou mesmo excluídos de um domínio, faz-se necessário que a ontologia associada seja adaptada a essas mudanças. Este trabalho adota a seguinte definição de evolução de ontologia:

Definição: *A **evolução da ontologia** é a adaptação temporal de uma ontologia para o surgimento de mudanças e a propagação dessas mudanças de forma consistente para os artefatos dependentes* [Stojanovic 2004].

Portanto, a evolução de uma ontologia envolve a manutenção da sua consistência após a mudança. Ao evoluir uma ontologia pode-se verificar a sua consistência **antes** ou **depois** da mudança [Stojanovic 2004]. Quando a verificação é feita **depois**, todas as mudanças podem ser checadas de uma só vez. Porém, se a checagem falhar é preciso retornar a ontologia ao estado inicial. A vantagem de verificar a consistência **antes** da mudança é que não é preciso guardar o estado inicial da ontologia. No entanto, a checagem à priori requer a definição de precondições necessárias à manutenção da consistência, além das restrições de consistência, que devem ser definidas para ambos os casos.

2.2. Consistência

No âmbito da Lógica, diz-se que uma teoria é consistente quando não contém contradições [Tarski and Helmer-Hirschberg 1941]. Este trabalho adota a definição dada por [Haase and Stojanovic 2005]:

Definição: *Uma **ontologia** O é consistente com respeito a um conjunto de condições de consistência K se, e somente se, para todo $k \in K$, O satisfaz a condição de consistência $k(O)$.*

Desse modo, entende-se que uma ontologia é consistente quando todas as restrições de consistência são respeitadas.

Exemplo: Suponha que o Departamento de Ciência da Computação da Universidade X ofereça cursos de graduação e mestrado. Considere que a ontologia utilizada para representar as entidades contenha a estrutura de conceitos apresentada na Figura 1.

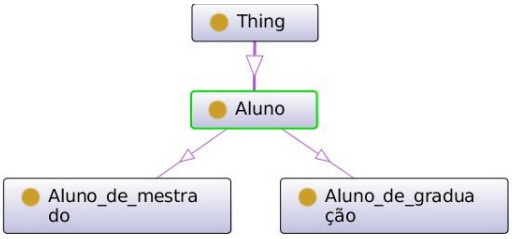


Figura 1. Exemplo de uma ontologia em OWL (*Web Ontology Language*). (Figura gerada pela ferramenta Protégé.)

Alguns anos após a utilização dessa estrutura, o referido Departamento adquire capacidade para oferecer um curso de doutorado. Para atender a essa demanda, um dos conceitos que poderiam ser refletidos para o sistema seria o de aluno de doutorado. Porém, o curso de doutorado participa do mesmo programa do mestrado, surgindo a figura do aluno de pós-graduação, um conceito que engloba os alunos de mestrado e doutorado. Dessa forma, a nova estrutura é ilustrada na Figura 2.

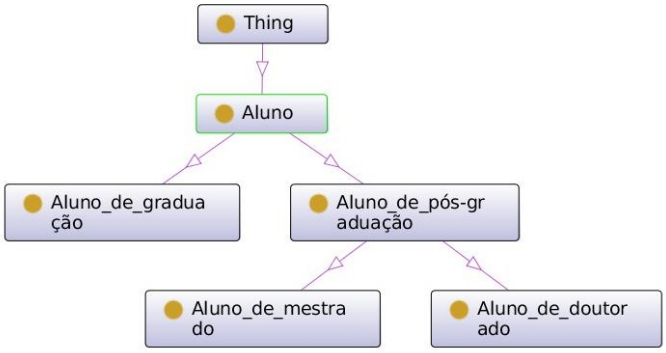


Figura 2. Exemplo de uma ontologia em OWL. (Figura gerada pela ferramenta Protégé.)

Observe que a inclusão do conceito de aluno de doutorado não necessariamente faz com que apenas este novo conceito seja incorporado na ontologia. Além disso, caso o conceito de aluno de doutorado fosse inserido na raiz *Aluno*, teríamos uma inconsistência pois *Aluno de pós-graduação* é um conceito que engloba o de aluno de doutorado. Pode-se dizer que de acordo com as necessidades do usuário, novos conceitos são vislumbrados e o conhecimento acerca de um domínio é aprimorado. Para que a ontologia não fique estagnada é preciso criar um mecanismo que permita a evolução da ontologia.

Deste exemplo, percebe-se que a evolução de uma ontologia não é um processo trivial, uma vez que a simples inclusão de um conceito pode causar inconsistências. Mesmo que a alteração de uma ontologia seja feita por um especialista, não há como garantir que todos os efeitos colaterais serão analisados. Faz-se necessário que a detecção de inconsistências seja um processo automatizado incorporado ao modelo de representação de contexto do *middleware*.

3. Trabalhos Relacionados

O uso de ontologias em ambientes ubíquos tem como motivação a possibilidade de modelar a semântica do contexto de forma independente da linguagem de programação, sistema operacional e/ou *middleware* utilizado.

O projeto CoBrA (*Context Broker Architecture*) propõe uma ontologia para ambientes pervasivos sensíveis ao contexto [Chen et al. 2003]. As principais contribuições estão relacionadas aos casos de uso modelados, juntamente com a separação de conceitos chave como localização, lugares, atividade e agentes. Porém, não há divisão entre os conceitos relacionados a domínio, como *Campus*, e conceitos mais genéricos e aplicáveis a vários domínios, como *Place*, pois ambas as classes são listadas em um mesmo bloco. Além disso, as classes e propriedades da ontologia são definidas de forma estática, não sendo tratado o aspecto evolutivo das ontologias a medida que novos conceitos são incorporados nos domínios.

Já o projeto SOCAM (*Service-Oriented Context Aware Middleware*) apresenta um modelo de contexto de duas camadas baseado em ontologia [Gu and Pung 2005]. Nesse modelo, a camada superior possui conceitos de alto nível e independentes de domínio. Já a camada inferior contém a ontologia específica de domínio, por exemplo o contexto do *escritório*, *carro*, etc. Essa divisão em camadas permite que a camada inferior seja ligada a camada superior de forma dinâmica, diminuindo o escopo de análise do contexto. Considerando-se que os dispositivos móveis possuem recursos limitados, a abordagem mostra-se pertinente pois tem como objetivo reduzir o custo de processamento das inferências e consultas realizadas à ontologia.

O DSM (*Dynamic Service Management*) [Min and Min 2010] apresenta um modelo de contexto baseado no projeto SOCAM. A diferença é que no DSM há uma terceira camada de ontologia, a qual é dedicada para os serviços. Esta camada gerencia a inclusão e exclusão de serviços. Quando um serviço é inferido da ontologia de domínio, ocorre uma busca pelo serviço inferido e, por fim, o serviço é inserido na ontologia de serviço. Caso haja alteração no domínio do contexto, a informação inferida anteriormente pode não ser mais necessária, nesse caso ocorre uma remoção na ontologia de serviço.

O projeto CANDEL (*Context As dyNamic proDuct Line*) apresenta uma forma de modelagem de contexto independente de domínio baseada em modelos de características [Jaroucheh et al. 2010]. O gerenciamento do contexto é realizado por componentes denominados CPCs (*Context Proxy Components*), que atuam como *proxies* das informações de contexto. Como podem haver várias características acerca de uma mesma informação de contexto, cabe aos serviços selecionar as características a serem utilizadas para prover as informações de contexto adequadas às aplicações. Dessa forma, o modelo trata os diferentes níveis de abstração das informações de contexto. Assim como em [Liu et al. 2010], a sublinguagem OWL DL (*Web Ontology Language Description Logic*) foi utilizada na implementação.

A partir da análise de trabalhos encontrados na literatura, constata-se que a utilização de ontologias para modelagem de informações de contexto em ambientes inteligentes vem recebendo crescente atenção e interesse da comunidade científica. Dentre os trabalhos analisados, entretanto, não foi encontrado suporte à evolução de ontologias, característica que, conforme discutido na seção 2, contribui para aumentar a flexibilidade

do *middleware* no suporte a aplicações sensíveis ao contexto.

4. *Middleware uOS*

O *middleware uOS* foi concebido com o objetivo de promover a adaptabilidade de serviços em um ambiente de computação ubíqua [Buzeto et al. 2010]. Seguindo a arquitetura SOA (*Service Oriented Architecture*), a DSOA também possui os papéis de consumidor, provedor e registro; com a diferença de que na DSOA (*Device Service Oriented Architecture*) quem assume esses papéis são os dispositivos e não os *Web Services*. Assim como em outros projetos, o gerenciamento de eventos segue o modelo *Publish-Subscribe*. Basicamente, o *middleware uOS* é composto por três camadas:

1. Camada de Rede: responsável por gerenciar as interfaces de rede do dispositivo;
2. Camada de Conectividade: responsável por coordenar a comunicação entre dispositivos;
3. Camada de Adaptabilidade: responsável por gerenciar as aplicações, os serviços disponíveis e o acesso aos mesmos.

A camada de Rede contém os módulos RADAR e *Connection Manager*. O RADAR descobre os dispositivos do ambiente por meio de varreduras e o *Connection Manager* gerencia conexões entre dispositivos com a mesma tecnologia de comunicação. Na camada de Conectividade, a comunicação entre dispositivos com diferentes tecnologias é obtida com a utilização de um *Proxy*, um dispositivo capaz de intermediar a comunicação. Por fim, a camada de Adaptabilidade contém um módulo chamado *Adaptability Engine*, que faz a intermediação entre as aplicações com o objetivo de selecionar o serviço mais apropriado para determinada aplicação. A camada de Adaptabilidade também contém o módulo *Application Deployer*, que permite a inclusão e exclusão de aplicações por meio das operações de *Deploy* e *Undeploy*.

5. Modelo para gerenciamento de ontologias em ambientes ubíquos

A Figura 3 ilustra os componentes responsáveis por gerenciar as informações de contexto e as mudanças que ocorrem na ontologia devido ao *Deploy* e *Undeploy* de aplicações no *middleware uOS*. As informações chegam ao gerenciador de contexto, o qual gera instâncias de conceitos preexistentes e inclui essas informações na ontologia. O gerenciador de contexto também é responsável por entregar informações de contexto atualizadas às aplicações que estão registradas para recebê-las. O mecanismo de registro segue o modelo definido pela arquitetura DSOA [Buzeto et al. 2010]. Utilizando a *uOS-Context API* (*Application Programming Interface*) (Seção 5.2), as aplicações podem alterar a ontologia e evoluir as informações de contexto existentes no *middleware*. Contudo, estas alterações são efetuadas somente após passarem por uma verificação de consistência, realizada pelo gerenciador de alterações.

As subseções a seguir descrevem os três componentes propostos para o gerenciamento de uma ontologia de contexto do *middleware uOS*.

5.1. Gerenciador de Contexto

O Gerenciador de Contexto é o componente responsável por controlar o fluxo das informações de contexto. As informações são essencialmente obtidas por meio de sensores e representadas no *middleware* por uma ontologia. A linguagem utilizada para a

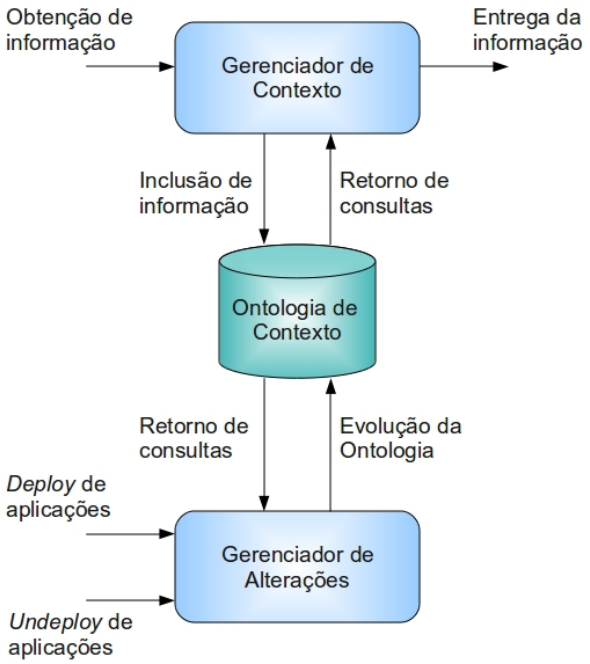


Figura 3. Gerenciamento do contexto do *middleware uOS* baseado em ontologia.

ontologia é a OWL (*Web Ontology Language*), por possibilitar um alto grau de expressividade semântica e ser recomendada pela W3C (*World Wide Web Consortium*). O nome da raiz de uma ontologia em OWL é *Thing*.

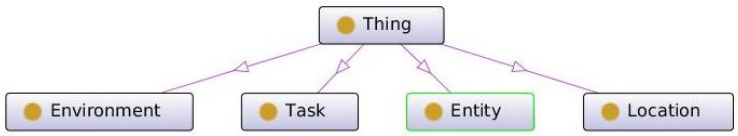


Figura 4. Ontologia de alto nível de abstração das informações de contexto. (Figura gerada pela ferramenta Protégé.)

Conforme ilustrado na Figura 4, o gerenciador de contexto do uOS contém uma ontologia de alto nível, a qual possui os conceitos considerados necessários para descrever o contexto de forma independente do domínio. A adoção dessa ontologia serve para padronizar a forma como é tratado cada tipo de conceito relacionado ao contexto. Existem também as ontologias de domínio, as quais extendem a ontologia de alto nível e são construídas e evoluídas a partir das aplicações por meio da *uOS-Context API*, detalhada a seguir.

5.2. *uOS-Context API*

Dependendo do domínio, as aplicações sensíveis ao contexto podem requerer que diferentes conceitos, com suas propriedades e relacionamentos, estejam presentes na ontologia. Portanto, faz-se necessário que a inserção de aplicações no *middleware* permita que tais conceitos sejam refletidos na ontologia. A utilização de uma API tem como vantagem a separação das alterações realizadas em relação à forma de persistência da ontologia

no *middleware*. Além disso permite que tais alterações ocorram de forma padronizada e controlada pelo gerenciador de alterações (Seção 5.3), o qual garante a consistência da ontologia após a mudança.

O formato de cada mudança segue o modelo *sujeito-predicado-objeto*. O *sujeito* refere-se ao tipo de elemento em que será realizada a mudança: classe, subclasse, propriedade, subpropriedade, *range*, domínio... O *predicado* corresponde ao tipo de mudança: adição, remoção, edição... E o *objeto* contém a definição do sujeito, por exemplo, a classe que está sendo inserida. O número de parâmetros do objeto varia de acordo com o sujeito. Para incluir uma propriedade, o domínio e o *range* relacionados a propriedade também podem ser passados por parâmetro.

A *uOS-Context API* consiste em um conjunto de interfaces de programação que tem como objetivo viabilizar a manipulação estrutural e semântica da ontologia de contexto contida no *uOS*. A API oferece aos desenvolvedores a possibilidade de que suas aplicações alterem a ontologia existente no *middleware*. Cada instância do *uOS* possui uma ontologia ditada pelos conceitos inerentes as aplicações utilizadas pelo usuário. Desse modo, a construção da ontologia torna-se dinâmica e capaz de evoluir de acordo com as necessidades do ambiente.

5.3. Gerenciador de Alterações

Uma das razões para a utilização de ontologias em ambientes ubíquos é permitir que as informações de contexto sejam compartilhadas não somente entre vários dispositivos mas também entre várias aplicações. Entretanto, o desenvolvimento de aplicações frequentemente ocorre de forma independente, tornando mais complexa a tarefa de compartilhamento das informações. Além disso, o fato das aplicações alterarem a ontologia de contexto existente no *middleware* possibilita o surgimento de inconsistências. O gerenciador de alterações tem como objetivo controlar as mudanças realizadas na ontologia. A verificação da consistência é feita antes das alterações serem efetuadas, de modo a não permitir que a ontologia fique em um estado inconsistente.

O modelo de consistência adotado utiliza regras semelhantes às definidas em [Stojanovic 2004], com algumas alterações relacionadas a especificidades da linguagem OWL. A seguir estão listadas regras estruturais para manutenção da consistência:

1. Todas as entidades (classes, propriedades e instâncias) possuem um identificador único.
2. A hierarquia de classes é um grafo acíclico direcionado. O mesmo vale para a hierarquia de propriedades.
3. Existe uma classe que é superclasse de todas as classes, que na linguagem OWL é a classe *Thing*.
4. Existe uma classe que é subclasse de todas as classes, que na linguagem OWL é a classe *Nothing*.
5. Todas as classes existentes na ontologia estão presentes na hierarquia de classes. O mesmo vale para as propriedades.
6. Uma propriedade pode estabelecer uma relação entre indivíduos ou relacionar um indivíduo a um valor.
7. Um domínio relaciona o sujeito de uma propriedade a uma classe.
8. Um *range* relaciona o objeto de uma propriedade a uma classe.

9. Todo indivíduo (instância) está associado a pelo menos uma classe.
10. Um indivíduo deve apenas ter propriedades da(s) classe(s) associada(s) ou de uma de suas superclasses.
11. A cardinalidade mínima deve ser menor que a cardinalidade máxima.
12. A cardinalidade é definida para uma propriedade de uma classe.

Além das regras estruturais, o gerenciador de alterações inclui regras de compatibilidade relacionadas à dependência entre as aplicações:

1. Caso uma ou mais aplicações sejam excluídas, a exclusão de uma entidade só pode ser realizada se não estiver relacionada a nenhuma outra aplicação incluída no *middleware*.
2. Caso uma aplicação tente inserir uma nova entidade no *middleware*, a aplicação só será aceita se a inclusão não infringir nenhuma das regras estruturais citadas acima.
3. Caso uma aplicação tente inserir uma entidade já existente no *middleware*, a aplicação só será aceita se as entidades forem idênticas.

6. Conclusões

Neste trabalho foram propostos os componentes responsáveis pelo gerenciamento de uma ontologia de informações de contexto em um ambiente ubíquo. Conforme visto nos trabalhos correlatos, algumas iniciativas de utilização de ontologias para ambientes sensíveis ao contexto foram realizadas. Contudo, a maneira como os *middlewares* para ambientes ubíquos gerenciam essas ontologias ainda se apresenta como potencial foco de investigação.

Mais precisamente, a proposta deste trabalho é permitir que os desenvolvedores construam aplicações capazes de modificar a ontologia utilizando uma API. Deixar a cargo dos desenvolvedores a responsabilidade de lidar com a persistência da ontologia, assim como aspectos relacionados a dependência de informações e consistência, tornaria muito complexa a tarefa de evoluir uma ontologia. A verificação de consistência deve ser feita pelo *middleware*, de modo a evitar que a inclusão de alterações na ontologia afete outras aplicações. Portanto, faz-se necessário incorporar funções de gerenciamento no *middleware*, o qual será encarregado de lidar com o gerenciamento da ontologia e das informações de contexto. Trabalhos futuros incluem análise de novas regras de consistência que permitirão maior controle do *middleware* sobre mudanças realizadas na ontologia.

Agradecimentos Os autores agradecem o apoio do DPP (Decanato de Pesquisa e Pós-Graduação) da Universidade de Brasília para a realização deste projeto.

Referências

Buzeto, F., de Paula Filho, C., Castanho, C., and Jacobi, R. (2010). DSOA: A Service Oriented Architecture for Ubiquitous Applications. In Bellavista, P., Chang, R.-S., Chao, H.-C., Lin, S.-F., and Sloot, P., editors, *Advances in Grid and Pervasive Computing*, volume 6104 of *Lecture Notes in Computer Science*, pages 183–192. Springer Berlin / Heidelberg.

Chen, H., Finin, T., and Joshi, A. (2003). An ontology for context-aware pervasive computing environments. *The Knowledge Engineering Review*, 18(03):197–207.

Dey, A. K. (2001). Understanding and using context. *Personal and Ubiquitous Computing*, 5:4–7.

Gruber, T. et al. (1993). A translation approach to portable ontology specifications. *Knowledge acquisition*, 5:199–199.

Gu, T. and Pung, H. (2005). A middleware for building context-aware mobile services. In *Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th*, volume 5, pages 2656–2660. IEEE.

Haase, P. and Stojanovic, L. (2005). Consistent evolution of owl ontologies. *The Semantic Web: Research and Applications*, pages 182–197.

Heflin, J. (2004). OWL Web Ontology Language-Use Cases and Requirements. *W3C Recommendation*, 10.

Jaroucheh, Z., Liu, X., and Smith, S. (2010). CANDEL: Product Line Based Dynamic Context Management for Pervasive Applications. In *2010 International Conference on Complex, Intelligent and Software Intensive Systems*, pages 209–216. IEEE.

Liu, C., Chang, K., Jason, J., and Hung, S. (2010). Ontology-Based Context Representation and Reasoning Using OWL and SWRL. In *8th Annual Communication Networks and Services Research Conference*, pages 215–220. IEEE.

Liu, L. and Zsu, M. (2009). *Encyclopedia of database systems*. Springer Publishing Company, Incorporated.

Min, D. and Min, H. (2010). Dynamic service management technique for various context information. In *Electronics and Information Engineering (ICEIE), 2010 International Conference On*, volume 2, page V2. IEEE.

Muthukrishnan, K., Lijding, M., and Havinga, P. (2005). Towards smart surroundings: Enabling techniques and technologies for localization. *Location-and Context-Awareness*, pages 350–362.

Peters, S. and Shrobe, H. (2003). Using semantic networks for knowledge representation in an intelligent environment. In *Pervasive Computing and Communications, 2003.(PerCom 2003). Proceedings of the First IEEE International Conference on*, pages 323–329. IEEE.

Saha, D. and Mukherjee, A. (2003). Pervasive computing: a paradigm for the 21st century. *Computer*, 36(3):25–31.

Stojanovic, L. (2004). *Methods and tools for ontology evolution*. PhD thesis, University of Karlsruhe.

Tarski, A. and Helmer-Hirschberg, O. (1941). *Introduction to Logic and to the Methodology of Deductive Sciences*. Oxford university press.

Zhao, R. and Wang, J. (2011). Visualizing the research on pervasive and ubiquitous computing. *Scientometrics*, pages 1–20.

Suporte à Colaboração em uma Infraestrutura Computacional Pervasiva para Auxílio às Atividades Clínicas

Marcelo Lopes Kroth[1], Alencar Machado[2], Iara Augustin[3]

[1, 2, 3] Grupo de Sistemas de Computação Móvel (GMob),
Universidade Federal de Santa Maria (UFSM) – Santa Maria – RS – Brasil

marcelo.tuco@ufsm.br, alencar.ufsm@gmail.com, august@inf.ufsm.br

Abstract. *In a hospital, many activities involve more than one clinician and collaboration to treat a patient is more the rule than the exception. Concepts and technologies of Ubiquitous or Pervasive Computing are being explored to meet the requirements of hospital settings, namely: mobility, interruption and collaboration. This article shows a service to support collaboration in a pervasive computing infrastructure to support the clinical activities.*

Resumo. *Em um hospital, muitas atividades envolvem mais de um médico e a colaboração para o tratamento de um paciente é mais a regra do que a exceção. Conceitos e tecnologias de Computação Ubíqua ou Pervasiva estão sendo explorados para atender aos requisitos dos ambientes hospitalares, quais sejam: mobilidade, interrupção e colaboração. Este artigo apresenta um serviço para dar suporte à colaboração em uma infraestrutura computacional pervasiva para auxílio às atividades clínicas.*

1. Introdução

As principais características do trabalho em um hospital são a mobilidade, interrupção frequente das atividades em andamento e a colaboração. Porém, grande parte dos sistemas foi projetada em um cenário de utilização onde os usuários estão sentados em suas mesas em frente a um computador. Este é um dos principais motivos da grande rejeição que os clínicos têm com tais sistemas [Miller 2004] [Jha 2009].

Atualmente, pesquisas na área de tecnologia da informação têm demonstrado grande interesse em entender melhor como os profissionais de saúde trabalham com o intuito de, ao introduzir novas tecnologias, oferecer aos clínicos uma maior qualidade no atendimento ao paciente [Barsottini 2006].

A Computação Ubíqua (*Ubiquitous Computing*) ou Pervasiva (*Pervasive Computing*) é um novo paradigma computacional com tecnologia de comunicação e informação em qualquer lugar, acessível por qualquer pessoa, disponível o tempo todo, onde os recursos computacionais devem estar integrados ao ambiente físico de forma transparente [Weiser 1991]. Por outro lado, a Computação Pervasiva não pode ser interpretada somente como um ambiente impregnado por dispositivos, servindo e se adaptando para um único usuário, mas também como um lugar onde vários usuários agem e interagem uns com os outros para a realização das atividades [Cabitza 2006]. Conceitos e tecnologias da Computação Pervasiva podem ser explorados para construção de sistemas integrados ao cotidiano dos profissionais da área da saúde.

Desenvolver um sistema integrado a esse cenário representa um desafio significativo, razão pela qual vêm sendo realizadas pesquisas com o objetivo de criar uma infraestrutura computacional que dê suporte a esses requisitos [Bardram 2007] [Vicentini 2010]. O projeto ClinicSpace tem como objetivo o desenvolvimento de uma ferramenta-piloto de auxílio às atividades do profissional de saúde utilizando tecnologias e conceitos de Computação Ubíqua [Ferreira 2009].

Este artigo apresenta as soluções criadas para modelar e implementar mecanismos para dar suporte à colaboração na arquitetura ClinicSpace e encontra-se organizado da seguinte forma: na seção 2 é resumida a arquitetura ClinicSpace, seguida da proposta do serviço de colaboração na seção 3, na seção 4 são apresentados os trabalhos relacionados, bem como uma análise das características de colaboração de cada projeto e na seção 5 são feitas as considerações finais e também as perspectivas futuras do trabalho.

2. Arquitetura ClinicSpace

O projeto ClinicSpace [Ferreira 2009] tem como ideia central a utilização de tecnologias de Computação Pervasiva para auxiliar os médicos na execução de suas tarefas em um ambiente hospitalar. O objetivo do projeto é a criação de uma ferramenta-piloto que permita aos clínicos a personalização da execução de suas tarefas, que são gerenciadas por uma infraestrutura computacional em um ambiente pervasivo.

Mesmo que uma das características de um sistema pervasivo seja a proatividade, o ClinicSpace leva em consideração a forma particular com que cada médico executa as suas tarefas. O equilíbrio entre a personalização e a proatividade visa diminuir a rejeição que os clínicos possuem aos sistemas atuais. Assim, espera-se que os profissionais da área da saúde se sintam mais confortáveis, pois terão a possibilidade de controle e adequação do sistema às suas necessidades cotidianas.

As atividades realizadas pelos clínicos foram decompostas no conceito de tarefas, as quais são auxiliadas por aplicações e seguem a forma particular de cada indivíduo realizá-las (personalização). Tarefas simples são compostas por subtarefas (operações básicas do sistema, que são aplicações Java disponibilizadas para composição das tarefas pelo clínico) e, quando agrupadas, formam uma tarefa composta, que segue um fluxo de execução ilustrado na Figura 1. Essa modelagem dá às tarefas as características de decomposição, recombinação e reuso [Vicentini 2010].

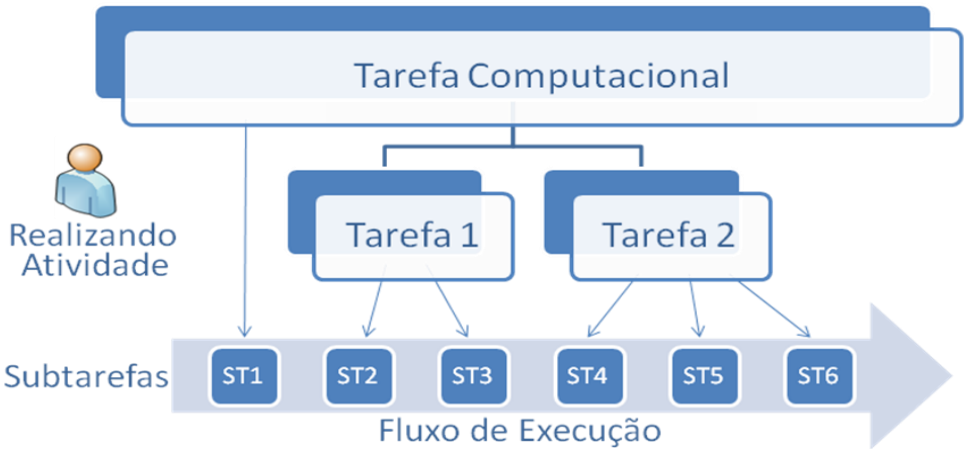


Figura 1 - Conceito de tarefas na arquitetura ClinicSpace

O *middleware* EXEHDA [YAMIN 2005] foi utilizado para dar o suporte ao gerenciamento das aplicações em um ambiente pervasivo na arquitetura ClinicSpace. Com isso foi possível que as tarefas herdassem as características de interrupção (podem ser interrompidas e retomadas posteriormente), contextualização (podem estar associadas a um contexto ou receber a associação dinamicamente), mobilidade e adaptação (devem migrar e se adaptar às diferentes características do ambiente de forma a acompanhar o usuário).

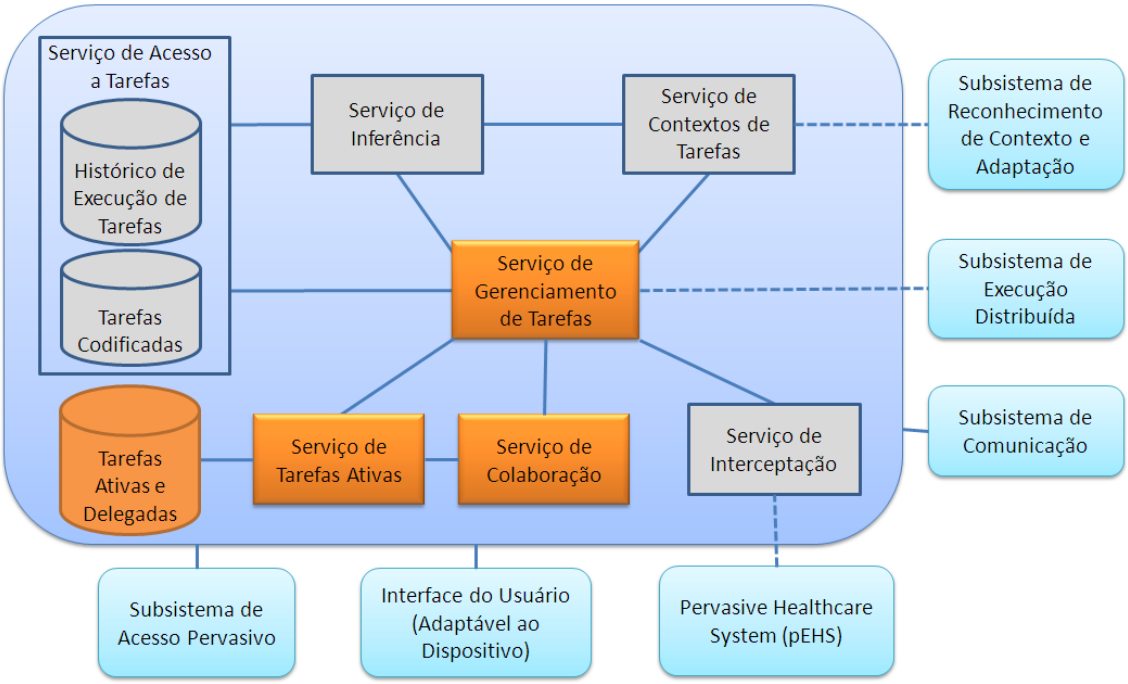


Figura 2 – Visão geral do Subsistema de Gerenciamento Distribuído de Tarefas

A Figura 2 apresenta uma visão geral do Subsistema de Gerenciamento Distribuído de Tarefas, adicionado ao *middleware* EXEHDA para dar suporte aos conceitos de atividades clínicas. O **Serviço de Acesso a Tarefas** é o responsável pelo acesso pervasivo ao repositório de tarefas e subtarefas de cada usuário. O **Serviço de Contextos de Tarefas** encapsula as informações de contexto obtidas pelo Subsistema de Reconhecimento de Contexto do EXEHDA. O **Serviço de Inferência** processa informações históricas sobre a execução das tarefas para inferir sobre a ativação de tarefas baseada em mudanças de contexto. O **Serviço de Gerenciamento de Tarefas** (SGT) controla a execução das tarefas do usuário. O **Serviço de Tarefas Ativas** (STA) controla as tarefas ativas de cada usuário. O **Serviço de Interceptação** é responsável por manipular os eventos gerados pelo pEHS (sistema de informação em saúde pervasivo).

3. Colaboração no ClinicSpace

Uma forma bastante comum de colaboração em ambientes hospitalares é a transferência de tarefas em execução entre os clínicos, por exemplo, na troca de plantão. Para isso, o sistema deve permitir aos usuários transferir suas tarefas não concluídas para outro profissional. A implementação deste requisito foi feita da seguinte maneira: quando um usuário recebe a tarefa que outro transferiu para ele, o sistema permite ao profissional que recebeu a tarefa escolher entre aceitá-la ou rejeitá-la. Ao rejeitá-la, a tarefa retorna

ao emissor. Já aceitando a tarefa, ela será incorporada a sua lista de tarefas interrompidas, podendo ser retomada do ponto onde foi parada.

Como o EXEHDA não incorpora os conceitos de tarefas e colaboração entre os usuários, o controle de usuários e suas tarefas ativas foi adicionado ao Subsistema de Gerenciamento Distribuído de Tarefas (SGDT). Assim, o EXEHDA controla o Ambiente Virtual do Usuário, e o SGDT controla a lista de tarefas ativas dos usuários. Com isso, é possível haver troca de tarefas entre os usuários, sem que haja necessidade de envolver diretamente os outros serviços do *middleware*.

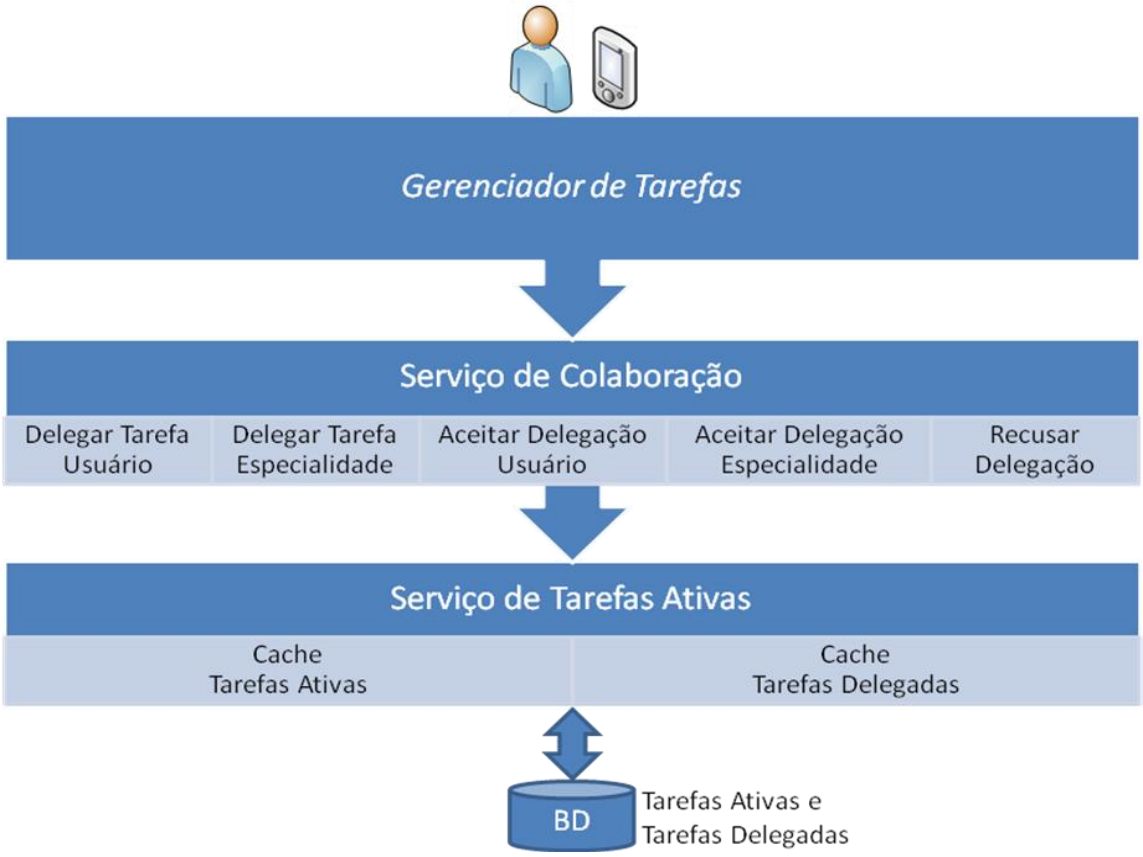


Figura 3 - Serviço de Colaboração

3.1 Arquitetura do Serviço de Colaboração

O **Serviço de Colaboração** (SC) é responsável pelo gerenciamento da delegação de tarefas entre os usuários do ClinicSpace. O SC disponibiliza uma API que é utilizada pelo Gerenciador de Tarefas para buscar as tarefas delegadas para o usuário, buscar as tarefas delegadas para a(s) especialidade(s) do médico, delegar tarefa para outro usuário, delegar tarefa para uma especialidade, aceitar uma tarefa delegada por outro usuário, recusar uma tarefa delegada por outro usuário e atribuir para si uma tarefa delegada para a sua especialidade médica.

A Figura 4 mostra o diagrama de sequencia da delegação de tarefa para uma especialidade. Quando o médico realiza uma delegação de tarefa através do gerenciador de tarefas é utilizada a API do Serviço de Colaboração que recebe como parâmetros a tarefa e a especialidade para qual será delegada. O Serviço de Colaboração, então, verifica se a tarefa está em execução e, se estiver, pausa a tarefa. No momento que a tarefa é pausada (pode ser que ela já esteja pausada no momento da delegação) o objeto

é serializado e persistido no banco de dados. O Serviço de Colaboração, então, faz uma requisição ao Serviço de Tarefas Ativas através de um protocolo próprio que posteriormente será transformado uma conexão HTTP.

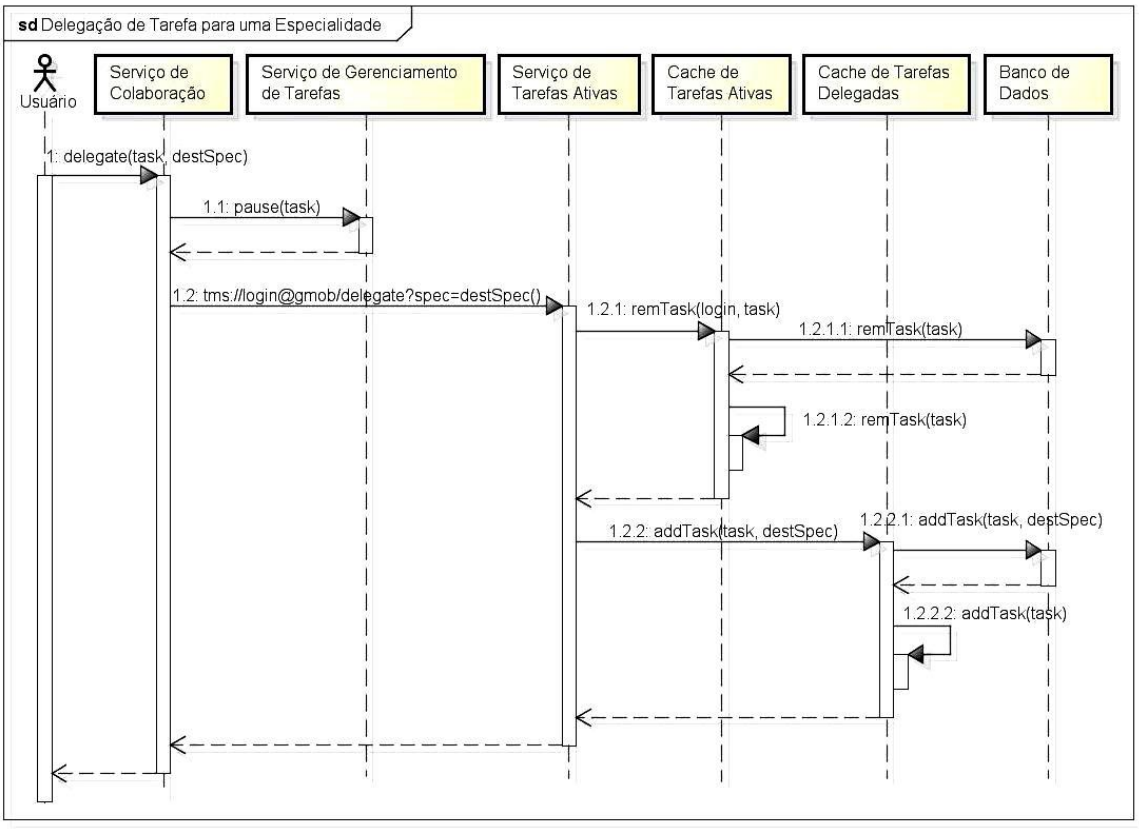


Figura 4 - Diagrama de Sequencia da Delegação de Tarefa

O **Serviço de Tarefas Ativas** é responsável por manter informações sobre as tarefas ativas de cada usuário. Uma tarefa é considerada ativa quando ela foi inicializada pelo Serviço de Gerenciamento de Tarefas e ainda não foi concluída nem cancelada. Foram adicionados ao STA mecanismos de persistência para as tarefas ativas e também das tarefas delegadas. A comunicação entre o SC e o STA é feita através de requisições HTTP. A tarefa é retirada da lista de tarefas ativas do usuário e adicionada na lista de tarefas delegadas da especialidade. Quando um médico, que possuir a especialidade para qual a tarefa foi delegada, atribuir a tarefa para si, a tarefa é removida da lista de tarefas delegadas e adicionada na lista de tarefas ativas do usuário. Quando o clínico retomar a tarefa, o objeto é buscado do banco de dados e desserializado, então o usuário pode continuar a tarefa do ponto onde ela parou.

O Serviço de Tarefas Ativas foi implementado no ClinicSpace como uma extensão do *HttpService* do EXEHDA, que é usado como base para todos os serviços do tipo cliente/servidor do *middleware*.

O formato utilizado para a URL é *protocol://user@cell/op*, onde:

- *protocol* – indica o protocolo usado;
- *user* – indica para qual usuário a operação está sendo solicitada;
- *cell* – indica a EXEHDAcel (área de atuação) a qual pertence o usuário;
- *op* – indica a operação que está sendo solicitada.

Exemplo de construção da url: *tms://tuco@gmob/delegate?spec=1*, onde *tms* é o protocolo (*task manager service*), que depois será transformado em HTTP, *tuco* é o usuário, *gmob* é a EXEHDAcel e *delegate* é a operação solicitada. Os demais parâmetros (*spec* é a especialidade) são passados utilizando o formato *query string*.

3.2 Estudo de Caso

Devido ao número de profissionais das mais diversas áreas envolvidos no ambiente clínico, a troca de informações entre os médicos é um ponto chave. A forma de trabalho é altamente colaborativa e o atendimento a um paciente envolve, em geral, várias especialidades [Bardram 2007]. A seguir é descrito um cenário para ilustrar o suporte à colaboração utilizando os conceitos do ClinicSpace.

O Dr. Kroth chega para trabalhar no hospital em seu último dia antes de começar as férias. Ele inicia o seu turno indo ao escritório localizado na ala do hospital, o sistema detecta a sua presença e abre uma tela de confirmação de identificação no computador da sala. Depois da verificação feita, a lista dos pacientes que estão sob o seu cuidado são mostradas na tela, além das tarefas que foram encaminhadas para a sua especialidade médica ou diretamente para o médico. Neste caso de uso existe uma aplicação que faz o gerenciamento das tarefas do usuário utilizando a infraestrutura do projeto (Figura 5).

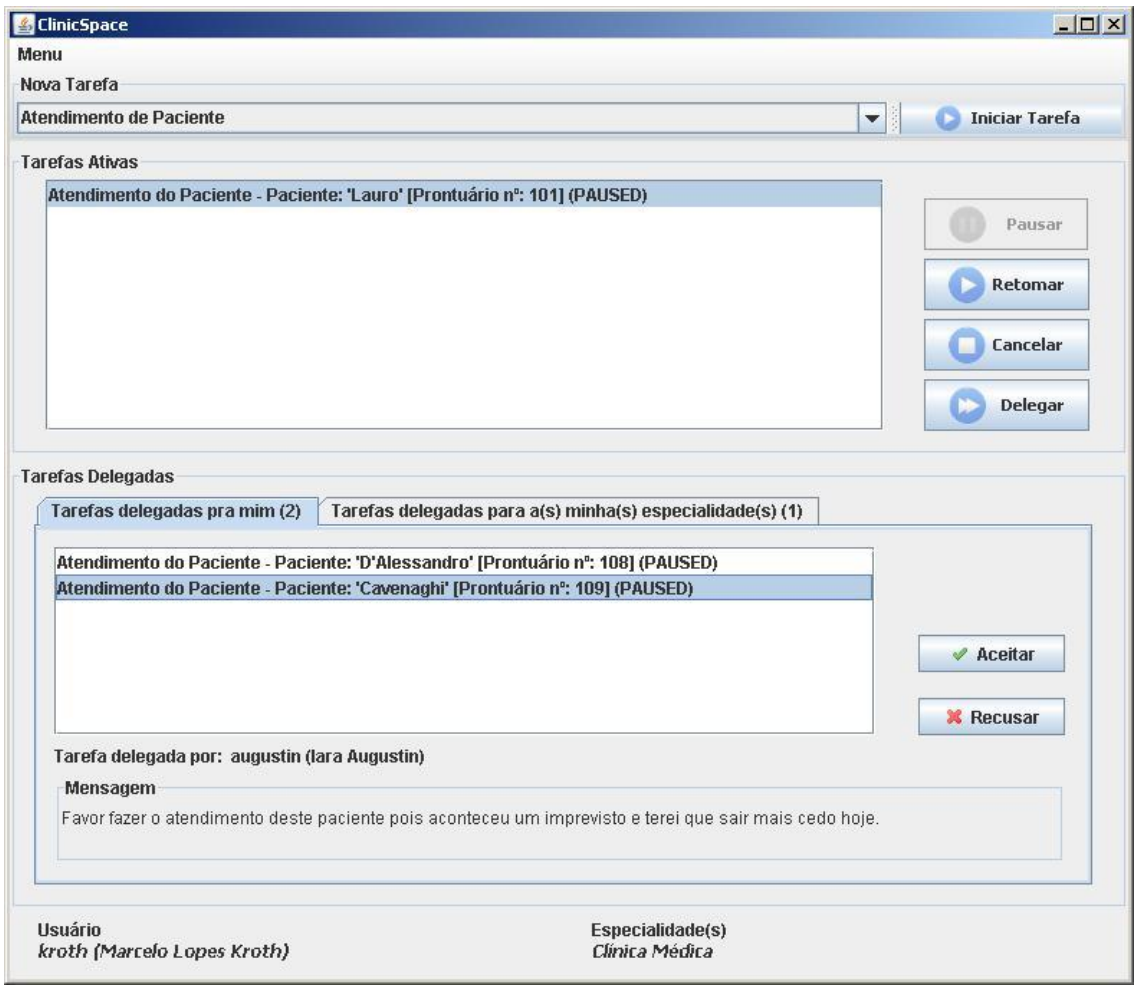


Figura 5 - Tela do Gerenciador de Tarefas

É possível pausar, retomar, cancelar e delegar as tarefas do usuário que estão ativas, bem como iniciar uma nova tarefa. Na mesma tela da aplicação também são mostradas as tarefas delegadas para o usuário e também para sua(s) a(s) especialidade(s), permitindo que o usuário aceite ou recuse a tarefa, no caso de delegação para o usuário e 'atribua pra ele', no caso de delegação para a sua(s) a(s) especialidade(s).

Após fazer o planejamento de suas atividades, o Dr. Kroth vai até o quarto do primeiro paciente que ele irá visitar hoje, chegando lá ele se aproxima de uma tela localizada junto ao leito e confirma uma mensagem que está perguntando se ele quer continuar o atendimento ao paciente. Ele termina a visita atualizando as informações no sistema e encaminhando a tarefa do atendimento para a Dra. Cameron, que já conhece o histórico do paciente. Neste estudo de caso, o ClinicSpace, utilizando informações de contexto, detecta a presença do médico no quarto do paciente e sugere a retomada da tarefa que está pausada envolvendo o paciente do quarto. Em seguida, ele encaminha a tarefa para a Dra. Cameron, utilizando o botão 'Delegar' na tela do gerenciador de tarefas. A Figura 6Figura 6 mostra a tela da delegação de tarefa onde o médico pode escolher delegar a tarefa para uma especialidade médica ou para algum profissional específico, informando uma mensagem para quem receber a tarefa.

O Dr. Kroth sai do quarto e, enquanto espera o elevador, retira um dispositivo móvel do bolso e consulta o resultado dos exames do segundo paciente que ele está indo visitar. Chegando ao quarto do paciente, os dados dos exames são mostrados na tela junto ao leito. Algumas anomalias no resultado dos exames, combinados com alguns sintomas do paciente, faz o Dr. Kroth tomar a decisão de pedir uma opinião para um colega de outra especialidade. Ele explica a situação ao paciente e, utilizando a tela junto ao leito, encaminha a solicitação para a especialidade médica desejada. Neste caso o médico aciona o botão 'Delegar' no gerenciador de tarefas e escolhe a especialidade médica desejada na tela da delegação de tarefa.

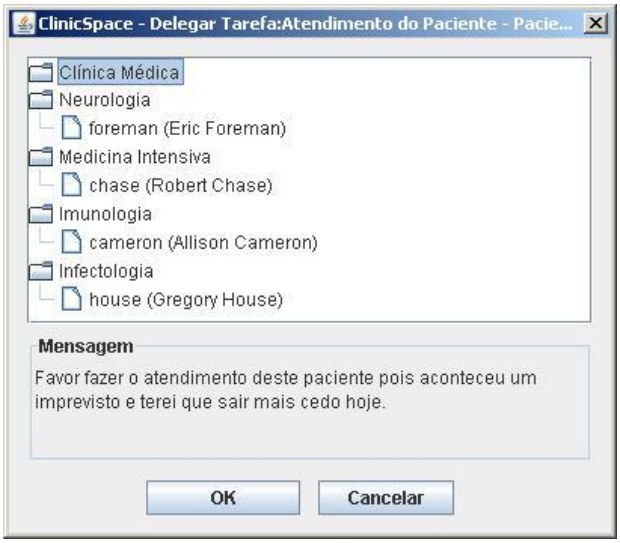


Figura 6 - Tela da Delegação de Tarefa

O Dr. Kroth volta ao escritório da ala do hospital, o sistema detecta sua presença na sala e lhe é mostrado o resultado dos exames de um paciente que o Dr. House lhe encaminhou. Ele escreve suas impressões sobre o caso e devolve ao Dr. House. O Dr.

Kroth nota que está se aproximando o final do horário do seu turno, ele verifica no sistema se existem atividades que ainda não estão finalizadas, o sistema lhe avisa e ele encaminha para os médicos ou especialidades mais adequados. Não tendo mais nenhuma atividade pendente, o Dr. Kroth está com o sentimento de dever cumprido, se despede dos colegas do hospital e começa a pensar nos detalhes das suas férias. Neste caso o Dr. House usou a funcionalidade 'Delegar' do gerenciador de tarefas e encaminhou a tarefa para o Dr. Kroth. Na tela do Dr. Kroth a tarefa apareceu na aba 'Tarefas delegadas pra mim' do gerenciador de tarefas. O Dr. Kroth aceita a tarefa, que passa a ser mostrada na lista de tarefas ativas. Agora o Dr. Kroth pode retomar a tarefa, analisar o resultado dos exames e delegar a tarefa de volta ao Dr. House.

4. Trabalhos Relacionados

Outros projetos serviram como fonte para as ideias utilizadas no ClinicSpace. Pode-se destacar o uso do *middleware* EXEHDA, o projeto Aura, que foi o precursor a utilizar o conceito de computação baseada em tarefas, e o projeto *Activity-based Computing* (ABC) que orientou a definição dos conceitos relacionados a atividades ou tarefas na área da saúde.

4.1 Projeto Aura

O projeto *Aura* [Garlan 2002] tem por objetivo promover uma arquitetura que gerencia um sistema pervasivo baseado nas atividades realizadas pelo usuário e possui mecanismos de busca e adaptação aos recursos disponíveis no ambiente. Foi o precursor ao utilizar o conceito de computação baseada em tarefas, permitindo que o usuário possa manter a continuidade de suas atividades mesmo se deslocando de um ambiente para outro, e motivou o desenvolvimento de vários outros projetos.

O conceito principal da arquitetura é a "aura" computacional do usuário, a qual se constitui em um sistema de identificação automática do usuário, de sua tarefa e dos requisitos necessários para sua realização. Baseado nas informações obtidas da "aura" do usuário, o sistema busca, de maneira proativa, os serviços e recursos disponíveis e mais adequados à realização da tarefa.

Um caso de colaboração apresentado no projeto foi o *Idealink* [Garlan 2002], uma aplicação desenvolvida utilizando a infraestrutura *Aura*. O *Idealink* é uma espécie de quadro negro distribuído e compartilhado utilizando mecanismos de colaboração síncrona, ou seja, suporte a múltiplas sessões onde os usuários podem escrever ou desenhar simultaneamente. O *Aura* possui suporte a alguns requisitos que podem ser utilizados para colaboração, porém deixa a cargo das aplicações essa responsabilidade, não possuindo de forma nativa essas características em sua infraestrutura computacional.

4.2 Projeto Activity-Based Computing (ABC)

O projeto ABC [Bardram 2007] tem como foco dar suporte computacional para o trabalho clínico em hospitais. A filosofia utilizada foi a da mobilidade e colaboração. Para tanto, foi desenvolvido um *framework ABC*, que é a infraestrutura de suporte para a Computação Pervasiva, com o objetivo de dar suporte à atividade humana, à colaboração e à mobilidade através de ambientes computacionais heterogêneos.

Através de funcionalidades direcionadas aos usuários, podem-se utilizar mecanismos de comunicação entre os profissionais de diferentes especialidades, permitindo que as informações estejam sempre disponíveis. A ideia básica do ABC é o compartilhamento das tarefas por todos os usuários autorizados e que, a qualquer momento, eles possam participar de uma atividade. Quando mais de um usuário participa de uma tarefa ao mesmo tempo, mecanismos de colaboração em tempo real ficam disponíveis, incluindo teleapontador (*telepointer*), audioconferência e informações sobre os participantes da atividade [Bardram 2007].

4.3 Comparativo entre os Projetos

As características de mobilidade, interrupção e colaboração relacionadas às atividades da área da saúde fazem com que ambientes clínicos tenham um grande potencial para pesquisas sobre colaboração pervasiva. Os projetos ABC e ClinicSpace inovam ao estudar tais características de forma integrada a sistemas pervasivos com o objetivo de melhorar o trabalho dos clínicos.

O *Aura* serviu de inspiração para vários outros projetos por ser o precursor a utilizar o conceito de computação baseada em tarefas e possui requisitos que podem ser utilizados para colaboração. A infraestrutura pode ser usada para o desenvolvimento de alguma aplicação com finalidade específica (por exemplo, um quadro colaborativo para auxiliar em diagnósticos usando como base o *Idealink*), porém a colaboração não é uma característica intrínseca às atividades na arquitetura do *Aura*.

O ABC tem como ponto forte a colaboração síncrona, utilizada principalmente para diagnósticos, possuindo mecanismos de comunicação entre os profissionais envolvidos, como compartilhamento de uma imagem de raios-X. A colaboração assíncrona é a mesma colaboração síncrona, ou seja, os profissionais autorizados podem participar ou modificar a atividade a qualquer momento.

O ClinicSpace possui o foco na atividade médica e tem o diferencial de permitir personalização das tarefas. Permite a colaboração assíncrona entre os médicos através da delegação de tarefas, que pode ser usada para solicitar a opinião de outro especialista ou encaminhar a um colega uma atividade que ainda não foi concluída, semelhante a um workflow.

5. Considerações Finais

O projeto ClinicSpace utiliza os conceitos de Computação Pervasiva para auxiliar os médicos na execução de suas tarefas em um ambiente hospitalar. O objetivo é permitir aos médicos a personalização da execução de suas tarefas, que são gerenciadas por uma infraestrutura computacional em um ambiente pervasivo. Uma das principais características do trabalho do médico em um ambiente clínico é a colaboração. Este artigo apresenta as soluções criadas para modelar e implementar mecanismos para dar suporte à colaboração na arquitetura ClinicSpace.

O projeto ClinicSpace ainda está em desenvolvimento. Os próximos passos da nossa pesquisa são analisar o impacto das alterações realizadas na arquitetura no desempenho das aplicações e avaliar os aspectos de usabilidade do sistema em um ambiente real.

Referências

Augustin, I., Lima, J. , Yamin, A. (2006) "Computação Pervasiva: como Programar Aplicações", X Simpósio Brasileiro de Linguagens de Programação (SBLP), RJ.

Bardram, J. E. Christensen, H. (2007) "Pervasive Computing Support for Hospitals: An overview of the Activity-Based Computing Project", IEEE Pervasive Computing, vol. 6, no. 1, pp. 44-51.

Barsottini, C., Wainer, J. (2006) "Análise do padrão de colaboração entre médicos por meio do prontuário do paciente", Congresso Brasileiro de Informática em Saúde Informática em Saúde e Cidadania, Florianópolis, SC, Brasil.

Cabitza, F., Locatelli M., Sarini M. and Simone C. (2006) "CASMAS: Supporting Collaboration in Pervasive Environments", Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications, pp. 286-295, March 13-17.

Ferreira, G., Augustin, I., Librelotto, G. , Silva, L., Yamin, A. (2009) "Middleware for management of end-user programming of clinical activities in a pervasive environment", In Workshop on Middleware for Ubiquitous and Pervasive Systems, volume 389, pp. 07–12.

Ferreira, G. (2009) "Adicionando ao Middleware EXEHDA o Suporte a Aplicações Orientadas a Atividades Humanas Cotidianas". Dissertação de Mestrado em Informática, PPGI, Universidade Federal de Santa Maria, Santa Maria.

Garlan, D., Siewiorek, D., Smailagic, A., Steenkiste, P. (2002) "Project Aura: Toward Distraction-Free Pervasive Computing", IEEE Pervasive Computing, v.1 n.2, pp. 22-31, April.

Jha, A., Desroches, C., Campbell, E., Donelan, K., Rao, S., Ferris, T., Shields, A., Rosenbaum, S., Blumenthal, D. (2009) "The Use of Electronic Health Records in U.S. Hospitals". In: New England Journal of Medicine. 360:1628-1638.

Miller, R., Sim, I. Physicians (2004) "Use of Electronic Medical Records: Barriers And Solutions". In: Health Affairs, 23, no. 2. pp. 116-126.

Vicentini, C., Machado, A. Ferreira, G., Lorenzi, F., Augustin, I. (2010) "PEHS – Arquitetura de um Sistema de Informação Pervasivo para Auxílio às Atividades Clínicas", Revista Brasileira de Computação Aplicada, Passo Fundo, pp. 69-80.

Weiser, M. (1991). "The Computer for the 21st Century". Scientific American, 265 (3), pp. 94–104, September.

Yamin, A., Augustin, I., Barbosa, J., Silva, L., Real, R., Schaeffer, A., Geyer, C. (2005) "EXEHDA: adaptative middleware for building a pervasive grid environment", In Artificial Intelligence and Applications - Self-Organization and Autonomic Informatics, volume 135, pp. 203–219. IOS Press.

Um Framework para Carregamento Dinâmico e Transição Suave entre Mapas Contextuais*

Danilo Inácio de Souza Resende, Heitor Menezes de O. Pereira, Ricardo C. Antunes da Rocha

[1]Instituto de Informática – Universidade Federal de Goiás (UFG)
Caixa Postal 131 – 74.001-970 – Goiânia – GO – Brasil

{daniloresende,ricardo}@inf.ufg.br, heitormzs@gmail.com

Abstract. *This paper presents a framework for building applications based on contextual maps: maps from heterogeneous map bases, which are dynamically discovered and loaded, according to the user context. The proposed framework enable a transparent integration of map services in a same user interface for map navigation and visualization. A framework instance defines tasks such as the communication protocol to a map service and rendering mechanism. The paper presents a case study with three map services: Google Maps, a simple prototype with bit-mapped maps and a service for maps based on symbolic location.*

Resumo. *Este artigo apresenta um framework para construção de aplicações baseada em mapas contextuais: mapas provindo de bases de mapas heterogêneas, descobertas e carregadas dinamicamente, de acordo como contexto do usuário. Por meio deste framework é possível integrar transparentemente serviços de mapas de diversas fontes por uma mesma interface de navegação e visualização. Uma instância do framework especifica, entre outras tarefas, o protocolo de comunicação com a base de mapas e como os mapas são renderizados na UI do usuário. O artigo apresenta um estudo de caso com três bases de mapas: Google Maps, um protótipo baseado em simples mapas bitmapeados e uma base de mapas baseados em localização simbólica.*

1. Introdução

Tradicionalmente, os serviços baseados em localização utilizam pontos geográficos baseados em latitude e longitude para fazer referência a uma certa localidade. Da mesma forma, os provedores de mapas estão fundamentados neste tipo de localização. Entretanto, locais de escopo geográfico bem limitado, como prédios, poderiam ser descritos por mapas que não são baseados em localização geográfica, mas em abstrações de localidade mais adequadas ao contexto do mapa. Por exemplo, o British Museum[1] oferece um mapa aos visitantes baseado em salas de visitação, onde cada uma mantém obras diferentes organizadas por diferentes critérios.

Uma aplicação que faça uso desses mapas deve ser capaz de utilizar outro tipo de inferência de localização, onde as informações para determinar a localização de um

*O autor Danilo Inácio de Souza Resende é aluno de iniciação científica pelo programa PIVIC/UFG.

[1]www.britishmuseum.org

determinado dispositivo são dadas a partir de pontos de referências próprios do mapa. Nessas aplicações, a própria interface com o usuário deve levar em consideração os pontos de referência. Este tipo de localização é chamada localização simbólica [Pradhan 2000].

O objetivo desse trabalho é implementar o conceito de *mapas contextuais*, proposto em [de Oliveira Pereira and da Rocha 2010], por meio de um framework para desenvolvimento de aplicativos na plataforma Android que permita carregar e exibir mapas de múltiplas fontes, e não apenas da base de mapas do Google Maps. Durante o carregamento de um novo mapa, também serão carregados as informações dos seus pontos de referência próprios, capazes de inferir uma localização.

Este artigo está organizado da seguinte forma. A seção 2 descreve um cenário para uso do conceito de mapas contextuais e elabora os principais requisitos do *framework*. A seção 3 apresenta a arquitetura de implementação do cenário proposto, e define a responsabilidade do *framework* e a sua interação com outros elementos do cenário, como um middleware sensível ao contexto e os serviços de mapas. As seções 4 e 5 descrevem a arquitetura interna do *framework* e a sua implementação na plataforma Android, respectivamente. A seção 6 avalia o *framework* proposto de acordo com três estudos de caso, enquanto que a seção 7 compara-o com outros trabalhos encontrados na literatura. Por fim, a seção 8 apresenta as conclusões deste trabalho e futuras direções de pesquisa.

2. Cenário

No cenário de referência deste trabalho, uma aplicação móvel de localização permite a um usuário se localizar dentro do campus da UFG, onde ocorre um evento aberto à comunidade chamado “Espaço das Profissões”. Este evento promove diversas atividades onde a comunidade interna e externa à universidade pode conhecer melhor cada curso de graduação.

Para chegar ao campus, um usuário utiliza os serviços providos pelo Google Maps, incluindo exibição de mapas e determinação de caminhos entre dois pontos. Entretanto, ao chegar na universidade, o serviço Google Maps deixou de ter utilidade pois não provia informações específicas do evento que estava ocorrendo, como local e horário das palestras.

Considere agora que a própria universidade possui um serviço de mapas interno à universidade que oferece mapas e primitivas de navegação especificamente relacionadas ao evento em questão. Como este serviço de mapas, um usuário poderia pesquisar qual estacionamento está mais próximo do estande do curso de bacharelado em Ciência da Computação.

Embora os dois serviços de mapas pudessem ser providos por aplicações distintas, é especialmente inconveniente ao usuário ter que trocar de aplicação para executar uma tarefa similar mas que aplicada a um contexto diferente (escopo de atuação do serviço de mapas da universidade). De fato, a mesma integração que existe entre os vários tipos de mapas providos pelo Google Maps poderia também ser oferecida com os mapas oferecidos pela universidade. O desafio, neste caso, é que o serviço Google Maps e correspondentes componentes de desenvolvimento de aplicações, são estaticamente ligados à base de provedores de mapas do Google e à sua própria semântica de mapas.

Dessa maneira, este cenário sugere a necessidade de um *framework* para

aplicações baseadas em mapas, que permita aos provedores de mapas oferecer componentes que permitam a integração dos seus mapas em aplicações e a troca suave de uma base de mapas para outra, seja por interação do usuário, seja por ação da aplicação. O *framework* deve ainda ser integrado com um middleware sensível ao contexto que permita o tratamento de mapas como contexto, disparar descoberta e carregamento de mapas contextuais, de acordo com a localização, assim como o controle da informação de localização. Com o *framework*, deseja-se evitar a integração explícita e estática de uma aplicação com diversas bases de dados e inter-relacionamento entre os mapas providos por cada base.

3. Arquitetura

A figura 1 ilustra a arquitetura de um sistema sensível ao contexto que faz uso do *framework* de mapas contextuais. Nesta arquitetura, quatro elementos interagem entre si: (i) a UI da aplicação, (ii) o *framework* de carregamento de mapas contextuais, (iii) um middleware sensível ao contexto e (iv) os serviços ou provedores de mapas.

Um serviço de mapas ou provedor de mapas é um servidor capaz de responder a requisições por mapas, tipicamente limitado a um certo escopo físico. Para permitir a transição suave entre os mapas de bases heterogêneas, o *framework* assume como premissa que todos os mapas providos pelos serviços devem possuir um sistema de coordenadas em comum. No protótipo implementado, assumimos coordenadas geográficas baseadas na tupla $(latitude, longitude, altitude)$, utilizada em diversos serviços de localização como GPS e bases de mapas como Google Maps. Portanto, todos os serviços de mapas devem oferecer primitivas de recuperação de mapas que permitam, no mínimo, responder a requisições baseadas neste sistema de coordenadas.

O middleware sensível ao contexto oferece serviços assíncronos para obtenção de informações contextuais relevantes para interação com os serviços de mapas. Em especial, o *framework* interage com o middleware para obter atualizações na localização de entidades exibidas nos mapas ou do próprio dispositivo, além da descoberta de serviços de mapas contextuais. Para descobrir serviços de mapas associados a uma localização, o *framework* registra no middleware sensível ao contexto o interesse por receber notificações contextuais de provedores de mapas no atual escopo em que a aplicação está executando. *Escopo de mapa* é uma abstração do domínio que um provedor de mapas é capaz de cobrir. Um escopo de mapa pode ser uma certa área física ou lógica (como a universidade ou museu), uma rede ou qualquer outro espaço que possa ser modelado por contexto. No protótipo desenvolvido, o escopo adotado foi o de rede, o que significa que uma troca da rede do dispositivo pode disparar uma troca do provedor de mapas.

O *framework* permite a interação transparente da UI do usuário e do middleware sensível ao contexto com um serviço de mapas. Uma instância do *framework* implementa a interação com um serviço de mapas em particular. O *framework* é independente da implementação do middleware, bastando que este ofereça uma interface assíncrona de comunicação (publish/subscribe).

4. Framework para Carregamento de Mapas Contextuais

O objetivo do framework é possibilitar o desenvolvimento de serviços de localização que utilizem diferentes provedores de mapas e diferentes semânticas, baseada no contexto do

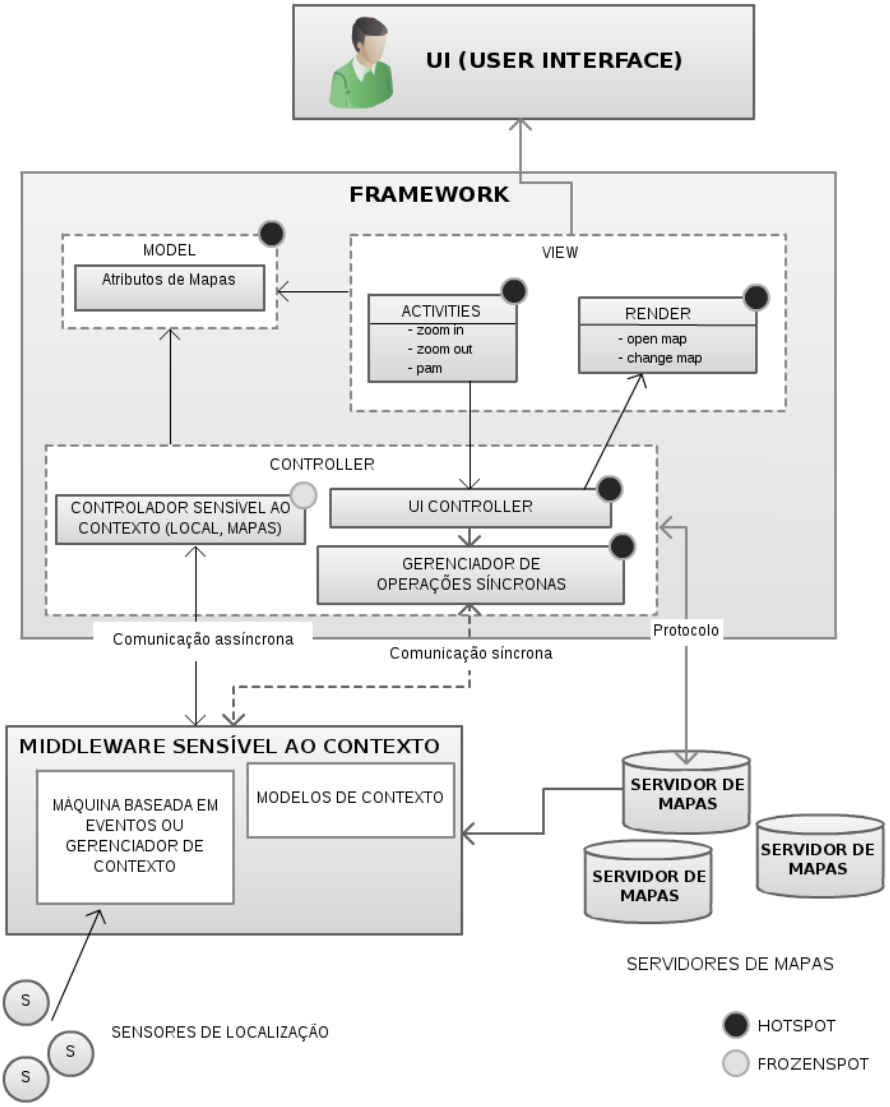


Figura 1. Arquitetura do *Framework*

usuário. Neste cenário, as responsabilidades do *framework* de mapas contextuais são: oferecer os componentes de interface de navegação nos mapas, renderizar os mapas obtidos na UI do usuário, implementar o protocolo de comunicação com os serviços de mapas e interagir com o middleware sensível ao contexto com o objetivo de obter notificações relativas à descobertas de mapas e atualização da localização do usuário. Dentre essas responsabilidades, tipicamente a renderização de mapas e o protocolo com os serviços de mapas representam os *hotspots* do *framework*, ou seja, devem ser implementados em cada instância particular do *framework*.

4.1. Arquitetura do Framework

A arquitetura do *framework* segue o padrão MVC (Model-View-Controller), com o qual é possível isolar as suas funcionalidades referentes à interface com usuário (navegação em mapas e renderização de mapas), de comunicação com o middleware sensível ao contexto e com os servidores de mapas. A figura 1 detalha a arquitetura interna do *framework* e

seus componentes `Model`, `View` e `Controller`, assim como a interação entre eles.

O componente `Model` é responsável por modelar em objetos os mapas que são manipulados pelo *framework*. Estes atributos serão específicos para cada instância do *Framework*. O componente `View` trata a renderização de um mapa e a resposta aos comandos da UI, como operações de navegação e *zoom in/out*. E o componente `Controller` possui os elementos responsáveis pela comunicação com o middleware sensível ao contexto e com os serviços de mapas. O protocolo de comunicação com os servidores pode ser específico para cada instância do *framework* e define também as regras de transferência de dados. Ele é implementado no *Gerenciador de Comunicação Síncrona.*

4.1.1. Componente `Controller`

O `Controller` é responsável por interagir com o middleware sensível ao contexto, registrando o interesse em receber notificações de situações contextuais que interferem no(s) mapa(s) exibido pela aplicação. Há no mínimo duas informações contextuais que podem interferir no funcionamento da aplicação: a localização e o mapa associado à localização. Do ponto de vista do funcionamento do *framework*, não há diferença entre obter a localização a partir de uma interação com a UI ou de uma notificação recebida por um middleware. Em um cenário estático, eventos de interface do usuário disparam a troca do mapa e/ou execução de operações com a base de mapas. O controlador registra junto ao middleware o interesse em receber notificações de tipos de mapas para uma determinada localização. O *framework* utiliza as notificações para atualizar no componente `View` a interface que permite ao usuário selecionar um mapa ou outro para exibição. Portanto, uma requisição a novo mapa pode ser disparada de duas formas:

- *Por interação do usuário* (eventos de UI): Componente `View` atualiza na interface os tipos de mapas disponíveis para a localização coberta pela aplicação, ele então pode selecionar qual tipo de mapa atende melhor aos seus requisitos. Neste caso, o middleware recebe o registro pelo interesse em todos os tipos de mapas que se aplicam a uma localidade.
- *Por mudanças no contexto*: a aplicação baseada no *framework* recebe as notificações da mudança de contexto, e verifica a existência ou alteração dos mapas existentes e decide exibir ou não um dos mapas disponíveis.

4.1.2. Componente `View`

O componente `View` é responsável pela apresentação gráfica da interface com o usuário, incluindo construir uma representação gráficas dos mapas recuperados do serviço de mapas. Cada componente `View` é associada a um controlador apresentado apenas como um atributo do tipo `Controller`. Este atributo é responsável por traduzir as operações de navegação (movimentação pelos mapas) ou detalhamento (operação de zoom) requeridas, capazes de trabalhar com semânticas diversas de localização, isto é, capaz de manipular diferentes valores obtidos por diversas bases de mapas.

Definida a semântica utilizada por um serviço de mapas, é necessário que esta

possua referência ao sistema de coordenadas geográficas, dessa forma é possível efetuar uma troca suave entre os mapas contextuais de interesse.

Considerando serviços baseados em mapas, é necessário definir o grau de detalhamento do mapa. No caso do Google Maps, por exemplo, essa informação é descrita como *zoom* do mapa. Para tratamento dessas informações o *framework* implementa métodos para manipular o nível de zoom dos mapas contextuais.

4.1.3. Componente `Controller`

O componente `Controller` é responsável pelas operações internas de controle dos mapas e externas de comunicação com outros componentes do sistema. As primitivas de localização, como existência e interseção, são definidas para cada semântica utilizada pelos mapas contextuais a partir do serviço oferecido. Dessa maneira, cada instância do *framework* define qual o tratamento para cada interação com o mapa. Tipicamente, o componente `Controller` traduz as operações de navegação do `View`, mencionadas anteriormente, em requisições à respectiva base de mapas. Por exemplo, uma instância do *framework* pode implementar os protocolos de comunicação utilizando diretamente *socket* ou utilizando HTTP.

5. Implementação

O *framework* de carregamento de mapas contextuais foi implementado na plataforma Android e testado tanto no emulador da plataforma como em dispositivos reais. O middleware sensível ao contexto foi implementado como um serviço Android e, portanto, independente do *framework*. Neste protótipo, utilizamos um proxy para um futuro serviço de contexto, implementado em outro trabalho e aproveitando as interfaces propostas em [da Rocha 2009]. Os componentes `Model` e `Controller` são classes tradicionais integradas à aplicação, enquanto que o componente `View` é implementado como uma *Activity*[2] Android. Devido à ausência de espaço, este artigo não discutirá as interfaces implementadas em cada componente.

Com o objetivo de facilitar a integração do *framework* com a implementação da API do Google Maps, a implementação da UI e do componente `View` adotou algumas das convenções dessa API. Desta forma, o conceito de camadas de mapas é implementado na forma de um objeto `Overlay` Android, fornecido pela API mencionada, assim como as operações comuns de navegação em mapas, como `setCenter(GeoPoint)`, `panEast`, `panNorth`, `setZoom(int)`, `zoomIn()` e `zoomOut()`.

Todas as mensagens internas trocadas entre os componentes e indicadas na figura 1 são implementadas como invocação direta de métodos de objetos, exceto as interações da UI com o componente `View`, que são implementadas como *Intents* Android.

A figura 2 ilustra a execução de uma aplicação com alguns pontos de referência, que também podem ser sobrepostos aos mapas. Quando a aplicação inicia um componente `View`, ela também inicializa o serviço de descoberta por meio do controlador sensível ao contexto, com o qual identifica novos serviços de mapas presentes na sua

[2]Componente de interface gráfica da plataforma Android

localização atual. Com essa informação, o componente `View` exibe na interface as camadas que representam cada tipo de mapa disponível, tipicamente associado a um diferente serviço de mapas. A figura 3 ilustra a sobreposição entre dois tipos de mapas e um conjunto de pontos de referência.

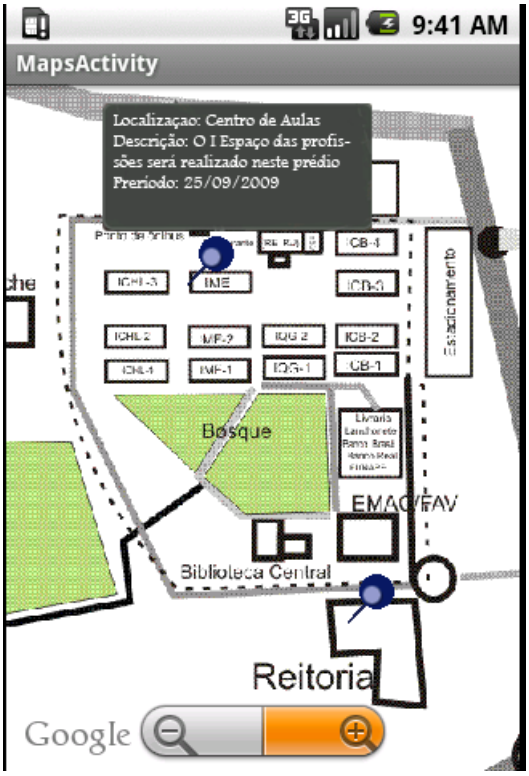


Figura 2. Exemplo de tela de uma aplicação exibindo mapas simbólicos

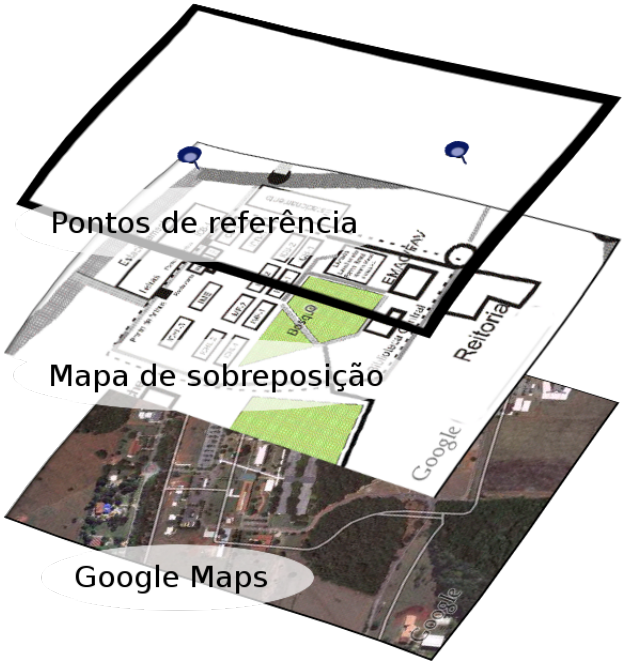


Figura 3. Exemplo de camadas

6. Avaliação

A avaliação foi feita a partir do desenvolvimento de três instâncias do *framework*. Todas as instâncias criadas usaram o Google Maps como provedor principal de mapas. As seguintes instâncias foram desenvolvidas: Instância Google Maps, onde as funções básicas do *framework* como navegação e zoom foram testadas, validando o modelo do *framework*; Instância Provedor de mapas simples bitmapeado, que responde a consultas por mapas simbólicos retornando uma imagem bitmap que será sobreposta aos mapas da base principal (Google Maps); E por último a Instância Provedor de Mapas Simbólicos, que responde consultas elaboradas de locais simbólicos com objetos de sobreposição aos mapas do Google Maps.

Os testes foram feitos em dois ambientes, a maquina virtual Dalvik (*Dalvik VM*) e o dispositivo móvel *Samsung Galaxy 5 I5500*, ambos executando a plataforma Android versão 2.1.

6.1. Instância Google Maps

O Google Maps fornece uma API nativa para manipulação de seus mapas no Android, estabelecendo seu próprio protocolo de comunicação e método de renderização. Essa API

foi usada para validar a estrutura de integração do *framework*, onde os mapas providos pelo Google Maps são usados como base da aplicação. Nessa instância não há uma base de mapas simbólicos de semântica geral, portanto não há sobreposição de mapas.

6.2. Instância Provedor de Mapas simples bitmapeado

A construção de um serviço de mapas é iniciada pela definição dos dados, valores e atributos utilizados e como estes são relacionados. Para o estudo de caso foram identificados dois principais atributos, a imagem estática da região e as coordenadas geográficas que delimitam a região referente ao mapa, relacionados por meio de um objeto que possua pelo menos esses dois atributos.

As informações de mapas contextuais foram associadas a um objeto serializável definido no `Model`, capaz de ser transmitido por meio de um canal de comunicação síncrona via socket. Este objeto possui um valor inteiro referente ao nível de zoom adequado, duas informações de posicionamento geográfico representando os pontos superior esquerdo e o inferior direito capazes de inferir a região representada pelo mapa contextual, por fim, três valores textuais, a primeira representando o nome do mapa, o segundo consiste no caminho de diretórios onde a imagem está salva na base de dados e a terceira é a informação da URL (Uniform Resource Locator) onde também pode ser encontrada a imagem do mapa contextual.

Outro componente presente no `Model` é um objeto que também representa as informações de um mapa contextual, porém este objeto é trabalhado apenas pelos componentes do *framework* associados a plataforma Android. Nele são encontrados as informações de nome e posições geográfica, tal como o objeto descrito anteriormente, e um atributo de valor Bitmap interpretado após o carregamento da imagem.

O servidor de mapas contextuais ou a base de mapas consiste em uma aplicação Java capaz de disponibilizar informações de mapas contextuais através de comunicação via socket, a partir de uma busca por posicionamento geográfico ou por nome (localização simbólica [Hu and Lee 2004]). A consulta é feita em uma lista de mapas, onde todos os elementos são percorridos para o casamento com a consulta. Devido a quantidade relativamente pequena de mapas, este método não causa impacto no desempenho do servidor.

O tratamento dado aos mapas obtidos pela base estática será de sobreposição aos mapas do Google Maps. Para isso é usado o próprio `MapView` disponibilizado pelo Google Maps. Este elemento consiste no principal elemento visual que será carregado no dispositivo para exibir as informações de mapas, e consistirá na visualização padrão da aplicação capaz de interagir com o usuário por meio de navegação e alteração do grau de detalhamento (*zoom in/out*) do mapa exibido.

A aplicação se comunica com servidor apenas quando for solicitado uma busca por mapas contextuais, recebendo uma lista de objetos com as informações dos mapas requisitados. Os mapas de sobreposição serão adquiridos através da URL dos objetos recebidos, que serão usados para requisitar a imagem do mapa no endereço especificado. E por fim a `View` atualiza a tela com a imagem obtida sobrepondo-a ao mapa do Google Maps fazendo ajustes necessário de zoom e posicionamento do centro do display.

6.3. Instância Provedor de Mapas simbólicos

Nesta instância temos uma base de mapas que responde a consultas por localização geográfica ou simbólica, assim como na instância anterior, porém o objeto resposta dessa é um arquivo KML[3], uma espécie de XML que descreve dados geográficos. Assim a aplicação é construída para manipular este arquivo e extrair as informações necessárias para fazer a sobreposição dos mapas.

As consultas por localização simbólica são encaminhas ao servidor em um objeto de texto que descreve a estrutura do local consultado [Durr and Rothermel 2003]. O servidor processa a consulta e retorna uma lista de arquivos KML com as informações suficientes para que a aplicação faça a sobreposição do mapa na tela.

7. Trabalhos Relacionados

Existem diversos *frameworks* e componentes para desenvolvimento de aplicações baseadas em mapas, principalmente para *Web*, onde essas aplicações são fortemente disseminadas. Citando dois desses *frameworks* que possibilitam o desenvolvimento para dispositivos móveis, que utilizam da J2ME (*Java 2 Platform, Micro Edition*), ou que são baseadas na plataforma Android temos, o Mobile Maps e a Google Maps API.

O Mobile Maps [Ericsson Labs] é um *framework* desenvolvido pela *Ericsson Labs*, que provê uma API para criação de aplicações baseadas em mapas para dispositivos móveis. A partir dos mapas obtidos do *TeleAtlas*[4] ou do *OpenStreetMaps*[5], este *framework* oferece uma API que implementa a interação com o usuário, sendo possível a navegação pelo mapa, operações de zoom, como também a criação de camadas (`Layer`), possibilitando a adição de elementos gráfico sobre o mapa.

O Google Maps API [Google Inc: Android Developers] é o mais popular *framework* para desenvolvimento de aplicativos baseados em mapas, tanto para *Web*, quanto para dispositivos móveis. Esta API oferece um objeto gráfico chamado de `MapView` responsável pela visualização de mapas providos pelo provedor do Google Maps através do protocolo HTTP. Utilizando localização baseada em IP ou *cellID*, a API infere o posicionamento geográfico do dispositivo por um serviço oferecido no servidor de mapas do Google.

Embora intensivamente utilizados, tais APIs são fortemente acopladas a um único serviço de mapas, assim como semântica de localização, impossibilitando a incorporação de outras fontes de mapas em uma aplicação que faça uso de localização simbólica. Em outras palavras, uma aplicação que faça uso de um desses serviços ou APIs fica fortemente dependente dos mapas descritos nas respectivas bases e, portanto, não é capaz de lidar com mapas de escopo limitado ou propósito específico.

8. Conclusões e Trabalhos Futuros

Este artigo apresentou um *framework* implementado em Android para o desenvolvimento de aplicações baseadas em mapas contextuais. Como próximos passos dessa pesquisa, integraremos o *framework* com um middleware sensível ao contexto baseado em CEP/esper

[3] `code.google.com/apis/kml/documentation/`

[4] http://www.teleatlas.com/index.htm

[5] http://www.openstreetmap.org/

e avaliaremos a instância de mapa contextual baseada em localização simbólica com o uso de sensores cricket de localização. Outro trabalho de pesquisa em andamento, é a aplicação de R-OSGi [Rellermeyer et al. 2007] para o carregamento dos serviços de mapas no *framework*.

Referências

da Rocha, R. C. A. (2009). *Context Management for Distributed and Dynamic Context-Aware Computing*. PhD thesis, Departamento de InformÃ¡tica, Pontificia Universidade Catolica do Rio de Janeiro, Rio de Janeiro.

de Oliveira Pereira, H. M. and da Rocha, R. C. A. (2010). Localizacao, mapas e contexto: Um framework para desenvolvimento de aplicações baseadas em mapas contextuais. In *II Simpósio Brasileiro de Computacao Ubiqua e Pervasiva (SBCUP)*.

Durr, F. and Rothermel, K. (2003). On a location model for fine-grained geocast. In Dey, A., Schmidt, A., and McCarthy, J., editors, *UbiComp 2003: Ubiquitous Computing*, volume 2864 of *Lecture Notes in Computer Science*, pages 18–35. Springer Berlin / Heidelberg.

Ericsson Labs. Mobile maps. Disponível na Internet em: <https://labs.ericsson.com/apis/mobile-maps/>, acesso em Abril, 2011.

Google Inc: Android Developers. Android developers: Location and maps. Disponível na Internet em: <http://developer.android.com/guide/topics/location/index.html>, acesso em Abril, 2011.

Hu, H. and Lee, D.-L. (2004). Semantic location modeling for location navigation in mobile environment. *Mobile Data Management, IEEE International Conference on*, 0:52.

Pradhan, S. (2000). Semantic location. *Personal and Ubiquitous Computing*, 4(4):213–216.

Rellermeyer, J. S., Alonso, G., and Roscoe, T. (2007). R-osgi: distributed applications through software modularization. In *Proceedings of the 8th ACM/IFIP/USENIX international conference on Middleware*, MIDDLEWARE2007, pages 1–20, Berlin, Heidelberg. Springer-Verlag.

SECOMU - XLI Seminário sobre Computação na Universidade

PREFÁCIO

A Sociedade Brasileira de Computação definiu os Grandes Desafios científicos da Computação para a próxima década, que exigirão esforços de pesquisa de longo prazo, por grandes equipes interdisciplinares, na solução de problemas de significativa relevância para o Brasil e para a humanidade. Em 2011, a discussão dos Grandes Desafios estará mais direcionada na importância dos mesmos no contexto da evolução da sociedade, visando diminuir as diferenças dos diferentes estratos da população no acesso à tecnologia, para propiciar uma maior participação da Computação no dia-a-dia de todas as pessoas.

O Seminário de Computação na Universidade (SECOMU) discute as questões políticas, sociais, econômicas e culturais associadas ao impacto que a Computação traz em áreas de grande relevância para o país. Essas questões são abordadas na forma de painéis que reúnem pesquisadores de Computação e áreas relacionadas, objetivando a discussão e reflexão de iniciativas e programas existentes, assim como na proposição em cooperação de soluções inovadoras e de longo prazo para problemas e desafios nacionais relevantes.

Este ano, o SECOMU aborda o tema "Computação para Todos: No Caminho da Evolução Social" do CSBC 2011, através da organização de quatro painéis que abordam aspectos sociais de grande importância e relevância para o país, são eles: programas e iniciativas de colaboração com outros países de língua portuguesa; programas e políticas nacionais e regionais de incentivo a pesquisa e inovação; e programas e iniciativas de inclusão digital. Um dos painéis será realizado em conjunto com o COMPUTEC 2011 - Computação e Mercado.

Paulo Roberto Freire Cunha
Uirá Kulesza
Coordenadores do SECOMU

COMISSÕES

Coordenação Geral

Paulo Roberto Freire Cunha, CIn-UFPE (coordenador geral)
Uirá Kulesza, DIMAp-UFRN (coordenador local)

SEMISH - XXXVIII Seminário Integrado de Software e Hardware

PREFÁCIO

Os Grandes Desafios (GD) da Computação definidos pela SBC têm guiado a programação e temática das últimas edições do Seminário Integrado de Software e Hardware (SEMISH). Este SEMISH 2011 não é exceção; os Grandes Desafios serviram de norte para que autores viessem a submeter trabalhos que estivessem explicitamente alinhados aos desafios definidos, como pode ser observado na chamada de trabalhos do SEMISH, amplamente divulgada na comunidade de Computação nacional.

Os 45 trabalhos submetidos para avaliação ao SEMISH foram observados, cada um, por pelo menos três especialistas em Computação, especialistas estes que compuseram o Comitê de Programa Técnico do SEMISH 2011. O Comitê é formado por representantes das Comissões Especiais (CEs) da SBC, bem como por renomados especialistas em Computação. Pelo seu trabalho voluntário e precioso, agradecemos aos membros do Comitê de Programa Técnico e revisores do SEMISH.

Após revisar os trabalhos submetidos, o Comitê de Programa Técnico passou a fase de seleção, onde 14 artigos foram aprovados para serem apresentados nas sessões técnicas do SEMISH, bem como serem publicados nos anais do CSBC 2011. Dado o expressivo número de submissões, i.e., 45 artigos, bons trabalhos acabaram não sendo selecionados. Acreditamos que os 14 trabalhos que constituem este SEMISH 2011 representam o estado-da-arte da Computação brasileira em relação aos Grandes Desafios definidos pela SBC. Neste contexto, gostaríamos de agradecer todos os autores que submeteram suas pesquisas ao SEMISH, e esperamos que os 14 trabalhos selecionados possam ser amplamente apreciados e debativos pela audiência do evento durante o CSBC 2011.

A programação do SEMISH é ainda complementada com quatro importantes palestras de pesquisados de renome internacional: Flavio Oquendo (University of South Brittany), Luiz Fernando Gomes Soares (PUC-Rio), William Feiereisen (Intel) e Jinesh Varia (Amazon). A estes palestrantes, fica também nosso agradecimento pela disposição e entusiasmo em compartilharem com as participantes do SEMISH 2011 seu precioso conhecimento.

Gostaríamos de agradecer também os coordenadores gerais do CSBC 2011, professores Martin Alejandro Musicante e Thaís Vasconcelos Batista, do DIMAp-UFRN, pela confiança em nosso trabalho e oportunidade em organizar esta edição do SEMISH. Por fim, mas não menos importante, deixamos um agradecimento especial a todos que vieram a participar do SEMISH 2011 nos dias 19 e 20 de julho. Esperamos que os trabalhos sejam frutíferos e inspiradores, e que resultem em parcerias importantes que avancem o estado-da-arte da Computação brasileira no contexto dos Grandes Desafios.

Lisandro Zambenedetti Granville (INF-UFRGS)
Nélio Alessandro Azevedo Cacho (ECT-UFRN)

COMISSÕES

Coordenação Geral

Lisandro Zambenedetti Granville, INF-UFRGS (coordenador geral)
Nélio Alessandro Azevedo Cacho, ECT-UFRN (coordenador local)

Comitê de Programa

Altigran Soares da Silva - UFAM
André Ponce de Leon F de Carvalho - ICMC-USP
Antônio Braga - UFMG
Artur Ziviani - LNCC
Celso Ribeiro - UFF
Clarisse de Souza - PUC Rio
Claudia Bauzer Medeiros - UNICAMP
Clodoveu Davis - UFMG
Eduardo Aranha - UFRN
Eduardo Figueiredo - UFMG
Fabio Protti - UFF
Fernando Castor - UFPE
Flavio Wagner - UFRGS
Geraldo Robson Mateus - UFMG
Germano Vasconcelos - UFPE
Gilberto Camara - INPE
Heloisa Camargo - UFSCar
Itana Maria de Souza Gimenes - UEM
José Palazzo Moreira de Oliveira - UFRGS
Leila Ribeiro - UFRGS
Lucia Drummond - UFF
Luciana Nedel - UFRGS
Luciano Paschoal Gaspary - UFRGS
Luis Lamb - UFRGS
Luiz Satoru Ochi - UFF
Marcel Oliveira - UFRN
Marcelo Finger - USP
Marcilio de Souto - UFPE
Marcos Gonçalves - UFMG
Marta Mattoso - UFRJ
Nelson Fonseca - UNICAMP
Nelson Rosa - UFPE
Renata Galante - UFRGS
Renato Tinós - FFCLRP-USP
Ricardo Torres - UNICAMP
Roberta Coelho - UFRN
Sergio Bampi - UFRGS
Soraia Musse - PUCRS
Teresa Ludermir - UFPE
Uirá Kulesza - UFRN
Vanessa Braganholo - UFF

Revisores

Adriana Vivacqua- DCC-IM/UFRJ
Alfredo Goldman - IME - USP
Altigran Soares da Silva - Universidade Federal do Amazonas
André Campos - UFRN
André Ponce de Leon F de Carvalho - ICMC-USP/S.Carlos
Antônio Braga - UFMG
Antonio Carlos Beck F. - Universidade Federal do Rio Grande do Sul
Artur Ziviani - LNCC
Carla Freitas - UFRGS
Carlos Morimoto - USP
Celso Ribeiro Instituto de Computação - UFF
Clarisse de Souza - PUC-Rio
Claudia Bauzer Medeiros - IC-UNICAMP
Clodoveu Davis - UFMG
Daniela Trevisan - Universidade Federal Fluminense
Débora Muchaluat-Saade - UFF
Eduardo Aranha - Universidade Federal do Rio Grande do Norte
Eduardo Figueiredo - Federal University of Minas Gerais (UFMG)
Eduardo Ogasawara - CEFET-RJ
Erika Cota - Universidade Federal do Rio Grande do Sul
Esteban Clua - Universidade Federal Fluminense
Fabio Protti - Universidade Federal Fluminense
Fernando Marson - Universidade do Vale do Rio dos Sinos
Flavia Delicato- UFRN
Flavio Wagner - UFRGS
Flávio Soares Corrêa da Silva - Universidade de São Paulo
Francisco Assis Moreira do Nascimento - Universidade Federal do Rio Grande do Sul
Geraldo Robson Mateus - UFMG
Germano Vasconcelos - UFPE
Gibeon Júnior - UFRN
Gilberto Camara - INPE
Gustavo Henrique Lima Pinto - Federal University of Pernambuco
Heloisa Camargo - UFSCar
Igor Moraes - Universidade Federal Fluminense
Itana Maria de Souza Gimenes - Universidade Estadual de Maringá
Jair Leite - Universidade Federal do Rio Grande do Norte
Jonice Oliveira - DCC/IM/UFRJ
José Palazzo Moreira de Oliveira - UFRGS
Jose Reginaldo Carvalho - Universidade Federal do Amazonas
Juliana Saraiva - Federal University of Pernambuco
Leandro Fernandes - Universidade Federal Fluminense (UFF)
Leônidas Brandão - Universidade de São Paulo
Leila Ribeiro - Universidade Federal do Rio Grande do Sul
Leonardo da Costa - UNIVERSIDADE FEDERAL FLUMINENSE
Leonardo Murta - UFF
Lisandro Zambenedetti Granville - UFRGS

Loana Nogueira - UFF
Lucia Drummond - Universidade Federal Fluminense
Luciana Nedel - UFRGS
Luciano Paschoal Gaspary - UFRGS
Luis Lamb - UFRGS - Instituto de Informática
Luiz Filipe Vieira - UFMG
Marcel Oliveira - UFRN
Marcelo Pimenta - Universidade Federal do Rio Grande do Sul
Marcelo Santos - IFSudesteMG
Marcilio de Souto - Universidade Federal de Pernambuco
Marcos Gonçalves - Universidade Federal de Minas Gerais
Marcos Vieira - UFMG
Marta Mattoso - COPPE/UFRJ
Márcio Kreutz - Universidade Federal do Rio Grande do Norte
Nelson Fonseca - UNICAMP
Nelson Rosa - UFPE
Pablo Alsina - Universidade Federal do Rio Grande do Norte
Pablo Luiz A. Munhoz - Universidade Federal Fluminense
Raimundo Barreto - Universidade Federal do Amazonas
Renata Galante - Universidade Federal do Rio Grande do Sul
Renato Tinós - USP
Ricardo Farias - UFRJ
Ricardo Torres- Unicamp
Roberta Coelho - Universidade Federal do Rio Grande do Norte
Rossana Baptista Queiroz - Pontifícia Universidade Católica do Rio Grande do Sul
Sergio Bampi - UFRGS
Soraia Musse - PUCRS
Taisy Weber - UFRGS
Tayana Conte - DCC/UFAM
Teresa Ludermir - Universidade Federal de Pernambuco
Uirá Kulesza - UFRN - Universidade Federal do Rio Grande do Norte
Vanessa Braganholo - UFF
Vinícius Cassol - Pontifícia Universidade Católica do RS

ARTIGOS

GreenWeb: Melhorando a Qualidade da Informação na Web 2.0

Jussara M. Almeida[1], Marcos A. Gonçalves[1], Raquel O. Prates[1], Daniel Hasan[1], Dílson Guimarães[1], Diogo R. de Oliveira[1], Fabiano Belém[1], Flavio Figueiredo[1], Hendrickson Langbehn[1], Henrique Pinto[1], Raquel Lara[1], Saulo Ricci[1], Fabrício Benevenuto[2]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais
Av. Antônio Carlos 6627, Prédio do ICEx, Pampulha, Belo Horizonte, MG

[2]Departamento de Ciência da Computação – Universidade de Ouro Preto
Campus Universitário – Morro do Cruzeiro, Ouro Preto, MG

{jussara,mgoncalv,rprates,hasan,dilsonag,renno,fmuniz,flaviov,reiter,
hpinto,raqlara,saulomrr}@dcc.ufmg.br, fabricio@iceb.ufop.br

Abstract. *This paper introduces the GreenWeb framework, which aims at improving the quality of information on the Web 2.0. GreenWeb has 4 main components: (1) metrics and methods to estimate information quality; (2) profiles of users' interests and system usage; (3) strategies to communicate content quality to users; and (4) methods to detect and reduce low quality content as well as to promote high quality content. In this paper, we present the challenges and solutions already developed for each such component.*

Resumo. *Este artigo apresenta o arcabouço GreenWeb, que visa melhorar a qualidade da informação na Web 2.0. O GreenWeb consiste de 4 componentes principais: (1) métricas e métodos para estimar qualidade da informação; (2) perfis de uso e de interesse dos usuários; (3) estratégias para comunicar a qualidade de um conteúdo para os usuários; e (4) métodos para detectar e reduzir conteúdo de baixa qualidade, assim como promover conteúdo de mais alta qualidade. Neste artigo, nós apresentamos os desafios e as soluções já desenvolvidas para cada componente.*

1. Introdução

A Web 2.0 tem como ênfase facilitar a interação e a colaboração entre usuários através da criação de comunidades virtuais e do estabelecimento de plataformas de distribuição de conteúdo. Ela é marcada por um maior envolvimento dos usuários que passaram a atuar não somente como consumidores, mas também como produtores e provedores de conteúdo [Boll 2007]. Tal conteúdo, muitas vezes criado de forma colaborativa e em diferentes tipos de mídia (p.ex: áudio, video, texto), é frequentemente chamado de *mídia social*. A mídia social é tipicamente composta por um *objeto*, que representa o principal veículo de disseminação de informação na aplicação (p.ex: um vídeo no YouTube, um artigo na Wikipedia[1]), e possivelmente uma série de atributos associados.

Tipicamente, as aplicações da Web 2.0 não impõem nenhum controle editorial sobre o conteúdo gerado pelos usuários e logo não fornecem nenhuma garantia de

[1] http://www.youtube.com e http://www.wikipedia.com, respectivamente

qualidade da informação disponibilizada. Embora o conceito de "qualidade da informação" seja intuitivo, uma definição explícita do mesmo é um desafio. Nós aqui consideramos um conteúdo com qualidade se a informação associada a ele é *relevante, atende as necessidades e/ou agregue valor a serviços e aplicações para um conjunto de usuários*. Logo, o conceito de qualidade vai além de aspectos sintáticos e semânticos do conteúdo e incorpora aspectos relacionados às necessidades informacionais dos usuários e características específicas dos serviços e aplicações. Por exemplo, a qualidade de um conteúdo pode ser avaliada sob a perspectiva do seu potencial como fonte de dados para suportar serviços de informação, tais como busca, recomendação, e propaganda. Neste contexto, o foco principal está nos atributos textuais associados aos objetos (p.ex: *tags*), dado que, a despeito da existência de técnicas de recuperação de informação multimídia, a maioria dos serviços ainda utiliza apenas estes atributos textuais [Boll 2007].

Estudos recentes indicam que existe uma grande quantidade de *lixo informacional* em aplicações da Web 2.0 [Suchanek et al 2008, Figueiredo et al 2009], possivelmente devido à facilidade e liberdade com que usuários criam e disponibilizam conteúdo nestas aplicações. Tal liberdade abre oportunidade para ações maliciosas e/ou oportunistas, que resultam na introdução de conteúdo de baixa qualidade (i.e., conteúdo poluído ou simplesmente poluição) no sistema. Exemplos incluem vandalismo na Wikipédia [Potthast et al 2010] e diferentes formas de *spamming* [Benevenuto et al 2009a, Koutrika et al 2008]. Conteúdo poluído incorre em custos extras para os administradores de sistemas, afeta a eficácia de serviços de informação e compromete a paciência do usuário e sua satisfação com o sistema. Isto porque os usuários não podem facilmente identificar a poluição sem ter contato com ela, o que leva ao consumo de recursos do sistema (p.ex: largura de banda [Benevenuto et al 2009a]).

Neste contexto, este artigo apresenta a proposta de um arcabouço, denominado *GreenWeb,* que visa fundamentar o desenvolvimento de técnicas e ferramentas para melhorar a qualidade de informação na Web 2.0, contribuindo para agregar valor a várias aplicações e serviços de informação. O GreenWeb foca em três pilares principais: (1) reduzir a poluição de conteúdo, (2) aumentar a qualidade da informação disponibilizada aos usuários, e (3) manter uma relação custo-benefício favorável para usuários e administradores de sistemas. Tendo aplicações e serviços da Web 2.0 como alvo, o GreenWeb enfatiza um ambiente que vem se mostrando promissor para a disseminação de informação, para a interação e a colaboração entre as pessoas e, em última instância, para a troca de conhecimentos e experiências, o que contribui diretamente para o crescimento da sociedade.

O GreenWeb aborda aspectos relacionados a quatro dos Grandes Desafios da Pesquisa em Computação, definidos pela Sociedade Brasileira de Computação [Carvalho et al 2006]. Em sua essência, ele aborda questões relacionadas ao *acesso universal ao conhecimento* (4º desafio), uma vez que o acesso à informação de maior qualidade pode estimular a participação dos usuários nos processos de produção e de uso do conhecimento. Ele também trata de aspectos relativos à *gestão de informação, sob a perspectiva de qualidade, em grandes bases de dados multimídia* (1º desafio), explorando várias técnicas de *modelagem computacional para representar as complexas interações entre usuários e entre os usuários e o sistema* (2º desafio). Por fim, ao abordar a detecção e o combate a ações maliciosas e oportunistas, ele também visa o *desenvolvimento de sistemas seguros e escaláveis* (5º desafio).

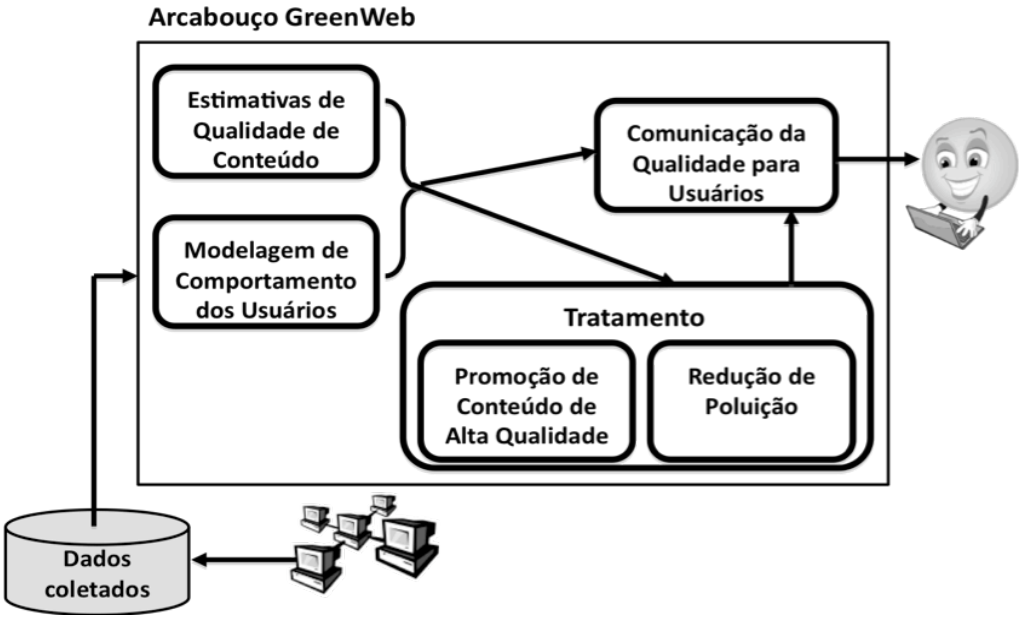


Figura 1: Componentes do Arcabouço GreenWeb

A seguir, a Seção 2 apresenta uma visão geral do arcabouço GreenWeb, seus principais componentes e desafios. As Seções 3 a 6 descrevem, mais detalhadamente, as soluções já desenvolvidas para estes componentes, como elas se posicionam frente ao estado-da-arte e os principais resultados já obtidos. A Seção 7 descreve dois protótipos desenvolvidos, enquanto conclusões e próximos passos são apresentados na Seção 8.

2. O Arcabouço GreenWeb

O arcabouço GreenWeb tem por objetivo fundamentar o desenvolvimento de soluções para agregar valor a diferentes serviços de informação da Web 2.0 a partir do combate à poluição e da promoção de conteúdo de mais alta qualidade. Para atingir esse objetivo, o GreenWeb é constituído dos seguintes 4 grandes componentes, mostrados na Figura 1:

1. **Estimativas de Qualidade do Conteúdo**: composto por técnicas e métricas para estimar a *qualidade da informação* associada a um dado conteúdo.

2. **Perfis de Uso e Interesse do Usuário**: composto por técnicas e modelos que representam aspectos relevantes sobre quem é o usuário e sobre o uso que ele faz do sistema. Informações sobre o perfil do usuário (p.ex: suas características e interesses) podem facilitar a identificação de suas necessidades informacionais e do conteúdo mais adequado (i.e., com maior qualidade) ao seu perfil. A análise de como os usuários interagem com o sistema (p.ex: as funcionalidades utilizadas, freqüência de uso, relacionamentos estabelecidos, etc.) pode ajudar na detecção de usuários maliciosos/oportunistas, que introduzem poluição no sistema [Benevenuto et al 2009a]. O conjunto de usuários analisados dependerá dos dados disponíveis (veja discussão abaixo). Por exemplo, dados coletados a partir da API de uma aplicação tipicamente são restritos a usuários que possuem conta no sistema. Por outro lado, dados coletados a partir de outras fontes, tais como servidores *proxy* ou mesmo de *logs* de acesso mantidos pela aplicação alvo podem viabilizar o estudo de uma população maior de usuários, incluindo aqueles que não têm conta no sistema.

3. **Estratégias de Tratamento**: este componente é dividido em dois subcomponentes principais que abordam o problema alvo sob perspectivas complementares: (1) Promoção de Conteúdo de Mais Alta Qualidade e (2) Detecção e Redução da Poluição no Sistema. Cada subcomponente é composto por mecanismos e técnicas voltados para atingir a estratégia definida.

4. **Comunicação da Qualidade para o Usuário**: composto por modelos e técnicas que permitam a comunicação, através da interface, da qualidade da informação sendo apresentada, apoiando o usuário na identificação de conteúdo de alta/baixa qualidade.

Os dois primeiros componentes, conjuntamente, fornecem subsídios para o desenvolvimento de soluções para promover conteúdo de mais alta qualidade e para detectar e reduzir a poluição no sistema (componente 3). As informações produzidas por todos estes componentes, por sua vez, podem subsidiar a definição, pelo componente 4, de quais aspectos devem ser comunicados a diferentes tipos de usuários. Por exemplo, o objetivo pode ser informar o usuário final sobre a qualidade de um dado conteúdo (p.ex: indicação de artigos de qualidade na Wikipédia), ou então alertar os administradores de um serviço sobre usuários fazendo uso malicioso ou oportunista do sistema (p.ex: indicar potenciais *spammers* para os administradores do YouTube).

Esta é uma descrição em alto nível do arcabouço GreenWeb, focada nos macrocomponentes principais. Cada macrocomponente é decomposto em vários subcomponentes. Por exemplo, muitas soluções propostas para os macrocomponentes exploram padrões identificados em dados coletados da aplicação alvo. Assim, faz-se necessário um subcomponente responsável pela coleta, processamento e armazenamento de dados em algum repositório (p.ex: um banco de dados).

O desenvolvimento dos componentes do GreenWeb enfrenta vários desafios, entre eles:

• Coleta, armazenamento e processamento de grandes volumes de dados: a identificação de padrões típicos de uso e perfis de usuário, bem como o desenvolvimento de técnicas para estimar qualidade de conteúdo dependem da análise de dados, obtidos de diferentes aplicações. Tais dados são também essenciais para direcionar o desenvolvimento de soluções de tratamento e de comunicação assim como para suportar a avaliação dos mesmos.

• Padrões dinâmicos e heterogêneos: tipicamente, a forma como as pessoas utilizam os sistemas e os conteúdos criados e acessados por elas variam conforme suas preferências pessoais e foco de uso. Tais padrões também tendem a variar com o tempo. Logo, as soluções desenvolvidas devem lidar com alta heterogeneidade e dinamicidade. As variações temporais implicam na necessidade de realizar coletas freqüentes a fim de manter o repositório de dados atualizado;

• Diferentes perspectivas de qualidade: a qualidade da informação associada a um conteúdo pode variar dependendo do usuário (grupo de usuários) alvo, da aplicação ou serviço (classe de aplicações ou serviços), ou ainda do uso feito da informação. Fundamentalmente, ela depende da necessidade informacional do usuário, que, por sua vez, pode variar dependendo do tipo de aplicação e serviço. Além disto, características da aplicação ou serviço tais como o tipo de mídia usado para disseminar informação e o seu público alvo podem também afetar a percepção de qualidade. Logo, as melhores estratégias de promoção de conteúdo, redução de poluição e comunicação de qualidade

podem depender da aplicação/serviço alvo, suas funcionalidades e detalhes da interface. Mais ainda, usuários da Web 2.0 cada vez mais utilizam serviços (p.ex: busca) para recuperação de informação e organização de seu conteúdo. Assim, a eficácia destes serviços no atendimento das necessidades dos usuários é também importante. Logo, a qualidade de um conteúdo pode ainda ser analisada sob a perspectiva do seu potencial como fonte de dados para suportar serviços de informação mais eficazes. Neste caso, ela pode depender do tipo de serviço: por exemplo, um conteúdo pode ter qualidade para suportar um serviço de busca, mas não um serviço de classificação de conteúdo [Almeida et al 2010]. Em suma, as diferentes perspectivas de qualidade implicam na sua alta dependência do domínio e contexto considerados. Logo, não é possível desenvolver uma solução única para todos os contextos, exigindo, pois, instanciações.

Logo, nós analisamos a aplicabilidade do arcabouço GreenWeb desenvolvendo soluções específicas para diferentes contextos. Vale ressaltar que nem sempre todos os componentes precisam ser instanciados, o que mostra a flexibilidade do arcabouço. Isto também aponta para a necessidade de uma investigação sobre quais componentes são de maior interesse, o que depende do contexto específico, da perspectiva de qualidade sob análise e da relação custo-benefício associada ao desenvolvimento. A seguir, apresentamos uma breve descrição das soluções já desenvolvidas para alguns contextos específicos. Em particular, discutimos soluções para estimar qualidade da informação (Seção 3), comunicar qualidade (Seção 4), promover conteúdo de qualidade (Seção 5) e reduzir poluição (Seção 6). Os modelos de perfis e padrões de uso já desenvolvidos foram aplicados na detecção de usuários poluidores, sendo, pois, discutidos na Seção 6. Apesar da existência de soluções alternativas para alguns dos problemas específicos discutidos a seguir, nós não temos ciência de nenhuma proposta de arcabouço abordando, de forma unificada, as várias perspectivas do problema de melhorar a qualidade da informação na Web 2.0. Esta é a principal contribuição deste artigo.

3. Estimativas de Qualidade da Informação

Estimar a qualidade da informação associada a um dado conteúdo é uma tarefa complexa pelo alto grau de subjetividade e pela inevitável necessidade de considerar aspectos relativos tanto aos usuários quanto à aplicação e ao contexto do estudo. Nesta seção, nós discutimos as soluções propostas para dois contextos específicos.

3.1. Qualidade de Atributos Textuais para Recuperação de Informação

Serviços de informação na Web 2.0 exploram majoritariamente atributos textuais como fontes de dados [Boll 2007]. Entretanto, embora existam vários estudos sobre os padrões de uso de *tags* [Heymann et al 2010, Santos-Neto et al 2010, Sigurbjornsson and van Zwol 2008] e sua qualidade para suportar busca, recomendação e classificação de objetos [Clements et al 2010, Schenkel et al 2008, Ramage et al 2009], os resultados obtidos não apontam um consenso. Enquanto alguns concluem que *tags* têm boa qualidade [Bischoff et al 2008], outros evidenciam problemas como *tag spamming* [Koutrika et al 2008] e uso frequente de termos sem significado ou com múltiplos significados [Suchanek et al 2008], que impactam negativamente os serviços de informação. Mais ainda, a maioria dos estudos anteriores focou apenas em *tags*, negligenciando o uso potencial de outros atributos, tais como título e descrição.

Considerando o foco em serviços de informação, nós argumentamos que um atributo textual de alta qualidade deve: 1) conter uma quantidade de conteúdo suficiente para ser útil; 2) prover uma boa descrição do conteúdo, o que é importante para serviços que exploram a semântica dos objetos (p.ex: recomendação); e 3) poder distinguir o objeto de outros para tarefas como separar os objetos em classes semânticas ou em níveis de relevância para uma dada consulta. Embora cada um destes três aspectos – quantidade de conteúdo, poder descritivo e poder discriminativo – esteja relacionado à qualidade de um atributo, eles não são igualmente importantes para todos os serviços [Almeida et al 2010]. Por exemplo, um bom poder discriminativo é importante para serviços de classificação, enquanto um bom poder descritivo pode ser mais importante para serviços de recomendação. Além disto, alguns serviços, tais como classificação, podem se beneficiar mais da presença de uma maior quantidade de conteúdo.

Assim, em [Figueiredo et al 2009] nós realizamos uma extensa caracterização da qualidade, considerando os três aspectos acima, de 4 atributos – título, *tags*, descrição e comentários – em 4 aplicações – YouTube, YahooVideo, LastFM e CiteULike[2]. Nossa análise foi feita em amostras com mais de 200.000 objetos (e seus atributos) coletados de cada aplicação. A quantidade de conteúdo foi estimada pelo número de termos distintos presentes em cada atributo associado a cada objeto analisado. Para estimar os poderes descritivo e discriminativo, optamos pelo uso de métricas heurísticas que, apesar de aproximadas e invariavelmente conterem limitações, podem ser computadas a baixo custo em grandes bases de dados. As heurísticas usadas são adaptações de um modelo de recuperação de informação em páginas Web estruturadas [Moura et al 2010].

Nós estimamos o poder descritivo de um termo *t* contido em um atributo *f* de um objeto *o* pelo *espalhamento* de *t* em *o*, definido como o número de atributos associados a *o* que contêm *t*. O poder descritivo de *f* é estimado pelo espalhamento médio de todos os termos de *f*. A intuição é que termos que aparecem em vários atributos associados ao mesmo objeto têm uma maior chance de serem relacionados ao seu conteúdo. Por exemplo, se o termo "Sting" aparece em 4 dos atributos de um objeto (espalhamento = 4), há uma alta chance de que ele seja relacionado ao famoso cantor.

Para estimar o poder discriminativo de um termo *t* contido em um atributo *f* de um objeto *o*, foi proposta a heurística Freqüência Inversa nos Atributos (FIA), baseada na métrica IDF, amplamente usada em recuperação de informação [Baeza-Yates and Ribeiro-Neto 2011]. FIA estima o poder discriminativo de *t* pelo inverso da freqüência de *t* em todas as instâncias do atributo *f* na coleção de objetos. A intuição é que termos que ocorrem em muitas instâncias de um dado atributo são pouco discriminativos. Por exemplo, a ocorrência do termo "music" no título de um vídeo do YouTube é pouco discriminativa se o mesmo ocorre nos títulos de vários outros vídeos.

Os principais resultados da caracterização são: (1) todos os atributos, **exceto título**, estão ausentes (i.e., com nenhum conteúdo), em uma fração não desprezível dos objetos coletados e, logo, podem **não** ser eficazes como fontes únicas de dados, já que, neste caso, os serviços não atingiriam muitos objetos; (2) considerando atributos não vazios, atributos criados e editados colaborativamente tendem a conter mais conteúdo se comparados àqueles editados somente pelo usuário que criou o objeto; (3) título e *tags*, tipicamente com menos conteúdo, têm, em geral, melhores poderes descritivo e

[2] http://video.yahoo.com, http://last.fm , http://www.citeulike.org, respectivamente.

discriminativo, seguidos de descrição e comentários; (4) todos os atributos contêm uma grande quantidade de termos sem significado (lixo) ou então com muitos significados.

As métricas desenvolvidas assim como os resultados da caracterização têm norteado o desenvolvimento de serviços de informação mais eficazes. Em [Figueiredo et al 2009], nós mostramos o uso destas métricas na avaliação da qualidade dos atributos para suportar classificação automática de objetos, concluindo que *tags* é o melhor atributo isolado graças ao seu bom poder discriminativo e quantidade de conteúdo razoável. Também concluímos que a combinação de múltiplos atributos pode trazer benefícios, uma vez que eles tendem a contribuir com diferentes informações sobre o objeto. Nós também temos explorado as métricas propostas na proposição de mecanismos de recomendação mais eficazes, conforme será discutido na Seção 5.

3.2. Qualidade de Artigos da Wikipédia

A estimativa da qualidade de artigos na Wikipédia e em outros ambientes colaborativos similares são essenciais para garantir a confiança do leitor no conteúdo ao qual ele está sendo exposto. Apesar de estudos indicando que certos artigos da Wikipédia têm uma qualidade similar àqueles da Enciclopédia Britânica [Giles 2005] e de esforços no sentido de definir critérios qualitativos para estimar essa qualidade [Dondio et al 2006, Santos e Prates 2010], tais soluções são baseadas em análise manual, e portanto não escalam frente ao volume e à velocidade com que o conteúdo é atualizado. Assim, soluções automáticas para produzir estimativas de qualidade [Dondio et al 2006, Rassbach et al 2007] são necessárias. Tais estimativas podem ser usadas como indicadores de documentos que necessitam revisão, para identificar vandalismo ou para recomendar artigos baseados em sua qualidade estimada.

Para abordar este problema, nós propusemos um método automático para estimativa de qualidade, tratando-o com um problema de *regressão* [Dalip et al 2009]. Ou seja, nós estimamos a qualidade dos artigos na Wikipédia como um valor numa escala contínua de qualidade, fazendo uso de regressão baseada em Máquinas de Vetores de Suporte (*Support Vector Machines*) [Vapnik 1995]. Nossa principal contribuição nesse trabalho foi um estudo detalhado de várias características dos artigos como fontes de evidência e seu impacto na estimativa da qualidade. As seguintes características, algumas das quais foram propostas por nós, foram consideradas:

- Características associadas à revisão do artigo, tais como: número de revisões, quantidade de mensagens de discussão, número de revisões feitas por usuários casuais e por usuários especialistas, estabilidade das revisões etc.
- Características de rede: número de *links* de entrada e saída, *PageRank* [Baeza-Yates and Ribeiro-Neto 2011], coeficiente de agrupamento, etc.
- Características de texto, que por sua vez foram divididas em quatro subgrupos:
    - Características relativas ao tamanho do artigo;
    - Características relativas à estrutura do artigo, tais como: número de seções, número e cobertura das citações, tamanho médio das seções;
    - Características de estilo, tais como: tamanho do maior e do menor parágrafo, uso de pronomes, advérbios;
    - Características de facilidade de leitura, baseadas no número, tamanho e

distribuição de palavras, sentenças e sílabas, usadas para estimar o grau de educação necessário para se entender um artigo.

Experimentos realizados com uma amostra da Wikipédia usando uma escala de qualidade proposta pelo próprio sistema demonstraram que o uso do método de aprendizado proposto junto com o conjunto de características analisadas apresentou melhores resultados do que as melhores abordagens disponíveis na literatura [Dondio et al 2006, Rassbach et al 2007]. As características relativas à estrutura do texto foram as mais eficazes, sendo também as mais fáceis de computar. Os melhores resultados foram obtidos combinando essas características com as de revisão.

4. Comunicação da Qualidade aos Usuários

Como discutido na Seção 2, a qualidade de um conteúdo depende do contexto no qual o usuário fará uso da informação associada. Nesta seção, apresentamos a investigação feita para o contexto específico da Wikipédia. Propostas anteriores de interfaces que apóiam os usuários na sua inferência sobre a qualidade do artigo [Pirolli et al 2009, Krieger et al 2009, Chevalier et al 2010] focam em uma melhor visualização de informações já disponibilizadas pela Wikipédia, tais como histórico, discussões, tamanho do artigo ou da página de discussão. Em contraste, a nossa proposta é utilizar informações novas, tais como as características discutidas na Seção 3.2.

Embora a análise das características propostas na Seção 3.2 tenha tido bons resultados, o nosso objetivo não é filtrar ou classificar conteúdo, mas ser capaz de informar ao usuário qual a qualidade do artigo sendo acessado, para que ele possa decidir se ele atende ou não suas necessidades. Assim, para gerar uma proposta da comunicação a ser feita, a primeira etapa envolveu uma análise semiótica da Wikipédia para identificar se (e quais) considerações sobre qualidade dos artigos são feitas pelo sistema, e como estas são apresentadas aos usuários. A partir desta análise, identificou-se que a Wikipédia adota 9 estratégias que visam obter (ou encorajar) artigos de maior qualidade [Santos e Prates 2010]. As estratégias são classificadas em 2 categorias: (1) ações tomadas pelos administradores da Wikipédia e (2) ações disponíveis aos usuários para que melhorem a qualidade do conteúdo. Identificou-se também vários problemas na forma em que estas estratégias são apresentadas aos usuários.

No passo seguinte, investigou-se na literatura propostas de indicadores quantitativos e qualitativos relativos a artigos da Wikipédia. A partir desta investigação, fez-se uma proposta da comunicação a ser feita aos usuários sobre a qualidade que consiste de **indicadores quantitativos**, que permitem que se tenha sempre uma avaliação atualizada de cada artigo. No entanto, estes indicadores apenas não são suficientes, pois eles se baseiam normalmente em aspectos estruturais do texto ou da interação sobre o texto, e o significado associado a ele pode não ser único em alguns contextos. Por exemplo, a cobertura de um artigo (distribuição de citações ao longo artigo) considera que quanto mais referências e melhor distribuídas, melhor a qualidade do artigo. No entanto, a qualidade das referências citadas não é considerada. Mais ainda, pode ainda não haver muitas referências para um assunto inovador, o que não implica que o texto tenha baixa qualidade. Assim, para cada indicador quantitativo, deve-se ter também **explicações qualitativas** associadas a ele. Estas explicações devem no mínimo definir o aspecto sendo considerado (e.g. fator de cobertura) e o impacto esperado disso na qualidade do artigo. Por fim, devem ser disponibilizadas **visualizações** que permitam uma visão mais detalhada ou mesmo complementar ao indicador quantitativo. Por

exemplo, para o caso do fator cobertura, pode-se mostrar visualmente como está a cobertura de cada seção. Vale ressaltar que a proposta foi feita para a Wikipédia, mas acredita-se que ela seja válida para outras enciclopédias colaborativas.

A proposta foi implementada no protótipo do GreenWiki (ver Seção 7). Foi feita uma primeira avaliação deste protótipo em ambiente controlado, incluindo observação da interação de 9 usuários com o sistema e entrevistas sobre suas experiências de uso [Pereira 2011]. A avaliação mostrou que os usuários conseguiram entender e utilizar sem dificuldades o painel disponibilizado. Além disso, a partir da interação, os usuários passaram a ter uma maior preocupação e uma visão mais crítica sobre aspectos de qualidade. Eles perceberam que a métrica quantitativa apenas não seria suficiente, pois poderia gerar falsos positivos, e que a informação complementar (i.e., explicações qualitativas e visualizações) era relevante para a avaliação sobre a qualidade do artigo.

5. Promoção de Conteúdo de Alta Qualidade

A promoção de conteúdo de qualidade pode ser abordada de várias maneiras. Serviços de busca podem ser otimizados para levar em consideração estimativas de qualidade na ordenação dos resultados de uma consulta, enquanto métodos de recomendação de conteúdo podem incluir métricas de qualidade como um de seus critérios. Nosso foco, até então, tem sido na recomendação de *tags* de qualidade. Este foco é motivado por: (1) *tags* são amplamente exploradas por vários serviços de informação, e 2) nossa caracterização da qualidade de atributos textuais apontam *tags* como um atributo promissor para esta tarefa (vide Seção 3.1). Logo, o nosso objetivo é desenvolver métodos para sugerir *tags* de qualidade para um dado objeto (i.e., termos relacionados ao seu conteúdo), visando melhorar a qualidade deste atributo e, indiretamente, a eficácia de serviços que dele dependam.

Os métodos de recomendação de *tags* existentes exploram tipicamente [Lipczak et al 2009, Menezes et al 2010, Sigurbjornsson and van Zwol 2008]: (1) regras de associação para inferir padrões de co-ocorrência de termos com *tags* previamente associadas ao objeto alvo; (2) termos extraídos de múltiplos atributos e (3) métricas de qualidade (p.ex: freqüência, entropia) para filtrar termos irrelevantes e promover termos com maior qualidade. Entretanto, a maioria dos métodos existentes explora no máximo duas destas três dimensões. Em [Belém et al 2010], nós desenvolvemos soluções que exploram as três dimensões conjuntamente. Nós estendemos métodos baseados em padrões de co-ocorrência para incluir tanto *tags* previamente atribuídas aos objetos quanto termos extraídos de outros atributos (título e descrição). Todos estes termos são então ordenados quanto à qualidade (ou relevância) para a tarefa de recomendação. Para tanto, utilizamos várias métricas heurísticas que tentam capturar a qualidade de um termo para um objeto alvo. O problema de recomendação então se reduz a projetar uma função que combina as métricas para ordenar os termos candidatos por qualidade.

As funções desenvolvidas são extensões de soluções disponíveis na literatura [Menezes et al 2010, Sigurbjornsson and van Zwol 2008], que se diferenciam por incluir métricas de poder descritivo, particularmente a métrica espalhamento. Avaliamos as soluções propostas, em um total de oito estratégias, utilizando bases de dados reais coletadas do YouTube, YahooVideo e LastFM. Assim como em trabalhos anteriores, nossa avaliação foi automatizada, utilizando parte das *tags* já atribuídas ao objeto alvo como gabarito: apenas termos do gabarito são considerados relevantes.

A Tabela 1 mostra alguns dos resultados obtidos, em termos da precisão nas 5 primeiras posições da ordenação, ou seja, em termos da fração dos termos nas 5 primeiras posições das recomendações que foram considerados relevantes conforme gabarito. A tabela mostra resultados médios e intervalos de confiança de 95%. Sum+ [Sigurbjornsson and van Zwol 2008], LATRE [Menezes et al 2010] e CTTR [Lipczak et al 2009], soluções consideradas estado-da-arte, exploram um subconjunto das três dimensões mencionadas acima. Mostramos resultados apenas para as 2 melhores heurísticas, LATRE+TS e SUM+TS, omitindo as demais por questões de espaço. LATRE+TS, estende LATRE, baseado somente em padrões de co-ocorrência, para incluir a métrica espalhamento (*Term Spread* ou TS*)* e também para extrair termos candidatos de múltiplos atributos textuais. Sum+TS estende o Sum+, baseado em co-ocorrência e em algumas métricas de qualidade, de forma similar. Note que a nossa melhor heurística produz melhorias de até 26% sobre o estado-da-arte, graças ao uso da métrica espalhamento e da exploração de múltiplos atributos textuais.

Tabela 1: Precisão nas 5 Primeiras Posições da Recomendação: Valores Médios e Intervalos de Confiança de 95% (Melhores Resultados em Negrito)

Estratégia		LastFM	YahooVideo	YouTube
Estado-da-Arte	Sum+	0.411 ± 0.001	0.484 ± 0.003	0.245 ± 0.002
	LATRE	0.405 ± 0.001	0.608 ± 0.003	0.285 ± 0.004
	CTTR	0.260 ± 0.001	0.465 ± 0.004	0.376 ± 0.002
Novas soluções heurísticas	Sum+TS	**0.418 ± 0.002**	0.674 ± 0.003	**0.475 ± 0.002**
	LATRE+TS	0.411 ± 0.001	**0.716 ± 0.003**	0.467 ± 0.003

6. Redução de Poluição

Diversas formas de poluição já foram detectadas em vários contextos: vandalismo na Wikipédia [Potthast et al 2010], lixo informacional em atributos textuais [Suchanek et al 2008], *spamming* em atributos textuais [Koutrika et al 2008]. O contexto escolhido aqui foi a poluição em Sistemas de Compartilhamento de Vídeos Online (SCVOs), com foco no sistema YouTube. Em particular, focamos em um recurso do YouTube ainda pouco investigado, as **vídeo-respostas**, que são vídeos postados como respostas a outros vídeos. O nosso interesse nasceu de uma investigação prévia que evidenciou a exploração deste recurso por dois tipos de usuários poluidores [Benevenuto et al 2009b]. ***Spammers*** são usuários que postam vídeos não relacionados em resposta a vídeos populares visando aumentar a visibilidade de seus próprios vídeos. **Promotores** são usuários que postam um grande número de vídeos, na sua maioria não relacionados, em resposta ao seu próprio vídeo, visando inflar, artificialmente, os contadores internos mantidos pelo YouTube (p.ex: número de vídeo-respostas) a fim de que seu vídeo venha a ser promovido para a primeira página da aplicação na lista de mais respondidos.

Assim, nós desenvolvemos um método automático para detectar *spammers* e promotores no YouTube [Benevenuto et al 2009a]. Note que o nosso foco não é a detecção de conteúdo poluído, mas ao contrário, a detecção de potenciais poluidores (*spammers* ou promotores). A idéia é que o método possa auxiliar administradores do sistema no esforço de detecção. Cabe a eles decidir sobre que políticas aplicar para reduzir a poluição gerada por eles (p.ex: suspensão de conta, e-mail de alerta, etc).

A primeira tarefa foi o desenvolvimento de um modelo do comportamento típico dos usuários, incluindo usuários poluidores e usuários legítimos, visando tentar diferenciá-los em um segundo momento. O comportamento de um usuário foi modelado por um conjunto de características que expressam seu comportamento no que tange o uso feito do sistema [Benevenuto et al 2009a]. As características são categorizadas em três grupos. **Características dos vídeos do usuário** capturam propriedades específicas dos vídeos postados e respondidos pelo usuário, tais como: duração média, números de visualizações e de comentários, número de vezes que o vídeo foi selecionado como favorito, números de honrarias e de links externos. **Características individuais** incluem número de amigos, número de vídeos postados, número de vídeos assistidos, número de vídeos adicionados como favoritos, número de vídeo-respostas postados e recebidos, tempo médio entre postagens, etc. Por fim, as **características das redes sociais** estabelecidas com outros usuários via interações de vídeo-resposta incluem coeficiente de agrupamento, *betweeness* (ou centralidade), reciprocidade, e assortatividade.

Nós coletamos dados do YouTube referentes a 829 usuários, que foram pré-classificados em 641 legítimos, 157 *spammers*, e 31 promotores. Esta pré-classificação exigiu a avaliação manual de mais de 20.000 vídeos. Para a detecção automática dos usuários nas três classes de usuários, utilizamos dois algoritmos de aprendizado de máquina supervisionado considerados estado-da-arte: SVM [Vapnik 1995] e Lazy Associative Classifier (LAC) [Veloso et al 2006]. Algoritmos de aprendizado supervisionado "aprendem" padrões a partir de um conjunto de treino (previamente rotulado nas classes corretas) para aplicá-los em um conjunto de teste (a ser classificado). Assim, utilizamos SVM e LAC para aprender padrões de combinações das características descritas acima que maximizassem a identificação dos usuários nas respectivas classes. Realizamos uma bateria de experimentos com a coleção de 829 usuários, utilizando um processo de validação cruzada com 5 partições (4 partições para treino do algoritmo e uma para teste, com 25 repetições).

A Tabela 2 mostra resultados obtidos usando o classificador SVM. Ela mostra as porcentagens de usuários de cada classe **(classe real)** que foram classificados como promotores, *spammers* e legítimos **(classe predita)**. Os resultados são valores médios de 25 execuções. Intervalos de confiança de 95%, omitidos por clareza, indicam um erro máximo de 5% sobre as médias reportadas. Estes resultados indicam que o algoritmo de aprendizado foi capaz de detectar corretamente quase todos os usuários promotores e legítimos, classificando aproximadamente 60% dos *spammer*s corretamente, enquanto cerca de 40% deles foram considerados usuários legítimos. Uma análise mais aprofundada desses resultados revelou que os *spammers* não detectados tinham um comportamento dual, ora agindo como *spammers* ora agindo como usuários legítimos, tornando a identificação automática muito difícil. Como trabalho futuro, pretendemos investigar estratégias para melhorar a eficácia da detecção de *spammers*. Uma possível abordagem seria tratar separadamente diferentes perfis de um mesmo usuário. De qualquer forma, considerando que a intenção é utilizar o método proposto como ferramenta de investigação para a implementação de políticas específicas de redução de poluição, muito provavelmente incluindo uma análise manual dos usuários suspeitos detectados, consideramos os resultados obtidos bastante satisfatórios.

Contudo, apesar dos bons resultados, métodos supervisionados têm a desvantagem de necessitar de dados de treinamento para o aprendizado dos padrões. No caso de SCVOs, isso pode ser muito custoso, pois pode envolver a verificação e

rotulação de milhares de vídeos. Visando reduzir este custo, nós desenvolvemos recentemente uma abordagem semi-supervisionada para o problema [Langbehn et al 2010]. Nessa abordagem, as características de vídeo, individuais e de redes são particionadas em "visões" distintas, e um classificador é treinado com um conjunto bem pequeno de instâncias de treino contendo apenas os atributos relativos às características de cada visão, gerando assim três classificadores diferentes. Esses classificadores são então aplicados a um conjunto de dados não rotulados. Quando esses classificadores concordam quanto à classificação de uma mesma instância desse conjunto com uma alta confiança, essa instância é incorporada ao treino. A idéia é, portanto, explorar as múltiplas visões para expandir um conjunto de treino originalmente reduzido e melhorar a classificação. Nós investigamos várias estratégias para combinar os classificadores gerados para cada visão. Os nossos melhores resultados indicam que conseguimos reduzir a necessidade de treino em até 80% mantendo as taxas de acerto do classificador em níveis muito próximos aos apresentados na Tabela 2 [Langbehn et al 2010].

Tabela 2: Classificação de Usuários do YouTube usando o Classificador SVM

Classe Real	Classe Predita		
	Promotor	*Spammer*	Legítimo
Promotor	96.13%	3.87%	0%
Spammer	1.40%	56.99%	41.91%
Legítimo	0.31%	5.02%	94.66%

7. Protótipos

Como prova de conceito, nós instanciamos as técnicas e métodos descritos nas Seções 3-6 em dois protótipos: GreenMeter e GreenWiki[3]. GreenMeter é uma ferramenta para estimar a qualidade de *tags* e para recomendar *tags* em aplicações da Web 2.0. Ela utiliza as métricas de qualidade e os métodos de recomendação apresentados nas Seções 3.1 e 5. A Figura 2 mostra uma tela do protótipo desenvolvido para a aplicação LastFM[4], sendo aplicado na página da artista "Nina Simone": a qualidade de cada *tag* é mostrada com uma cor, em uma escala de vermelho (pior qualidade) a verde (melhor qualidade). O medidor indica a qualidade média de todas as *tags* da nuvem de *tags*. A figura também mostra as *tags* recomendadas pelo GreenMeter, "jazz", "soul", e "piano", que parecem descrever bem a famosa pianista e cantora.

GreenWiki é uma ferramenta para apresentar aos usuários indicadores da qualidade dos artigos da Wikipédia, seguindo a proposta de comunicação da qualidade apresentada na Seção 4 e as estimativas discutidas na Seção 3.2. Na sua versão atual, o GreenWiki implementa dois indicadores: cobertura (distribuição das citações ao longo do artigo) e estabilidade (número de edições feitas no artigo em um determinado período sobre número de edições total do artigo). Maiores cobertura e estabilidade indicam um artigo com melhor sua qualidade. Na interface foram acrescentados um medidor para cada indicador, seguindo o mesmo padrão do GreenMeter. Ao clicar no medidor, outra tela é aberta mostrando as explicações e as visualizações associadas. A

[3] http://sites.google.com/site/greenmeterdemo/ e http://www.dcc.ufmg.br/projetos/greenwiki/mediawiki.

[4] LastFM ordena as *tags* pela sua popularidade, mostrando *tags* mais populares em fontes maiores.

Figura 3 mostra a tela do GreenWiki com os indicadores apresentados no canto superior direito e a tela que será aberta ao seu clicar no indicador de estabilidade.

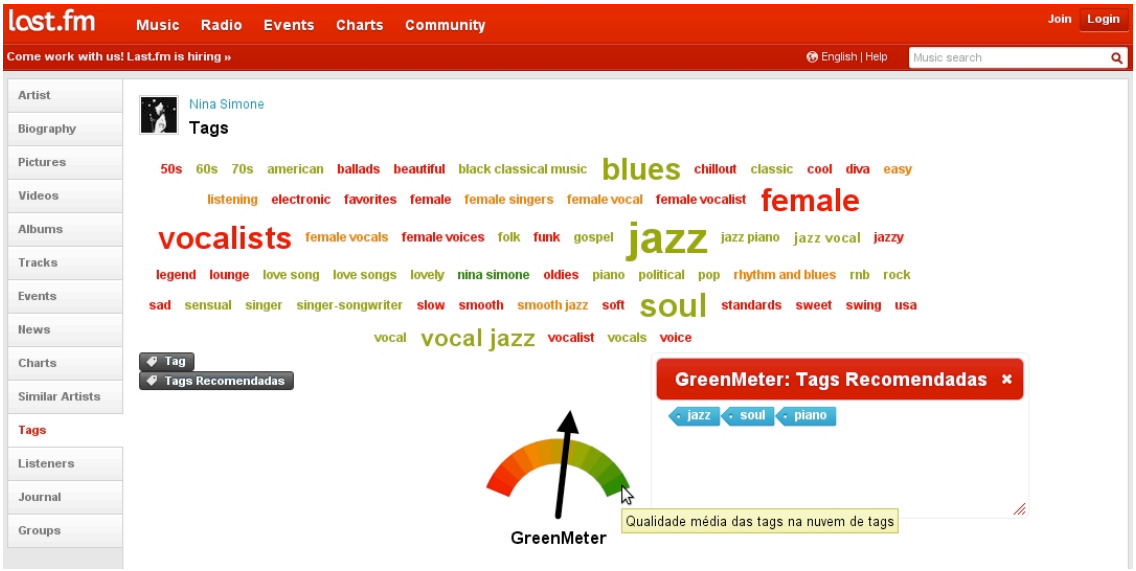


Figura 2: GreenMeter: Estimador de Qualidade e Recomendador de Tags (estudo de caso: LastFM)

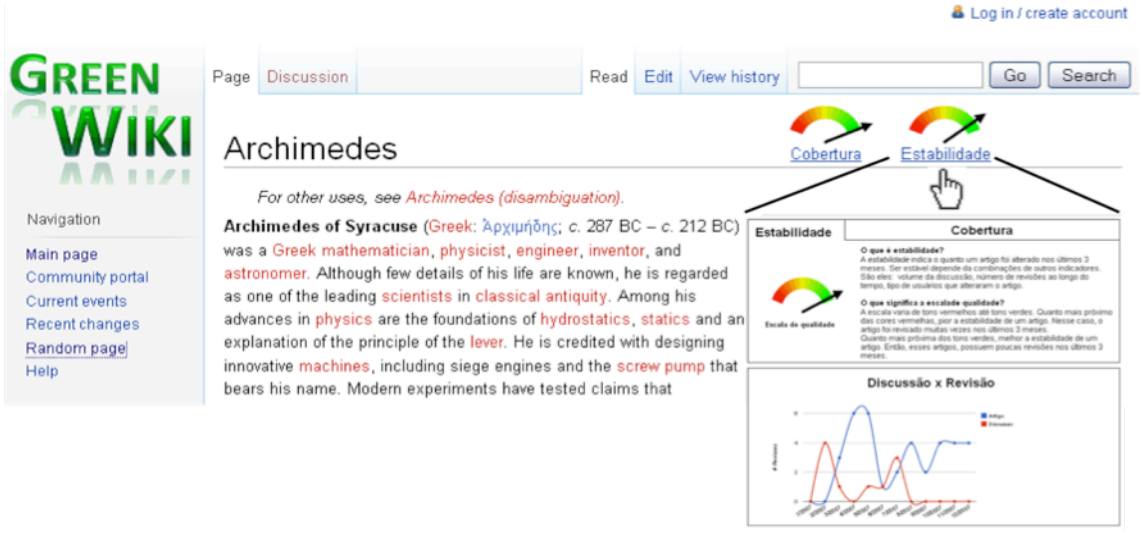


Figura 3: GreenWiki: Estimador de Qualidade de Artigos do Wikipédia

8. Considerações Finais

Este artigo apresentou o GreenWeb, um arcabouço que visa melhorar a qualidade da informação em aplicações e serviços da Web 2.0 através da promoção de conteúdo de maior qualidade e da redução de conteúdo poluído. Soluções desenvolvidas para 3 instanciações do arcabouço foram apresentadas: (1) qualidade de atributos textuais na Web 2.0; (2) qualidade de artigos da Wikipédia e (3) qualidade de vídeos no YouTube.

Como trabalho futuro, pretendemos estender o arcabouço para outros contextos, incluindo: detecção de usuários vândalos na Wikipédia e de *spammers* em atributos textuais assim como o uso das métricas de qualidade propostas no projeto de serviços classificação automática de objetos e em serviços de recomendação de conteúdo.

Pretendemos ainda estender nossas soluções para conteúdo multimídia explorando técnicas de processamento e de recuperação de informação especializadas.

Agradecimentos

Este trabalho é desenvolvido como parte do Instituto Nacional de Ciência e Tecnologia para Web (MCT/CNPq proc.53.3871/2008-6), com o apoio do CNPq e da FAPEMIG.

Referências

Almeida, J. M., Gonçalves, M. A., Figueiredo, F., Belém, F. and Pinto, H. (2010) "On the Quality of Information for Web 2.0", In: IEEE Internet Computing, v. 14.

Baeza-Yates, R. and Ribeiro-Neto, B. (2011) "Modern Information Retrieval", Addison-Wesley Professional, second edition.

Belém, F., Martins, E., Almeida, J. M., Gonçalves, M. A. and Pappa, G. (2010) Exploiting Co-Ocurrence and Information Quality Metrics to Recommend Tags in Web 2.0 Applications", In: Proc. ACM CIKM.

Benevenuto, F. , Rodrigues, T., Almeida, V., Almeida, J. and Gonçalves, M. (2009) "Detecting Spammers and Content Promoters in Online Video Social Networks", In: Proc. ACM SIGIR.

Benevenuto F., Rodrigues T., Almeida, V. Almeida, J. and Ross, K. (2009) "Video Interactions in Online Video Social Networks", ACM TOMCCAP, 5(4) article 30.

Bischoff, K., Claudiu-S, F., Wolfgang, N. and Raluca, P. (2008) "Can All Tags Be Used for Search?", In: Proc. ACM CIKM.

Boll, S. (2007) "MultiTube – Where Web 2.0 and Multimedia Could Meet", In:.IEEE Multimedia, 14(1).

Carvalho, A., Brayner, A. Loureiro, A., Furtado, A. et al. (2006) "Grandes Desafios da Computação – 2006 a 2016", Relatório disponível em http://www.sbc.org.br .

Chevalier, F.; Huot, S. & Fekete, J.-D. (2010). WikipediaViz: Conveying article quality for casual Wikipedia readers. 2010 IEEE Pacific-Vis, pp. 49--56.

Clements, M., de Vries, A. P. and Reinders, M. (2010) "The Task Dependent Effect of Tags and Ratings on Social Media Access", In: ACM TOIS, 28(4).

Dalip, D., Gonçalves, M., Cristo, M. and Calado, P. (2009) "Automatic Quality Assessment of Content Created Collaboratively by Web Communities. A Case Study of Wikipedia", In: Proc. JCDL.

Dondio, P., Barrett, S., Weber, S., and Seigneur, J. (2006) "Extracting Trust from Domain Analysis: A Case Study on the Wikipedia Project", Autonomic and Trusted Computing. Springer Berlin / Heidelberg.

Figueiredo, F., Belém, F., Pinto, H., Almeida, J. M., Gonçalves, M. A. , Fernandes, D., Moura, E. and Cristo (2009) "Evidence of Quality of Textual Features on the Web 2.0", In: Proc. ACM CIKM.

Giles J. (2005) "Internet Encyclopaedias Go Head to Head", In: Nature 438, 7070.

Krieger, M.; Stark, E. M. & Klemmer, S. R. (2009) "Coordinating tasks on the commons" In: Proc. CHI.

Heymann, P. Paepcke, A. and Garcia-Molina, H. (2010) "Tagging Human Knowledge", In Proc. ACM WSDM.

Koutrika, G., Effendi, F., Gyöngyi, Z., Heymann, P. and Garcia-Molina, H. (2008) "Combating spam in tagging systems: An evaluation", In: ACM TWEB 2(4).

Langbehn, H., Ricci, S., Gonçalves, M. A., Almeida, J., Pappa, G. and Benevenuto, F. (2010) "A multi-view approach for detecting spammers and content promoters in online video social networks", In: Journal of Information and Data Management v.1.

Lipczak, M., Hu, Y., Kollet, Y. and Milios, E. (2009). "Tag Sources For Recommendation In Collaborative Tagging Systems", In: Proc. PKDD.

Menezes, G., Almeida, J. Belém, F., Gonçalves, M., Lacerda, A., Moura, E., Pappa, G., Veloso, A. and Ziviani, N. (2010) "Demand-Driven Tag Recommendation", In: Proc. PKDD.

Moura, E., Fernandes, D., Ribeiro-Neto, B., Silva, A. and Gonçalves, M. (2010) "Using Structural Information to Improve Search in Web Collections", In: JASIST, 61.

Pereira, R. L. dos Santos. (2011) "Qualidade de Artigos na Wikipedia para seus Usuários – Análise e Proposta da Interação", Dissertação de mestrado, DCC/UFMG.

Pirolli, P.; Wollny, E. & Suh, B. (2009) "So you know you're getting the best possible information", In: Proc. CHI.

Potthast, M., Stein, B. and Holfeld, T. (2010) "Overview of the 1st International Competition on Wikipedia Vandalism Detection", In: Notebook Papers of CLEF 2010 LABs and Workshops.

Ramage, D., Heymann, P., Manning, C. and Garcia-Molina, H. (2009) "Clustering the Tagged Web", In: Proc. WSDM.

Rassbach, L., Pincock, T., and Mingus, B. (2007) "Exploring the Feasibility of Automatically Rating Online Article Quality. http://upload.wikimedia.org/wikipedia/wikimania2007/d/d3/RassbachPincockMingus07.pdf.

Santos, R. L. e Prates, R. O. (2010) "Estratégias para Comunicar Qualidade na Wikipédia", In: Proc: IHC.

Santos-Neto, E., Figueiredo, F. Figueiredo, Almeida, J., Mowbray, M., Gonçalves, M. and Ripeanu, M. (2010) "Assessing the Value of Contributions in Tagging Systems", In:Proc. IEEE 2nd International Conference on Social Computing, 2010.

Schenkel, R., Crecelius, T. , Kacimi, M., Michel, S., Neumann, T., Parreira, J.X. and Weikum, G. (2008), "Efficient Top-k Querying Over Social-Tagging Networks", In: Proc. SIGIR.

Sigurbjornsson, B. and van Zwol R. (2008) "Flickr Tag Recommendation Based on Collective Knowledge", In: Proc. WWW.

Suchanek, F. M., Vojnovic, M. and Gunawardena, D. (2008) "Social Tags: Meaning and Suggestions", In: Proc. ACM CIKM.

Vapnik, V. (1995) "The Nature of Statistical Learning Theory", Springer.

Veloso, A., Meira Jr, W. and Zaki, M. (2006) "Lazy Associative Classification", In: Proc. ICDM.

BiO4SeL: Um Algoritmo Autonômico Biologicamente Inspirado para Roteamento em Redes de Sensores Sem Fio

Levi Bayde Ribeiro[1], **Miguel Franklin de Castro**[1]

[1]Grupo de Redes de Computadores, Engenharia de Software e Sistemas (GREat)
Departamento de Computação – Universidade Federal do Ceará – Fortaleza-CE

{levi,miguel}@great.ufc.br

Abstract. *One of the main challenges in a Wireless Sensor Network (WSN) is to make its organization and routing energetically efficient. It is required to perform these task in an autonomic and self-sustainable way. This paper presents BiO4SeL (Bio-inspired Optimization for Sensor network Lifetime), a self-organization and routing algorithm for WSN in which the main objectives are autonomy and network lifetime optimization. BiO4SeL is based on Swarm Intelligence, more specifically in Ant Colonies. Results show that BiO4SeL accomplishes its objectives when compared to other algorithms used for WSN (and ad-hoc) routing: AODV and ARAMA.*

Resumo. *Um dos principais desafios das Redes de Sensores Sem Fio (RSSF) é realizar de forma energeticamente eficiente sua organizaçãoo e o seu roteamento. É necessário que estas tarefas sejam executadas de forma autônoma e auto-sustentável. Este artigo propõe o BiO4SeL (Bio-inspired Optimization for Sensor network Lifetime), um algoritmo de auto-organização e roteamento para RSSF, cujos principais objetivos são autonomia e otimização do tempo de vida da rede. O BiO4SeL é baseado em Inteligência de Enxames, mais especificamente em Colônias de Formigas. Os resultados mostram que o BiO4SeL atinge seus objetivos quando comparados a outros algoritmos utilizados para roteamento em RSSF: AODV e ARAMA.*

1. Introdução

No contexto de Computação Ubíqua e Pervasiva, sensores são dispositivos computacionais importantes e potencialmente bastante numerosos. Eles coletam informação do ambiente, como temperatura, movimentação, humidade, dentre outros, e comunica-se de forma direta ou através de seus vizinhos para enviar seus dados até a Estação Base (EB, ou BS, *Base Station*), ou sorvedouro (*sink*). Eles podem ser utilizados em uma grande variedade de aplicações. Por exemplo, podem ser distribuídos em uma floresta para monitoramento de incêndios.

Uma das mais abordadas questões em redes de sensores é seu tempo de vida, ou seja, o tempo de utilização da rede até o primeiro nó ficar sem energia. Durante a operação, alguns nós podem ser mais requisitados que outros para encaminhar a informação de outros nós. Isto poderia gerar locais sem cobertura na área de abrangência da rede. Além disso, os nós devem fazer balanceamento de carga para maximizar seu tempo de vida.

Manter a cobertura da rede completa pelo máximo de tempo possível é o objetivo de diversos algoritmos de roteamento para redes de sensores. Este problema é particularmente importante em aplicações críticas, como monitoramento de radioatividade, pois nelas as áreas "cegas" são inaceitáveis. Soluções para este problema, ou para parte dele, podem ser encontradas na literatura. A maioria delas é baseada no nível de roteamento. Muitas delas, entretanto, têm fortes restrições, como conhecimento de localização, processamento centralizado, etc.

Este artigo apresenta o BiO4SeL (*Biologically-inspired Optimization for Sensor network Lifetime*, ou Otimização Biologicamente Inspirada para o Tempo de Vida de Redes de Sensores), um protocolo de roteamento baseado em colônias de formigas distribuído e autônomo. Seu objetivo é maximizar o tempo de vida em redes de sensores. Para isto, aplica o conceito Inteligência de Enxame, formando inteligência a partir de comportamente rudimentar distribuído pelos sensores.

BiO4SeL, assim como outras abordagens, inspira-se na idéia de criar caminhos na rede para enviar os dados dos nós para a EB, distribuindo o consumo de energia pelos nós ao longo deste caminho. BIO4SeL, entretanto, diferencia-se de outros algoritmos por utilizar informação de nível energético de bateria não somente para construir a tabela de roteamento incialmente, mas também para atualizá-la a medida que a energia é consumida. Em consequência disto, a responsabilidade pelo encaminhamento de pacotes tende a ser igualmente distribuída. Assim, mantém-se a variância do mapa energético da rede o menor possível, aumentando o tempo de vida da rede. Experimentos de simulação comparando BiO4SeL ao ARAMA (*Ant-based Routing Algorithm for Manets*, ou Algoritmo de Roteamento baseado em Formigas para RSSF) [Hussein et al. 2005] e AODV (*Ad-hoc On-demand Distance Vector*, ou Vetor de Distância Ad-hoc sob demanda) [Perkins et al. 2003] mostram como o BiO4SeL alcança seus objetivos, apresentando melhor organização de inicialização da rede e cenários mais densos (testados até 320 nós).

O restante deste artigo está organizado como segue. A Seção 2 aborda alguns trabalhos relacionados em roteamenteo para redes de sensores. Na Seção 3, a abordagem BiO4SeL é introduzida e sua especificação, mostrada. A Seção 4 traz a descrição dos experimentos e a discussão sobre seus resultados, enquanto a Seção 5 contém as conclusões e propostas para trabalhos futuros.

2. Trabalhos Relacionados

Inicialmente, Inteligência de Enxames foi aplicada a grafos como uma abordagem de Inteligência Artificial para a solução do problema do Caixeiro Viajante utilizando-se formigas [Dorigo et al. 1991]. Depois disto, este paradigma tornou-se cada vez mais popular. No domínio de redes de comunicações, ele foi inicialmente utilizado para o roteamento de pacotes, com o algoritmo AntNet [Caro and Dorigo 1997]. Fazendo-se um paralelo entre a Natureza e as redes, para algoritmos de roteamento, cada formiga é um pacote de controle. Quando um caminho deve ser criado de um nó para outro, algumas formigas são enviadas. Elas movem-se e, probabilisticamente, selecionam o próximo salto, baseadas na distribuição feromônica ao longo do caminho ou em alguma outra heurística do caminho. Quando a formiga alcança seu destino, ela calcula o feromônio a ser depositado, baseado nesta heurística. Depois disso, a formiga é enviada de volta. Para seguir os mesmos saltos, elas mantêm uma lista dos nós visitados durante sua ida até o destino. Durante

seu retorno, elas aprimoram os feromônios em cada salto, aumentando a probabilidade de aquele caminho ser o escolhido para as próximas formigas. Este algoritmo é baseado em tabelas de roteamento probabilísticas.

O ARAMA (*Ant-based Routing Algorithm for Manets*, ou Algoritmo de Roteamento baseado em Formigas para RSSF) [Hussein et al. 2005] utiliza estas tabelas de roteamento probabilísticas para enviar pacotes em redes Ad-hoc. A principal diferença entre esta estratégia e o AntNet é a influência da quantidade energética das baterias dos nós influenciarem no cálculo da deposição feromônica.

Uma variante do ARAMA, o AOER (*Ant-Based On-Demand Energy Routing*, ou Roteamento baseado em Formigas por Demanda Energética) [Shuang et al. 2007], almeja ajustar o roteamento para aumentar o tempo de redes Ad-hoc. Nesta abordagem, as formigas coletam informação sobre a potência de bateria remanescente em cada nó, assim como a capacidade de reenvio e contagem de saltos. Esta informação é utilizada para construir uma tabela de roteamento probabilística inversa. Ela é inversa, pois quanto mais formigas seguem pelo caminho, menos poder de energia ele tem. Além disso, uma formiga não guarda o caminho visitado, reduzindo seu tamanho e, consequentemente, a energia consumida para enviá-la, recebê-la e processá-la. Infelizmente, o artigo não provê detalhes de implementação suficientes que possam permitir uma comparação com o BiO4SeL.

Além de formigas, outras estratégias podem ser encontradas na literatura para economia energética em redes de sensores. Uma destas estratégias, também relacionada ao BiO4SeL, é o balanceamento de carga, uma solução herdada do domínio de telecomunicações. Utilizar alguma infraestrutura torna este processo mais simples. Por exemplo, abordagens encontradas em [Akyildiz et al. 2002, Al-Karaki and Al-Mashaqbeh 2007, Iqbal et al. 2006] empregam agrupamento (*clustering*) para conseguir agregação de dados e revezamentos de nó cabeça, consequentemente economizando gasto energético.

Após um profundo estudo e comparação de vantagens e desvantagens destas abordagens, este artigo propõe um novo protocolo de roteamento baseado em formigas, chamado BiO4SeL (*Biologically-inspired Optimization for Sensor network Lifetime*, ou Otimização Biologicamente Inspirada para o Tempo de Vida de Redes de Sensores). Seu objetivo é aumentar o tempo de vida de uma Rede de Sensores, como descrito na próxima sessão.

3. O Protocolo BiO4SeL

Esta seção descreve o protocolo de roteamento BiO4SeL (*Biologically-inspired Optimization for Sensor network Lifetime*, ou Otimização Biologicamente Inspirada para o Tempo de Vida de Redes de Sensores), assim como sua implementação na ferramenta Network Simulator (NS-2) [NS-2 Webpage 2010], versão 2.32 (também testado na versão 2.33). Assume-se cada nó ter uma identificação única e utilizar o IEEE 802.15.4 [IEEE 2003] nas camadas PHY e MAC. Nas simulações, os identificadores dos nós são números começando por 0.

Em resumo, o protocolo funciona como segue. Compreende-se por três estágios: Reconhecimento (*Bootstrap*), Descoberta Inicial de Rotas e Troca de Dados. No primeiro estágio, cada nó reconhece seus vizinhos utilizando uma sinalização específica. Após

isto, na fase de Descoberta Inicial de Rotas, formigas são utilizadas para criar as rotas iniciais na rede, de cada nó até a EB. Assim, na fase de Troca de Dados, dados de aplicação começam a fluir, levando formigas de carona (*piggyback*) para aprimorar a probabilidade das rotas. A medida que os dados são encaminhados, a energia das baterias dos sensores ao longo dos melhores caminhos diminui mais rápido que em outros nós. Uma das diferenças entre o BiO4SeL e outras abordagens baseadas em formigas é o fato de levar esta energia em consideração para a evaporação feromônica. Quanto menos poder remanescente houver na bateria, mais rápido o feromônio no nó decrementa, tornando este nó menos provávelç de ser escolhido. Isto causa a seleção de rotas menos escolhidas, mas com mais energia, mesmo formando um caminho mais longo (mais saltos). O pseudocódigo deste protocolo é mostrado no Algoritmo 1.

Algorithm 1 Algoritmo do Protocolo BiO4SeL

```
1: {Fase de Reconhecimento}
2: nós enviam ihello
3: if nó recebe ihello then
4:     guarda informação de bateria do vizinho
5: end if
6:
7: {Fase de Descoberta Inicial de Rotas}
8: nós criam suas tabelas de roteamento
9: EB envia pacotes iant
10: if nó recebe iant then
11:     diminui a estimativa energética dos vizinhos
12:     if já recebida (mesmo identificador) then
13:         descarta iant
14:     else
15:         if saltos do pacote < distância atual à EB then
16:             atualiza distância à EB
17:         else
18:             decrease feromônio
19:         end if
20:     end if
21: end if
22:
23: {Fase de Troca de Dados}
24: eventualmente, cada nó envia hello
25: eventualmente, cada nó envia dados
26: while nó recebe pacote do
27:     decrementa estimativa energética dos vizinhos
28:     if pacote = hello then
29:         if previamente na tabela de roteamento then
30:             conserta estimativa energética dos vizinhos
31:         else
32:             envia requestIHello
33:         end if
34:     end if
35:     if pacote = requestIHello then
36:         guarda informação de bateria
37:         envia respRqIHello
38:     end if
39:     if pacote = respRqIHello then
40:         guarda informação de bateria
41:     end if
42:     if pacote = dados then
43:         if previamente recebido then
44:             descarta
45:             envia formiga negativa no caminho inverso
46:         else
47:             calcula próximo salto
48:             ajusta feromônio
49:             encaminha dados
50:         end if
51:         if EB then
52:             descarta
53:         end if
54:     end if
55:     if pacote = formiga negativa then
56:         decrementa feromônio
57:         if primeiro nó do laço then
58:             descarta
59:         else
60:             encaminha
61:         end if
62:     end if
63: end while
```

3.1. Fase de Reconhecimento

Esta fase inicia-se quando o sensor é ligado. Neste momente, ele envia uma mensagem `ihello` para todos os seus vizinhos, ou seja, uma mensagem de inundação com Tempo de vida (TTL) igual a 1. Ela é composta poelo tipo do pacote, `hlTp`, e sua própria informação de bateria. Esta informação muda de um tipo de sensor para outro, e é composta de energia inicial (`InitEn`) e taxa de consumo específicas do modelo nos possíveis estados do sensor: ocioso (`IdEnCte`), envio (`SndEnCte`) e recepção (`RcvEnCte`).

Assim como o BiO4SeL leva em consideração a energia do vizinho para calcular o feromônio relacionado a ele, a informação de bateria de cada vizinho é mantida e atualizada em cada nó.

3.2. Fase de Descoberta Inicial de Rotas

Esta fase consiste em inundar a rede com uma quantidade configurável $iant_cnt$ de formigas (`iant` packets) de cada EB para a rede. Elças são compostas pelo tipo de pacote (`iaTp`) e identificador de inundação (`broadId`). Este identificador, juntamente com o Id do nó fonte, é armazenado em cada salto ao longo do caminho. Se um nó receber a mesma *iant* novamente, ela não será reencaminhada (detecção de *loops*).

Estas formigas agem diferentemente das de outros protocolos. Na primeira chegada de uma `iant` a cada nó, ela salva a distância do nó até a EB, com o próximo nó sendo o nó do qual ela veio. Além disso, ela mantém, na tabela de roteamento, o mínimo de saltos para alcançar a EB. A partir da segunda chegada, elas podem modificar o feromônio em certos saltos. Ainda que incrementar feromônios seja impossível nesta fase, eles podem ser decrementados se um nó receber uma `iant` de um vizinho com distância (saltos) a EB maior que a guardada no nó. Isto representa uma formiga andando na direção da EB e não procurando por nós remanescentes. Como consequência, a probabilidade de escolha desse salto seria reduzida, por pertencer a um caminho reverso ou maior que o necessário. Depois de algumas rodadas de `iants`, o feromônio é concentrado no sentido dos nós para a EB.

Quando ocorre a situação citada acima, a taxa de decréscimo feromônico em cada nó é calculada de acordo com a Equação 1, como segue:

$$F_d^v = F_d^v - Min\{K \times (2 - S_d^v); 0.9\} \times (F_d^v - F_{min}) \quad (1)$$

Sendo F_d^v o feromônio relacionado ao destino (sorvedouro) d pelo vizinho v, K o coeficiente arbitrário de decréscimo feromônico, S_d^v a relação de saltos do vizinho v para o destino d, e F_{min} é o mínimo feromônico para um salto, escolhido arbitrariamente.

A relação de saltos referida na Equação 1 é dada por $S_d^v = S_d^{min}/S^a$, sendo S_d^{min} o mínimo de saltos, dentre todos os vizinhos, para o destino d, dado por $S_d^{min} = Min\{S_d^v\}, \forall$ vizinho v, e S^a a quantidade de saltos da formiga. S_d^v representa o quão longe o caminho da formiga está do melhor caminho previamente alcançado. Assim, o feromônio em cada nó é diretamente influenciado por sua distância até a EB. Esta equação também é útil para evitar ciclos, pois caminhos mais longos são desencorajados.

Além disto, cada nó pode estimar a quantidade de potência de bateria disponível em cada vizinho ao longo do tempo. Isto é possível pelo fato de o nó saber como se processa o consumo energético deles pelas informações do `ihello`. Isto é feito para cada mensagem enviada ou recebida. Esta estimava é útil para fazer uma redução feromônica mais aproximada nos saltos. Periodicamente, esta estimativa é atualizada com dados reais provenientes de formigas de carona.

Quando uma mensagem é enviada a um vizinho, a estimativa energética para ele é atualizada de acordo com a fórmula $E^v = E^v - (C_{rcp}^v \times FC_{rcp}^n) \times T$, sendo E^v a estimativa energética atual do vizinho v, C_{rcp}^v o coeficiente de consumo energético para o estado de recepção do vizinho v (informado no pacote `ihello`), FC_{rcp}^n o fator de correção de recepção do vizinho v, calculado na próxima fase e inicialmente 1, e T o tamanho total do pacote, em bits. Este fator de correção é utilizado para tornar a estimativa tão aproximada quanto possível, pois compara estimativas prévias com valores reais para

calcular as futuras.

De maneira similar, quando uma mensagem é recebida por um vizinho, a fórmula utilizada é $E^v = E^v - (C^v_{env} \times FC^v_{env}) \times T$, sendo C^v_{env} o coeficiente de consumo energético para o estado de envio do vizinho v (também contido no `ihello`), e FC^v_{env} o fator de correção de envio para o vizinho v, calculado na próxima fase e inicialmente 1.

Além disso, o nó mantém o *timestamp* (marcação de tempo) para a última mensagem recebida deste vizinho. Assim, ele pode estimar o consumo energético durante o tempo ocioso, com a fórmula $E^v = E^v - C^n_{oci} \times (TS_0 - TS_{-1})$, sendo C^n_{oci} o coeficiente de consumo energético no estado ocioso para o nó n, e TS_0 e TS_{-1} os *timestamps* para o atual e último pacotes recebidos, respectivamente.

BiO4SeL inclui uma estratégia de retransmissão de pacotes. Em um ambiente muito populoso, um reencaminhamento de `iant` poderia colidir e se perder. Desta forma, se, dentro de um tempo pré-definido, o nó não receber `iants` de todos os seus vizinhos, ele reenvia a formiga, com um *delay* aleatório. Isto acontece somente uma vez para cada nó e cada formiga.

3.3. Fase de Troca de Dados

Nesta fase, o foco é o envio de informação dos nós para a EB. Neste ponto, o caminho inicial e a tabela de roteamento já estão calculadas, somente esperando para os dados serem enviados. A cada envio de informação, a tabela é consultado e algum vizinho é escolhido probabilisticamente, baseado nos feromônios dos saltos. Quando o pacote é encaminhado, as energias e os feromônios são atualizados e ao nó continua esperando pelo próximo pacote.

A cada intervalo de tempo $\Delta\tau_a$, formigas de anúncio, ou `hellos`, são enviados de cada nó. Seu tempo de vida é um salto, e seu objetivo é informar aos vizinhos sobre o nó estar vivo e seu nível energético atual. Com estas informações, os vizinhos podem atualizar suas tabelas de roteamento. Se um nó receber esta mensagem de um vizinho já em sua tabela de roteamento, o nó atualiza-a com um novo tempo de expiração para o remetente e sua potência de bateria. Além disso, atualiza o fator de erro na estimativa de energia gasta. Para isto, utiliza as seguintes fórmulas:

$E^n_{rcp} = C^v_{rcp} \times Cnt^v_{rcp}$, sendo E^n_{rcp} o consumo energético de recepção de pacotes desde o último `hello`, e Cnt^v_{rcp} o contador de pacotes recebidos do vizinho v neste mesmo tempo.

De maneira semelhante, $E^n_{env} = C^v_{env} \times Cnt^v_{env}$, sendo E^n_{env} o consumo energético de envio de pacotes desde o último `hello`, e Cnt^v_{env} o contador de pacotes enviados ao vizinho v neste mesmo tempo.

Por fim, $E^v_{tot} = E^v_{rcp} + E^v_{env}$, sendo E^v_{tot} a energia total gasta desde o último `hello`.

Por fim, os coeficientes de correção podem ser calculados, como segue:

- $FC_{rcp} = FC_{rcp} + (((E^v_{rea} - E^v) \times (E^v_{rcp}/E^v_{tot}))/Cnt^v_{rcp})$
- $FC_{env} = FC_{env} + (((E^v_{rea} - E^v) \times (E^v_{env}/E^v_{tot}))/Cnt^v_{env})$

Sendo E^v_{rea} a energia real do vizinho v (informado no pacote `ihello`).

Se o nó receptor não tiver o remetente em sua tabela de roteamento, ele requisita, via pacote `requestIHello`, ou pedido de anúncio de inicialização, o pacote `InitHello` contendo as informações do novo vizinho. Se o vizinho moveu-se, ele deve limpar sua vizinhança e substituí-la com uma nova. Para apagar os vizinhos antigos, há um temporizador para expirá-los.

O `InitHello` carrega a informação de bateria, de um modo similar ao *ihello*. Como o nó receptor não reconhece o nó enviando o `hello`, ele deve ter se movido para aquela localização. Recebendo o pacote, ele salva o perfil de seu novo vizinho, atualizando sua tabela de roteamento. Esta é uma mensagem em *unicast*, e de somente um salto. Após isto, ele responde com o `respRqIHello`, resposta a requisição de anúncio de inicialização, um pacote de inundação e de somente um salto. Como todos ou, ao menos, algum vizinho será novo para ele, o novo poderá salvar suas informações. Além disso, todos receberam seus `hello`, e todos enviarão seus próprios `requestIHello`. Desnecessário responder mais de uma vez. Assim, somente uma mensagem será enviada.

Esta fase, entretanto, é caracterizada pela transmissão de dados. Cada pacote de dados enviado vai juntamente com uma formiga de carona, `pbAnt`. Esta formiga é somente composta por um campo, o `pb_hop`. Ao enviar um pacote de dados, este campo leva a distância, em saltos, até a EB. Esta distância é utilizada na deposição feromônica, a cada nó.

As fórmulas para ela estão a seguir, nas quais todas as constantes arbitrárias e os pesos citadas foram obtidas durante os experimentos, com o objetivo de otimizar o protocolo.

$$F_d^v = F_d^v + ((maxF - F_d^v) \times (c + e) \times fc)$$

Sendo F_d^v o feromônio do vizinho v com relação ao destino d, $maxF$ o máximo feromônio possível em um salto, c a constante de caminho no cálculo feromônico, e a constante energética e fc a constante arbitrária associada ao cálculo.

Assim, o feromônio é adicionado de uma porcentagem do restante a ser colocado, ou seja, da subtração entre o máximo de feromônio e o atual. Esta porcentagem é calculada a partir de um balanço entre o quão bom o caminho é, ou seja, o menor possível (em saltos), e quanta energia o próximo salto possui. Utiliza-se, ainda, uma constante arbitrária para diminuir ou aumentar a mudança no feromônio.

A constante de caminho associada ao cálculo feromônico é calculada de acordo com as fórmulas $c = cc \times pc$ e $cc = minH/(cH + cnH)$, sendo cc a variável de caminho, pc o peso do caminho no cálculo, $minH$ a mínima quantidade de saltos até o destino a partir da fonte, calculado na fase de reconhecimento e contido na formiga de carona em cada pacote de dados, cH a quantidade de saltos contada pelo pacote, cnH o mínimo de saltos até o destino a partir do nó corrente, calculado na fase de reconhecimento e salvo no prórpio nó.

A constante energética, por sua vez, utiliza as fórmulas $e = ce \times pe$, $pe = 1 - pc$ e $ce = E^v/EI^v$. Para elas, ce é a variável energética, pe o peso da energia no cálculo, E^v a estimativa atual da energia do vizinho v, EI^v a energia inicial do vizinho v. Assim, a constate arbitrária da energia (ce) é dada pelo restante de energia do vizinho, e o peso da energia (pe) é sempre o complemento do peso do caminho (pc).

Como consequência da atualização do feromônio em um salto, o feromônio total até aquele destino deve ser recalculado. Descreve este cálculo a equação $F_d = F_d - (Fa_d^v - F_d^v)$, sendo F_d o feromônio total para o destino d, F_d^v o feromônio ao destino d passando pelo vizinho v, e Fa_d^v o feromônio anterior deste salto, antes da atualização.

Ainda durante o reencaminhamento, a cada certa quantidade (EvC, variável obtida durante os testes de otimização do protocolo) de pacotes de dados recebidos de um vizinho v e indo para uma EB d, o feromônio somente naquele salto é evaporado. Para isto, utiliza-se a seguinte equação.

- $F_d^v = minF + (Ev^{EvI} \times (F_d^v - minF))$
- $Ev = E^v / EI^v$

sendo F_d^v o feromônio ao destino d passando pelo vizinho v, $minF$ o mínimo feromônio possível, Ev a porcentagem de evaporação, EvI o índice de evaporação, E^v a estimativa atual de energia do vizinho v, EI^v a energia inicial do vizinho v.

Nesta fórmula, a porcentagem de evaporação feromônica é diretamente proporcional à energia restante no vizinho. Esta decisão foi tomada com o objetivo de fazer as rotas variarem, e diminuir a probabilidade de uma rota com pouca energia ser utilizada. Deste modo, espera-se aumentar o tempo de vida da rede como um todo, dividindo-se a carga pela rede inteira. Esta porcentagem é aplicada à diferença entre o feromônio atual e o mínimo. Além disto, é modificada pelo índice de evaporação, uma variável de configuração do protocolo, obtida por meio de testes. Tem o intuito de calibrar esta porcentagem para uma evaporação ideal. Este tipo de variável é utilizada em protocolos com este fim.

Novamente, como consequência da atualização (evaporação) do feromônio em um salto, o feromônio total até aquele destino deve ser recalculado, como na equação descrita.

Há uma chance, pE, de o salto não ser escolhido por esta probabilidade, mas pela maior energia restante nos vizinhos. Esta chance é dada pela fórmula $pE = 1 - (0,8 \times ce + 0,2 \times cf)$, sendo $cf = (F_d^v - minF)/(maxF - minF)$. Assim, quanto mais próximo estiver do máximo de energia ou do máximo feromônico, menor a probabilidade de haver uma escolha aleatória, pois é menor a sua necessidade. Isto somente ocorre quando o melhor caminho já está abaixo de dois terços de sua energia original. Esta é uma das técnicas utilizadas para garantir uma melhor distribuição das rotas pela rede.

Ao receber um pacote de dados, o nó salva uma entrada para o pacote, em uma fila. Esta entrada é similar a aquelas utilizadas pelas `iants`, consistindo no id do nó fonte e da mensagem e tempo de expiração. Também há um temporizador para eliminar entradas antigas. Ao receber um pacote de dados, o nó testa a presença da entrada. Se não estiver lá, é adicionada. Se ela estiver, a `negAnt` (*negative ant*, formiga negativa) é enviada pelo caminho inverso da mensagem. Esta formiga evapora o feromônio no ciclo, desencorajando futuras formigas a ir por este caminho.

4. Resultados e Discussão

Para verificar a efetividade do BiO4SeL, alguns experimentos foram feitos utilizando-s o simulador de redes NS-2. Nestes experimentos, foi utilizado o modelo de nós móveis existente, e ele foi extendido para adicionar um novo algoritmo de roteamento. A maioria dos parâmetros IEEE 802.15.4 foram deixados em seus parâmetros padrão.

Os outros parâmetros utilizados foram os seguintes.

Tabela 1. *Parâmetros do protocolo*

Parâmetro	Valor
Tempo de envio de cada `ihello`	$t = 0,01 \times id$
Quantidade de formigas de inicialização (`iant`) enviadas	$iant_cnt$ = 5
Intervalo entre cada envio de *iant*	0,5s
Intervalo entre cada envio de *hello*	7,5s a 12,5s
Tempo de expiração do contador de *hello*	13s
Tempo de expiração das entradas de *unicast*	0,5s
Tempo de expiração do contador de entradas de *unicast*	1s
Mínimo de feromônio ($minF$)	0.000001
Mínimo de feromônio ($maxF$)	0.01
Feromônio inicial (iF)	0.0001
Intervalo de envio de *hello* ($hlInt$)	10s
Expiração do contador de pacotes (EvC)	2 pacotes
Índice de evaporação (EvI)	3
Peso do caminho (pc)	0,1
Constante arbitrária associada ao cálculo feromônico (fc)	0,1
Coeficiente de diminuição feromônica (K)	0,6

Legenda: id - identificador único de cada nó.

Foram testados cenários com diferentes quantidades de nós. Para cada uma delas, foi experimentalmente definido o tamanho do cenário para ter, ao menos, 90% dos cenários aleatoriamente gerados plenamente conectados, ou seja, sem nós inalcançáveis. As configurações resultantes são apresentadas na Tabela 2.

Tabela 2. *Cenários de simulação para os experimentos.*

Quantidade de nós	Tamanho do Cenário
10	20x20
20	30x30
40	50x50
80	70x70
160	100x100
320	140x140

Nos experimentos, o BiO4SeL foi comparado a duas outras abordagens relacionadas. A primeira é o AODV (*Ad-Hoc On-Demand Distance Vector*) [Perkins et al. 2003], um padrão *de-facto* para protocolos de roteamento em MANETs, geralmente utilizado em comparações em outros trabalhos da mesma área encontrados na literatura. A outra é o ARAMA (*Ant-based Routing Algorithm for Manets*) [Hussein et al. 2005], formalmente apresentado na Seção 2, utilizado pela facilidade de reprodução, devido a quantidade de detalhes de descrição do protocolo no artigo.

Para cada configuração mostrada na Tabela 2, 100 cenários diferentes foram gerados, aleatoriamente. Para cada cenário, a simulação foi repetida 50 vezes, mudando-se as sementes de aleatoriedade. Apesar disso, para o AODV não houve repetição, pois ele é um algoritmo determinístico. Os resultados apresentados nesta seção incluem um intervalo de confiança considerando $\alpha = 0,95$.

Em cada cenário, há somente um transmissor (nó captando do ambiente) e somente um receptor (Estação Base ou sorvedouro), ambos localizados em cantos opostos

da área. Este tipo de operação é possível para aplicações nas quais eventos são monitorados baseados em *threshold*, ou seja, quando nenhum evento de exceção é sensoriado, nenhum dado é transmitido pelo nó. Por exemplo, este é o caso do monitoramento de presença para um único ator em uma área (um animal monitorado dentro de uma grande jaula). Neste caso, todos os outros nós que não estão produzindo dados são reencaminhadores (roteadores).

Para observar o comportamento do consumo energético em um curto período de simulação, a geração de dados do nó transmissor foi deixada com uma vazão relativametne alta, não necessariamente o padrão para este tipo de aplicação (um pacote de 76 KB a cada 0,4 s). Pacotes de dado começam a ser enviados a cada 28 s de simulação, quando as fases iniciais (Reconhecimento e Descoberta Inicial de Rotas) estão completas. Utilizou-se 200 s como tempo total de simulação. Com a vazão escolhida, este tempo é suficiente para observar os eventos requeridos (morte do primeiro nó e da rede). Os parâmetros para cada algoritmo (BiO4SeL, ARAMA and AODV) foram escolhidos para haver comportamentos comparáveis entre eles.

4.1. Experimento 1: Conectividade dos Cenários

Neste experimento, foi validada a efetividade dos algoritmos na construção de redes plenamente conectas, sem conhecimento prévio de vizinhos, destinos ou posicionamento geográfico. Os resultados para estes experimentos são mostrados na Tabela 3, contendo a porcentagem destes cenários gerados para cada quantidade de nós testada.

Tabela 3. Percentual de Cenários com Rede Completamente Conexa (Experimento 1).

	Quantidade de Nós no Cenário					
Algoritmo	10	20	40	80	160	320
BiO4SeL	100%	100%	100%	100%	93%	97%
AODV	100%	99%	100%	98%	98%	95%
ARAMA	97%	70%	8%	1%	0%	0%

Os resultados mostram que o BiO4SeL fica mais eficiente em construir tabelas de roteamento com o aumento do cenário. A chave para isto é a combinação de dois detalhes do protocolo. O primeiro é a descoberta de rotas ser inicializada pela EB ao invés de pelos nós, deixando-o escalável. O outro é o mecanismo de desencorajamento de formigas de retorno implementado, evitando ciclos e fazendo as formigas moverem-se para frente.

Um fato interessante a se notar é o ARAMA não ser efetivo em sua descoberta de rotas com grandes cenários como o é nos pequenos. De fato, os resultados mostados no artigo que o descreve se baseiam em pequenos cenários, os melhor adaptados para ele.

4.2. Experimento 2: Tempo de Vida da Rede de Sensores

Estes experimentos validam o tempo de vida da rede de sensores, ou seja, a quantidade de tempo desde a disposição da rede até o momento da morte do primeiro nó, o término de sua energia. Com isto, há a possibilidade de criar-se uma falha de cobertura na rede. A Figura 1 mostra a o tempo de vida da rede para cada tamanho de cenário (10, 20, 40, 80, 160 e 320 nós).

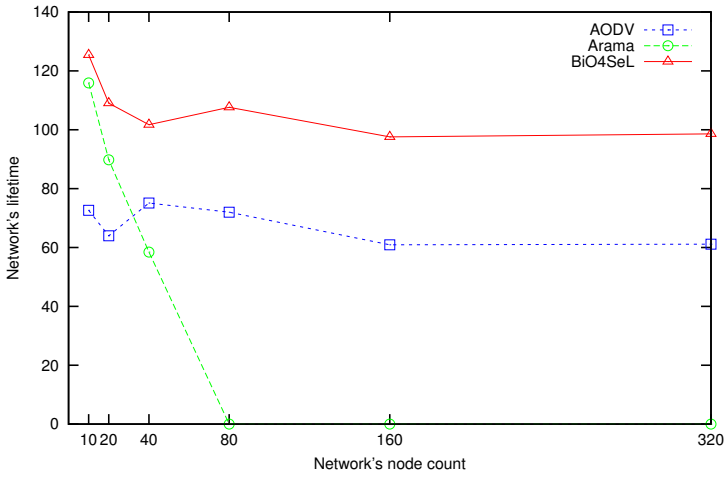


Figura 1. Tempo de Vida da Rede de Sensores (Experimento 2).

Como o ARAMA não conseguiu construir cenários conexos a partir de 80 nós, considera-se seu tempo de vida como zero nestes casos. A Figura 1 mostra como o BiO4SeL alcança melhores valores de tempo de vida em todos os cenários, mesmo quando o ARAMA é eficiente em construir seus cenários. Além disso, o comportamento relativamente linear do BiO4SeL e AODV sugerem sua escalabilidade no crescimento dos cenários.

4.3. Experimento 3: Potência de Bateria dos Nós no Momento do Tempo de Vida

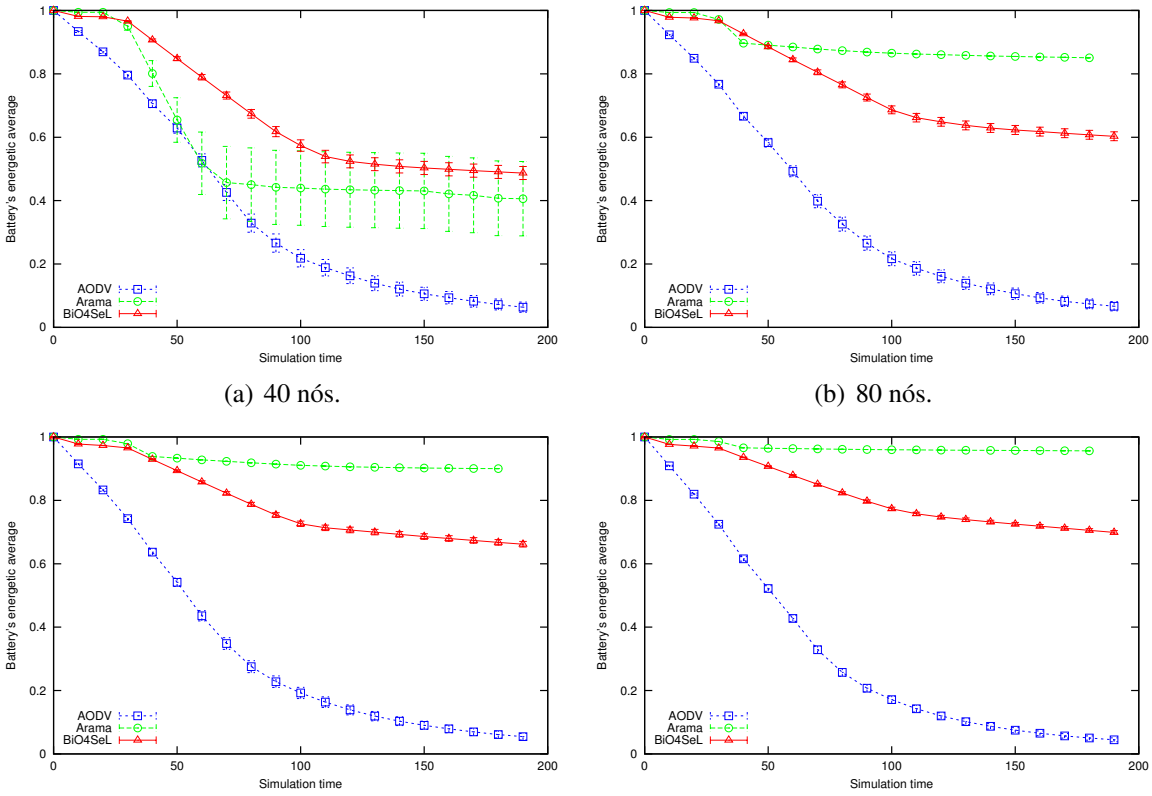
Este experimento tem por objetivo validar como a potência da bateria é economizada na rede. Isto é possível quando a rede reduz a quantidade de dados produzida e encaminhada. Considerando os dados de aplicação constantes para cada algoritmo testado, a economia energética é alcançada quando a quantidade de mensagens de sinalização é reduzida. A Figura 2 mostra o nível médio de potência da bateria disponível nos nós da rede no momento de falha do primeiro nó. Quatro cenários são mostrados: 40 nós (Figura 2(a)), 80 nós (Figura 2(b)), 160 nós (Figure 2(c)) e 320 nós (Figure 2(d)).

A Figura 2(a) mostra como o BiO4SeL mantém a média energética mais alta se comparado aos outros protocolos, para 40 nós. No cenário de 160 nós, ARAMA mantém a mais alta. Isto acontece por sua bateria não ser consumida depois de certo tempo, dado ele não encontrar rotas na rede. Assim, somente consome a bateria próximo a fonte, até a morte dos nós adjacentes, depois disso não conseguindo mais enviar dados. O BiO4SeL reduz a necessidade de sinalização enviando formigas por carona nos pacotes de dados e optimiza as mensagens. Assim, reduz a quantidade destes pacotes trafegados na rede, mantendo uma boa taxa de entrega, como comprovado pelo próximo experimento.

4.4. Experimento 4: Taxa de Entrega dos Pacotes

Este experimento mostra quão efetiva foi a entrega dos pacotes nos algoritmos. A Tabela 4 mostra a porcentagem média de pacotes enviados por um nó e recebidos pela Estação Base, para cada tamanho de cenário.

Como pode ser visto na Tabela 4, BiO4SeL mantém a melhor taxa de entrega de pacotes, não importa qual o tamanho do cenário testado. Como o ARAMA não alcança conectividade em cenários com 80 nós, ele tem taxa de entrega nula.



(a) 40 nós. (b) 80 nós.

(c) 160 nós. (d) 320 nós.

Figura 2. Média de Potência de Bateria dos Nós no Momento do Tempo de Vida (Experimento 3).

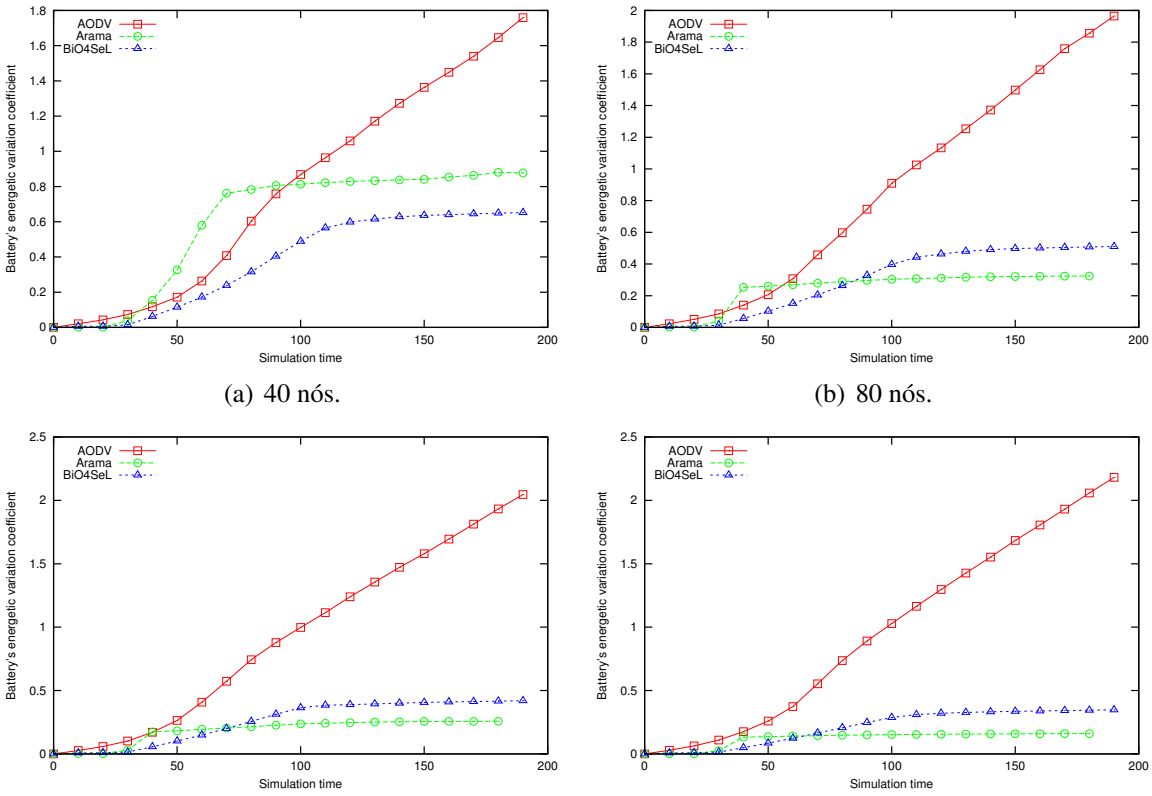
Tabela 4. Taxa de Entrega dos Pacotes (Experimento 4).

	Quantidade de Nós no Cenário					
Algoritmo	10	20	40	80	160	320
BiO4SeL	98%	96%	95%	93%	87%	75%
AODV	79%	58%	57%	52%	44%	39%
ARAMA	79%	77%	92%	0	0%	0%

4.5. Experimento 5: Distribuição de Carga de Roteamento

Este experimento mostra os efeitos do algoritmo BiO4SeL na distribuição da responsabilidade de encaminhamento de pacotes entre os nós da rede com relação aos protocolos AODV e Arama. Como o BiO4SeL desencoraja a passagem de pacotes através de nós com baixos níveis de bateria, através de seu esquema de aceleração da evaporação de feromônios, a consequência esperada é que nós com mais níveis de bateria encaminhem mais pacotes, enquanto os pacotes com pouca bateria seriam poupados dessa tarefa. Após um certo tempo, é esperado que os nós apresentem uma distribuição de níveis de bateria mais uniformes.

Para verificar esse experimento, foi calculado o coeficiente de variação dos níveis de bateria entre todos os nós, nos diferentes cenários. O resultado é apresentado na Figura 3.



(a) 40 nós. (b) 80 nós.

(c) 160 nós. (d) 320 nós.

Figura 3. Coeficiente de Variação dos Níveis de Bateria dos Nós da Rede (Experimento 5).

Os resultados mostraram que o BiO4SeL efetivamente reduziu o coeficiente de variação no cenário com 40 nós (Figura 3(a)). Nos demais cenários (Figuras 3(b)–3(d)), o BiO4SeL só não obteve menor coeficiente de variação do que o Arama. No entanto, a razão pela qual o Arama tem, nestes cenários, baixo coeficiente de variação é o fato deste protocolo não ter conseguido configurar rotas. Desta forma, a maioria dos nós, sem ter demanda de ncaminhamento de pacotes, apresentam um comportamento de gasto de bateria uniforme, apenas através do envio periódico de *hellos*. Apenas os nós próximos ao sorvedouro e os nós vizinhos àquele que produz dados apresentam gasto energético diferenciado. No entanto, tando as formigas que tentam estabelecer rotas próximo ao sorvedouro quanto os pacotes de dados produzidos que não podem ser encaminhados ao sorvedouro por falta de rotas acabam por se extinguir sem percorrer outros nós, por colisão, por exemplo.

5. Conclusões e Trabalhos Futuros

Este artigo apresenta o projeto, implementação e experimentação de um novo protocolo de roteamento biologicamente inspirado, chamado BiO4SeL, cujo objetivo principal é maximizar o tempo de vida de uma rede de sensores sem fio. O protocolo inspira-se na inteligência de enxame encontrada em colônias de formigas com o objetivo de agregar autonomia, distribuição e eficiência ao algoritmo.

O algoritmo foi modelado utilizando o simulador de redes NS-2. Os resultados

obtidos a partir da comparação do BiOSeL com as abordagens de roteamento *Ad-hoc* AODV e ARAMA mostraram que o BiO4SeL foi eficiente para atingir seus objetivos. O algoritmo mostrou melhor fator de escala com relação ao aumento de tamanho dos cenários (em área geográfica e quantidade de nós) do que os outros dois. O tempo de vida da rede de sensores foi efetivamente aumentado em todos os cenários comparados, ao mesmo tempo em que manteve uma melhor taxa de entrega de pacotes e melhor conectividade. Os experimentos mostraram também que o BiO4SeL conseguiu melhorar o consumo de energia da rede como um todo através da redução da sinalização e do emprego de *piggybacking* sempre que possível.

O BiO4SeL apresentou seus melhores resultados em cenários onde apenas alguns nós produzem dados, o que pode ser considerado seu cenário ideal de aplicação. Outros testes devem ser realizados com o intuito de avaliar como o algoritmo se comporta quando uma quantidade maior de nós produzem dados ao mesmo tempo.

Além disso, apesar do algoritmo ter sido projetado para suportar mobilidade de nós e oferecer suporte a mais de um sorvedouro, novos experimentos precisam ser realizados com o objetivo de verificar se o bom desempenho é mantido.

Referências

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38:393–422.

Al-Karaki, J. N. and Al-Mashaqbeh, G. A. (2007). Energy-centric routing in wireless sensor networks. *Microprocess. Microsyst.*, 31(4):252–262.

Caro, G. D. and Dorigo, M. (1997). AntNet: a mobile agents approach to adaptive routing. Technical Report IRIDIA/97-12, Université Libre de Bruxelles, Belgium.

Dorigo, M., Maniezzo, V., and Colorni, A. (1991). The ant system: An autocatalytic optimizing process. Technical Report 91-016,, Politecnico di Milano, Italy. (Revised version Sub' IEEE Transactions on Systems, Man, and Cybernetics).

Hussein, O., Saadawi, T., and Lee, M. J. (2005). Probability routing algorithm for mobile ad hoc networks' resources management. *IEEE Journal on Selected Areas in Communications*, 23(12):2248–2259.

IEEE (2003). *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*. IEEE 802.15.4.

Iqbal, M., Gondal, I., and Dooley, L. (2006). Online load balancing for energy-aware anycast routing. In *APCC '06: Asia-Pacific Conference on Communications*, pages 771–775, Washington, DC, USA. IEEE Computer Society.

NS-2 Webpage (2010). `http://isi.edu/nsnam/ns/`. Last access on December, 06, 2010.

Perkins, C., Belding-Royer, E., and Das, S. (2003). Ad hoc on-demand distance vector (AODV) routing. IETF RFC 3561.

Shuang, B., Li, Y., Li, Z., and Chen, J. (2007). An ant-based on-demand energy routing protocol for ad hoc wireless networks. In *WiCom '07 : Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1516–1519, Washington, DC, USA. IEEE Computer Society.

Influence of Shared Memory and Network Topology in the Consensus Dynamics of a Naming Game

Thaís G. Uzun, Reginaldo J. Da Silva-Filho, Matthias R. Brust, Carlos H. C. Ribeiro

Postgraduate Program in Computing and Electronical Engineering – Technological Institute of Aeronautics (ITA) – São José dos Campos, SP – Brazil

thaisgobet@gmail.com, reginaldo@aluno.ita.br,
matthias.brust@gmail.com, carlos@ita.br.

Abstract. *In the Naming Game, individuals or agents exchange pairwise local information in order to communicate about objects in a common environment. The goal of the game is to reach a consensus about naming these objects. In this paper we extend the classical naming game with a globally shared memory accessible by all agents. Although the extended naming game is non-deterministic in its word selection, we show that consensus towards a common vocabulary is reached in diverse network topologies. More importantly, we show the qualitative and quantitative influence of the external source of information, i.e. the shared memory, on the consensus dynamics.*

1. Introduction

The natural emergence of a common language among individuals remains a phenomenon yet to be explained. However, a deeper understanding of the evolutionary processes of language formation is indispensable for developing autonomous multi-agent systems where each agent can potentially have different origins and where no knowledge about the language used in an open-ended environment is provided. To put in a question: How can these agents build a common language through local agreements and reach a consensus about the meaning of their vocabulary on a global scale?

A promising model for a deeper understanding of the common language phenomenon is the *naming game* [Steels 1998]. It describes a model in which individuals can reach a consensus on how to name different objects. All individuals (or agents) exist in the same environment, sense the same set of objects, and are able to invent or create words for these objects. An interaction between two agents is a word transmission from one agent (the *speaker*) to a second agent (the *hearer*), with a resulting outcome (usually success or failure). The goal of the game is to reach, after a number of interactions, an agreement among the agents about the object-word association used for a single object [De Vylder and Tuyls 2006]. Interestingly, a self-organized vocabulary or even a common language with syntactic and semantic levels can be built using such simple local communication process [Steels 1996]. The naming game is therefore a microscopic model for the interaction dynamics among autonomous agents that communicate without any centralized control [Baronchelli et al. 2008]. Such distributed model could be used to understand how large populations reach an agreement with respect to the usage of a

certain word, how new language constructs are established, how rumors and opinions spread, how words propagate in social networks, and even to provide a basis to an emergent communication system where symbol grounding and intentionality are natural outcomes from a particular system dynamics [Steels *et al*. 2007].

Besides its application in modeling the language formation process for individuals, agents or robots (in particular in the field of Artificial Intelligence), the naming game is of relevance to understand the consensus dynamics of collaborative tagging systems of web sites like *delicious* and *flickr* [Marlow et al. 2006] that have become increasingly popular in recent years. The users of such sites can attach keywords or tags to provide information (e.g., favorite sites on the Internet). In a recent study [Golder and Huberman 2005], it is shown how collaborative tagging can lead to both regularities regarding users' activities, tag frequency and keyword usage, and stabilities concerning relative proportions between tags for a given URL and strings that define the location of programs or files in the Internet. Although it is potentially possible to have a constantly increasing number of tags, these findings indicate convergence to a name descriptor (the collection of tags) and concept (the contents in the location itself).

Different variants of naming games played by humans can help to overcome one of the most challenging problems for search engines: Image labeling. The ESP game [Ahn and Dabbish 2004] aims to use humans' perceptual abilities in order to create valuable output in the process of image labeling. Two players are shown the same image but they are not able to communicate. They are then asked to describe the image with labels under a given time constraint (e.g. Google Image Labeler uses 2 minutes). As soon as they use a common label, it is saved in the database to index the image, the players earn points accordingly, and the next image is shown. The objective is to get as many points as possible. While the ESP game is initially designed for a two-player game, in a broader context, the label consensus dynamics of the naming game can be directly used to improve the description accuracy of the images.

Research on the naming game uses mainly the introduced communication model above and focuses on showing its convergence empirically [Baronchelli et al. 2006, Steels 1996, Steels 1998]. Convergence of a deterministic naming game, however, has been mathematically proven [De Vylder and Tuyls 2006]. One common characteristic of these models is that their dynamics are influenced only by the local memories of the agents involved. There is no common access memory, implying that the dynamics of these models is completely uncoupled from any influence of an environment external to the one where the negotiation occurs. The consensus, when reached, is a consensus which belongs to a specific population, and makes sense only in that context. From a sociological point of view, such an arrangement can be plainly artificial, or at least very difficult to establish [Carrington et al. 2007].

The variation of the naming game model introduced in this paper differs from these approaches in that it enables agents to access a shared (global) memory with a given probability p (see Fig. 1). The reason for introducing a shared memory originates from the fact that the real world consists of central access points like books, media, and conferences where individuals build a common vocabulary even without a single direct interaction. Additionally, often an individual tends to search for an external reference before even emitting an opinion about a given subject. The shared memory extension might thus be important for modeling and understanding e.g. the influence of the press and media on the consensus of the group of individuals.

Since classical naming games that allow only local negotiations tend to converge [De Vylder and Tuyls 2006], it appears reasonable that an extended version using a shared memory should behave similarly. Although one of our contributions is to show that the extended naming game in fact converges, the focus is on the role and impact of a shared memory on the convergence behavior itself. Knowledge about this influence enables the possibility to control the convergence and, thus, to trigger the outcomes. Another contribution of this paper shows that against common sense expectation, the shared memory is not solely responsible for triggering the consensus word, thus giving importance backing to the importance of local interactions.

Moreover, we also consider that systems as diverse as the World Wide Web are best described as networks with complex topologies [Barabási and Albert 1999]. In fact, a common property of many large networks is that the vertex connectivity follows a scale-free power law distribution. A model based in preferential attachment to nodes that has this property is the Barabási-Albert model (BA) [Barabási and Albert 1999], and the influence of this topology on the consensus dynamics is also analyzed in this paper.

The paper is organized as follows. A detailed description of the shared memory-based model is given in Section II. The model has been implemented as a proof-of-concept prototype. Empirical results are shown and discussed in Section III. Section IV presents a discussion about the influence of the network topology in the dynamics behavior, and also presents empirical results about the model presented in this paper, when applied to BA networks. Finally, Section V presents the main conclusions of the paper.

2. A Non-Deterministic Naming Game Model

This section describes formally the model proposed in Section I. A population of N agents is considered. Each agent has access to a local memory, which can contain potentially any number of words about a given subject. A word can be a composition of alphabetic elements, but also any other kind of unique identifier. Furthermore, all agents have reading access to the common external (shared) memory. This shared memory contains — prior to the beginning of the game — C distinct words ($C \geq 1$). The objective of the game is to reach a steady state (consensus), i.e. a state in which all agents have the same word in their local memories.

At t=0, all agents have empty local memories. At each successive time step (*t=1,2,3,…*) two agents are randomly selected, one playing the role of the speaker and the other as the hearer. The negotiation dynamics is as follows:

1. The speaker randomly selects one of the words in its own local memory. If the local memory is empty, two actions are possible: a) with probability λ the speaker chooses a word from the shared memory whereby the selected word is added to the local memory of the speaker, or b) with probability 1- λ a new word is created locally and selected.

2. The speaker transmits the selected word to the hearer.

3. If the hearer does not have the transmitted word in its local memory, the interaction is considered a failure and the hearer adds the transmitted word to the local memory. If the hearer has the transmitted word in the local memory, two actions are possible. In both cases, the negotiation is considered a success: a) with probability λ the agents involved consult the shared memory. If the transmitted word exists in the shared memory, the

speaker and hearer remove all other words from their memories, or b) with probability *1-λ* both agents remove all words, besides the transmitted one, from their local memories.

The model has three inputs: the number of agents *N*, the probability *λ*, and the number of words *C* in the shared memory. The probability *λ* represents the tendency for the agents to check the shared memory. When *λ=0*, the model is reduced to the standard naming game described e.g. in [Baronchelli et al. 2006], i.e. without any external influences which are represented by the shared memory. When *λ=1*, the game can be interpreted as a controlled version of the naming game having *C* possible words.

3. Simulation and Results for MF Topologies

The extended naming game communication model has been implemented as a proof-of-concept prototype. Empirical results obtained by simulation are evaluated and discussed in this section.

We assume that the agents are in a fully connected network, where each agent can communicate with all others. This topology is referred in Statistical Mechanics as a Mean Field topology (MF), and is also assumed in the communication models described in [Baronchelli et al. 2006, De Vylder and Tuyls 2006, Steels 1996, Steels 1998]. The number of agents N is set to 100 in all simulations. The values of *λ* vary from 0.0 to 1.0, and we tested values of *C* as 1, 5, 10, 50, 100 and 500. For each combination (*λ*,*C*) the game was executed 1000 times. The results shown are averages over these runs.

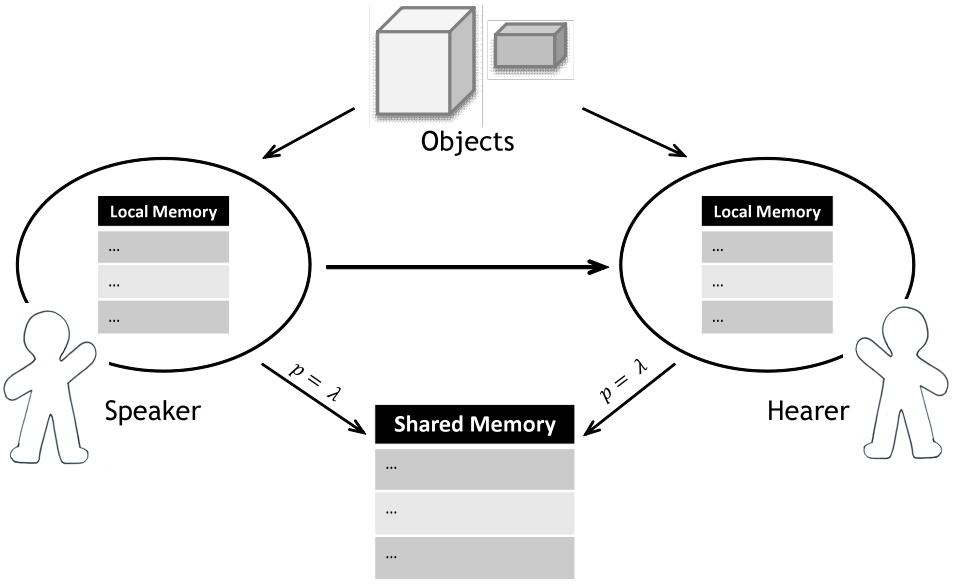


Figure 1. Illustration of the extended naming game using a shared memory that agents are able to access with probability λ.

There are three default measurements for the naming game, see e.g. [Baronchelli et al. 2007]. The first one is the variation of the total number of words as a function of time $N_w(t)$. For a given time step *t*, the value of $N_w(t)$ is the sum of the number of words in the local memories of all agents. Second, we define $N_d(t)$ as a function that gives the number of different words at time *t*, i.e. it is the number of elements of the set containing all the words in the model at time *t*. Third, we define the success rate *S(t)* as follows: In a given interaction between two agents, the value 1 is assigned if the

interaction is a success and 0 if it is a failure. It is important to note that for a given execution the success rate *S(t)* can only have values either 0 or 1.

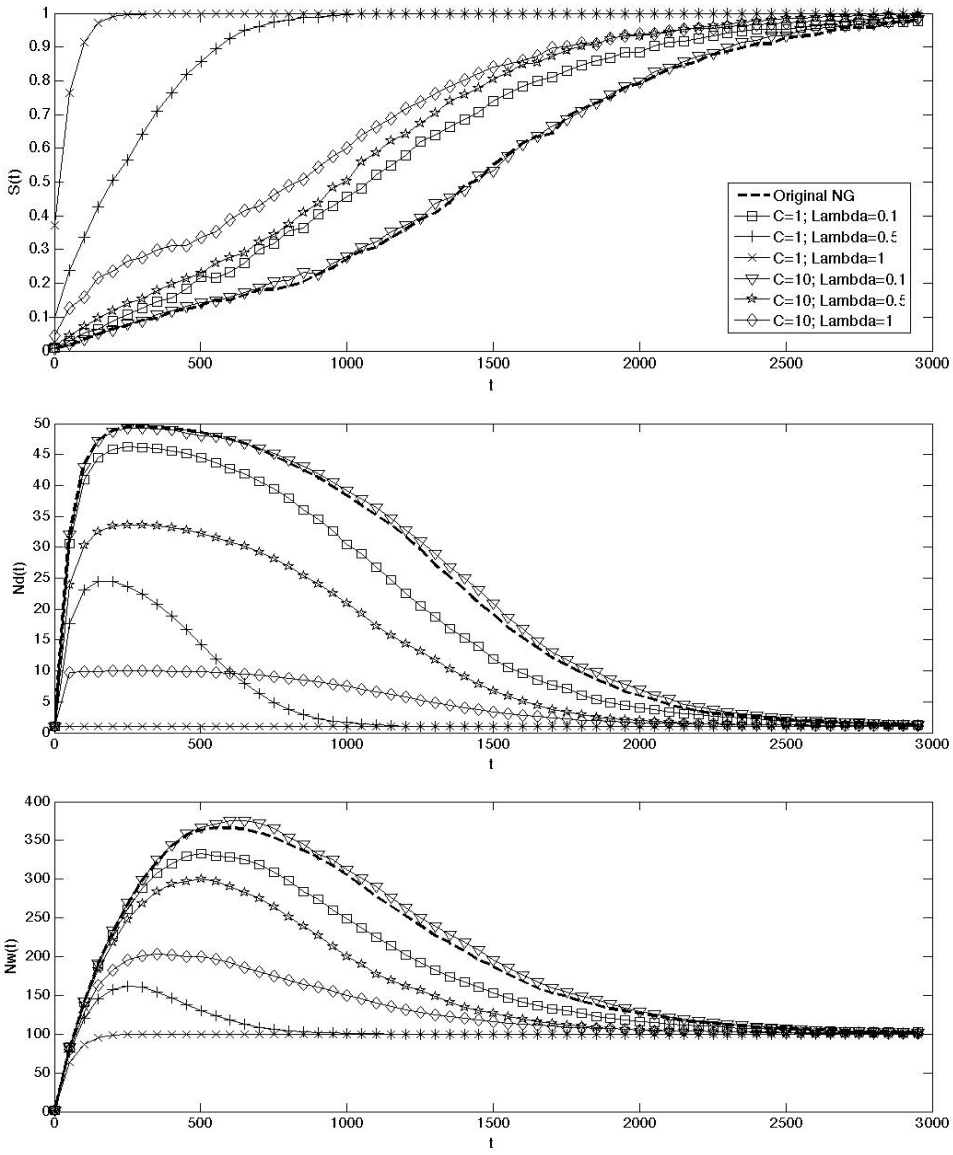


Figure 2. Curves for N_w, N_d and S as a function of time for N=100 agents playing the game described by the introduced model for a *MF* topology.

Figure 2 is an overview of the behavior of the basic properties of the system using the introduced shared memory in a MF topology. It shows that the system dynamics is influenced by both λ and C. The dark blue curves show the results for the standard naming game [Baronchelli et al. 2006], which occurs when λ =0. It is also possible to see that the system clearly undergoes a disorder/order transition. At the beginning, the total number of words in the system $N_w(t)$ grows smoothly, indicating that unsuccessful interactions occur, a fact that can be confirmed by the low value of $S(t)$. On the other hand, the number of different words $N_d(t)$, grows significantly, quickly reaching its maximum value. This means that new words are introduced. Still at the beginning, $N_d(t)$ begins to decrease, although somewhat moderately, while the value of $N_w(t)$ is still increasing. This means that although successful interactions start to

occur, failures are still predominant. After $N_d(t)$ reaches its maximum, no new words are further created. Instead, the initially created words spread all over the network. The difference compared to the initial phase is in the fact that the rate for creating new words is steadily decreasing. The value of S(t) grows moderately at first, but when the existing words are propagated to the majority of the agents, some become very popular, and the success rate starts to grow at a faster pace. With the more frequent occurrence of successful interactions, both the total number of words and the number of different words decrease, eventually leading to a consensus state, where $N_d=1$ and $N_w=N$.

An important issue is to analyze how the input parameters λ and C influence the behavior of the system. In other words, it is important to verify how the shared memory affects the game dynamics. The most clearly affected property is the maximum value of N_d(t), max(N_d), which is the maximum number of distinct words in the system. For a fixed value of C, max(N_d) decreases for increasing values of λ. On the other hand, for a fixed value of λ, max(N_d) increases for increasing values of C. Figure 3 shows the behavior of max(N_d) with respect to λ and C.

Observe that checking of shared memory by the agents as described in Section II is done with probability λ. It can potentially happen in two situations: (a) an agent receives a transmitted word or (b) an agent is selected as speaker and does not have a word in its local memory. In the latter case, the agent can choose one of the words of the shared memory with probability λ or invents a new word with probability $1-\lambda$.

In the classical naming game with N agents [Baronchelli et al. 2006], max(N_d) is approximately $N/2$, meaning that on average half of the agents invent new words. This happens because the inventing agents were chosen as speakers while their local memories were empty. With the introduction of the shared memory, this behavior is expected as well, so that on average $N/2$ agents are chosen as speakers while their local memories are empty. Amongst these agents, $\lambda N/2$ choose a word from the shared memory for transmission, while $(1-\lambda)N/2$ will introduce (invent) new words, ideally distinct ones. Then, the average maximum number of distinct words expected in the system obeys $\max(N_d(N,\lambda,C)) \leq (1-\lambda)N/2 + NC_d(N,\lambda,C)$, where $NC_d(N,\lambda,C)$ represents the maximum possible number of words chosen by the $\lambda N/2$ agents amongst the C words of the shared memory, in other words $NC_d(N,\lambda,C) = C$ if $\lambda N/2 > C$ and $NC_d(N,\lambda,C) = \lambda N/2$, if $\lambda N/2 \leq C$.

Figure 4 shows the variation of the time in which the number of distinct words in the system reaches its maximum value $t_{\max(Nd)}$. For a fixed C, $t_{\max(Nd)}$ decreases for increasing values of λ. For a fixed value of λ, $t_{\max(Nd)}$ increases for increasing values of C. Figure 5 shows the behavior of the average convergence time t_{conv} for the game. We say that the system has converged when every agent has exactly one word, which is the same for all of them, that is, when $N_w=N$ and $N_d=1$. For $C=1$ (only one word in the shared memory), the convergence time always decreases when λ increases. For other values of C, the convergence time is maximum for some λp, increasing in the interval $[0, \lambda p)$ and decreasing in $(\lambda p, 1]$. In general, for a fixed λ, t_{conv} increases for increasing C. When $\lambda=0$, the convergence time obviously does not depend on C and its value is approximately 2,500, in fact the same registered in [Baronchelli et al. 2006] and, thus, indirectly validating the implementation.

The curves for the maximum number of words in the system, max(N_w) are shown in Figure 6. For a fixed value of λ, max(N_w) increases for increasing values of C.

When $C=1$, the value of $\max(N_w)$ always decreases when λ increases. For other values of C, $\max(N_w)$ also reaches its maximum value for some λp, increasing in the interval $[0, \lambda p)$ and decreasing in $(\lambda p, 1]$. Figure 7 shows the behavior of the property $t_{\max(Nw)}$, the time in which the total number of words in the system $N_w(t)$ reaches its maximum value. For a fixed value of λ, $t_{\max(Nw)}$ always increases for increasing values of C.

For all simulations executed, convergence was observed to a state in which all the agents have the same word, i.e. a steady state. Interestingly, the resulting *consensus word* is not always amongst the C words in the shared memory. To analyze this result we define the parameter P_{shared} as the quotient between the number of executions in which the consensus word is also in the shared memory and the total number of executions. It means the probability that a system with inputs N, λ and C converges to a word in the shared memory. The behavior of P_{shared} is shown in Figure 8. Remarkably, the shared memory only contains the consensus word in all executions when $\lambda > 0.5$. For $\lambda < 0.5$ the ratio depends on the number of words C in the shared memory whereby more words mean a lower ratio.

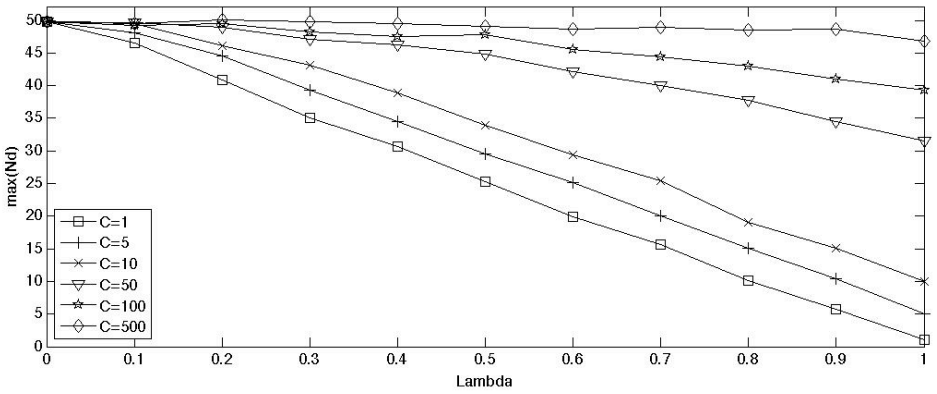


Figure 3. Variation of the maximum number of distinct words in the system $\max(N_d)$ with respect to λ and C for the MF topology.

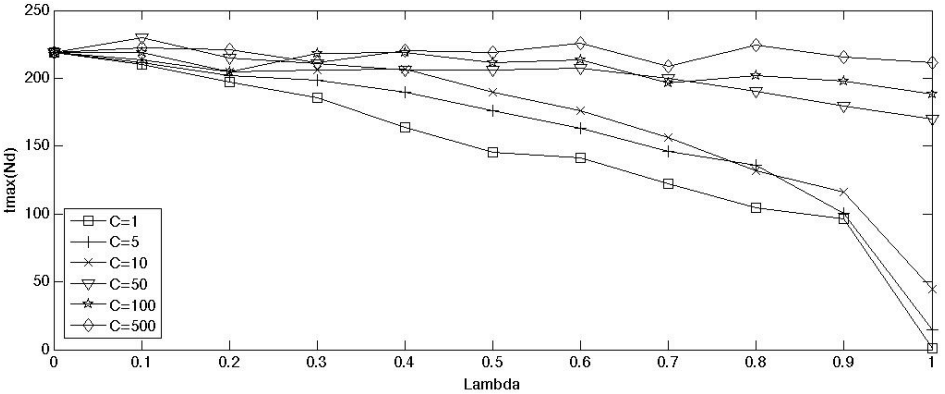


Figure 4. Time when $N_d(t)$ reaches its maximum value $t_{\max(Nd)}$ as a function of λ for the MF topology.

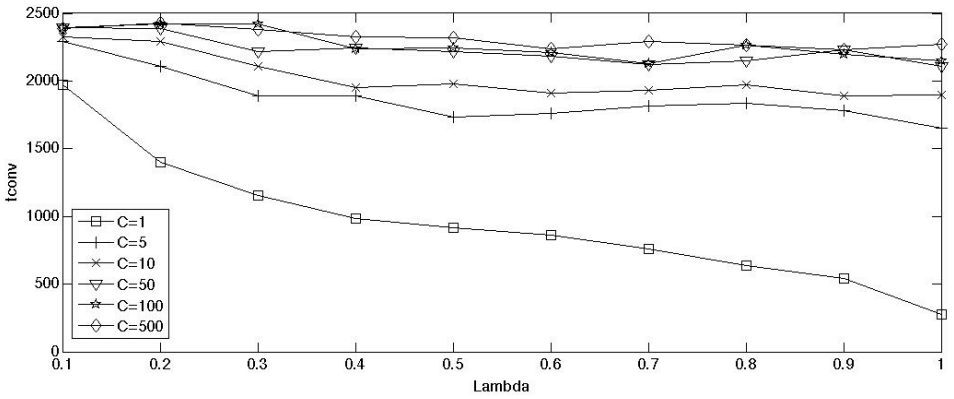


Figure 5. Average convergence times for the proposed model for the MF topology.

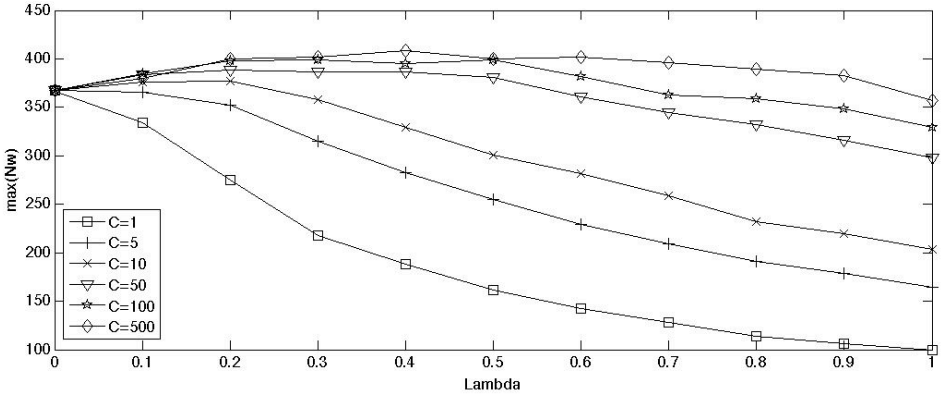


Figure 6. Maximum number of words in the system max(N_w) for the MF topology.

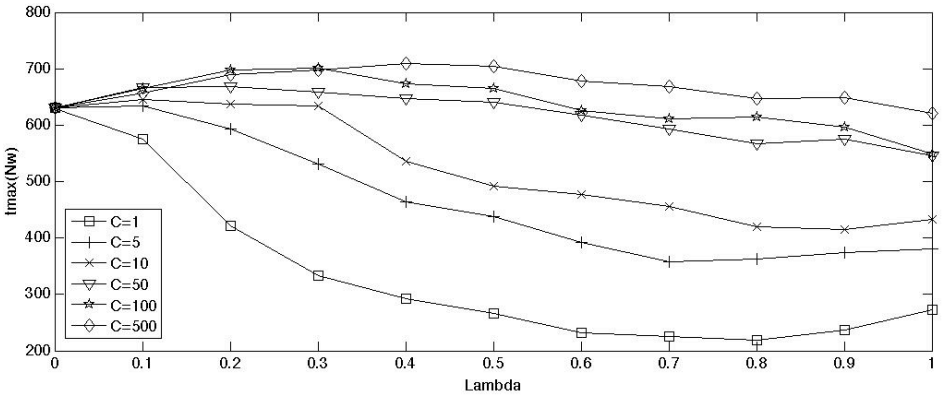


Figure 7. Time in which the total number of words in the system is maximum ($t_{max(Nw)}$) for the MF topology.

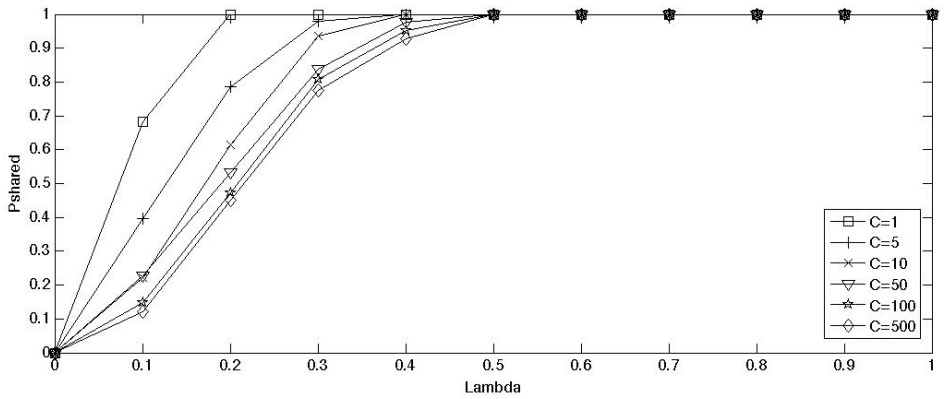


Figure 8. Behavior of P_{shared} as function of λ and C.

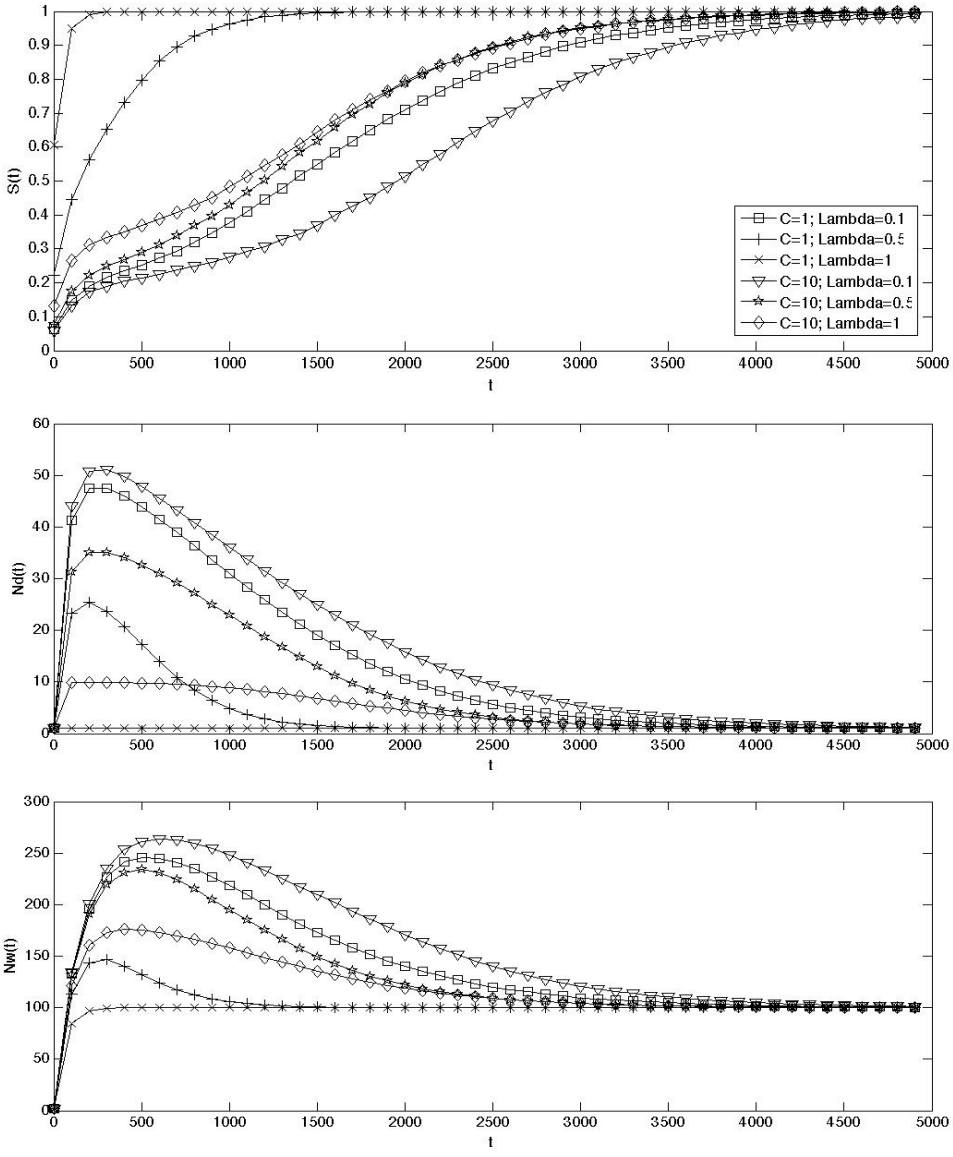


Figure 9. Curves for $N_w(t)$, $N_d(t)$ and S as function of time for $N=100$ agents playing the game described by the introduced model for a BA topology.

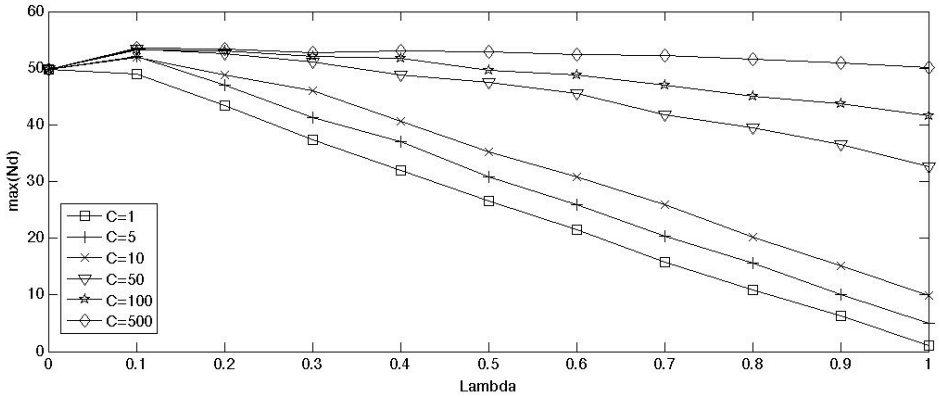


Figure 10. Behavior of the maximum number of distinct words in the system max(N_d) with respect to λ and C for the BA topology.

In order to explain this phenomenon, we consider that invention of words only occur in the very beginning of the game. When λ increases, the number of agents that choose words from the shared memory instead of inventing new ones increases at the same time. Thus, some of the words which were initially exclusively in the shared memory become popular from the very beginning of the game. This explains the fact that the consensus word has a relatively high probability of belonging to the shared memory.

4. Simulation and Results for BA topologies

In the previous section, we assumed that the agents were in a fully connected (MF) network. However, several works report on how the underlying topology influences the consensus behavior [Baronchelli et al. 2006, Baronchelli et al. 2007, Brust et al. 2008, McIntyre and Steels 1999]. In this section, we investigate how the network topology affects the properties of the extended naming game.

Many topologies of large networks, from the WWW to citation patterns in science, display that, independently of its constituents, the probability $P(k)$ that a vertex in the network interacts with k other vertices decays as a power law that follows $P(k) \sim k^{-\gamma}$. This property is called scale-free [Barabási and Albert 1999]. We consider here the scale-free BA model from [Barabási and Albert 1999], which has become one of the most used models for complex heterogeneous networks. A BA topology is a very simple construct. Starting from a small set of m interconnected nodes, new nodes are introduced one by one. Each new node selects m older nodes for connection according to the preferential attachment rule, i.e., the probability of connecting to a node is proportional to its degree. When a predefined network size is reached, this procedure stops. It can be shown that the obtained network follows a power law distribution $P(k) \sim k^{-\gamma}$, with $\gamma = 3$ [Barabási and Albert 1999]. We thus consider here BA networks with N=100 agents and average degree $\langle k \rangle = 15$. The values of λ and C are the same as in Section III and, for each combination (λ, C), the game was executed 500 times. The results were averaged over these runs. Figure 9 shows the behavior of the parameters $N_d(t)$, $N_w(t)$ and $S(t)$. The system dynamics behaves as in the MF case, influenced by the values of both λ and C.

The most clearly influenced parameter is the maximum value of N_d(t), max(N_d). For a fixed value of C, max(N_d) decreases for increasing values of λ. Also, for a fixed value of λ, max(N_d) increases for increasing values of C, as in the MF case. A comparison between Figure 9 and Figure 2 shows that the maximum total number of words is smaller than in the MF case, while the number of different words remains almost the same. This is due to the fact that the network topology does not influence the creation of new words, but the network has an average connectivity smaller than the MF network. The nodes have access only to the words of their neighbors' inventories, whereas in the MF case all nodes have potential access to all words in the game. In this way, the average memory size of any agent is smaller than in the MF case, as it has more limited access to the existing words. The behavior of max(N_d) with respect to λ and C is shown in Figure 10. Figure 11 shows the behavior of the convergence time t_{conv}. Comparing Figures 11 and 5, one can notice that t_{conv} for BA networks is larger than in the MF case. We can explain this fact also by the smaller average connectivity of the BA networks, resulting in a more local spreading of words, thus taking longer to reach consensus. Figure 12 shows the curves for the maximum number of words in the system, max(N_w). We can notice, comparing once again with the same parameter in the MF case (Figure 6), that the BA case has lower memory use, due to the fact that each node has restricted access to the existing words in the network, as already mentioned in this section. The curves for the time in which $N_w(t)$ reaches its maximum, $t_{\max(Nw)}$ are shown in Figure 13. The behavior of the parameter P_{shared}, defined in the last section, is shown in Figure 14. In the BA case, words are propagated in a more local way than in the MF case. For BA networks, when $\lambda > 0.4$, the shared memory contains a consensus word with high probability, while in the MF this occurs for λ larger than 0.5. This difference happens because, in the BA case, the words are propagated in a more localized form than in the MF case.

When, in a BA network, an agent invents a random word, it can only be present in various network nodes if it is propagated step by step during various interactions. The word presence in different nodes increases the chances of it being the consensus word. On the other hand, when more than one agent selects a same word of the shared memory, few interactions are necessary (relatively to when the word in question was invented) for the word to be in various different network nodes. Thus, in the BA network, the difference between a probability of consensus on a word of the shared memory and the probability of consensus on a random invented word is larger than the same difference for a MF network. In a MF network, an invented word may be in various nodes of the network after few interactions, because every node is connected to all the others. Thus, there is less difference (relatively to the same pattern in a BA network) in the consensus probability of a random word and of a word in the shared memory.

As a final remark, we observe that, regarding the variation of the size of the shared memory C and probability λ, the model maintains qualitatively the same characteristics found in the previous section, with respect to the variation of max(N_d), t_{conv}, max(N_w), $t_{\max(Nw)}$ and P_{shared}.

5. Discussion and Conclusion

In this paper, we introduce a shared external memory into the original naming game model [Steels 1998] and analyzed the resulting consensus dynamics both for completely connected (MF) and scale-free (BA) topologies. The memory can be interpreted as the

role of a dictionary, a popular reference (book, encyclopedia, etc.), the press, or a search engine, in a social network of simple communicating agents.

Results show that if the agents follow the communication rules described in the extended naming game consensus is always reached, *i.e.*, the agents reach an agreement on the vocabulary about the objects in their environment. That happens without centralized control, as the agents only have reading access to the shared memory. We have empirically shown the degree of impact of the external shared memory on the consensus dynamics. It is noteworthy observing that, although the shared memory does not completely determine the consensus word (a possible indication of the importance of local interactions), it has enormous influence in defining it, even for low access probabilities. For further investigations we can consider the idea that some entity has total access to the shared memory (reading and writing). This entity could to a certain extent determine and manipulate the outcome of the game, bringing to the fore an interesting discussion on related media control phenomena for which the model could be understood as a first simplification. As the value of λ does not even have to be the same for all agents, it might be more realistic, as far as a study on social communications is concerned, to consider that each agent has its own value of λ, simulating different likelihoods of being influenced by an external source of information. The characteristics of the external shared memory (including but not limited to the value of *C)* are in a certain way determined by the entities that control the external memory. In other words, if these entities allow agents to access only a limited set of possible words, this will result in lowering *C*. On the other hand, with a low value of *C*, it is easier to predict the outcome of the game: the consensus word will very likely be amongst those words that are "interesting" to the entities that control the memory, even if the system has small λ. Further studies will consider socially-related concepts such as trust and reputation in the line initiated in [Brigatti 2007], and the inclusion of utility measures as considered in the multi-agent games literature.

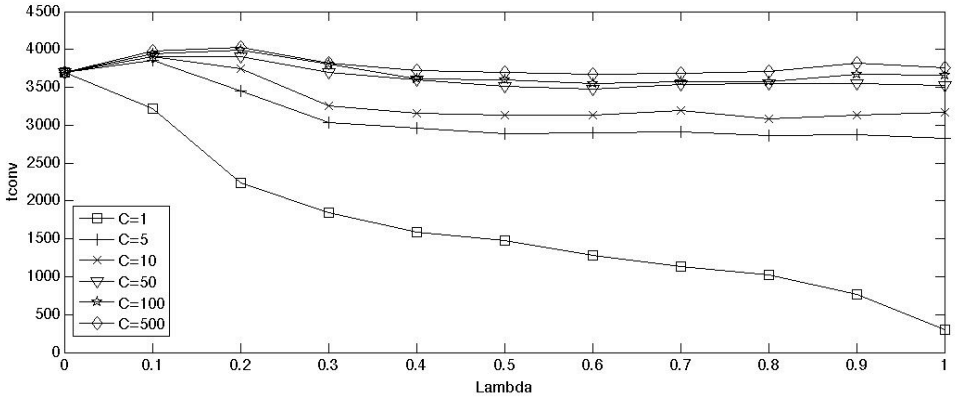


Figure 11. Average convergence times for the proposed model for the BA topology.

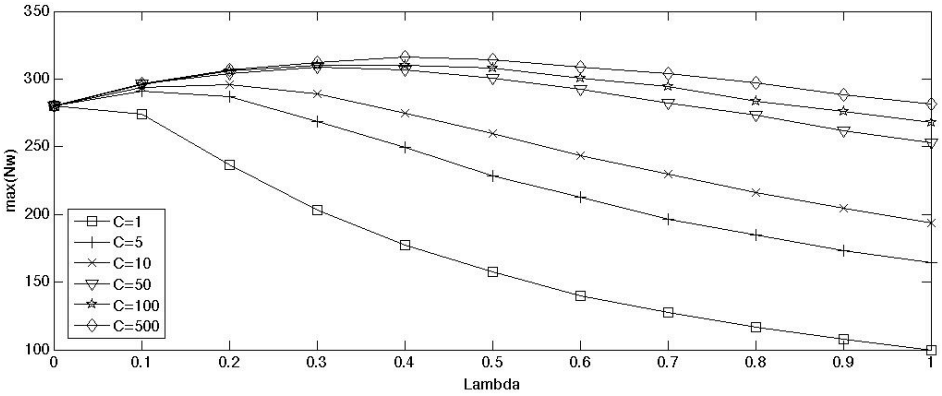


Figure 12. Maximum number of words max(N_w) for the BA topology.

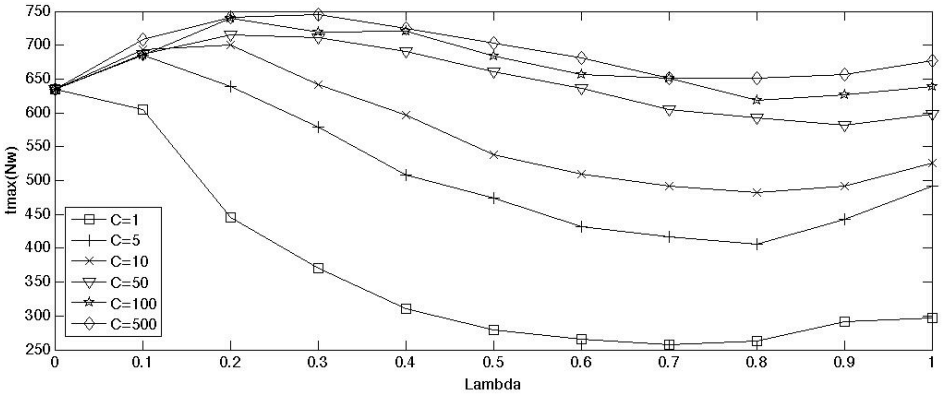


Figure 13. Average time in which the total number of words in the system is maximum $t_{\max(N_w)}$ for the BA topology.

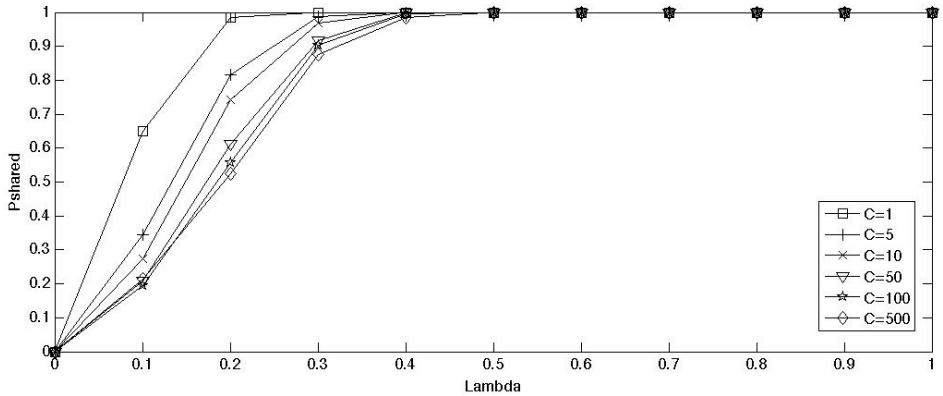


Figure 14. P_{shared} versus λ and C for the proposed model in BA networks.

6. References

Ahn, L.V. and Dabbish, L. "Labeling images with a computer game," *Conference on Human Factors in Computing Systems*, 2004.

Barabási, A. and Albert, R. “Emergence of scaling in Random Networks”, *Science*, 286, pp. 509-512, 1999.

Baronchelli, A., Barrat, A., Dall’Asta, L. and Loreto, V. “Non-equilibrium dynamics of language games on complex networks”, *Physical Review* E, vol. 74, 2006.

Baronchelli, A., Barrat, A., Dall’Asta, L. and Loreto, V. "The role of topology on the dynamics of the Naming Game," *The European Physical Journal Special Topics*, vol. 143, pp. 233-235, 2007.

Baronchelli, A., Felici, M., Loreto, V., Caglioti, E. and Steels, L. "Sharp transition towards shared vocabularies in multi-agent systems," *Journal of Statistical Mechanics: Theory and Experiment*, 2006.

Baronchelli, A., Loreto, V. and Steels, L. "In-depth analysis of the Naming Game dynamics: the homogeneous mixing case," vol. 19, 2008.

Brigatti, E. "Consequence of reputation in an open-ended Naming Game," *Physical*

Review E, 78(4):046108, 2008.

Brust, M.R., Ribeiro, C.H. and Mesit, J. "Avoiding Greediness in Cooperative Peer-to-Peer Networks," *Collaborative Computing: Networking, Applications and Worksharing*, E.B. Joshi and J.B. D., Orlando, USA: Springer Berlin Heidelberg, pp. 370-378, 2008.

Carrington, P., Scott, J. and Wasserman, S. *Models and Methods in Social Network Analysis*, Cambridge Univ. Press, 2007.

De Vylder, B. and Tuyls, K."How to reach linguistic consensus: a proof of convergence for the naming game," *Journal of theoretical biology*, vol. 242, pp. 818-31, 2006.

Golder, S.A. and Huberman, A. "The Structure of Collaborative Tagging Systems," Social Computing Lab, 2005.

Marlow, C., Naaman, M., Boyd, D., and Davis, M. "HT06, tagging paper, taxonomy, Flickr, academic article, to read," *Conference on Hypertext and Hypermedia*, 2006.

McIntyre, A., and Steels, L. "Spatially Distributed Naming Games," *Advances in Complex Systems*, vol. 1, pp. 301-323, 1999.

Steels, L. "Self-Organizing Vocabularies," *Proceedings of the Fifth International Workshop on the Synthesis and Simulation of Living Systems*, C.G. Langton and K. Shimohara, Nara, Japan, 1996.

Steels, L. "The Origins of Ontologies and Communication Conventions in Multi-Agent Systems," *Autonomous Agents and Multi-Agent Systems*, vol. 1, pp. 169 - 194, 1998.

Steels, L., Loetzsch, M. and Spranger, M., "Semiotic Dynamics Solves the Symbol Grounding Problem", Available from Nature Precedings <http://hdl.handle.net/10101/npre.2007.1234.1>, 2007.

Configuração automática de elementos em modelos heterogêneos de simulação

Alex Guimarães Cardoso de Sá[1], **Braulio Adriano de Mello**[2]

[1]Departamento de Ciência da Computação – Universidade Federal de Lavras (UFLA)
Lavras – MG – Brasil.

[2]Ciência da Computação – Universidade Federal da Fronteira do Sul (UFFS)
Chapecó – SC – Brasil.

{alexgsa@comp.ufla.br, braulio@uffs.edu.br}

Abstract. *The automatic configuration of heterogeneous simulation elements can improve the demanded time and reduce erros for the project of a model. However, this is a very complex task due to the variability of the elements used to make models. Thus, this work presents an automatic configuration tool for these kinds of elements performing inside the Distributed Co-simulation Backbone (DCB) architecture. For this job, the configurator handle informations about elements in a distributed data base to generate configuration files for each element which is integrated at the simulation model. The automatic building of these files is the focus of this work. This paper also presents a solution for integration of real elements in simulated models. This integration is advantaged when the elements have a complex behavior. During the study case, this paper shows a discussion about the correct variability precision representation to describe the systems behavior in the context of Precision Agriculture.*

Resumo. *A configuração automática de elementos heterogêneos de simulação contribui com agilidade e redução de erros na tarefa de construção de modelos. No entanto, esta é uma atividade complexa devido à variabilidade das características dos elementos utilizados na composição dos modelos. Este trabalho apresenta um módulo de configuração automática de elementos de simulação para a arquitetura DCB (Distributed Co-simulation Backbone). O configurador traduz informações do elemento, gerenciadas por um repositório distribuído de elementos, em arquivos de configuração, cujo formato é reconhecido pelos módulos de execução de modelos heterogêneos do DCB. A geração automática desses arquivos é o foco deste trabalho. O trabalho também apresenta uma solução para integração de elementos reais em modelos simulados. A integração de elementos reais é vantajosa quando a sua representação é muito complexa ou inviável. Nos estudos de caso o trabalho apresenta experimentos na representação da variabilidade de precisão em elementos usados na composição de modelos no contexto da Agricultura de Precisão.*

1. Introdução

As técnicas de modelagem e simulação de sistemas contribuem para identificar e corrigir problemas, diminuir custos de produção, detectar falhas antes da construção de protótipos (em sistemas novos) e validar o comportamento completo de sistemas. De acordo com os

recursos utilizados para a sua construção, modelos podem ser considerados homogêneos ou heterogêneos. Ao contrário de modelos homogêneos, os modelos heterogêneos utilizam partes que se diferenciam em termos de linguagem de descrição, interface e modo de tratamento do tempo. Essas características podem onerar a representação de sistemas reais que demandam a composição de partes (elementos) distintas. Essa dificuldade também está sofre influência da dinamicidade, distribuição e/ou precisão requeridas pelo modelo a ser construído.

Um dos grandes desafios encontrados na construção de modelos heterogêneos está na comunicação entre elementos simulados com tecnologias heterogêneas (distintas) de construção. Este desafio está afinado com o segundo dos grandes desafios da pesquisa em computação, a Modelagem de Sistemas Complexos. Grandes esforços podem ser gastos na tentativa de estabelecer meios de comunicação entre elementos que possuem partes distintas. A bibliografia retrata alguns estudos que propõem soluções para a simulação heterogênea. Dentre eles está a *High Level Architecture* (HLA) [IEEE 2009], que apresenta técnicas de interoperabilidade entre elementos (federados) de um modelo (federação). Porém, a interferência no código dos elementos, necessária para cooperação distribuída, pode gerar problemas difíceis ou impossíveis de tratar. Existem trabalhos, no contexto da HLA, que tem por objetivo evitar modificações nos códigos dos elementos [Tolk 2002].

Já a arquitetura do *Distributed Co-simulation Backbone* (DCB) [Mello et al. 2005], objeto de estudo desse trabalho, faz uso de técnicas de cooperação entre elementos heterogêneos (e/ou distribuídos) de simulação que preservam a integridade do elemento. A ideia central do DCB é que o elemento se torne independente do restante do modelo, comunicando-se apenas com seu *gateway* [Strassburger et al. 1998], que o preserva de primitivas de comunicação. O *gateway* é o módulo do DCB que controla a comunicação do elemento com o restante do modelo. A existência de um *gateway* privativo para cada elemento facilita o reuso mesmo sob restrições de propriedade intelectual [Page et al. 1999]. A partir de seus métodos é possível atribuir a responsabilidade das tarefas para os serviços internos do DCB. Outro aspecto que motiva o uso do DCB é a existência de um ambiente de suporte para gerenciamento de repositórios distribuídos de elementos [Mello and Parreiras 2009].

A solução apresentada neste trabalho permite gerar automaticamente os arquivos de configuração de elementos para composição de modelos reconhecidos pela arquitetura do DCB. Esses arquivos de configuração definem a cooperação realizada entre cada elemento do modelo do sistema. Entretanto, quando essa configuração é feita sem o uso de ferramentas que elevem o nível de abstração, a necessidade de conhecimentos sobre detalhes internos do ambiente de simulação consomem mais esforço do projetista para a integração de elementos. Isso torna a configuração manual trabalhosa e demorada, o que justifica o desenvolvimento de um configurador que reduz o esforço no trabalho de composição de elementos. O módulo de configuração apresentado neste trabalho está integrado ao Gerenciador de Repositórios Distribuídos de elementos [Mello and Parreiras 2009].

Como segunda parte da configuração de modelos, este trabalho apresenta um módulo que é responsável por gerar automaticamente os arquivos de controle de interface (*gateways*) de cada elemento. A construção desse segundo módulo é justificável pelo mesmo motivo, reduz a necessidade de conhecimentos mais detalhados da arquite-

tura do DCB, de programação e de tarefas realizadas pelos *gateways* (estrutura interna), para que possa construir um modelo de simulação. Além disso, esses arquivos preservam a implementação original dos elementos.

Porém, a representação de um sistema real através de modelos geralmente sofre redução de fidelidade ou perda de detalhes importantes. Há situações em que o sistema real possui partes com interface simples e com comportamento interno complexo de difícil representação. Nesses casos pode ser vantajoso integrar elementos reais em modelos onde as demais partes são simuladas, elevando o grau de heterogeneidade, além da composição de elementos simulados distintos em termos de interface, sincronização do tempo ou tecnologia de desenvolvimento.

Assim, de modo complementar à geração automática da configuração de modelos, este trabalho apresenta também um módulo que permite a integração de elementos reais em modelos simulados à luz da arquitetura do DCB. Experimentos realizados demonstram comparativos do comportamento entre as duas versões de um mesmo modelo, uma delas com todos os elementos simulados e a outra onde os elementos simulados cooperam com um elemento real.

O artigo esta organizado nas seguintes seções: a seção 2 explora as principais características da plataforma de simulação DCB e questões de projeto relacionadas ao Gerenciador de Repositórios Distribuídos. A seção 3 delimita os métodos de configuração mais comuns na literatura. Já na seção 4 é apresentado o configurador e, em seguida, seu estudo de caso realizado sobre o modelo específico para a Agricultura de Precisão. Considera-se na seção 5, a variabilidade dos níveis de precisão do sensor de passagens de sementes. Os estudos sobre a integração de elementos reais ao modelo simulado são apresentados na seção 6. Considerações finais e referências são apresentadas nas duas últimas seções.

2. Plataformas Utilizadas

Essa seção explora as principais características plataformas e seus respectivos componentes, no qual esse trabalho baseia-se. A primeira subseção define as características principais do DCB. E a segunda mostra as funcionalidades dos componentes presentes no Gerenciador de Repositórios Distribuídos de elementos.

2.1. O DCB

O DCB é uma plataforma de simulação que provê de recursos necessários para cooperação de elementos heterogêneos e/ou distribuídos. Todas suas atividades são transparentes ao componente simulado pelo fato de que todas elas são realizadas respeitando políticas internas do próprio DCB. Quatro módulos principais, identificados na Figura 1, são responsáveis pelo controle da cooperação entre elementos: o DCBS, o DCBR, o DCBK e o Gateway.

O DCBS realiza tarefas de envio de mensagens para outros elementos do modelo. O DCBR decodifica/recebe mensagens advindas de elementos origem. Já o DCBK realiza atividades para manter a comunicação entre elementos, respeitando o tempo global do modelo. A necessidade da utilização do Gateway está na comunicação de interfaces distintas dos componentes. A Figura 1 apresenta a estrutura geral do DCB. O único módulo

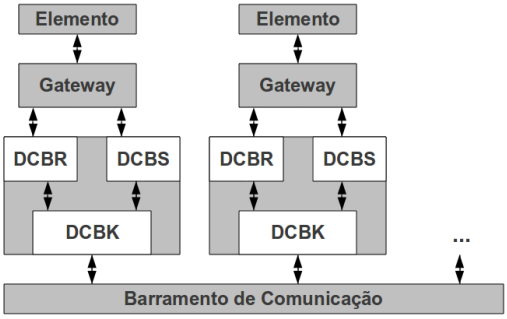


Figura 1. Arquitetura geral do DCB.

interno do DCB que executa operações remotas é o DCBK. Os demais realizam gerenciamento local para garantir a correta cooperação entre o elemento e o restante do modelo.

2.2. Gerenciador de Repositórios Distribuídos

O ambiente de gerenciamento de repositórios distribuídos de elementos foi desenvolvido com o principal objetivo de cadastrar, localizar e gerenciar elementos para facilitar a sua busca, identificação e seleção para que possam ser reutilizados em novos modelos. Os principais benefícios do gerenciador são a agilidade, a facilidade e a confiabilidade na construção e execução de modelos de simulação.

O cadastro de elementos no repositório é requisito para reuso na composição de novos modelos. Quando disponível no repositório, o elemento pode ser identificado pelas suas características e mais facilmente integrado/sincronizado ao modelo. Por isso, é recomendável que todos os elementos do sistema tenham código livre e aberto para ajuste de funcionalidades [Amory et al. 2002].

O módulo de cadastro está integrado ao Gerenciador de Repositórios Distribuídos e atende os requisitos da arquitetura do DCB. O trabalho de integração exigiu modificações no formato de armazenamento de informações sobre atributos e tipo de sincronização do elemento. A Figura 2 mostra a interface de cadastro de elementos e de seus respectivos atributos.

O registro de um elemento tem diversas informações associadas a ele, tais como: porta de entrada, IP, atributos de entrada e de saída e tipo de sincronização no tempo. As condições para geração automática do modelo dependem diretamente das informações sobre o elemento registradas no cadastro (repositório). O configurador, a partir dos dados de cadastro do elemento, gera arquivos em XML e *gateways* no formato identificado pelo DCB para carga e execução de modelos heterogêneos.

3. Métodos de Configuração

A configuração de elementos para a construção de modelos pode ser realizada manualmente ou empregando recursos computacionais para elevação do nível de abstração. A vantagem de recursos computacionais para esse fim recai principalmente sobre a redução de esforço e da necessidade de conhecimento técnico sobre o ambiente de simulação em uso.

Algumas das formas de automatizar etapas de configuração são, por exemplo a configuração por tabelas de vinculação, a configuração por meios de recursos gráficos e

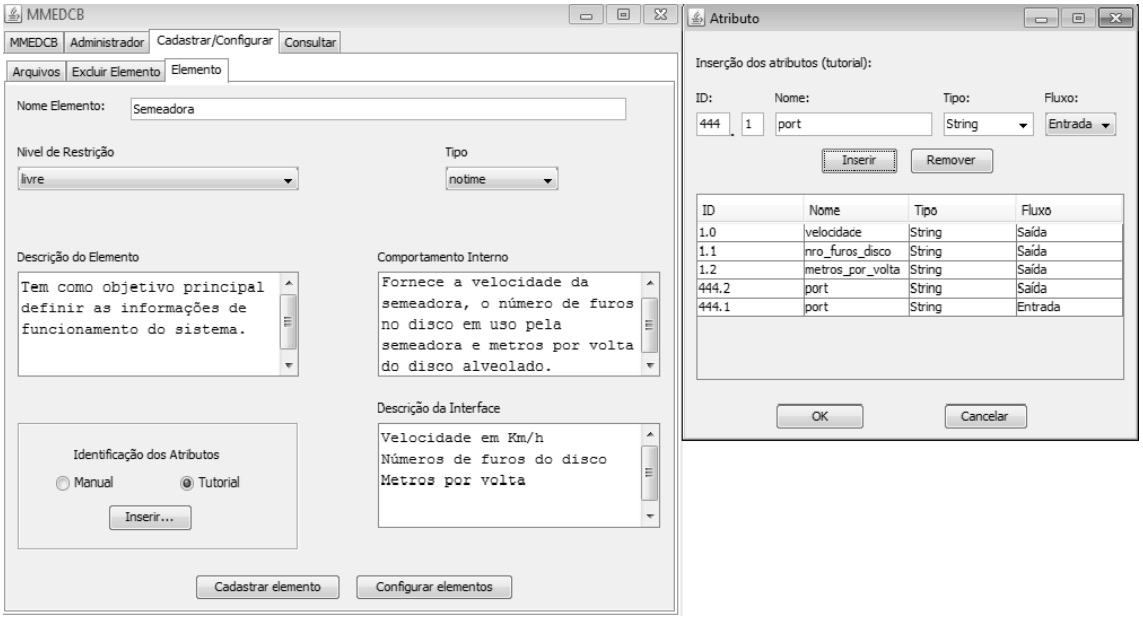


Figura 2. Cadastro de um elemento e de seus atributos no gerenciador.

a configuração por regras. O primeiro modo destaca-se por apresentar todas as peculiaridades do elemento a ser cadastrado em uma tabela, na qual define todas as vinculações de atributos de saída do elemento com os atributos de entrada de outros elementos. O configurador proposto nesse trabalho utiliza esse primeiro tipo de configuração. No segundo, os elementos são representados através de componentes gráficos e toda a configuração é feita por representações gráficas dos elementos e atributos que definem os esquemas para relacioná-los. São exemplos desse tipo de configuração o ambiente ASDA– Ambiente de Simulação Distribuída Automática [Bruschi 2002], o Tangram [Souza et al. 2003] e o wSimul [Merhi 2010] que visam, através de módulos gráficos, realizar a configuração do modelo. A terceira estratégia de configuração é o uso de um conjunto de regras as quais os elementos e os modelos devem respeitar para poder ser realizada a cooperação. A arquitetura HLA segue essa técnica.

4. Módulo de configuração automática de elementos

Esta seção apresenta o trabalho de especificação e construção do módulo de configuração automática. Um estudo de caso também é feito, onde o configurador foi utilizado para gerar os arquivos de configuração e os arquivos *gateways* do SEP (Supervisor Eletrônico de Plantio) [Mello and Caimi 2008]. Esses arquivos foram gerados para que o SEP possa ser executado no DCB usando apenas os arquivos gerados automaticamente pelo configurador. A possibilidade de geração automática do modelo completo sustenta os benefícios no uso de recursos computacionais com a finalidade de agregar agilidade, facilidade e correção aos modelos.

4.1. Persistência da Configuração de Elementos

O configurador permite gerenciar os atributos de entrada e saída dos elementos cadastrados no (s) repositório (s) reconhecidos pelo Gerenciador Repositórios Distribuídos, sejam eles remotos ou locais. Para que isso aconteça de forma transparente ao usuário projetista do modelo, uma interface gráfica é apresentada a fim de melhorar a integração

desse usuário com as tarefas do configurador. Assim, ações como a geração do arquivo em XML (configuração) do elemento que está sendo configurado são transparentes ao usuário. A Figura 3 mostra a interface gráfica de configuração de elementos.

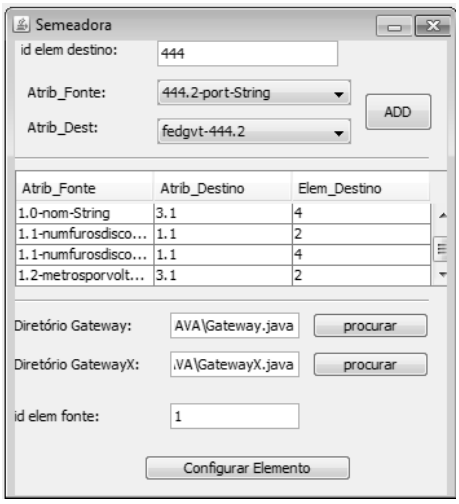


Figura 3. Interface Gráfica de configuração de elementos.

A configuração automática de elementos tem como requisito o cadastro prévio do elemento no banco de dados do Gerenciador de Repositórios Distribuídos de elementos. Esse banco de dados utiliza persistência em XML para registrar os novos elementos. Para os elementos serem configurados, os seus dados, em questão, são carregados pelo configurador que, então, permite ao usuário a visão dos atributos na forma de uma lista, como demonstrado na da Figura 3 e condiciona os dados dos elementos para que esses sejam manipulados corretamente. Para tornar a tarefa de configuração intuitiva, o configurador apresenta na interface, os atributos de saída do elemento em fase de configuração e os atributos de entrada dos demais elementos. O projetista vincula os atributos de saída do elemento (origem) aos atributos de entrada dos elementos (destino) conforme a especificação do modelo em construção. Essa vinculação tem por objetivo definir a cooperação entre dois ou mais elementos do modelo, que está em processo de design.

Para cada elemento escolhido pelo usuário, o configurador mostra uma nova interface como a da Figura 3. É possível, a partir disso, monitorar e controlar quais atributos de entrada estão conectados aos atributos de saída de cada elemento que está sendo configurado. A consistência das conexões entre atributos não é validada pelo configurador. O estudo, especificação e desenvolvimento de estratégias para garantir consistência das conexões é identificada como perspectiva de continuidade do trabalho.

A tarefa de selecionar o atributo fonte e destino pode ser feita repetidamente, para que ocorra vinculação de atributos e, consequentemente, a troca e mensagens entre esses elementos. O registro, das informações de identificação (id) dos elementos destino, é feito no momento da vinculação. Essas informações são adicionadas em uma tabela, para melhor visualização do projetista, com três colunas: a do atributo fonte (“Atrib_Fonte”), a do atributo destino (“Atrib_Dest”) e o id do elemento destino (“Elem_Destino”), que somente é válido se for um inteiro.

O configurador, após esse processo, já pode gerar o arquivo em XML de

configuração do elemento, necessitando apenas da identificação do elemento fonte (“id elem fonte”), essencial para cada elemento do modelo. Porém é ainda necessário esperar a geração dos arquivos de controle de interface para que o processo continue.

Este método de configuração de elementos é bem distinto do ASDA e de outros configuradores, que propõem submódulos de especificação gráfica de modelos. Porém, da mesma forma, esse ambiente busca afastar o usuário da tarefa de transcrição do modelo em um programa de simulação e também da análise estatística dos resultados mostrando as informações da configuração em uma simples tabela.

Todavia, como o módulo apresentado neste trabalho visa especificar detalhes de eventos peculiares à arquitetura do DCB. A composição das características de independência dos elementos, a distribuição geográfica sem que elementos necessitem executar ações de comunicação e a integração de elementos heterogêneos (reais ou simulados) identifica a contribuição do trabalho.

Um ponto relevante no estudo da arquitetura configurador proposto é sua similaridade em relação à outras aplicações que modelam eventos, como por exemplo o Composer [Guimarães et al. 2008], que é um *software* utilizado para mapear eventos para TV digital. O Composer também gera um arquivo no formato XML, bem como o configurador desenvolvido nesse trabalho, para definir quais são os eventos que irão ocorrer ou estão ocorrendo de acordo com a transmissora do canal. Esses eventos são feitos para promover a iteratividade entre usuário e TV. Esta característica demonstra que caso hajam modificações na estrutura de controle de eventos para o DCB, o configurador pode ser utilizado para outros fins.

4.2. Geração Automática dos Gateways

Além dos arquivos de configuração, o DCB requer o ajuste dos arquivos de interface denominados *gateways*. Cada elemento possui dois *gateways* particulares que reconhecem as portas de comunicação com o elemento identificadas na etapa de modelagem como atributos de entrada e saída. Para a geração automática dos *gateways* é necessário preencher os dois campos de diretórios, onde se encontram os *templates* dos *gateways*. Para cada novo elemento, o configurador deve gerar um arquivo GatewayX, onde X é o identificador do elemento, e realizar uma alteração no arquivo Gateway informando a existência do novo elemento. O DCB permite que múltiplos elementos sejam executados em *threads* distintas em uma mesma máquina virtual, por isso a existência de um arquivo Gateway para redirecionar mensagens ao destino correto, conforme a configuração do modelo.

Nos arquivos dos *templates* estão contidas informações que não se alteram dentro dos *gateways*, ou seja, esses arquivos são padrões ou modelos de *gateways*. O primeiro diretório contém o *template* do *gateway* genérico (Gateway), responsável pelas tarefas de inicialização e redirecionamento de tarefas para o *gateway* específico (GatewayX). O segundo diretório contém o *template* do *gateway* específico do elemento que cumprirá tarefas de atualização do tempo virtual local do elemento, conversão de tipos de atributos, caso seja necessário, e atualização de atributos. A ideia de *templates* de *gateways* foi inspirada dos trabalhos de [Sperb 2003] e de [Parreiras 2008]. O configurador também permite editar o conteúdo de ambos os arquivos de *gateways* para a inclusão de ações não previstas pelo configurador.

Preenchidos todos os campos de diretórios dos *templates* dos *gateways* e feita todas as conexões dos atributos fonte e destino, o elemento fonte pode ter sua identificação registrada em um campo da interface. O módulo tem em sua tela quase todas essas informações relevantes que podem ser passadas a um método. Com a adição dos atributos de entrada do elemento como parâmetro desse método é possível gerar o arquivo em XML de configuração desse elemento e seus respectivos *gateways*, seguindo um padrão de modelagem.

Ao final da configuração, um arquivo de saída é gerado para cada elemento, em um formato XML padronizado e dentro de um diretório previamente selecionado pelo usuário, onde constam as informações de cooperação entre elementos do modelo. Também são abertas duas janelas para cada elemento, uma com o *gateway* genérico e outra com o específico, para que possam ser feitas modificações quando os elementos apresentam características não previstas ou exceções. Neste caso é necessário intervenção do projetista no código dos *gateways*, antes de salvá-los como arquivos. Os códigos fontes dos *gateways* são escritos na linguagem de programação Java.

O uso do configurador reduz drasticamente a complexidade do trabalho de construção de modelos. Muitos dos detalhes relacionados ao modo como o vínculo entre atributos deve ser feito são mantidos consistentes pelo configurador. Isso reduz a probabilidade de erros. Outra vantagem é a possibilidade de configurar o comportamento (cooperação) entre múltiplos elementos simultaneamente. Não é necessário concluir a configuração de um elemento para iniciar a configuração do próximo. Esta característica adiciona flexibilidade sem perda de consistência ao trabalho do projetista de modelos.

4.3. Geração automática dos arquivos do modelo SEP

Para validar o configurador proposto, foram gerados automaticamente os arquivos de configuração e os arquivos *gateways* de um modelo de supervisor eletrônico de plantio, o SEP. Isso foi feito para que esse modelo fosse executado no DCB, utilizando somente os arquivos gerados pelo configurador. A Figura 4 mostra a arquitetura geral do SEP. O SEP possui quatro elementos principais: a Semeadora, o Disco Alveolado, o Sensor de Passagem de Sementes e o Centro de Controle.

O modelo construído para realização dos experimentos possui dez unidades do conjunto de elementos formados pelo disco alveolado e pelo sensor. Cada unidade possui autonomia na execução de ações e enviam dados de funcionamento do conjunto disco-sensor para o único elemento de controle (centro de controle) do sistema. Primeiramente, os elementos foram modelados e cadastrados no Gerenciador de Repositórios Distribuídos de elementos.

Em seguida, o configurador automático foi utilizado para compor as regras de cooperação entre os elementos do SEP e geração dos arquivos de configuração e interface (XML e *gateways*). Pequenos ajustes foram realizados no arquivo do DCB que identifica o número de diferentes elementos disparados em *threads* distintas da mesma JVM.

Depois de gerados os arquivos de configuração, o modelo foi executado no DCB tendo como entrada os dados cujos comportamentos eram conhecidos. Os resultados da execução confirmaram os resultados esperados. Além disso, a partir das abordagens utilizadas no configurador, os códigos dos elementos, os arquivos em XML e os arquivos *gateways* gerados com a configuração não foram modificados.

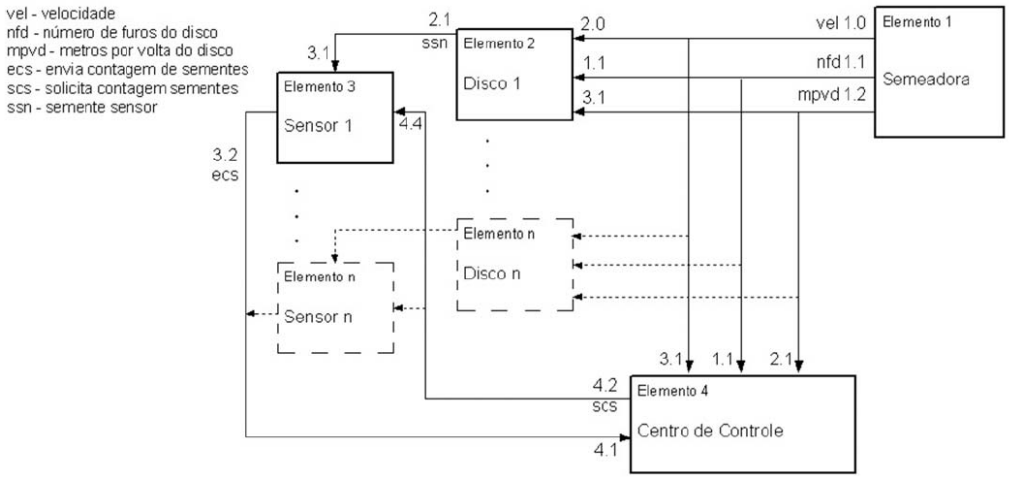


Figura 4. Arquitetura do SEP.

A configuração dos elementos foi feita também manualmente como indicador comparativo do esforço necessário em comparação à configuração automática de modelos. A partir disso pode-se comprovar a redução expressiva do esforço para o projetista do sistema simulado (ou usuário). Pode ser percebido também que a reutilização de componentes é uma das vantagens do gerenciador. A maior facilidade de incluir itens ao modelo a ser criado e em seguida configurá-los de acordo com as preferências do usuário beneficia a fidelidade em relação ao sistema real.

5. Variabilidade dos níveis de precisão do sensor de passagem

Um dos desafios na representação por modelos de sistemas para agricultura de precisão é, justamente, a avaliação dos níveis de precisão dos componentes do sistema. As condições severas de trabalho dos equipamentos e o excesso de resíduos (de qualquer tipo) comprometem expressivamente os níveis de precisão. Para isso, o modelo foi alterado para integrar recursos capazes de estimar a variabilidade na precisão das medidas. A independência entre elementos, característica da arquitetura do DCB, tornou esta tarefa simples. O elemento e o *backbone* do DCB não foram alterados.

O método responsável pelo envio de contagens parciais de sementes para o centro de controle recebeu uma alteração para monitorar e controlar a variabilidade da precisão usando distribuição de probabilidade. A distribuição de Poisson foi utilizada para representar a variabilidade de precisão. Em estudos que aplicam técnicas de simulação em sistemas que demandam altos custos, como a agricultura de precisão, essa distribuição de probabilidade tem sido sugerida [Chowdhury and Koval 2005].

Para o cálculo de lambda, parâmetro necessário ao modelo de Poisson, foram estimados 15 anos de atividade com tempo de inatividade em 30 porcento. Foram utilizados dados de estimativas de estudos já realizados sobre a avaliação de desempenho em equipamentos agrícolas [Pacheco 2000]. Neste caso, para o experimento realizado, o lambda teve valor igual a 3,33. Para o cálculo dos dos fatores que representam a perda de precisão do modelo foi utilizado o algoritmo da Figura 5.

O algoritmo de Poisson segue a ideia de iterar uma variável qualquer (“auxiliar”),

```
Algoritmo Poisson (lambda){

    limite = e^(-lambda);
    auxiliar  = 1;
    interações = 0;

    Faça{
      iterações = iterações + 1;
      auxiliar = auxiliar * Sorteia-Nro-Real(0,1);
    Enquanto (auxiliar > limite);

  Retorna (Iterações – 1);
}
```

Figura 5. Algoritmo para simular variabilidade na precisão dos sensores.

multiplicando-a por um número real aleatório entre zero e um (“Sorteia_Nro_Real(0,1)”), até que essa variável seja maior que o limite calculado durante a inicialização (“limite”). Para calcular esse limite, o algoritmo recebe como parâmetro o valor de lambda. O número de iterações (“iterações”) menos uma unidade é o valor do tempo de espera do sensor para enviar a contagem do número de sementes para o centro de controle. Esse fator representará a perda de precisão do sistema em campos agrícolas. No modelo, essa característica foi colocada em duas unidades disco-sensor.

Os dados utilizados na configuração da semeadora para o experimento foram: velocidade de 12km/h, 20 furos em cada disco alveolado e uma volta do disco significando 4,5 metros percorridos. Com isso, são 4,444 sementes esperadas por metro e média monitorada de 4,410 incluindo os recursos de variabilidade na precisão do sensor de sementes. Dessa forma, os testes realizados sobre o modelo do SEP obtiveram os resultados contidos na Tabela 1. A segunda coluna da tabela mostra os resultados para a representação da perda de precisão (RPP) do sensor em função das condições severas de trabalho. Esses resultados obtiveram uma média de 4 segundos e um desvio padrão de 1,414. Já para os tempos de quedas para 2000 sementes, a média dos tempos foi de 65,5 segundos e seu desvio padrão foi de 0,972.

Tabela 1. Testes de simulação do modelo.

Testes	RPP (s)	Tempo (s) p/ 2000 sementes	Perda de desempenho
1	4	67	10,00%
2	4	67	10,00%
3	2	65	8,00%
4	4	67	10,00%
5	4	67	10,00%
6	2	65	8,00%
7	6	68	12,00%
8	5	66	9,00%
9	6	67	12,00%
10	3	66	9,00%

Para calcular a porcentagem de perda de desempenho dos modelos (quarta coluna), considerou-se antes da inclusão do código que reduz a precisão do modelo, testes para prever qual é o tempo necessário para passarem 2000 sementes nos sensores. O tempo exato foi de 60 segundos.

6. Integração de elementos reais em modelos simulados

A integração de elementos reais com elementos simulados contribui com a fidelidade do modelo em relação ao sistema real e reduz o esforço de construção do modelo. O uso de elementos reais é especialmente apropriado quando a imitação do elemento é muito difícil ou inviável. A provável necessidade de ajustes proprietários na interface entre o elemento real e o modelo é um exemplo de dificuldade para integração de elementos reais. O protocolo de comunicação na interface do elemento real também não pode ser alterado ou simplificado. Se um elemento real gera um volume grande de dados constantemente informações importantes podem ser perdidas durante a transmissão.

Problemas de integração e tratamento de dados no uso de elementos reais podem ser resolvidos com recursos que certificam o recebimento e envio de dados. Um protocolo específico para este fim viabiliza esta solução. O uso de um protocolo padrão também contribui com soluções para detectar eventuais problemas de funcionamento do elemento real, por exemplo, implementando políticas de tempo limite de resposta do elemento (*timeout*). Em [Reynolds 1998] são apresentados alguns dos principais problemas envolvendo a integração de elementos reais em modelos de simulação.

A arquitetura do DCB favorece a integração de elementos reais, no entanto, exige a construção de uma interface específica entre o elemento e o *gateway* do DCB, para que ambos consigam se comunicar adequadamente. Esta seção apresenta uma solução para integração de elementos reais que combinam partes de *hardware* e *software* em modelos suportados pelo DCB. A solução é composta de uma controladora para interface de *hardware* e um protocolo de comunicação para gestão da cooperação entre sinais ou dados recebidos e/ou enviados pelo elemento real.

De acordo com [Fummi et al. 2009], não há metodologia consolidada para o uso de ambientes de simulação heterogênea (co-simulação) em função, essencialmente, do principal desafio que é a complexidade dos modelos [Oraw et al. 2007]. Em alguns casos, quando a representação do elemento real é inviável, a integração do elemento real ao modelo onde os demais elementos são simulados é uma alternativa que substitui o esforço de representação de um elemento complexo pelo trabalho de integração ao modelo.

A integração de elementos reais em modelos simulados ainda é pouco explorada como recurso para tratamento de problemas complexos. Por isso, os ambientes de simulação dedicam pouca ênfase para esta alternativa. O ambiente de interesse deste trabalho, o DCB, proporciona facilidades para a integração de elementos reais em modelos simulados, pois preserva a independência de elementos em relação ao seu comportamento interno e ao uso de recursos de comunicação para simulação distribuída. Toda cooperação do elemento com o ambiente externo ocorre localmente com o uso de portas de comunicação gerenciadas por componentes dedicados do DCB, exclusivas ao tratamento de interface. A flexibilidade para adequação do DCB à interface do elemento também é considerada.

No entanto, a integração de elementos reais exige recursos não comuns para a integração de elementos simulados. Sistemas embarcados, atuadores, sensores, entre outros, são exemplos de elementos reais que exigem recursos intermediários de interface não necessários para integração de elementos simulados essencialmente baseados em tecnologias de descrição de comportamento (linguagens).

6.1. Módulo para integração de elementos reais

No contexto do presente estudo foi desenvolvido um trabalho que demonstra a integração de elementos reais ao DCB [Chaves et al. 2009]. O principal objetivo foi construir um módulo de entrada e saída (*hardware* e *software*) capaz de cooperar com interfaces de comunicação tais como RS232, porta paralela e USB de um lado, com o *gateway* do DCB do outro. O recurso foi validado na integração de um sensor real de passagem de sementes do sistema de Supervisão Eletrônica de Plantio (SEP) já modelado.

O módulo possui uma controladora capaz de cooperar com elementos reais dentro de um conjunto reduzido de requisitos. A construção de um padrão genérico de controladora é muito difícil. Neste trabalho foi desenvolvida uma controladora que suporta a adaptação de sensores compostos por *leds* infravermelhos. Os *leds* geram tensão positiva quando detectam a passagem de uma semente.

Além da controladora, o módulo possui a parte de *software* que é um protocolo de comunicação entre elementos reais e DCB. Neste trabalho, o protocolo desenvolvido, baseado no MODBUS [Mobdus sd.], tem como objetivo principal padronizar as trocas de dados entre DCB e o elemento real. O protocolo permite controle sobre a comunicação, envio e recebimento de dados, execução de funções, e possui um padrão próprio de mensagem. Também é verificado pelo mesmo, a integridade das mensagens através do cálculo do CRC (*Cyclic Redundance Check*) [Griffiths and Stones 1987]. A Figura 6 apresenta o formato da mensagem utilizada pelo protocolo.

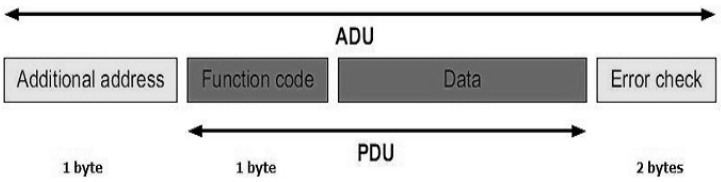


Figura 6. Formato das mensagens utilizada pelo protocolo.

O primeiro campo contém o endereço do elemento real de destino. O campo "Function code" carrega o código da função que será executada. O campo "Data" contém os dados necessários para a execução da função. E o campo "Error check" contém o código para verificação de erros.

O uso de um protocolo de comunicação para integração de elementos reais a modelos de representação facilita a integração de elementos, pois permite o reuso de soluções de integração para elementos similares ou que utilizam os mesmos padrões de comunicação. O uso de protocolo reduz a necessidade de modificações nos *gateways* do DCB e também o *overhead* na medida em que a parte de *hardware* do módulo de integração mantém uma tabela de endereços entre dois elementos reais permitindo que um elemento A possa cooperar com um elemento B sem interferência do DCB.

Experimentos foram realizados com a versão do SEP que integra um sensor real. Resultados equivalentes ao modelo simulado foram alcançados. Para isso, uma fonte de tensão foi utilizada como elemento real no lugar dos sensores infravermelhos como medida de controle sobre a frequência dos sinais enviados pelo elemento real.

7. Conclusões e trabalhos futuros

A configuração de modelos heterogêneos de simulação é uma tarefa complexa diante da necessidade de integração de elementos diferentes. A etapa de configuração define os critérios de cooperação entre elementos do modelo. Por exemplo, eventos que definem a troca de mensagens, especificação das origens e destino, tradução de dados para formato e tipos compatíveis com o destino, identificações dos elementos que receberão mensagens e atributos de saída serão utilizados. O configurador automático desenvolvido e apresentado neste trabalho aumenta o nível de abstração do trabalho de configuração de modelos para execução no DCB.

A relevância da configuração automática de elementos está em agilizar, facilitar e aumentar a confiabilidade do projeto de sistemas com o uso de recursos computacionais. Como resultado há redução de esforço do projetista, pois reduz a quantidade de detalhes no trabalho de configuração de elementos. Isso permite ao projetista preocupar-se com necessidades de maior importância, como por exemplo, a fidelidade do modelo em relação ao sistema real.

O configurador automático apresenta, ao projetista, uma interface contendo todas as informações sobre comportamento, atributos de interface, tipo de dados, e demais informações relevantes para o trabalho de modelagem. Assim, o configurador utiliza as relações de cooperação definidas pelo projetista na interface para gerar os arquivos de configuração em persistência XML e os arquivos *gateways* para tratamento da interface entre DCB e elemento.

Foram identificadas questões relevantes de projeto durante o desenvolvimento do gerador de arquivos de configuração (XML e *gateways*) e na construção do modelo SEP com dez unidades de disco-sensor realizada no estudo de caso. As soluções implementadas no configurador contribuíram com a geração dos arquivos de configuração com reduzida interferência do usuário projetista, facilitaram a modelagem e tratamento da variabilidade de precisão do modelo de simulação considerando o cenário da agricultura de precisão, criaram condições favoráveis para representar essa variabilidade e formas de gerenciar a quantidade de mensagens trocadas por tantos elementos em relação aos limites físicos do elemento real.

Estão previstos estudos para habilitar o módulo de configuração atual com funções para que ele também permita o cadastro de novas características nos *gateways*, reduzindo ou anulando a necessidade de modificações diretas no código. A versão atual do configurador gera os *gateways* de elementos simples. Outra funcionalidade relevante é a habilidade do configurador verificar se há inconsistências nas conexões entre atributos. Esta funcionalidade, posta como perspectiva irá contribuir para a geração de modelos mais corretos.

A integração de elementos reais ao modelo de simulação completa este trabalho. Pode ser necessário adaptar elementos reais ao modelo de simulação quando o elemento apresenta complexidade muito elevada. Com o desenvolvimento de um módulo que proporciona os recursos essenciais de comunicação ao modelo, essa integração foi realizada. Esse módulo possui uma controladora que oferece uma interface reconfigurável para cooperação de um elemento real com um simulado. A comunicação entre elementos reais e simulados é administrada por um protocolo que delimita regras para cooperação.

A utilização de um protocolo específico entre o modelo e o elemento real proporciona um fluxo de mensagens controlado entre emissor e receptor. Além disso, como a controladora desenvolvida possui uma interface configurável, ela permite que mais de um elemento real possa ser integrado ao modelo. Um estudo de caso comparando o modelo do SEP com elemento real e sem elemento real foi realizado. O comportamento de ambos foi equivalente.

No contexto dos grandes desafios da pesquisa em computação, um desafio específico é o esforço para simular mudanças ocorridas no mundo real em tempo de execução do modelo. O conjunto das contribuições apresentadas neste trabalho apontam, como perspectiva, para a busca de alternativas que permitam compor recursos ou ferramentas para viabilizar a construção, preferencialmente colaborativa, de modelos com condições de suportar a modificação dinâmica.

Referências

Amory, A. M., Moraes, F. G., Oliveira, L. A., Hessel, F. P., and Calazans, N. L. V. (2002). Desenvolvimento de um ambiente de co-simulação distribuído e heterogêneo. In *Proceedings of VIII Workshop Iberchip*.

Bruschi, S. (2002). *Um ambiente de simulação distribuída automático*. Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação (ICMC) - Universidade de São Paulo (USP).

Chaves, T. M., Mello, B. A., and Caimi, L. L. (2009). Adaptação de elementos reais na interface do DCB para simulação de modelos heterogêneos. In *Proceedings of XXXV Latin-American Conference on Informatics - CLEI*.

Chowdhury, A. and Koval, D. (2005). Development of probabilistic models for computing optimal distribution substation spare transformers. In *IEEE Transactions on Industry Applications*, 41(6):1493 – 1498.

Fummi, F., Loghi, M., Poncino, M., and Pravadelli, G. (2009). A co-simulation methodology for HW/SW validation and performance estimation. In *ACM Transactions on Design Automation of Electronic Systems*, 14:23:1–23:32.

Griffiths, G. and Stones, G. C. (1987). The tea-leaf reader algorithm: An efficient implementation of CCR-16 and CCR-32. In *Communications of the ACM*, v.30:617-620.

Guimarães, R. L., Costa, Monteiro, R. M. R., and Soares, L. F. G. (2008). Composer: Authoring tool for itv programs. In *Proceedings of the 6th European conference on Changing Television Environments*, EUROITV '08, pages 61–71, Berlin, Heidelberg. Springer-Verlag.

IEEE. (2009). IEEE draft standard for Modeling and Simulation (M&S) – High Level Architecture (HLA) - Framework and Rules. *IEEE Unapproved Draft Std P1516/D5, Apr 2009*.

Mello, B. A. and Parreiras, A. A. (2009). Distributed management of elements for modeling and simulation of heterogeneous models. In *Summer Computer Simulation Conference (SCSC)*, pages: 49–53. Society for Modeling and Simulation International.

Mello, B. A., Souza, U. R. F., Sperb, J. K., and Wagner, F. R. (2005). Tangram: Virtual integration of IP components in a distributed co-simulation. In *IEEE Design and Test of Computers*, 22:462–471.

Mello, B. A. and Caimi, L. L. (2008). Simulação na validação de sistemas computacionais para a agricultura de precisão. *Revista Brasileira de Engenharia Agrícola e Ambiental*, 12:666 – 675.

Merhi, E. (2010). wsimul: Um sistema para simulação computacional. Em *Anais do Simpósio de Mecânica Computacional (SIMMEC)*.

Modbus (sd.). Modbus application protocol specification. Obtido em: www.mobdus.org.

Oraw, B., Choudhary, V., and Ayyanar, R. (2007). A co-simulation approach to model-based design for complex power electronics and digital control systems. In *Proceedings of the 2007 Summer Computer Simulation Conference*, SCSC, pages 157–164, San Diego, CA, USA. Society for Computer Simulation International.

Pacheco, E. P. (2000). Seleção e custo operacional de máquinas agrícolas. Relatório Técnico, Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA Acre), Ministéio da Agricultura e do Abastecimento.

Page, E. H., Buss, A., Fishwick, P. A., Healy, K., Nance, R. E., and Paul, R. J. (1999). Web-based simulation: Revolution or evolution? *In ACM Transactions on Modeling and Computer Simulation*, 10:3–17.

Parreiras, A. A. (2008). Geração automática da configuração de elementos para execução no DCB. Trabalho de conclusão de curso (graduação em Ciência da Computação), Universidade Regional Integrada do Alto Uruguai e das Missões.

Reynolds, P. F. (1998). Heterogeneous distributed simulation. In *Proceedings of the 1988 Winter Simulation Conference.*

Souza, U. R. F., Sperb, J. K., de Mello, B. A., and Wagner, F. R. (2003). Tangram – Virtual integration of heterogeneous ip components in a distributed co-simulation environment. In *Proceedings of the 16th Symposium on Integrated Circuits and Systems Design*, SBCCI '03, Washington, DC, USA. IEEE Computer Society.

Sperb, J. K. (2003). Geração de modelos de co-simulação distribuída para a arquitetura DCB. Dissertação de mestrado (Ciência da Computação), Universidade Federal do Rio Grande do Sul - UFRGS.

Strassburger, S., Schulze, T., Klein, U., Henriksen, J. O., and D-Magdeburg (1998). Internet-based simulation using off-the-shelf simulation tools and HLA. In *Proceedings of the 1998 Winter Simulation Conference*, pages 1669–1676.

Tolk, D. A. (2002). Avoiding another green elephant – A proposal for the next generation HLA based on the model driven architecture. In *Proceedings of the 2002 Fall Simulation Interoperability Workshop.*

Analisando a Simplicidade do Laptop da OLPC: Desafios e Propostas de Soluções de Design

Maíra Codo Canal[1], Leonardo Cunha de Miranda[2,3], Leonelo Dell Anhol Almeida[2], M. Cecília C. Baranauskas[2,3]

[1]Departamento de Computação
Universidade Federal de São Carlos (UFSCar)
13565-905 – São Carlos, SP – Brasil

[2]Instituto de Computação
Universidade Estadual de Campinas (UNICAMP)
13083-852 – Campinas, SP – Brasil

[3]Núcleo de Informática Aplicada à Educação
Universidade Estadual de Campinas (UNICAMP)
13083-873 – Campinas, SP – Brasil

maira.canal@dc.ufscar.br, professor@leonardocunha.com.br,
leonelo.almeida@ic.unicamp.br, cecilia@ic.unicamp.br

Abstract. *Previous studies reported problems regarding interaction design with the educational laptops from OLPC (One Laptop Per Child). However, those results point to low-level issues and also do not propose solutions for the reported problems. This paper describes and discusses results obtained in a study involving 21 subjects aiming at a qualitative evaluation of the XO laptop design using an approach based on the Laws of Simplicity proposed by John Maeda. The results reveal design problems regarding XO's hardware and software and for most cases we point out suggestions to solve them.*

Resumo. *Estudos publicados na literatura sinalizam que o laptop da OLPC possui problemas relacionados com o seu design da interação. No entanto, esses resultados apontam questões em alto nível e, ainda, não apresentam propostas de soluções para os problemas relatados. Este artigo descreve e discute os resultados alcançados em um estudo realizado com 21 sujeitos com o intuito de avaliar de maneira qualitativa o design do laptop XO usando como referencial as Leis da Simplicidade definidas por John Maeda. Os resultados revelam problemas no design tanto de hardware quanto de software do XO e, em sua maioria, são apontadas sugestões para contorná-los.*

1. Introdução

As primeiras ideias de utilizar tecnologias computacionais na Educação surgiram na década de 1960. Kay e Papert introduziram em 1968 o conceito do Dynabook, computador que possui características semelhantes aos computadores portáteis atuais. Ambos acreditavam que o computador portátil desempenharia um papel fundamental não só na Educação, mas também em outros aspectos da sociedade contemporânea [Press 1992]. Baseado no conceito do Dynabook e inspirado na teoria da aprendizagem

Construcionista [Papert 1980,1993] surgiu em 2001 a One Laptop Per Child (OLPC) [OLPC 2010], uma associação, sem fins lucrativos, criada por pesquisadores do Massachusetts Institute of Technology (MIT). Liderada por Nicholas Negroponte, a OLPC iniciou em 2005 o projeto de disseminação de um *laptop* de baixo custo com fins educacionais denominado *laptop* XO ou, simplesmente, XO.

Diversos países já adquiriram milhares de unidades do XO como, por exemplo, Uruguai, Peru, México, Colômbia e Argentina. Nesses países foram realizadas implantações piloto que relatam mudanças positivas no contexto sócio-educacional como, por exemplo, aumento de matrículas na escola, maior participação dos estudantes nas atividades de sala de aula [Kraemer *et al.* 2009] e aumento do interesse dos alunos pela leitura e escrita [Hourcade *et al.* 2008]. Apesar dos benefícios relatados, Hourcade *et al.* (2008) apontam que ainda existem desafios de *hardware* e *software* relacionados com o *design* da interação desse *laptop*, que precisam ser tratados.

No contexto brasileiro, esse *laptop* pode ser usado como um importante instrumento de acesso ao conhecimento não só de crianças em idade escolar, mas também de seus familiares, e seu entorno. Pois, ainda que o XO seja uma tecnologia desenvolvida para o ensino, também possibilita seu uso no trabalho e no lazer, com recursos para colaboração e compartilhamento. Esse olhar diferenciado para fora dos muros da escola, como apontado por Miranda *et al.* (2008), pode promover a inclusão digital e social de um número significativo de brasileiros. Entretanto, entendemos que o uso efetivo e universal do XO só será possível se as suas interfaces de usuário forem concebidas considerando a pluralidade de habilidades e competências dos possíveis usuários desse *laptop* em toda a sua extensão.

Coincidentemente, essa iniciativa com os *laptops* de baixo custo acontece concomitantemente à proposta do Desafio nº 4 da Sociedade Brasileira de Computação (SBC) [Baranauskas e Souza 2006], que aponta como necessária a realização de estudos que colaborem, de diferentes maneiras, com a diminuição das barreiras que dificultam o pleno acesso do cidadão brasileiro ao conhecimento. Em nossa leitura, o uso XO pode promover não só o acesso ao conhecimento das crianças, como originalmente proposto, mas das pessoas ao seu redor, de maneira geral. Este trabalho investiga essas possibilidades sob a ótica da Interação Humano-Computador (IHC), apresentando os resultados de uma avaliação de *design* de *hardware/software* do XO, usando como referencial as Leis da Simplicidade [Maeda 2006].

Este artigo está organizado da seguinte maneira: a Seção 2 apresenta trabalhos relacionados com o uso e a avaliação do XO; a Seção 3 descreve as Leis da Simplicidade e apresenta o método utilizado para a realização deste trabalho; a Seção 4 apresenta os resultados do experimento; a Seção 5 discute os resultados sob a ótica de IHC; e a Seção 6 apresenta as considerações finais.

2. Trabalhos Relacionados

Diferentes trabalhos já foram realizados com o XO ao redor do mundo com o intuito de avaliar as potencialidades e as barreiras de uso desse *laptop*, com os mais variados perfis de usuários. Por exemplo, o estudo apresentado por Nugroho e Lonsdale (2010) sintetiza as técnicas de avaliação e as principais barreiras encontradas em iniciativas de uso do XO em vinte países. A Figura 1 apresenta o XO de três maneiras diferentes: a)

com as antenas “fechadas”; b) com as antenas “abertas”; e c) no formato de um Tablet PC. Como pode ser observado, a tela do XO permite uma rotação de 180º, o que flexibiliza o seu uso e permite que o XO fique parecido com um Tablet PC.



(a) (b) (c)

Figura 1. XO de três maneiras diferentes (a) antenas “fechadas” (b) antenas “abertas” (c) semelhante a um Tablet PC.

No Uruguai, Hourcade *et al.* (2008) realizaram testes em sala de aula com vinte crianças da 1ª a 6ª série do Ensino Fundamental. Com relação aos *softwares* do XO, a maioria das crianças apontou que não gostou do “gerenciador de arquivos” – no Sugar[1] esse aplicativo[2] chama-se *Diário* –, uma vez que tiveram dificuldades para entender o funcionamento de sua interface. Em outra atividade, as crianças tiveram que escrever um texto no editor do *laptop* – no Sugar esse aplicativo chama-se *Escrever* – e, em seguida, enviá-lo por *e-mail*. As crianças tiveram pouca dificuldade para escrever o texto no editor. Entretanto, não obstante a ajuda dos professores e pesquisadores, somente três das vinte crianças – apenas 15% – foram capazes de finalizar a tarefa, visto que a maioria teve dificuldade no uso do *touchpad* e com a autenticação do aplicativo *web* ao tentar enviar o *e-mail*. Ainda com esse público, em outra atividade, durante a utilização do aplicativo *Etoys*[3] [Freudenberg *et al.* 2009] foram reportados problemas na interpretação dos ícones do Sugar e com o tamanho reduzido dos elementos da interface do *Etoys*.

Ainda no Uruguai, devido ao fato de as crianças poderem levar o *laptop* para casa, houve a possibilidade dos pais utilizarem o XO e alguns testes com os adultos também foram realizados [Flores e Hourcade 2009]. Enquanto as crianças sentiam-se livres para explorar o ambiente, os adultos apresentaram dificuldades para se adaptar a essa interface (*Sugar*). Além disso, muitos se sentiram frustrados por nem terem conseguido abrir o *laptop*, sem auxílio. Os adultos também encontraram dificuldades para manusear o *touchpad* e para pressionar algumas teclas do teclado.

Também no Brasil, alguns trabalhos foram realizados para explorar o potencial de uso dessas máquinas. Testes de usabilidade com o aplicativo de desenho – no Sugar

[1] O Sugar é a *Graphical User Interface* (GUI) *default* do XO que roda sob uma distribuição Fedora Linux. Quando mencionamos a palavra “interface”, neste artigo, estamos nos referindo a esse ambiente gráfico ou, quando for o caso, a interface de um aplicativo, em específico, executado sob essa GUI.

[2] Quando falamos em “aplicativos”, neste artigo, estamos nos referindo as Atividades – *softwares* – disponíveis no Sugar. Os programas recebem essa nomenclatura, pois segundo a OLPC são desenvolvidos sob uma nova abordagem, por exemplo, privilegiando recursos de colaboração.

[3] O *Squeak Etoys* é um aplicativo que possibilita a criação de objetos com simulação de situações. O ambiente permite criar projetos utilizando linguagem de programação.

esse aplicativo chama-se *Pintar* – foram realizados com seis crianças, de 12 a 16 anos, da 6ª a 8ª série do Ensino Fundamental, por 1 hora [Martinazzo *et al.* 2008]. Alguns dos problemas identificados nesse trabalho foram, por exemplo, o mau funcionamento do "desfazer", que não funcionou nos desenhos de forma livre e a dificuldade dos usuários em sair do *Pintar*. Outro estudo apresentou uma avaliação de três diferentes modelos de *laptops* educacionais de baixo custo, que incluía uma versão beta do XO. Nesse trabalho foram identificadas barreiras de acessibilidade e problemas de usabilidade desse *laptop* [Miranda *et al.* 2007]. Um dos problemas reportados nesse estudo, que chama a atenção, foi a dificuldade inicial para sua operação fundamental: abrir o XO.

Os trabalhos citados acima são alguns exemplos de estudos realizados nos quais é possível detectar lacunas em relação a avaliação do XO, principalmente sob a ótica de IHC, considerando nosso contexto sociocultural. Dessa maneira, é relevante a adoção de uma metodologia que não privilegie apenas a identificação de problemas, mas que também viabilize a proposição de soluções.

3. Método

A metodologia adotada para a investigação deste trabalho consiste na adaptação das Leis da Simplicidade a um processo de avaliação de elementos de *design*, conforme descrito nas seções a seguir.

3.1. As Leis da Simplicidade

As Leis da Simplicidade foram propostas por John Maeda, pesquisador do MIT [Maeda 2006], que advoga a favor da simplicidade em *design* na era digital, e propõe dez Leis que considera úteis na busca da simplicidade no *design* de tecnologia. As Leis são classificadas em três níveis: básico (1ª a 3ª), intermediário (4ª a 6ª) e profundo (7ª a 10ª). Segue uma síntese das Leis:

- **Simplicidade básica**
  - **1ª Lei (reduzir):** a forma mais fácil de conseguir a simplicidade é por meio de uma redução consciente, ou seja, quando tudo o que pode ser eliminado é removido da solução de *design*;
  - **2ª Lei (organizar):** a organização tem o poder de fazer com que um sistema de muitos se assemelhe ao de poucos. Assim, por exemplo, tudo o que se apresenta separado por categorias, e não misturado, leva à percepção de simplicidade;
  - **3ª Lei (tempo):** a simplicidade pode ser transmitida através do uso do tempo. Caso a espera seja necessária, algo deve ser feito para torná-la tolerável; por exemplo, se o usuário tem uma ideia de quanto do processo já foi concluído, este parecerá mais simples.
- **Simplicidade intermediária**
  - **4ª Lei (aprender):** o conhecimento torna a realização de tarefas mais fácil. Nesse sentido, não utilizar instruções para a realização de tarefas pode tornar sua execução mais demorada;
  - **5ª Lei (diferenças):** simplicidade e complexidade necessitam uma da outra. Enquanto que sistemas invariavelmente simples podem tornar-se monótonos, sistemas demasiadamente complexos exigem grande esforço.

Assim, simplicidade e complexidade devem ser articuladas em soluções de *design* que se adequem às necessidades das pessoas;

- **6ª Lei (contexto):** o que reside na periferia da simplicidade definitivamente não é periférico. Assim, uma meta para alcançar a simplicidade é buscar o significado de tudo que está ao redor do foco de *design*, e não considerar uma única direção;

- **Simplicidade profunda**
  - **7ª Lei (emoção):** é melhor mais emoção do que menos em soluções de *design*. Um sistema personalizável é um exemplo no qual a emoção é utilizada para alcançar a simplicidade, por meio da promoção da empatia;
  - **8ª Lei (confiança):** na simplicidade nós confiamos; por exemplo, quando um sistema conhece as preferências e necessidades de seus usuários, transmite, portanto, confiança no uso;
  - **9ª Lei (fracasso):** algumas coisas podem nunca ser simples ou são desejáveis de se mantê-las complexas como, por exemplo, relacionamentos interpessoais;
  - **10ª Lei (a única):** a simplicidade consiste em subtrair o óbvio e acrescentar o significativo. Essa Lei simplifica as anteriores e expressa o senso consciente de simplicidade e significado em *design*.

3.2. Sujeitos e Procedimentos do Estudo

Para a atividade de avaliação participaram alunos de graduação e pós-graduação em Ciência da Computação da Universidade Estadual de Campinas (UNICAMP) que cursavam, em junho de 2010, a disciplina Projeto de Interfaces de Usuário. O experimento foi gravado em áudio/vídeo, com prévio consentimento dos participantes, de modo a viabilizar ao nosso grupo de pesquisa uma análise aprofundada pós-atividade. Previamente à realização da atividade, os participantes assistiram a uma aula expositiva de cerca de uma hora sobre a temática da simplicidade. Essa aula envolveu uma explicação detalhada sobre as Leis da Simplicidade, bem como, a apresentação de exemplos de problemas e de soluções de *design* utilizando as Leis.

Após essa aula, com o intuito de proporcionar aos participantes um entendimento claro de como se daria a realização da avaliação, foi apresentada uma explicação da atividade e uma breve revisão de cada uma das dez Leis da Simplicidade. Em seguida, os alunos foram convidados a se organizarem – sem intervenção dos pesquisadores – em grupos. Após essa definição, cada grupo recebeu um XO[4] e um formulário de avaliação, que foi preenchido por cada um dos grupos ao final da atividade. Esse formulário foi estruturado para que os grupos pudessem, para cada uma das dez Leis, descrever os problemas encontrados e sugerir possíveis soluções.

Os formulários foram identificados por grupo, ou seja, sem nenhuma identificação dos participantes do estudo. Participaram desse experimento um total de 21 alunos distribuídos em 7 grupos de 3 pessoas. A divisão em grupos com três membros objetivou o enriquecimento da experiência de uso e, consequentemente, da

[4] O *hardware* utilizado foi o OLPC XO-1 CL1(C2) com *firmware* v. Q2E18. O sistema operacional (SO) instalado era uma distribuição Linux Fedora (OLPC *build* 767) com Sugar v. 0.82.1.

identificação dos problemas e a proposição de soluções. Neste artigo esses grupos são identificados por G1, G2, ..., G6, G7.

As atividades consistiram na realização de cinco tarefas que foram realizadas por cada grupo no espaço de uma hora, ou seja, a realização das tarefas e a atividade de avaliação foram realizadas neste período de tempo. As tarefas foram especificadas com base no entendimento de que essas tarefas são relativamente simples ao mesmo tempo em que são fundamentais para o acesso ao conhecimento via XO, seja no contexto de uso educacional, de trabalho ou familiar. As tarefas são identificadas neste artigo por T1, T2, T3, T4 e T5. As tarefas, nessa ordem, foram as seguintes: i) abrir o *laptop*; ii) ligar o XO; iii) rotacionar a tela; iv) tirar foto com a câmera embutida no XO; e v) editar texto. Em relação a essas tarefas, cabe destacar que as três primeiras possuem uma relação maior com o *design* da interação de *hardware* e as duas últimas possuem um foco maior na relação dos usuários com a interface do Sugar e de alguns aplicativos. Além disso, as tarefas foram especificadas nessa sequência por se entender que, dessa maneira, a complexidade aumenta gradativamente.

4. Resultados

Esta seção apresenta primeiramente uma visão geral dos resultados e, em seguida, descreve-os e discute-os; a partir desse ponto do artigo, detalham-se os principais problemas de *design* identificados durante a avaliação e as soluções propostas pelos participantes. A Tabela 1 apresenta um extrato dos problemas e das soluções de *design* descritas pelos participantes. Esses dados foram consolidados a partir dos questionários preenchidos pelos grupos durante a atividade. Vale destacar que a 6ª Lei não aparece nessa tabela, pois os participantes não identificaram problemas relacionados com essa Lei.

Abrir o *laptop* (T1): Apesar de essa tarefa parecer trivial e fácil de ser realizada, os resultados indicam que o mecanismo de abertura[5] foi o item de *hardware* com maior recorrência de problemas, tendo violado sete das Leis.

Alinhado com a 1ª Lei (reduzir), o G6 sinalizou a dificuldade que tiveram para abrir o *laptop*. Para esse grupo uma proposta que facilitaria a abertura do XO seria adotar mecanismo com apenas uma trava. O G6 entende que esse mecanismo seria mais simples do que a solução atual com duas travas (as duas antenas do XO). Esse grupo também achou interessante a solução adotada em proteger com uma "tampa" os conectores que permitem ligar dispositivos de entrada/saída externos ao *laptop* como, por exemplo, *pendrives*, fones de ouvido e microfones. O G6 estava se referindo às antenas, que ao mesmo tempo em que são empregadas como mecanismo para abrir/fechar o XO, também fazem a proteção de 2 portas USB no lado direito, e 1 porta USB e 2 conectores de entrada/saída de áudio – microfone e fone – no lado esquerdo.

[5] Para abrir essa versão do *laptop*, primeiramente, é necessário "levantar" as duas antenas do XO, que também funcionam como travas, no sentido frontal para, então, no sentido oposto "levantar" a tela do XO.

Tabela 1. Exemplos de resultados da avaliação.

Lei	T*x*	G*x*	Problema Identificado	Proposta de Solução
1. reduzir	T1	G6	dificuldade para abrir o *laptop*	mecanismo para abrir o *laptop* com apenas uma trava
	T4	G7	dificuldade para identificar a área de trabalho efetiva do *touchpad*	deixar mais claro onde é a área de trabalho do *touchpad*
2. organizar	T4	G2	ícones dispostos em círculo parecem completamente irrelevantes	organizar os ícones em grupos
		G4	alguns ícones não representam o aplicativo vinculado	melhorar os ícones
	T5	G5	existem botões com rótulos confusos	utilizar o rótulo padrão dos teclados convencionais
3. tempo	T1	G5	abrir o *laptop*	adotar uma trava convencional
	T4	G4	falta de noção do tempo restante para carregamento dos aplicativos	usar barra de progresso como metáfora na interface
4. aprender	T1	G4	dificuldade para abrir o XO	melhoria do mecanismo de abertura
	T4	G2	interface sem ajuda	colocar instruções mais visíveis, mesmo que simples, na interface
		G6	funcionalidades associadas aos ícones da interface não são muito claras	melhorar a qualidade dos ícones ou colocar rótulos textuais
5. diferenças	T1	G3	abrir o *laptop*	acrescentar setas ou ícones na carcaça do *laptop*
	T2	G1	tela de inicialização exibe elementos e informações desnecessárias	exibir menos informações na tela de inicialização
	T4	G4	funções da interface que só ficam disponíveis com o uso do *mouse*	apresentar teclas de atalho mais facilmente visíveis
7. emoção	T4	G1	faltam elementos gráficos que despertem a emoção das crianças	adicionar elementos que despertem mais as emoções
		G7	interface de *software* é muito impessoal	redesenho com um artista/*designer*
8. confiança	T4	G1	interface do Sugar não é usual	procurar seguir padrões de interfaces já estabelecidos
		G2	as ações realizadas no ambiente gráfico possuem poucos *feedbacks*	prover *feedback* das ações por meio de sons e LEDs
9. fracasso	T1	G4	abrir o *laptop* não é tão fácil	indicação mais clara de como abri-lo
		G5	aparência de maleta está conflitante com a forma de abertura do *laptop*	metáfora de maleta consistente com a abertura do XO
	T5	G1	uso de muitos símbolos nas teclas que não sugerem nenhum significado	melhorar os símbolos utilizados nas teclas
		G6	ausência de certas teclas e a presença de outras de iconografia mística	disponibilizar um teclado mais próximo do convencional
10. a única	T1	G5	local de abertura do *laptop* não é óbvio; as travas do *laptop* não parecem travas	mudar a maneira de abrir o *laptop*

Na 2ª Lei (organizar), para o G6, a disposição das entradas de dispositivos na lateral do XO foi uma boa solução, pois no momento em que o *laptop* é aberto seus conectores de interface externa passam a ficar livres para o uso. Alinhado com a 3ª Lei

(tempo), o G5 sinalizou que “não é intuitiva a maneira como o *laptop* deve ser aberto para uso”. Para esse grupo o XO deveria utilizar uma trava convencional, ou seja, semelhante às travas adotadas nos *notebooks* convencionais. Para o G6 abrir o *laptop* demora demais. No que diz respeito à 4ª Lei (aprender) o G4, G6 e o G7 comentaram a dificuldade que tiveram para aprender a abrir o *laptop*. O G6 comentou que “abri-lo pela primeira vez foi um desafio. Não há muito que indique onde e como abrir”. Sugerem que seja adicionada uma indicação de como abrir o *laptop*.

Os grupos G3 e G4 comentaram, a respeito da 5ª (diferenças), 8ª (confiança) e 9ª (fracasso) Leis, que a abertura do XO não é uma tarefa simples. O G3 e o G4 apontaram a necessidade de indicações sobre como realizar a abertura. O G3 sugeriu a adição de uma seta ou ícone na carcaça para indicar a abertura. Com relação à 8ª Lei, para o G6, o *laptop* “parece ser frágil”. Sobre a 9ª Lei, para o G5, a aparência de maleta está conflitante com a forma de abertura do *laptop* (isso ocorre porque a abertura do XO é feita pelo lado oposto à alça). No que diz respeito à 10ª Lei (a única), o G5 comentou que “o local de abertura do *laptop* não é óbvio. A parte central, que fica próxima à alça, tem mais aparência de trava do que as próprias travas”.

A Figura 2 apresenta dois momentos ocorridos durante a T1, que exemplificam a dificuldade dos participantes em abrir o XO. A Figura 2a mostra o instante em que um dos grupos acaba removendo a bateria do *laptop* ao tentar abri-lo. A Figura 2b mostra o momento em que um participante de outro grupo auxilia os colegas na execução da T1.



(a) (b)

Figura 2. Participantes tentando abrir o XO (a) acabam removendo a bateria ao tentar abri-lo (b) recebendo auxílio de um participante de outro grupo.

Ligar o *laptop* (T2): Para essa tarefa duas Leis foram violadas.

O G6 sinalizou para a 3ª Lei (tempo), que o *laptop* demora demais para ser inicializado e sugeriu que a memória RAM fosse expandida. No que tange a 5ª Lei (diferenças) o G1 reportou que “a tela de inicialização exibe diversos elementos e informações com complexidade desnecessária”. Para esse grupo, poderiam ser exibidas “menos informações na tela de inicialização”. O G1 se referiu às informações textuais que geralmente são apresentadas na tela quando do carregamento do SO Linux. De forma positiva, no que diz respeito à 4ª Lei (aprender), o G4 comentou sobre o emprego de alguns símbolos universais como, por exemplo, o símbolo gravado sob o botão utilizado para ligar e desligar aparelhos.

Rotacionar a tela (T3): Para essa tarefa nenhuma Lei foi violada. Entretanto, seguem algumas considerações relevantes.

Com relação à 1ª Lei (reduzir), o G7 questionou se o recurso mecânico adotado no *laptop* que permite a rotação da tela seria, de fato, um recurso necessário. Esse questionamento sugere que o grupo G7 entende que o recurso não é necessário no XO. No entanto, uma provável utilidade para esse recurso seria, por exemplo, ler notícias da Internet com o *laptop* no formato apresentado na Figura 1c. Em oposição ao entendimento do G7, por sua vez, o G4, em relação à 4ª Lei (aprender), avaliou de forma positiva a existência do mecanismo para girar a tela.

Ao deixar o *laptop* parecido com um Tablet PC surgiram problemas relacionados com *affordances*[6] [Norman 1990,1999], uma vez que esse posicionamento sugere que os usuários podem interagir com o XO através de comandos via toque de tela. A Figura 3 mostra o exato momento em que um dos participantes tenta usar a tela como se ela fosse uma tela sensível ao toque, que não é o caso dessa versão de *hardware* do XO. Vale lembrar que, atualmente, existe uma tendência de artefatos digitais – por exemplo, iPod, iPhone, iPad e, até mesmo, celulares em geral – de adotar telas sensíveis a (múltiplos) toques, nos mais diferentes contextos de uso. Essa é, inclusive, uma promessa para novas versões do XO.



Figura 3. Participante tentando interagir com o XO via toques na tela.

Tirar foto (T4): Assim que os participantes começaram a interagir com o XO para executar essa tarefa, foram identificados problemas relacionados com o *touchpad* e com o Sugar e seus aplicativos. Os problemas relacionados com o Sugar foram vinculados a essa tarefa, pois foi a partir dela que os participantes começaram a ter um contato efetivo com a interface gráfica do XO. Na realização dessa tarefa foram violadas oito das Leis.

No que diz respeito à 2ª Lei (organizar), G1, G2 e G7 entendem que os "aplicativos não são organizados por grupos" e sugerem que sejam criados grupos de aplicativos como, por exemplo, jogos, textos e Internet. O G3, sugere que o texto que descreve a funcionalidade correspondente aos ícones do menu poderia aparecer perto dos símbolos como um rótulo. Ainda em relação à 2ª Lei, para o G4, alguns ícones não ilustram bem a função do aplicativo. Comentam, por exemplo, sobre o ícone com o formato de um "olho" que ao ser ativado carrega o aplicativo usado para tirar foto e gravar áudio/vídeo (*Gravar*). O G4 sugere que sejam usadas melhores metáforas na produção dos ícones. Já o G6 comentou que a interface principal poderia ser melhorada, pois é difícil reconhecer o que um aplicativo faz apenas observando seu ícone. Sugere

[6] Segundo Norman as *affordances* "contam" ao usuário que ações podem ser realizadas sobre um objeto e, até certo ponto, como realizá-las.

que os ícones sejam maiores, e que os mesmos possam ser exibidos lado a lado, ao invés do formato circular empregado atualmente (Figura 4a).



(a) (b)

Figura 4. Sugar *e Touchpad* (a) "*desktop*" do Sugar (b) área de trabalho do *touchpad* delimitada pela circunferência pontilhada em vermelho.

Sobre a 3ª Lei (tempo), os grupos G1, G3 e G4 reportaram que os aplicativos demoram demasiadamente para serem carregados e que não existe *feedback* de quanto do processo de carregamento de um aplicativo já foi concluído, para tornar a espera tolerável. O G1 sinalizou que, quando os aplicativos não são fechados, o tempo de resposta aumenta significativamente. Para minimizar esse problema, os grupos G1 e G4 sugerem que barras de progresso ou semelhantes sejam incluídas na interface. Ainda em relação ao desempenho do XO, os grupos G2 e G7 relataram que o XO é muito lento, por exemplo, para abrir e fechar aplicativos, e o G6 expressou que "certas respostas dos aplicativos demoram demais". Os grupos G2 e G7 sugerem que a configuração de *hardware* seja melhorada e o G6 sugere, especificamente, que a memória RAM seja aumentada.

No que diz respeito à 4ª Lei (aprender), o G2 entende que a interface não provê muita ajuda. Esse grupo vê como necessário colocar instruções mais visíveis, mesmo que simples, na interface. O G5 entende que há emprego de metáforas no *hardware* que confundem os usuários. Esse grupo exemplifica, comentando que parte do *touchpad* não é funcional. Por sua vez, o G4 avaliou de forma positiva a existência do *touchpad*. Ainda em relação à 4ª Lei, o G6 comentou que as funcionalidades associadas aos ícones da interface não são muito claras. Sugere que sejam realizadas melhorias na qualidade dos ícones ou que rótulos textuais sejam associados ao ícone. Para o G7, a interface é muito diferente de uma interface convencional de *desktop*. Para esse grupo, ao usar o *laptop*, tiveram a impressão de que estavam mexendo em um computador pela primeira vez. Esse mesmo grupo também sinalizou que os "ícones não ajudam". Sugerem que os ícones e as cores sejam melhorados.

A respeito da 5ª Lei (diferenças), o G4 sinalizou que o acesso a algumas funções da interface só podem ser realizadas com o uso do *mouse*. Para esse grupo "a intenção de tornar o acesso simples acabou complicando o processo". Sugere que as teclas de atalho estejam visíveis. No que diz respeito à 7ª Lei (emoção), o G1 entende que faltam elementos gráficos que despertem a emoção das crianças. Para esse grupo, talvez seja necessário adicionar mais animações e elementos nos aplicativos que sejam interessantes para as crianças. Para o G5, o logotipo do XO localizado no centro da

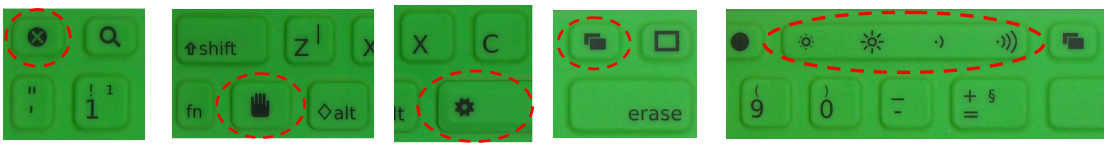
interface não tem nada de muito emotivo. Para esse grupo, poderia ser criada uma área personalizável no lugar desse logo, onde fosse possível colocar uma foto e algumas informações pessoais do usuário. Para o G7 a "interface de *software* é muito impessoal". Foi sugerido por esse grupo que a interface seja redesenhada por um artista/*designer*.

Com relação à 8ª Lei (confiança), para o G1 a interface do Sugar não é familiar e devido a isso, não transmite confiança. Esse grupo sugere que a interface gráfica do XO siga padrões já estabelecidos, especialmente para facilitar o trabalho de instrutores, professores, entre outros. Para o G2, as ações realizadas na interface possuem pouco *feedback*. Esse grupo citou, por exemplo, a falta de *feedback* auditivo ao se tirar uma foto. Sugerem que sejam adicionados *feedbacks* sonoros na interface e, até mesmo, LEDs na carcaça do XO. Ainda em relação à 8ª Lei, para o G4, a descoberta de algumas funções da interface não é tão simples. No que diz respeito à 10ª Lei (a única), o G7 considerou "estranha" a escolha dos aplicativos disponíveis no XO, não conseguindo estabelecer um *rationale* para ela.

Editar texto (T5): Os alunos identificaram problemas no uso do teclado durante a realização desta tarefa (cinco das Leis foram violadas).

Com relação à 1ª Lei (reduzir), G3, G5 e G7 entendem que o teclado do XO possui teclas demais. O G5 e G7 sugerem que as teclas sem funcionalidade ou não presentes em teclados convencionais sejam removidas. Ainda, o G5 sugere que as demais teclas sejam aumentadas de tamanho.

No que diz respeito à 2ª Lei (organizar), os grupos G1 e G5 entendem que o teclado possui teclas com funções muito diferentes aparecendo relativamente próximas. Visando minimizar a confusão dessa proximidade, o G1 sinaliza que as teclas deveriam ser reorganizadas. Já o G5 sugere que as teclas utilizem rótulos padrões como encontramos nos teclados convencionais uma vez que, por exemplo, a tecla *Esc* nesse *laptop* é um "X" na cor branca dentro de um círculo preto, como demonstrado na Figura 5a. Além disso, o G5 avaliou a existência de outras teclas com rótulos confusos como, por exemplo, as teclas apresentadas na Figura 5b (*grab key*), Figura 5c (*view source key*) e na Figura 5d (*bulletin board key*). Para o G7, alguns botões estão agrupados de forma não esperada. Esse grupo sinalizou a junção das teclas de controle da intensidade do brilho do vídeo e volume do som (Figura 5e).



(a) (b) (c) (d) (e)

Figura 5. Algumas teclas do XO (a) *Esc* (b) *grab* (c) *view source* (d) *bulletin board* (e) controles de vídeo e áudio.

Com relação à 3ª Lei (tempo), o G5 comentou que a digitação com o teclado do XO é difícil. No que diz respeito à 4ª Lei (aprender), o G3 entende que há emprego de metáforas no *hardware* que confundem os usuários. Esse grupo exemplifica, comentando que a tecla "mão" não possui funcionalidade (Figura 5b) e que o botão esquerdo com "X" (Figura 5a), sugere "Cancelar". Vale notar que comentários semelhantes foram realizados pelo G5 para a 2ª Lei. Como comentado anteriormente,

ainda em relação à 4ª Lei, o G5 entende que o *design* do *laptop* não considerou a experiência e o conhecimento prévios do usuário com o uso de outros dispositivos. Citam como problemas a disposição das teclas e os rótulos usados.

No que diz respeito à 7ª Lei (emoção), o G3 avaliou como "fantástico poder interagir com outras pessoas". Esse grupo estava se referindo aos recursos de digitação de texto de forma colaborativa que o XO possui. A respeito da 9ª Lei (fracasso), o G1 comentou que algumas teclas fazem uso de símbolos que não sugerem aos usuários seus significados. Para esse grupo é necessário melhorar os símbolos utilizados nas teclas. Para o G6 "a ausência de certas teclas e a presença de outras de iconografia mística não ajuda no uso". Esse grupo ainda comentou que "como computeiros, esperávamos que lidar com o teclado seria algo fluente, quando não foi".

5. Discussão

O conjunto das Leis da Simplicidade constitui um arcabouço rico e desafiador aos *designers* da interação e aos avaliadores. Algumas dessas Leis parecem conflitantes entre si, tais como a 1ª (reduzir) e a 9ª (fracasso) Leis. Entretanto, tais conflitos aparentes provocam a reflexão do *designer* para uma análise mais profunda sobre o equilíbrio tênue entre simplicidade e às expectativas das pessoas quando interagem com artefatos digitais. Tais paradoxos não são exclusividade das Leis da Simplicidade e podem ser encontrados em outros conjuntos de recomendações como, por exemplo, dois dos princípios de *design* de Nielsen (1993): "Usuário está sempre certo" e "Usuário não está sempre certo". No contexto deste trabalho, os alunos no papel de avaliadores, puderam experimentar a complexidade da atividade de *design* que não se restringe a soluções absolutas em relação à simplicidade nem à complexidade isoladamente, mas considera a sua coexistência como na 5ª Lei.

O fato dos participantes desse estudo possuírem experiências prévias em abrir *laptops* convencionais, fez com se sentissem frustrados ao tentar abrir o XO. Isto se deveu à falta de indicações para abertura dada a falta de semelhança com *notebooks*. Esse problema, já relatado em pesquisas anteriores com pessoas com pouca proficiência no uso de computadores, também foi considerado relevante pelos participantes desse estudo. Com isso concluímos que o problema não está relacionado à falta de proficiência nem a experiências anteriores com dispositivos correlatos e que, por isso, merece atenção.

Em relação ao *touchpad*, identificamos a dificuldade de percepção de onde é a área de trabalho do *touchpad*. Para esse problema os participantes sugeriram a remoção da área "não funcional", localizada nas suas laterais. Assim, é importante a adição de relevo às bordas da área de sensibilidade para que os usuários tenham *feedback* sobre o limite da movimentação. Isso é especialmente relevante para que usuários com pouca habilidade motora possam fazer uso do *touchpad*. Os participantes desse experimento também se sentiram incomodados pelo fato do *touchpad* não ser de toque, como são os de *notebooks*, ou seja, o *touchpad* somente é usado para movimentação do cursor, sendo que os cliques são realizados por meio de botões. Em relação a esses botões, vale observar que o botão de confirmação possui um "X" que confunde com "Cancelar". Em relação ao teclado, fica clara a necessidade de um *redesign* dos símbolos de algumas teclas. Além disso, existe a necessidade de melhorias no formato, tamanho e na

mecânica das teclas. Dessa maneira, ficaram nítidos os problemas de usabilidade e de ergonomia impostos pelo *design* do *touchpad* e do teclado do XO.

Também foi observado pelos participantes desse experimento que os aplicativos demoram em serem carregados e não há *feedback* – por exemplo, visual ou auditivo – em relação ao processo de carregamento dos aplicativos; ou seja, não se tem uma noção exata do tempo restante para o carregamento dos programas. A sugestão de vários grupos foi a inclusão na interface de barras de progresso com o intuito de tornar a espera tolerável, já que o *laptop* é lento para carregamento dos aplicativos. Assim, ficaram explícitos os problemas de usabilidade da interface do Sugar e de alguns aplicativos.

É importante enfatizar a necessidade de se tratar com maior urgência os problemas de acessibilidade, uma vez que a visão ampla da acessibilidade está "em sintonia com a diversidade e a amplitude da questão social atrelada à inclusão digital no país" [de Souza *et al.* 2010]. A falta de recursos de acessibilidade no Sugar – *e.g.*, leitor de telas e lupa – indica uma negligência para o uso por pessoas com deficiências visuais. Além disso, a necessidade de se utilizar dispositivos apontadores – *e.g.*, *touchpad* e *mouse* – para interagir com o Sugar pode impor restrições de uso para pessoas com deficiência motora também. Vale considerar, ainda, que apesar de outras pesquisas terem avaliado uma versão beta do XO há quatro anos – *e.g.*, [Miranda *et al.* 2007] – pode-se constatar, dessa vez em conjunto com usuários finais, que os recursos de *hardware/software* da versão de produção do XO ainda apresentam problemas para boa interação usuário-sistema. Mudanças devem ser realizadas para que o *design* e o desempenho do XO não comprometam a experiência do usuário e imponham barreiras para o acesso ao conhecimento.

A utilização das Leis da Simplicidade como ferramental para avaliação do *design* do XO permitiu a identificação de diversas barreiras, algumas identificadas anteriormente em outras pesquisas, assim como novos achados. A identificação tanto de problemas já conhecidos como de novos sugerem a eficácia dessa abordagem. Ainda, faz-se necessária a realização de novos estudos que permitam a análise da abordagem utilizada neste trabalho frente a outras como, por exemplo, avaliações de usabilidade, clarificando a potencialidade de sua aplicação em outros contextos, bem como a determinação das curvas de aprendizado e de quantidade de avaliadores necessárias.

Ao concluir a discussão dos resultados deste estudo, pode-se constatar que as Leis da Simplicidade de Maeda, em conjunto com a metodologia adotada neste trabalho, promoveram efetivo suporte à identificação e descrição detalhada dos diversos problemas de *design* desse *laptop*, ao mesmo tempo, auxiliaram na proposição de potenciais soluções para tratar os problemas identificados. Dessa forma, este trabalho, além de expandir o conhecimento acerca dos problemas de *design* do *laptop* da OLPC também, de forma diferenciada, propõe diversas soluções para tratá-los.

6. Conclusões

Este artigo endereçou, sob a ótica de IHC, a promessa de acesso ao conhecimento via o *laptop* da OLPC. Para tal, apresentamos uma avaliação do *design* do XO norteado pelas Leis da Simplicidade de John Maeda. Com base nessas Leis e na metodologia descrita neste trabalho, foram levantados problemas de *design* desse *laptop* e, para a maioria dos problemas identificados, foram indicadas possíveis soluções de *design* que poderiam ser

consideradas num futuro *redesign* de *hardware* e *software* do XO. Vale ressaltar, ainda, que não tivemos a pretensão de esgotar a lista de problemas/soluções de *design* do XO-1 apenas com a realização deste estudo.

A utilização do XO como artefato digital mediador do acesso universal à ubiquidade informacional, certamente passa pelas suas interfaces de usuário. O acesso ao conhecimento via essa tecnologia computacional não deve estar limitada ao espaço da escola. Compreendemos que os alunos, ao levarem esse *laptop* para casa, poderão colaborar com a promoção da cidadania através da inclusão digital de, por exemplo, seus pais, familiares, amigos e vizinhos. Para isso, esse artefato também deve ser desenhado considerando esses potenciais usuários. Entendemos que a inserção de tecnologias dessa natureza em nossa sociedade gera impactos e desdobramentos de cunho social. Tratar os desafios aqui apresentados, em consonância com o Desafio nº 4 da SBC, potencializa o acesso a informação e ao conhecimento para todos os brasileiros, na maior extensão possível e sem discriminação, via XO.

Para caminhar em direção a um XO que possa ser utilizado por todos, faz-se necessário que as barreiras de interação levantadas neste trabalho sejam efetivamente tratadas. Entendemos que o acesso universal do cidadão brasileiro ao conhecimento – em toda a sua diversidade – dependerá diretamente de que as recomendações sugeridas neste artigo sejam implementadas pela OLPC em novas versões do XO-1 ou, até mesmo, consideradas no *design* da geração futura desse *laptop*. Compreendemos ainda que a implementação dessas melhorias, além de maximizar as possibilidades de uso do XO no Brasil, também, indiretamente, poderia contribuir com outros projetos ao redor do mundo como, por exemplo, o Projeto Ceibal do Governo uruguaio [Ceibal 2010].

Como trabalhos futuros seria interessante a aplicação das Leis da Simplicidade, com a metodologia adotada neste trabalho, em outros artefatos digitais da mesma natureza do XO como, por exemplo, o Classmate PC, que é o *laptop* do Programa Um Computador por Aluno (UCA) do Governo Federal [PROUCA 2010], e o Portátil Magalhães [Magalhães 2011], que é o *laptop* adotado pelo Governo português no Projeto Magalhães [Magalhães 2010]. Ainda, como resultado da experiência adquirida por meio da adoção das Leis da Simplicidade, espera-se a proposição de uma metodologia que propicie mecanismos de *design* e avaliação de artefatos digitais em geral. Por fim, uma análise desses equipamentos deveria ser colocada em perspectiva, de forma a avaliar em que medida tais recursos tecnológicos – e suas interfaces – de fato promoveriam ao nosso cidadão, o acesso almejado à cultura digital.

Agradecimentos

Este trabalho contou com o apoio da Fundação de Amparo à Pesquisa do Estado de São Paulo – FAPESP (processos #2010/11004-9 e #2007/02161-0) e do Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq (processo #475105/2010-9). Agradecemos também a One Laptop per Child Association, Inc. (OLPC).

Referências Bibliográficas

Baranauskas, M.C.C. e Souza, C.S. (2006) “Desafio nº 4: Acesso Participativo e Universal do Cidadão Brasileiro ao Conhecimento”. In: *Computação Brasil*, ano VII, nº 23, p. 7.

Ceibal (2010) “Plan Ceibal”, http://www.ceibal.edu.uy.

de Souza, C.S., Monteiro, I. e Intrator, C. (2010) “Questões Científicas, Técnicas e Éticas do Desafio IV: Reflexões sobre um Assistente para a Navegação na Web”. In: *XXX Congresso da Sociedade Brasileira de Computação*, p. 1-15.

Flores, P. e Hourcade, J.P. (2009) “One Year of Experiences with XO Laptops in Uruguay”. In: *Interactions*, Vol. 16, No. 4, p. 52-55.

Freudenberg, B., Ohshima, Y. e Wallace, S. (2009) “Etoys for One Laptop Per Child”. In: *7th International Conference on Creating, Connecting and Collaborating through Computing*, p. 57-64.

Hourcade, J.P., Beitler, D., Cormenzana, F. e Flores, P. (2008) “Early OLPC Experiences in a Rural Uruguayan School”. In: *Extended Abstracts on Human Factors in Computing Systems*, p. 2503-2512.

Kraemer, K.L., Dedrick, J. e Sharma, P. (2009) “One Laptop per Child: Vision vs. Reality”. In: *Communications of the ACM*, Vol. 52, No. 6, p. 66-73.

Maeda, J. (2006) “The Laws of Simplicity”, MIT Press.

Magalhães (2010) “Iniciativa Magalhães”, http://www.iniciativa-magalhaes.com.

Magalhães (2011) “Portátil Magalhães”, http://www.portatilmagalhaes.com.

Martinazzo, A.A.G., Patrício, N.S., Biazon, L.C., Ficheman, I.K. e Lopes, R.D. (2008) “Testing the OLPC Drawing Activity: An Usability Report”. In: *8th IEEE International Conference on Advanced Learning Technologies*, p. 844-846.

Miranda, L.C., Hornung, H.H., Romani, R., Baranauskas, M.C.C. e Liesenberg, H.K.E. (2008) “Estendendo a Idéia do Projeto UCA ao Desenvolvimento Comunitário: Reflexão e Estratégias”. In: *XXVIII Congresso da Sociedade Brasileira de Computação*, p. 107-116.

Miranda, L.C., Hornung, H.H., Solarte, D.S.M., Romani, R., Weinfurter, M.R., Neris, V.P.A. e Baranauskas, M.C.C. (2007) “Laptops Educacionais de Baixo Custo: Prospectos e Desafios”. In: *XVIII Simpósio Brasileiro de Informática na Educação*, p. 358-367.

Nielsen, J. (1993) “Usability Engineering”, Morgan Kaufmann.

Norman, D.A. (1999) “Affordances, Conventions, and Design”. In: *Interactions*, Vol. 6, No. 3, p. 38-43.

Norman, D.A. (1990) “The Design of Everyday Things”, Doubleday.

Nugroho, D. e Lonsdale, M. (2010) “Evaluation of OLPC Programs Globally: A Literature Review”, http://wiki.laptop.org/images/a/a5/OLPC_Lit_Review_v4_Aug2010.pdf.

OLPC (2010) “One Laptop per Child”, http://www.laptop.org.

Papert, S. (1980) “Mindstorms: Children, Computers and Powerful Ideas”, Basic Books.

Papert, S. (1993) “The Children’s Machine: Rethinking School in the Age of the Computer”, Basic Books.

Press, L. (1992) “Dynabook Revisited - Portable Computers Past, Present and Future”. In: *Communications of the ACM*, Vol. 35, No. 3, p. 25-32.

PROUCA (2010) “Programa Um Computador por Aluno”, http://www.uca.gov.br.

Avaliação de Acessibilidade dos Sites de Municípios Brasileiros

Rodrigo Gonçalves de Branco[1], **André Pimenta Freire**[2], **Débora Maria Barroso Paiva**[1]

[1]Faculdade de Computação – Universidade Federal de Mato Grosso do Sul (UFMS)
Caixa Postal 549 – 79070-900 – Campo Grande – MS – Brazil

[2]Department of Computer Science – University of York - York
United Kingdom, U.K.

{rodrigo.branco,dmbpaiva}@gmail.com, apfreire@cs.york.ac.uk

Abstract. *The use of the Internet to disseminate information has reached levels never seen before. However, the way in which information is provided affects directly its access. The concept of web accessibility is concerned with ensuring access to information to all user groups, regardless of barriers or disabilities. The full achievement of this goal is very challenging and requires awareness of developers and governmental initiatives. Therefore, tools for accessibility ratings help accelerate the development and monitoring of accessible sites. The use of accessibility metrics is a challenge and presents numerous opportunities for research. This paper discusses results of an evaluation about accessibility in Brazilian municipalities web sites considering the Accessibility Model for Electronic Government in Brazil (e-MAG) and metrics. The results present the main data from automatic evaluations and the correlation between these results and social indicators from the Human Development Indexes of these municipalities. The analysis reveals that there is no evidence of a correlation between human development indexes and levels of accessibility, suggesting that the accessibility of governmental websites is a problem to be tackled in all parts of the country.*

Resumo. *O uso da Internet para disseminar informação alcança níveis nunca vistos antes. No entanto, a forma como essa informação é fornecida afeta diretamete seu acesso. O conceito de acessibilidade web assegura o acesso à informação a todos os grupos de usuários, independentemente de barreiras ou deficiências. O alcance completo desse objetivo é extremamente difícil e requer conscientização dos desenvolvedores e iniciativas governamentais. Portanto, ferramentas para avaliação e classificação de acessibilidade ajudam a acelerar o desenvolvimento e monitoria de sites acessíveis. O uso de métricas de acessibilidade é um desafio e apresenta inúmeras oportunidades de pesquisa. Este artigo discute os resultados de uma avaliação sobre acessibilidade em web sites de municípios brasileiros considerando o Modelo de Acessibilidade de Governo Eletrônico (e-MAG) e métricas. Os resultados apresentam os principais dados obtidos a partir de avaliações automáticas e a correlação entre estes resultados e o indicador social de índice de desenvolvimento humano desses municípios. A análise revelou que não há correlação entre os índices de desenvolvimento humano os municípios e os níveis de acessibilidade. Isso sugere que o problema da acessibilidade para informação pública afeta o país como um todo, e não só municípios com problemas sociais específicos.*

1. Introdução

O termo acessibilidade pode ser definido sob diferentes perspectivas, dependendo da área de conhecimento a que se refere. Segundo a norma NBR 9050/94 [NBR 9050 1994], que trata principalmente de construção civil, acessibilidade é a possibilidade e condição de alcance, percepção e entendimento para a utilização com segurança e autonomia de edificações, espaço, mobiliário, equipamento urbano e elementos. Em termos computacionais, Dias [Dias 2002], considerando a acessibilidade digital, define o termo como capacidade de um produto ser flexível o suficiente para atender às necessidades e preferências do maior número possível de pessoas, além de ser compatível com tecnologias assistivas usadas por pessoas com necessidades especiais.

Observa-se que acessibilidade é a maneira como tornamos algo (seja um produto, uma informação, etc) acessível para que qualquer pessoa possa, a princípio, usufruir desse benefício. Percebe-se que nos últimos anos a preocupação em tornar uma informação ou produto acessível aumentou consideravelmente, no entanto, alcançar plenamente os objetivos de acessibilidade não é trivial. O assunto tem recebido atenção especial de diversos pesquisadores [Brajnik 2006, Lazar et al. 2004, Parmanto and Zeng 2005] e órgãos governamentais em diversos países.

Disponibilizar *sites* acessíveis vai além de propósitos comerciais. A medida que as informações são divulgadas apenas via Internet, mecanismos devem ser fornecidos para que as pessoas interessadas acessem o conteúdo. A visão não é apenas disponibilizar conteúdo, mas sim permitir a utilização do recurso por todos, independente de barreiras ou limitações. O objetivo é, através da Computação, permitir a Inclusão Social e Digital de toda a Sociedade.

No Brasil, existe o Decreto-Lei 5.296 de 2 de dezembro de 2004 [Decreto-lei 5296 2004] que estabelece normas gerais e critérios básicos para a promoção da acessibilidade, incluindo o conteúdo web. Para amparar o Decreto-Lei 5.296, foi criado em janeiro de 2005, o e-MAG (Modelo de Acessibilidade para Governo Eletrônico do Brasil) com o intuito de auxiliar na padronização da acessibilidade dos *sites* e portais do governo eletrônico e facilitar sua implementação [e-Mag 2008]. Em maio de 2007, o e-MAG foi institucionalizado no âmbito do Sistema de Administração dos Recursos de Informação e Informática (SISP), tornando sua observância obrigatória nos *sites* e portais do governo brasileiro.

Em 2008, Freire et al [Freire et al. 2008a] propuseram uma abordagem de avaliação de acessibilidade baseada em métricas e realizaram a avaliação de aproximadamente 2.000 sites de municípios brasileiros utilizando as normas do *Web Content Accessibility Guidelines 1.0 (WCAG)* do W3C [WCAG 2009a]. Como principal resultado, os autores observaram que os sites apresentavam um alto índice de barreiras de acessibilidade.

Na pesquisa descrita neste artigo, é apresentado um avanço em relação ao trabalho realizado por Freire [Freire 2008] com a implementação da avaliação baseada em métricas utilizando o modelo brasileiro e-MAG, já que a solução fornecida pelo Governo Brasileiro (a ferramenta ASES [ASES 2009] não aplica nenhum tipo de métrica existente na Literatura. Além disso, foi efetuada uma análise com o intuito de verificar se há uma relação entre o valor obtido para as métricas e o índice IDH (Índice de Desenvolvi-

mento Humano) dos municípios (IDH-M). Considerou-se a hipótese de que municípios com valores maiores de IDH teriam os melhores resultados em termos de avaliação de acessibilidade.

Este artigo está organizado da seguinte forma: a segunda seção apresenta conceitos referentes ao tema desta pesquisa e trabalhos relacionados; a terceira seção descreve a metodologia de avaliação utilizada e os resultados obtidos. A quarta seção apresenta as conclusões.

2. Acessibilidade na Web

A Internet oferece às pessoas acesso a informações e serviços. Porém, esta tecnologia não pode atingir todo seu potencial se projetistas e desenvolvedores não seguirem as regras e recomendações de acessibilidade. De fato, a acessibilidade é uma importante vantagem para todos os usuários, não somente para pessoas com dificuldades de acesso ou portadores de deficiência [Pühretmair and Miesenberger 2005].

O princípio chave da acessibilidade web é projetar aplicações que sejam flexíveis para reunir diferentes necessidades, preferências e situações do usuário. Ao considerar a acessibilidade, é possível (1) fazer com que pessoas com deficiência e idosos tornem-se independentes ao usarem a Internet; (2) que a aplicação esteja mais adequada a leis e regulamentações de acessibilidade e (3) a audiência do *site* seja aumentada [W3C/WAI 2008, Paddison and Englefield 2003].

2.1. Regulamentações e métricas

A principal regulamentação sobre acessibilidade Web foi criada pelo W3C (*World Wide Web Consortium*), por meio das diretrizes apresentadas no documento WCAG (*Web Content Accessibility Guidelines*). Este documento é uma recomendação que estabelece pontos a serem considerados para que o conteúdo Web seja acessível.

Em 1999 a WAI, organização vinculada ao W3C, publicou a primeira versão do WCAG [WCAG 2009a]. O documento foi dividido em 14 diretrizes, cada uma com seus respectivos *checkpoints*. Cada *checkpoint*, por sua vez, possuía uma prioridade. As prioridades são divididas em 3 grupos: As prioridades 1 (devem ser atendidas), prioridades 2 (deveriam ser atendidas) e prioridades 3 (poderiam ser atendidas). Se algum *checkpoint* com prioridade 1 não for atendido, um ou mais grupos de usuários ficarão impossibilitados de acessar a informação no documento. Se algum *checkpoint* com prioridade 2 não for atendido, um ou mais grupos de usuários terão dificuldades em acessar a informação no documento. Se um *checkpoint* com prioridade 3 não for atendido, um ou mais grupos de usuários poderão ter dificuldade em acessar alguma informação no documento. Se todos os *checkpoints* com prioridade 1 forem atendidos, o documento tem o nível de conformidade "A". Se todos os *checkpoints* com prioridade 1 e 2 forem atendidos, o documento tem o nível de conformidade "AA". Por fim, se todos os *checkpoints* com prioridade 1, 2 e 3 forem atendidos, o documento tem o nível de conformidade "AAA".

Em 2008, a mesma organização publicou a segunda versão do documento [WCAG 2009b]. O WCAG 2.0 possui 12 diretrizes, organizadas em 4 princípios: percepção, compreensão, operação e robustez. Cada diretriz possui critérios de sucesso testáveis, nas mesmas prioridades do WCAG 1.0. Independentemente da versão, o

WCAG é o documento mais importante atualmente e serve de base para muitos outros documentos.

O e-MAG é o modelo brasileiro de acessibilidade baseado no WCAG 1.0, que tem o intuito de adequar as *guidelines* de acessibilidade ao contexto brasileiro. Sua primeira versão foi lançada em janeiro de 2005. Em dezembro do mesmo ano, foi lançada a versão 2.0, que é utilizada atualmente. O e-MAG descreve como a sua implementação pode ser percebida na visão do cidadão e na visão técnica.

A *visão do cidadão* separa os princípios de acessibilidade em áreas parecidas com os princípios do WCAG. São elas: percepção, operação, entendimento e compatibilidade. A Cartilha Técnica, que se refere à *visão técnica*, apresenta as diretrizes que reúnem as recomendações necessárias para serem implementadas nos *sites* acessíveis. As recomendações obedecem a níveis de prioridade. Os níveis de prioridade são utilizados para determinar o grau de acessibilidade da aplicação web. Na Prioridade 1 estão as exigências básicas, ou seja, se não forem implementadas, os usuários ficarão impossibilitados de acessar o conteúdo web. Na Prioridade 2 constam normas e recomendações que garantem o acesso às informações do documento e, caso não sejam implementadas, os usuários terão dificuldade ao acessar a informação. Na Prioridade 3 estão normas e recomendações que, sendo implementadas, facilitarão o acesso aos documentos web.

As diretrizes do WCAG só oferecem três níveis de classificação de sítios web (A, AA e AAA) para os níveis de conformidade. Essa classificação não oferece um nível detalhado o suficiente para mostrar diferenças entre sítios em diferentes níveis. Por exemplo, dois sítios em nível A podem ter níveis de acessibilidade bastante distintos, com um sítio próximo de nível AA e outro mais distante. Desta forma, diversos pesquisadores tem proposto o uso de métricas quantitativas para a avaliação de acessibilidade de sítios web. Uma breve descrição das métricas encontradas na literatura utilizadas na avaliação realizada neste trabalho é apresentada a seguir.

POT - Métrica de problemas potenciais: nesta métrica, proposta por Sullivan & Matson [Sullivan and Matson 2000], foi considerado o conceito de problemas potenciais, de forma a estabelecer um valor referente ao número de barreiras encontradas no site em relação ao número de barreiras potenciais, como ilustrado na Equação 1, onde Bp representa o número de barreiras e Pp representa o número total de problemas em potencial. Por exemplo, uma imagem sem texto alternativo é considerada uma barreira. Dessa forma, como existe o conceito de barreira relacionada a imagem, toda imagem é considerada uma barreira em potencial. Os valores variam entre 0 e 1. Em uma página onde metade das barreiras potenciais fossem realmente barreiras, o resultado da métrica seria 0,5.

$$problemas_potenciais = \frac{B_p}{P_p} \tag{1}$$

Métrica WAB (Web Accessibility Barrier): nesta métrica, proposta por Hackett et al. [Parmanto and Zeng 2005], também são considerados os conceitos de problemas potenciais, tendo dido estabelecido um peso para cada um deles, como ilustrado na Equação 2. O cálculo da métrica leva em conta o número de páginas em um site e estipula uma atribuição de pesos, correspondente ao inverso da prioridade definida pelo WCAG 1.0. Os

autores verificaram que, em média, sites com melhor classificação nos níveis estabelecidos pelo WCAG 1.0 tinham um índice WAB menor (com menos barreiras). Os resultados fornecidos por esta métrica fornecem um valor que aproxima-se a um "somatório ponderado" dos índices de percentual de barreiras, com o peso definido pela prioridade de cada diretriz individualmente. Se duas diretrizes fossem avaliadas, ambas com índice de falha 0,5, e uma com prioridade 1 e outra 2, o resultado da métrica seria um índice acumulado de 0,75. Uma desvantagem dessa métrica é que não há uma definição clara de limitante superior para os valores obtidos, que dependem do número de diretrizes consideradas na avaliação.

$$WAB = \frac{\sum_{\forall j \in J} \sum_{\forall i \in I} W_i(\frac{B_{ij}}{P_{ij}})}{|J|} \tag{2}$$

Métrica UWEM (Modelo de agregação de resultados de testes): nesta métrica, desenvolvida no contexto do projeto UWEM [WAB Cluster 2009], o objetivo é definir a probabilidade de que, ao utilizar um site, possa ser encontrada uma barreira que impeça o usuário de completar a tarefa. Também são utilizados os conceitos de problemas potenciais e pesos, como ilustrado na Equação 3. Contudo, sugere-se que os pesos sejam definidos com base em estudos com usuários. Esta métrica também propõe um método para agregar resultados de índices de falha, mas utilizando um produtório. O resultado almejado pelo desenvolvimento da métrica é uma aproximação de uma probabilidade de que um usuário encontre alguma barreira enquanto percorre a página. É importante ressaltar que até o momento da publicação deste artigo não havia evidências experimentais de que a métrica realmente cumpre esse requisito.

$$UWEM1.1 = 1 - \Pi_{t \in T} 1 - \frac{B_{tj}}{P_{tj}} W_t \tag{3}$$

Métrica A3 (Modelo melhorado de agregação de resultados de testes): nesta métrica, proposta por Bühler et al [Bühler et al. 2006], foram sugeridas algumas melhorias em relação à métrica UWEM, de forma a utilizar algumas propriedades de operações com probabilidades [Bühler et al. 2006].

$$A3 = 1 - \Pi(1 - W_i)^{\frac{B_i}{P_i} + \frac{B_i}{P_b}} \tag{4}$$

Métrica WAQM (Accessibility Quality Metric): esta métrica, desenvolvida por Vigo et al. [Vigo et al. 2007], considera as diretrizes do WCAG 1.0, classificada de acordo com os seguintes princípios: perceptível, operável, inteligível e robustês. Ao contrário das outras métricas, esta também considera problemas identificados como *warnings* na contagem. O resultado é um somatório de índices de acessibilidade para diretrizes relacionadas aos quatro princípios definidos no WCAG 2.0. Enquanto as métricas citadas anteriormente apresentam valores "bons" considerados como próximos de 0 (baixo índice de barreira), a métrica WAQM apresentou mudanças para que os resultados próximos de 100 fossem resultados melhores, para que a interpretação fosse mais intuitiva.

$$A = \frac{1}{N} \sum_{x \in X} N_x \sum_{y \in Y} \frac{N_{x,y} \sum_{z \in Z} W_z A(x,y,z)}{N_x} \tag{5}$$

$$A_{x,y,z} = \begin{cases} \frac{-100}{b} \frac{B_{x,y,z}}{P_{x,y,z}} + 100, if \frac{B_{x,y,z}}{P_{x,y,z}} < \frac{a-100}{a-100/b} \\ -a(\frac{B_{x,y,z}}{P_{x,y,z}}) + a, otherwise \end{cases} \tag{6}$$

Como principal trabalho relacionado, podem-se citar os resultados obtidos por Ferreira et al [Ferreira et al. 2007]. Os autores afirmam que, apesar dos esforços para promover a acessibilidade, a imposição do Decreto-Lei afetou muito pouco as organizações públicas brasileiras. Foi realizada uma avaliação de acessibilidade em que se observou que em um conjunto de 87 organizações que participaram da pesquisa em março de 2007, 22% tinham seus *sites* adaptados, 31% estavam em adaptação e 47% dos *sites* não estavam adaptados. Destes *sites* não adaptados, 36,58% das organizações conheciam o Decreto, 46,34% não conheciam, e as demais não souberam responder. Em outra pesquisa envolvendo 613 participantes ligados ao desenvolvimento de aplicações Web, distribuídos entre indústria, academia e governo, Freire et al [Freire et al. 2008b, Freire et al. 2008a], verificaram que 40,29% dos participantes nunca ouviram falar do Decreto e 32,95% já ouviram falar, mas não o conhecem ou conhecem vagamente.

3. Avaliação de acessibilidade dos sites

De forma a atender ao principal objetivo deste trabalho, buscou-se analisar os sites dos 5552 municípios brasileiros em relação às métricas de acessibilidade propostas na literatura e em relação ao e-MAG 2.0. A metodologia utilizada e os resultados obtidos são apresentados nas próximas subseções.

3.1. Metodologia de avaliação

A primeira tarefa realizada foi a adaptação de uma ferramenta para avaliação de acessibilidade baseada em métricas e considerando o e-MAG, pois observou-se que não havia uma ferramenta disponível que atendesse a esta situação. Existem diversas ferramentas que apóiam a avaliação de acessibilidade web e a escolha de uma delas para o desenvolvimento deste trabalho dependeu de fatores tais como: credibilidade, disponibilização do código fonte, qualidade dos artefatos gerados, dentre outros. Assim, decidiu-se usar a ferramenta Hera, desenvolvida por um grupo de pesquisadores espanhol [Sidar 2009]. A ferramenta Hera adaptada pode ser executada em http://developer.ledes.net/~rodrigo/heraemag/. A Figura 1 apresenta a tela principal da ferramenta, em que é possível visualizar a opção de realizar avaliação de acessibilidade usando o e-MAG.

A ferramenta EGOVMeter [Bittar 2006] foi utilizada para identificar os municípios com sites válidos. A URL desses municípios foi registrada em um banco de dados. Cada registro do banco de dados foi avaliado manualmente para eliminar ruídos (por exemplo, sites que estavam disponíveis mas que apresentavam apenas a informação "Site em Construção"). Assim, considerando os 5552 municípios brasileiros, constatou-se que 3288 possuíam web site e 2225 possuíam web site válido. Com relação aos web sites inválidos, os principais problemas detectados foram:



Figure 1. Tela principal da ferramenta Hera

- 24 acusavam erro de acesso negado
- 134 acusavam erro de servidor, como página não encontrada, entre outros
- 46 acusavam erro de configuração de servidor, exibindo a página inicial do Apache
- 271 estavam em manutenção, em construção ou eram pouco expressivas
- 5 apresentavam redirecionamento javascript que não puderam ser resolvidos
- 54 não puderam ser validados pela ferramenta porque a URL possuía caracteres com ˜, -, entre outros
- 367 tiveram problemas em função dos frames existentes na página, que impediam a análise automática
- 30 apresentavam conteúdo divergente do esperado para a página, como página do provedor de acesso, páginas particulares, entre outras
- 132 apresentavam uma página de login, impossibilitando a análise da página.

Em seguida, o banco de dados contendo as URLs dos 2225 sites foi submetido à avaliação de acessibilidade usando a ferramenta Hera.

3.2. Resultados da avaliação

Com a execução de avaliacão automática de acessibilidade com a Hera para as URLS dos 2225 municípios, observou-se, em relação ao e-MAG, que apenas 51 sites não apresentaram erros de prioridade 1. Nenhum site conseguiu evitar erros de prioridade 2 e 3. Estes dados são importantes no sentido de expressar claramente o quanto os sites dos municípios brasileiros estão distantes de atingir a solicitação de observância do e-MAG no setor público.

Foi observado que os sites que utilizam algum CMS (*Content Management System*), em geral, apresentam bons resultados. Uma explicação plausível é que essas ferramentas possuem os recursos pré-programados, livrando o programador de eventuais erros que possam surgir. Os principais CMSs encontrados pelas páginas observadas foram o Joomla! (http://www.joomla.com.br/) e o Drupal (http://www.drupal.com.br/).

O Programa das Nações Unidas para o Desenvolvimento (PNUD) [PNUD 2009] é o órgão que disponibiliza informações sobre o IDH-M. O índice representa uma medida geral e sintética do desenvolvimento humano dos municípios, partindo do pressuposto de que para aferir o avanço de uma população não se deve considerar apenas a dimensão econômica, mas também outras características sociais, culturais e políticas que influenciam a qualidade da vida humana. Ao todo, o órgão possui informações sobre 5507 municípios e os dados relativos ao IDH-M encontrados no site do PNUD se referem ao ano 2000. Neste trabalho foi realizada uma análise considerando-se o IDH-M e os valores obtidos para as métricas relativas a cada município.

Considerando os 2225 sites, 47 não estavam presentes no banco de dados IDH-M e não foram considerados na análise. A distribuição entre as regiões brasileiras dos demais sites foi a seguinte:

- 174 sites pertencem a região Centro-Oeste
- 422 sites pertencem a região Nordeste
- 93 sites pertencem a região Norte
- 832 sites pertencem a região Sudeste
- 657 sites pertencem a região Sul

A figura 2 apresenta a distribuição dos municípios, de acordo com sua faixa no IDH. 24 municípios possuiam IDH entre os valores 0,45 e 0,55. 322 municípios possuiam IDH entre os valores 0,55 e 0,65. 673 municípios possuiam IDH entre os valores 0,65 e 0,75. 1136 municípios possuiam IDH entre os valores 0,75 e 0,85. Finalmente, 23 municípios possuiam IDH entre os valores 0,85 e 0,95. Nenhum município possuia IDH acima de 0,95.

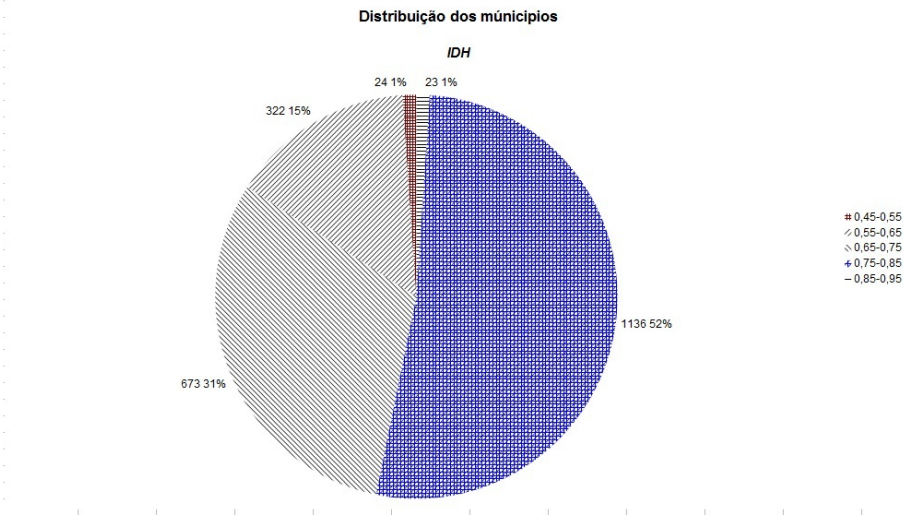


Figure 2. Distribuição dos Municípios entre as faixas de valores de IDH

Foram utilizadas diferentes métricas para verificar se alguma delas poderia fornecer alguma perspectiva diferente da medida de conformidade com diretrizes de acessibilidade e a correlação com o IDH dos municípios. Somente foi realizada a avaliação automática das páginas. Para cada métrica, apresentamos o valor médio obtido para todos os sites e o índice de correlação de Pearson entre as variáveis "métrica de acessibilidade"

e “IDH-M”. Índices de correlação próximos de 1 indicam que há alta correlação entre as variáveis (e.g. municípios com alto IDH obteriam altos níveis de acessibilidade).

A seguir, são apresentados os resultados obtidos na análise da correlação entre os dados obtidos com a avaliação de acessibilidade baseada em métricas e os dados do IDH-M. O valor médio obtido para a métrica POT foi 0,194912. Os possíveis resultados para esta métrica variam entre 0 e 1, sendo que o valor mais próximo de 0 indica melhor acessibilidade. A Figura 3 mostra a relação entre o resultado obtido para a métrica POT e o IDH-M. O índice de correlação de Pearson foi igual a r = 0,033, o que demonstra uma correlação muito baixa entre as duas variáveis.

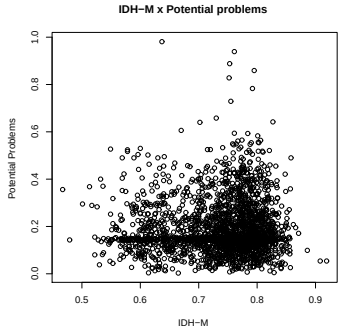


Figure 3. Correlação entre o IDH-M e a métrica POT

O valor médio obtido para a métrica WAB foi 4,384023. Os possíveis resultados para esta métrica não têm limite superior, sendo que valores mais baixos indicam melhor acessibilidade. A Figura 4 mostra a relação entre o resultado obtido para a métrica WAB e o IDH-M. O índice de correlação de Pearson foi igual a r = 0,011.

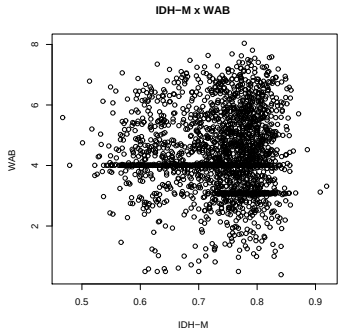


Figure 4. Correlação entre o IDH-M e a métrica WAB

O valor médio obtido para a métrica UWEM foi 0,386253. Os possíveis resultados para esta métrica variam entre 0 e 1, sendo que o valor mais próximo de 0 indica melhor acessibilidade. A Figura 5 mostra a relação entre o resultado obtido para a métrica UWEM e o IDH-M. O índice de correlação de Pearson para este caso foi de r = 0,009.

O valor médio obtido para a métrica A3 foi 0,417178. Os possíveis resultados para esta métrica variam entre 0 e 1, sendo que o valor mais próximo de 0 indica melhor acessibilidade. A Figura 6 mostra a relação entre o resultado obtido para a métrica A3 e o IDH-M. O índice de correlação de Pearson nesse caso foi igual a r = 0,008296403.

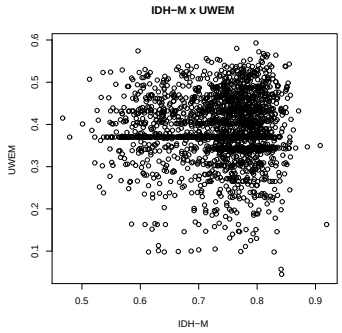


Figure 5. Correlação entre o IDH-M e a métrica UWEM

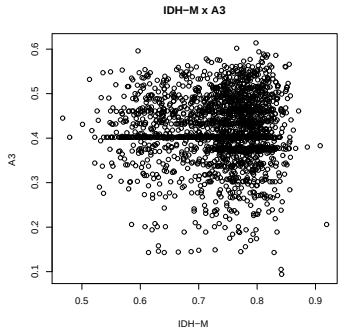


Figure 6. Correlação entre o IDH-M e a métrica A3

O valor médio obtido para a métrica WAQM foi 45,79791. Diferentemente das outras métricas, os possíveis resultados para esta métrica variam entre 0 e 100, sendo que valores mais altos indicam melhor acessibilidade. A Figura 7 mostra a relação entre o resultado obtido para a métrica WAQM e o IDH-M. O índice de correlação de Pearson foi igual a r = 0.039.

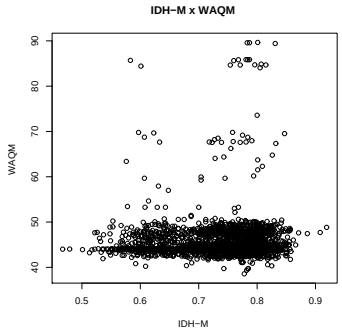


Figure 7. Correlação entre o IDH-M e a métrica WAQM

Os resultados mostram claramente que não existe evidência de que haja alguma relação entre o IDH-M e os índices de acessibilidade de um site, portanto não significa que se o site de um município que tem um IDH-M maior terá o melhor índice de acessibilidade. Assim, não foi confirmada a hipótese de que municípios que oferecem melhores condições em termos de qualidade de vida aos seus habitantes oferecem sites mais

acessíveis.

A avaliação baseada no modelo e-MAG é apresentada a seguir. Os resultados mostram a média dos valores obtidos pelos *checkpoints*, agrupados por níveis de prioridades. A Figura 8 mostra o resultado obtido com a avaliação de todos os sites. Por exemplo, para todos os sites avaliados, para o nível de prioridade 1, em média (a) 15 *checkpoints* não foram avaliados porque dependiam da intervenção humana; (b) 5,5 *checkpoints* não foram avaliados porque não eram aplicáveis; (c) 0,1 *checkpoints* foram avaliados de forma positiva (foram implementados nos sites) e (d) 2,4 *checkpoints* foram avaliados de forma negativa (não foram implementados nos sites).

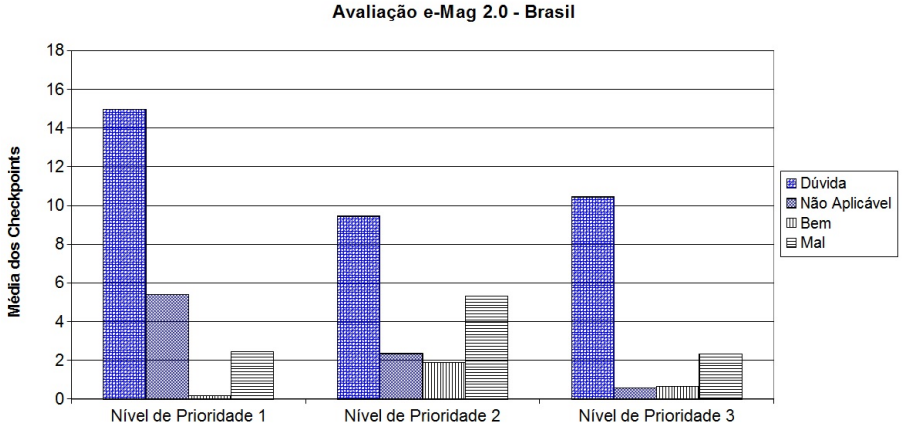


Figure 8. Avaliação do e-MAG para todos os municípios

Os resultados reforçam a necessidade da avaliação humana, pois a maioria dos *checkpoints* não pôde ser avaliada automaticamente. Os resultados também mostram que, por se tratarem de informações públicas e que devem estar disponíveis a todos os cidadãos, a quantidade média de erros é alta, impactando diretamente no nível de acessibilidade dos sites.

4. Conclusões

Neste trabalho, o principal objetivo proposto foi realizar avaliação de acessibilidade dos municípios brasileiros de acordo com métricas e o e-MAG, fazendo uma relação entre os resultados obtidos e o IDH-M. Considera-se que as principais contribuições do trabalho foram (a) a implementação de uma ferramenta de avaliação baseada no modelo e-MAG, incluindo o cálculo de métricas de acessibilidade e (b) a análise de acessibilidade dos sites de municípios brasileiros.

Pesquisas sobre o tema "acessibilidade", devem ser motivadas, pois, neste trabalho, verificou-se que a introdução deste tema é fundamental para a promoção da inclusão de todas as pessoas na "Sociedade da Informação". Acessibilidade é hoje uma das grandes questões a serem tratadas para enfrentar o desafio de garantir acesso universal e participativo ao conhecimento.

O estudo sobre métricas de acessibilidade propostas na literatura mostrou que a necessidade de verificar quantitativamente e monitorar a acessibilidade é crescente. Prin-

cipalmente no que tange à fiscalização, ferramentas de avaliação são necessárias para auxiliar avaliadores a garantir a acessibilidade nos sites que desenvolvem.

Os resultados obtidos da avaliação dos sites das prefeituras municipais mostraram que o Brasil ainda possui poucos municípios com sites e desses, nem todos possuem informações relevantes. Os resultados também mostram que um grande ndice de desenvolvimento humano não necessariamente implica em um site com um bom índice de acessibilidade.

Apesar de a pesquisa ter sido limitada ao uso de avaliação automática, os números obtidos pela análise das métricas é alarmante. Por exemplo, a média dos valores da métrica WAB foi acima de 4. Esse valor pode incluir situações em que, por exemplo, 4 diretrizes de prioridade 1 falharam em 100% dos casos analisados. Considerando que somente testes automáticos foram realizados, os resultados para uma avaliação manual completa seriam ainda mais graves.

O estudo não proporciona indicativos completos dos níveis de conformidade das páginas com o e-MAG, uma vez que somente avaliação automática das páginas foi realizada, dado que foi efetuada uma análise de grande escala. A análise apresentada, entretanto, apresenta um indicativo em relação aos problemas identificados automaticamente. Como trabalho futuro, os autores pretendem realizar análise manual de uma amostra representativa das páginas para verificar o nível de conformidade.

É importante frisar que as ferramentas de avaliação automática têm o objetivo de auxiliar um avaliador humano, e seus resultados não podem ser usados de forma isolada. A ferramenta Hera está em constante evolução e as principais atividades futuras a serem desenvolvidas são:

- Melhorias na interface;
- Adaptação da ferramenta Hera para outros modelos de acessibilidade, inclusive o modelo WCAG 2.0;
- Permitir a avaliação de acessibilidade de sites utilizando o modelo de avaliação proposto em outros países;

Outro trabalho futuro refere-se à constante avaliação dos sites dos municípios do Brasil e sites de e-gov no sentido de avaliar o quanto estão atendendo ao Decreto-Lei 5.296.

A versão da Ferramenta Hera que utilizamos possui algumas limitações. O mecanismo de aquisição dos *sites* não é dos melhores, inclusive impactando diretamente na performance, levando vários segundos para analisar um site. Além disso, caso o *site* não esteja bem codificado, a ferramenta se "perde" na análise, e muitas vezes não consegue completá-la. Devido ao volume de dados, foi considerado inviável realizar uma avaliação manual de cada *site*.

Apesar das limitações do estudo em relação ao uso restrito de ferramentas automáticas, a análise da acessibilidade e sua correlação com o IDH proporciona uma interessante perspectiva em relação às políticas no país. Verifica-se o tratamento de acessibilidade nos governos públicos municipais não tem relação direta com o índice de desenvolvimento humano. Isso indica que a necessidade de uma abordagem mais efetiva para acessibilidade é um problema comum para munícipios das diversas regiões do país.

AGRADECIMENTOS

Os autores agradecem ao CNPq (proc. 551017/2007-4), à Fundect (proc. 23/200.094/2008) e à CAPES pelo apoio financeiro. Os autores agradecem também ao pesquisador Thiago Jabur Bittar pela importante colaboração em fornecer a lista de URLs dos municípios da ferramenta EGOVMeter.

References

ASES (2009). Ases - avaliador e simulador de acessibilidade de sítios. http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG/ases-avaliador-e-simulador-de-acessibilidade-sitios.

Bittar, T. J. (2006). Abordagem para diagnóstico de problemas em aplicações de e-gov - um retrato dos municípios brasileiros. Master's thesis, Universidade Federal de São Carlos.

Brajnik, G. (2006). Web accessibility testing: When the method is the culprit. In *ICCHP*, pages 156–163.

Bühler, C., Heck, H., Perlick, O., Nietzio, A., and Ulltveit-Moe, N. (2006). Interpreting results from large scale automatic evaluation of web accessibility. In *ICCHP*, pages 184–191.

Decreto-lei 5296 (2004). Regulamenta as leis n°s 10.048, de 8 de novembro de 2000, que dá prioridade de atendimento às pessoas que especifica, e 10.098, de 19 de dezembro de 2000, que estabelece normas gerais e critérios básicos para a promoção da acessibilidade. Publicado no D.O.U, nº 232.

Dias, A. C. (2002). *Usabilidade na Web: criando portais mais acessíveis*. Alta Books, 2ª edition.

e-Mag (2008). Governo eletrônico. http://www.governoeletronico.gov.br/acoes-e-projetos/e-MAG.

Ferreira, S. B. L., dos Santos, R. C., and da Silveira, D. S. (2007). Panorama da acessibilidade web brasileira. In *RCA -Revista de Controle e Administração. Publicação da Controladoria Geral do Município do Rio de Janeiro*, volume III, pages 205–234.

Freire, A. P. (2008). Acessibilidade no desenvolvimento de aplicações web: um estudo sobre o cenário brasileiro. Master's thesis, Universidade de São Paulo.

Freire, A. P., Bittar, T. J., and Fortes, R. P. M. (2008a). An approach based on metrics for monitoring web accessibility in brazilian municipalities web sites. In *Proceedings of the 2008 ACM symposium on Applied computing*, SAC '08, pages 2421–2425, New York, NY, USA. ACM.

Freire, A. P., Russo, C. M., and de Mattos Fortes, R. P. (2008b). The perception of accessibility in web development by academy, industry and government: a survey of the brazilian scenario. *The New Review of Hypermedia and Multimedia*, 14(2):149–175.

Lazar, J., Dudley-Sponaugle, A., and Greenidge, K. (2004). Improving web accessibility: A study of webmaster. *Computers and Human Behavior*, 20(2).

NBR 9050 (1994). *Acessibilidade a edificações, mobiliário, espaços e equipamentos urbanos*. Rio de Janeiro - RJ.

Paddison, C. and Englefield, P. (2003). Applying heuristics to perform a rigorous accessibility inspection in a commercial context. In *Proceedings of the 2003 conference on Universal usability*, CUU '03, pages 126–133, New York, NY, USA. ACM.

Parmanto, B. and Zeng, X. (2005). Metric for web accessibility evaluation. *JASIST*, 56(13):1394–1404.

PNUD (2009). Pnud brasil. http://www.pnud.org.br/pnud/.

Pühretmair, F. and Miesenberger, K. (2005). Making sense of accessibility in it design — usable accessibility vs. accessible usability. *Database and Expert Systems Applications, International Workshop on*, 0:861–865.

Sidar (2009). Revendo a acessibilidade com estilo. http://www.sidar.org/hera/.

Sullivan, T. and Matson, R. (2000). Barriers to use: usability and content accessibility on the web's most popular sites. In *CUU '00: Proceedings on the 2000 conference on Universal Usability*, pages 139–144, New York, NY, USA. ACM.

Vigo, M., Arrue, M., Brajnik, G., Lomuscio, R., and Abascal, J. (2007). Quantitative metrics for measuring web accessibility. In *W4A '07: Proceedings of the 2007 international cross-disciplinary conference on Web accessibility (W4A)*, pages 99–107, New York, NY, USA. ACM.

W3C/WAI (2008). Introduction to web accessibility. 21. http://www.w3.org/WAI/intro/accessibility.php.

WAB Cluster (2009). Unified web evaluation methodology uwem version 1.0. http://www.wabcluster.org/uwem1/.

WCAG (2009a). Web content accessibility guidelines 1.0. http://www.w3.org/TR/WCAG10/.

WCAG (2009b). Web content accessibility guidelines (wcag) 2.0. http://www.w3.org/TR/WCAG20/.

Um Jogo com Reconhecedor de Voz para o Ensino de Crianças com Dificuldade de Aprendizagem em Leitura e Escrita.

Elton Sarmanho Siqueira[1], Ellton Sales Barros[1], Leonardo Brandão Marques [2], Dionne Cavalcante Monteiro[1]

[1] Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará (UFPA)
Rua Augusto Corrêa, 01, CEP: 66075-110 – Belém – Pará – Brazil.

[2] Laboratório de Estudos do Comportamento Humano – Universidade Federal de São Carlos (UFSCar)

{eltonsarmanho, elltonsalesbarros, dionnecm, leobmarques}@gmail.com

Abstract. *This article shows an education game with style RPG based on teaching tasks which help children with problem in reading and writing. Resources such as mouse, keyboard, console XBOX 360 and speech recognition are used to accomplish the teaching tasks issued by the game.*

Resumo. *Este artigo tem como objetivo mostrar um jogo educacional de estilo RPG baseado em tarefas de ensino que possibilitarão ajudar crianças com deficiência em leitura e escrita. Recursos como mouse, console XBOX 360, teclado e reconhecedor de voz são usados para realização de tarefas de ensino emitidas pelo jogo.*

1. Introdução

Desde a década de 80, vem sendo desenvolvido e aprimorado no Brasil um programa de ensino que auxilia no processo de aprendizagem de leitura e escrita com o objetivo de melhorar o desempenho escolar de crianças que possuem dificuldades em ler e escrever, com e sem histórico de fracasso [Souza e Rose, 2006].

Durante a fase de alfabetização algumas crianças apresentam problemas de leitura e escrita, tais dificuldades as prejudicam posteriormente em outras disciplinas. Entretanto, a remediação precoce e rápida dessas dificuldades tem sido principal objetivo desse programa de ensino.

Segundo [Azevedo e Marques, 2001] à leitura e escrita são dois repertórios inicialmente distintos e que envolvem uma série de relações diferentes, ambos são tratados como tarefas distintas e complexas. Disso decorre que dificilmente um método de ensino, por melhor que seja, não será igualmente eficiente para todos os aprendizes. Por isso, tem-se necessidade contínua de desenvolvimento de recursos instrucionais personalizados e eficientes para aqueles que têm mais dificuldades.

O foco do programa de ensino de leitura e escrita é a aquisição das relações básicas entre as palavras impressas, as figuras que as representam e o nome falado das figuras (estímulo sonoro). Investigações experimentais para o ensino de leitura e escrita foram realizadas e deram origem a um programa individualizado para o ensino dessas

habilidades [Rose, Souza e Hanna, 1996] e vem sendo sistematicamente avaliado e aperfeiçoado [Souza e Rose, 2006]. A versão original, aplicada com materiais convencionais, deu origem a uma versão informatizada [Rosa Filho *et al*., 1998], que garante aplicação **padronizada** a todos os aprendizes e tem se mostrado igualmente eficaz em estudos de laboratório [Souza e Rose, 2006]. Contudo, há indícios de que esses procedimentos, ao serem transpostos do laboratório de pesquisa para salas de aula, não desempenham de forma consistente um papel motivador para algumas crianças, o que reduz sua eficiência [Santos, 2001].

Mediante a estes indícios, os jogos educativos entram como um fator motivacional no processo de ensino e aprendizagem [Gee, 2007]. Diversos pesquisadores apontam a importância do jogo como recurso para educar e desenvolver a criança, desde que respeitadas às características de atividades pedagógicas além das lúdicas. Através dos jogos a criança aprende a agir, sua curiosidade é estimulada, adquire iniciativa e autoconfiança, e um elevado desenvolvimento da linguagem, do pensamento e da concentração [Marques, 2010]. Junto a este fato, as tecnologias de informação (computadores robustos, dispositivos de realidade virtual, consoles modernos entre outros), cada vez mais estão sendo utilizadas para desempenhar um papel motivador no processo de ensino-aprendizagem, aplicando recursos de interação com usuário, como o reconhecimento e síntese de voz, sensores de movimento e uso de dispositivos tradicionais, como teclado, mouse e joystick [Tori e Kirner, 2006].

Mediante ao conteúdo apresentado, o presente trabalho chega com o propósito de demonstrar um jogo educativo computadorizado com finalidade de ajudar crianças com dificuldades de leitura e escrita, em que o processo de aprendizado feito pelo jogo é realizado através de tarefas providas pelo programa de ensino especializado com apoio de recursos tecnológicos como *engine* de reconhecimento de voz, console do *XBOX* 360, *GamePad Wireless* e fone de ouvido.

2. Trabalhos Relacionados

Ao proceder à revisão bibliográfica foram encontrados alguns trabalhos que abordassem sobre jogos eletrônicos como forma de aprendizado. A seguir apresentam-se dois trabalhos *fortemente relacionados* [Wazlawich, 2008, pag. 127] com o trabalho proposto.

O trabalho desenvolvido por [Scardovelli *et al*., 2004] consiste em um jogo chamado "Aventuras de Jack" (figura 2.1) que auxilia crianças em adquirir capacidades linguísticas necessárias para reconhecer palavras mediante a identificação de letras. O processo de identificação ocorre mediante a escolha de uma letra (no canto superior esquerdo da figura 2.1) por vez, e caso a formação da palavra (apresentada como "barreira") esteja correta a criança prossegue o fluxo normal do jogo.



Figura 2.1 - Processo de Letramento [Scardovelli *et al*., 2004]

Há também o trabalho desenvolvido por [Amante, Oliveira e Slaets, 2003] que mostra alguns jogos computacionais norteados por um processo didático, que tem objetivo ajudar crianças com dificuldades de aprendizagem de escrita (figura 2.2 e 2.3).



Figura 2.2 – Monta Palavra [Amante, Oliveira e Slaets, 2003]

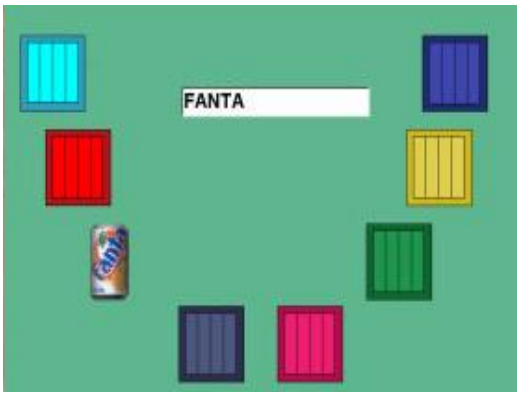


Figura 2.3 – Jogo das Marcas [Amante, Oliveira e Slaets, 2003]

O jogo **monta palavra** (figura 2.2) tem como objetivo explorar a capacidade da criança de reconhecer as letras que compõem as palavras, mas sem a necessidade de organizá-las. Enquanto o **jogo das marcas** (figura 2.3) possui a finalidade de trabalhar com o reconhecimento de palavras do universo cotidiano. A criança nesse patamar consegue distinguir palavras contidas em algumas marcas de produtos comerciais, com os quais ela tem contato no dia a dia. Nesse jogo, o educador digita uma palavra referente a uma das marcas comerciais que conste numa das oito caixas organizadas na tela do jogo, fazendo a criança indicar em qual caixa esta o produto que corresponde à palavra escolhida pelo educador.

Os trabalhos anteriores e os pesquisados em literaturas da área possuem o mesmo objetivo, ensinar a reconhecer e escrever palavras de forma lúdica. Mas verifica-se a existência de pontos fracos nestes trabalhos:

- Não promovem nenhum processo de aprendizado em leitura.
- Não ocorre nenhuma diversificação das atividades de ensino dentro destes jogos, fazendo com que a criança perca o interesse mais rápido do mesmo.

O trabalho proposto utiliza o processo de aprendizagem dos trabalhos mencionados e diferencia-se mediante ao processo de reconhecimento de voz e o uso de um programa de ensino especializado, assim, suprimindo os pontos fracos ditos

anteriormente, proporcionando um software mais consistente no que diz respeito aprendizagem de leitura e escrita.

3. Programa de Ensino e GEIC

O jogo discutido neste trabalho utiliza um programa de ensino para que o mesmo tenha caráter educacional. Além disso, este programa possui sua forma informatizada através do sistema chamado GEIC.

3.1. Programa de Ensino

O programa de ensino denominado *Aprendendo a ler e a escrever em pequenos passos* [Rose *et al*., 1989], apresenta algumas características planejadas para ajudar pessoas com dificuldades em leitura e escrita. A primeira característica diz respeito ao fato do programa ser **individualizado**, o que permite que cada pessoa cumpra as atividades de ensino de acordo com seu próprio ritmo. Outra característica é a **divisão dos repertórios** a serem ensinados em pequenas unidades chamadas de **tarefas de ensino**. O aprendiz pode repetir as unidades de ensino caso não tenha alcançado o critério de aprendizagem previamente estabelecido, o que garante que o progresso ao longo do programa só ocorrerá depois da aquisição dos repertórios previamente ensinados, fazendo com que o programa seja um produto de um arranjo de contingências para promover a aprendizagem [Skinner, 1972].

O programa ensina relações entre **palavras impressas, figuras e palavras ditadas** por meio de um procedimento de emparelhamento com o modelo de referência. Neste procedimento, o aprendiz deve escolher entre duas ou mais alternativas (por exemplo, apontar uma figura entre outras duas ou apontar uma palavra impressa entre duas) qual é alternativa equivalente com modelo de referência. A organização da estrutura de ensino baseia-se nas estratégias e pesquisas realizadas por [Reis, Souza e Rose, 2009] sobre **responder relacionamentos** para encadear os diferentes formatos de apresentação e relacionamento dos estímulos.

Um dos formatos de treino é de construção da resposta, ou *CRMTS* (*constructed-response matching-to-sample*) [Reis, Souza e Rose, 2009], no qual se apresenta um estímulo modelo que pode ser visual (figura ou palavra impressa) ou auditivo (palavra ditada) e os diferentes estímulos de escolha (sílabas, palavras ou figuras). A tarefa consiste em escolher, entre os estímulos de escolha disponibilizados, qual deles apresenta **equivalência** com o estímulo modelo.

Na figura 3.1, verifica-se uma tarefa que envolve cópia com construção da resposta. Nesta tarefa, uma palavra localizada na parte superior da tela é apresentada como estímulo modelo e na parte inferior da tela, existe um conjunto de letras aleatoriamente posicionadas, como estímulo de escolha. O aprendiz deve compor a palavra selecionando, na ordem correta, as letras de acordo com o modelo apresentado.

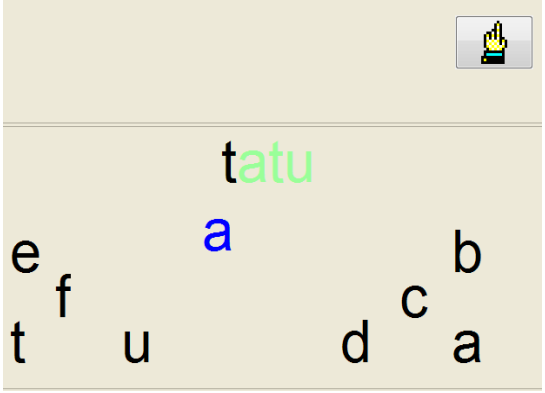


Figura 3.1 – Exemplo de CRMTS [Orlando, Capobianco, Bela, de Souza e Teixeira, 2009].

3.2. GEIC

O GEIC (Gerenciador de Ensino Individualizado por Computador) [Orlando, Capobianco, Bela, de Souza e Teixeira, 2009] é um sistema Web que viabiliza a autoria e a aplicação remota (à distância) do programa de ensino citado no tópico anterior. Além disso, possui componentes para gerenciamento de recursos humanos (especialistas de domínio, tutores e alunos) e para consulta de resultados e geração de relatórios.

Apesar de concebido para implementar um currículo de ensino de leitura e escrita, o sistema apresenta potencial para o ensino de outros conteúdos que possam ser apresentados no formato de tentativas discretas, tais como matemática, linguagem musical e até mesmo para reabilitação auditiva de indivíduos surdos e usuários de implante coclear.

O GEIC terá um papel fundamental para este trabalho por fornecer a estrutura de ensino para o jogo, fazendo este último projetar de forma fiel as tarefas, ocasionando-se qualidade no aprendizado mediante o emprego do programa de ensino.

4. Arquitetura de Comunicação

Neste trabalho houve o desenvolvimento de um ambiente interoperável, uma vez que o jogo utiliza o programa de ensino citado no tópico 3.1, localizado no GEIC (sistema construído na plataforma Java) que utiliza Web Services (WS) para expor os serviços de autenticação do usuário, obtenção de módulos de ensino e registro da sessão de ensino. Desta forma, o jogo (implementado na plataforma .Net) promoveu a construção de um lado cliente, para que o mesmo pudesse utilizar os serviços disponibilizados. A figura 4.1 mostra a infraestrutura construída para a comunicação entre os sistemas.

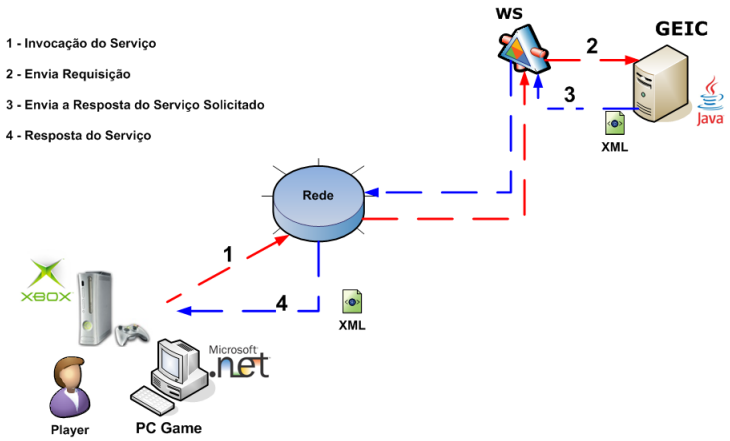


Figura 4.1 – Infra-Estrutura de comunicação

De acordo com a figura 4.1 acima, o aprendiz identificado como *Player*, pode usar o jogo tanto no computador como no console XBOX 360 da Microsoft, sendo que ambos devem estar conectados a uma rede com acesso a *Internet*.

Para que o usuário possa utilizar os serviços disponíveis pelo GEIC, o mesmo deve invocar um serviço através de uma requisição que conterá os dados relacionados ao tipo de serviço desejável. Após o WS receber a requisição, este encaminha ao GEIC a requisição e este processará o pedido enviando uma resposta ao WS. Após estes processos, o WS enviará a resposta ao cliente que solicitou o serviço, finalizando todo procedimento de chamada de serviço.

5. Jogo "ALERV– Aprendendo a Ler e a Escrever com Reconhecedor de Voz"

O jogo desenvolvido neste trabalho denominado **ALERV** (**A**prendendo a **L**er e a **E**screver com **R**econhecedor de **V**oz), implementado no estilo clássico dos RPG's, tem como objetivo ajudar crianças que possuem dificuldade no processo de aprendizagem em leitura e em escrita através da utilização do reconhecedor de voz. O jogo além de obter as tarefas de ensino gerenciadas pelo GEIC, proporcionará a criança desafios dentro de cenários (pequenos labirintos, exploração do mapa e mini jogos), tendo a capacidade de evoluir o personagem.

O ALERV foi construído utilizando a linguagem C# [Liberty e Xie, 2008] tendo como base para o desenvolvimento das rotinas o *Framework XNA Game Studio* 3.0 [XNA, 2010] desenvolvido pela *Microsoft*. Também ocorreu a utilização de Ferramentas como *Mappy* [MappyWin, 2010] para construção dos cenários e *Reiner Prokein* [Reiner's TileSets, 2010] como fonte das texturas de uso não comercial usadas para criação dos personagens. Como diferencial na interface homem máquina, foi integrada a *engine* reconhecedora de voz pt-Br (português Brasileiro) denominada **Coruja** [Klautau, Batista, Silva e Neto, 2010], em que a mesma consegue verificar o nível de confiança do som emitido pelo usuário, analisando se o mesmo emitiu o som equivalente a palavra ou a imagem apresentada na tela. Desta forma, o jogo proporciona exclusivamente um recurso que avalia a leitura da criança, na qual a mesma deve atingir um nível desejável e, caso não seja alcançado, o jogo poderá repetir o procedimento para que a criança possa tentar novamente atingir o nível de confiança necessário.

O ALERV apresenta funcionalidades de caráter simples para interação com a criança, em que a mesma conversa com os personagens, realiza tarefas de ensino e as tarefas características do próprio jogo. Na figura 5.1 são mostradas as funcionalidades do jogo.

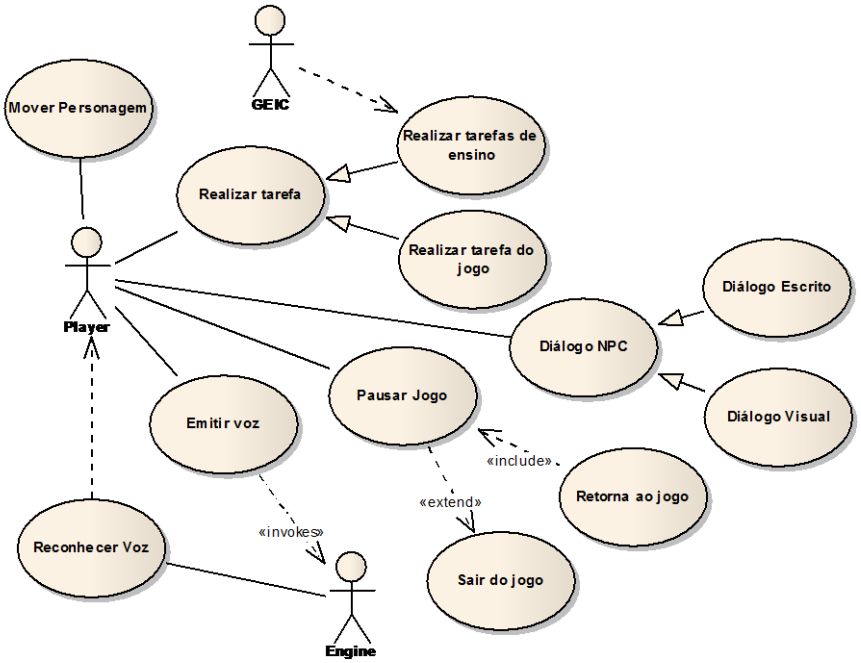


Figura 5.1 – Caso de uso de funcionalidades do jogo.

5. 1. Interface interativa

A interface interativa controla a comunicação entre o jogo e o jogador, atribuindo graficamente um estado novo mediante a escolha do usuário. A interface foi organizada de acordo com a divisão que [Clua e Bittencour, 2005] sugere: interface *out game* e *in game*.

A interface *out game* (Figura 5.2) é referente aos menus presentes fora do jogo, como o *menu* inicial, novo jogo e opções. A interface deste jogo teve todas as telas relacionadas através de um fluxo de menus.

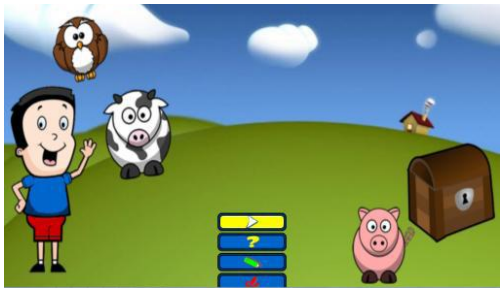


Figura 5.2 – Menu principal do Jogo.

A interface *in game* (Figura 5.3) é aquela que exibe informações sobre o usuário em tempo de jogo como estatística de habilidades, nível de vida e missões, tanto em andamento, como as concluídas.



Figura 5.3 – Tela mostrando informações sobre o personagem.

5.2. Abordagem Geral do Jogo

Uma das funcionalidades do jogo consiste em realizar as tarefas de ensino mediante a interação do jogador. Tais tarefas seguem uma nomenclatura para saber qual o **tipo de tarefa, os estímulos de referência** e **quais os eventos de escolha possíveis**. Para os dois últimos, são adotadas legendas para facilitar a compreensão, em que a sigla T representa texto, a sigla F significa figura e, por último, a sigla S representa o som. Essas tarefas podem ser realizadas tanto pelo mouse como pelo *GamePad* do *XBOX*. Logo abaixo serão apresentadas as tarefas pertinentes ao jogo.

- **Tarefa CR**: responsável pelo exercício da **escrita**. O jogador deve através de um **estimulo de referência** (podendo ser uma F, T ou até S), somente interagir com eventos do tipo T. A figura 5.4 mostra uma tarefa CR com estímulo de referência do tipo T. O jogador deve clicar (usando botão esquerdo do mouse ou botão X do *GamePad*) em cima de uma letra que pertence ao conjunto das letras aleatoriamente posicionadas (localizadas na metade inferior da tela), fazendo uma correspondência à palavra localizada na metade superior da tela (estímulo de referência). Na Figura 5.4, às palavras clicadas são reposicionadas, ficando abaixo das letras da palavra de referência e alterando sua cor original de preto para verde.
- **Tarefa NOM**: responsável pelo exercício da **fala**. Nesta tarefa o jogador deve através de estímulo de referência (podendo ser uma F, T ou S) somente falar. Na figura 5.5, mostra-se a tarefa NOM com estímulo de referência do tipo F (figura localizada na parte superior da tela). Esta tarefa possui uma peculiaridade, pois envolve um mecanismo de reconhecimento de voz, na qual possibilita avaliar o som emitido pelo o usuário.
- **Tarefa MTS**: responsável pelo exercício de **relacionamentos entre objetos equivalentes**. Possibilita a criança conhecer um objeto antes de iniciar processo de leitura e escrita do mesmo. O jogador deve através de estímulo de referência (podendo ser uma F, T ou S) somente selecionar eventos do tipo T ou F. Na Figura 5.6, é apresentada uma tarefa MTS com estímulo de referência do tipo F (figura localizada na metade superior da tela), em que os eventos de escolha são figuras (três figuras localizadas na metade inferior da tela). A interação desta tarefa ocorre igualmente como na tarefa CR, sendo que o jogador deve escolher dentre as três figuras apresentadas (eventos de escolha) a que corresponde à

imagem de referência. Após o clique do mouse ou aperto do botão X do *GamePad*, ocorre estímulo sonoro de acordo com a escolha da criança.

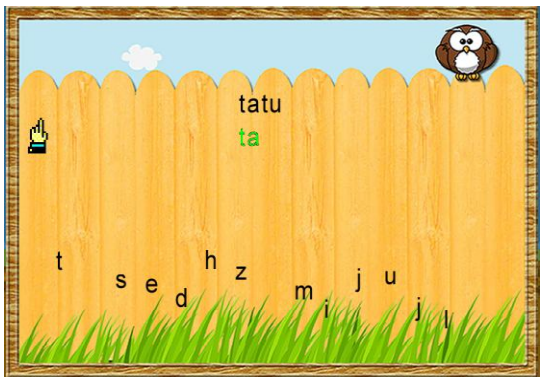


Figura 5.4 – Tarefa CR

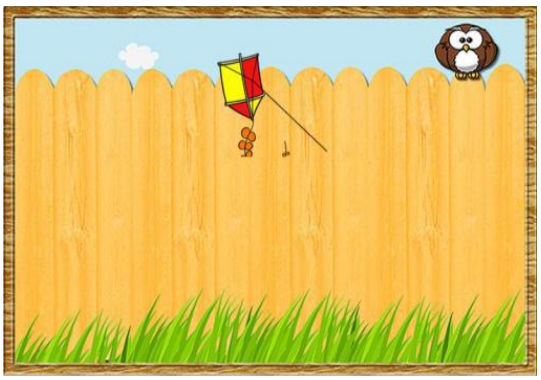


Figura 5.5 – Tarefa NOM



Figura 5.6 – Tarefa MTS

Uma das características marcantes do ALERV, que se diferencia dos outros jogos lidos nas bibliografias, é o elevado numero de tarefas de ensino proporcionado pelo mesmo, uma vez que as tarefas MTS, CR e NOM podem utilizar várias combinações de estímulos, tornando aprendizado da criança consistente em comparação aos outros jogos pesquisados.

Devido ao fato do jogo em questão ser do tipo *RPG*, proporciona grande número de atividade, mostrando certos atributos não encontrados na maioria dos jogos educacionais. Alguns elementos de *RPG* são mostrados durante o jogo tais como: aumento e a diminuição da barra de vida; aumento do nível de intelecto, sorte e velocidade e a possibilidade de ganhar acessórios para aquisição de roupas, proporcionando, desta forma, um maior interesse por parte da criança. Além disso, o aumento das habilidades e o incremento de acessórios citados anteriormente estão atrelados ao **aumento do aprendizado da criança**, fazendo o jogo cumprir o seu papel **motivador** no processo de ensino-aprendizagem.

6. Resultados

Nesta seção são apresentados os resultados para validação do trabalho desenvolvido. Os testes foram aplicados em seis especialistas da área da psicologia (dois psicólogos do Laboratório de Estudos do Comportamento Humano da UFSCar e quatro da Escola

Experimental de Primatas da UFPA). O ALERV foi apresentado aos especialistas sem nenhuma informação sobre a sua dinâmica.

Para a validação do trabalho, utilizou-se o questionário desenvolvido por [Savi, Wangenheim e Ulbricht, 2010] com algumas adaptações que tiveram o intuito de ajustá-lo ao público alvo. Além disso, perguntas contidas no questionário estão em forma de **afirmações** para os especialistas indicarem o quanto concordam com elas de acordo com a escala Likert [Likert, 1932] de cinco pontos variando "discordo fortemente" até "concordo fortemente". Logo abaixo, mostram-se as perguntas pertencentes ao questionário, no qual foi dividido de acordo com a estrutura do modelo de avaliação de jogos educacionais (avaliação da motivação, experiência do usuário e conhecimento).

Tabela 6.1 – Tabela da Estrutura da Motivação.

Avaliação da motivação	
1	Logo no início o jogo prendeu minha atenção.
2	O designer da interface do jogo é atraente.
3	A interação dos Menus está sendo mostrado de forma clara e objetiva.
4	Os sons incluídos no jogo estão adequados.
5	Ficou claro como o conteúdo do jogo está relacionado com coisas que eu já sabia.
6	O conteúdo do jogo é relevante para meus interesses.
7	Eu poderia relacionar o conteúdo do jogo com coisas que já vi, fiz ou pensei.
8	O conteúdo do jogo será útil para mim.
9	O jogo foi mais difícil de entender do que eu gostaria.
10	O jogo tinha tanta informação que foi difícil identificar e lembrar os pontos importantes.
11	O conteúdo do jogo é tão abstrato que foi difícil manter a atenção nele.
12	As atividades do jogo foram muito difíceis.
13	Eu não consegui atender uma boa parcela do material do jogo.
14	Complementar os exercícios do jogo me deu um sentimento de realização.
15	Os textos de feedback depois das tarefas de ensino e/ou desafios, me ajudaram a sentir recompensado pelo meu esforço.
16	Eu me senti bem ao completar as fases contidas no jogo.

Tabela 6.2 – Tabela da Estrutura da Experiência do usuário

Experiência do usuário	
17	Eu não percebi o tempo passar enquanto jogava.
18	Me senti estimulado a aprender com o jogo
19	Eu gostei do jogo e não me senti ansioso ou entediado.
20	O jogo me manteve motivado a continuar utilizando-o.
21	Minhas habilidades melhoraram gradualmente com a superação dos desafios e com as tarefas de ensino.
22	O jogo oferece novos desafios num ritmo apropriado.
23	O jogo mostra de forma clara a diferença entre as *tarefas de ensino* e as *tarefas do jogo*.
24	Este jogo é adequadamente desafiador, onde as *tarefas de ensino* não são muito fáceis nem muito difíceis.
25	Este jogo é adequadamente desafiador, onde as *tarefas do jogo* como: busca por objetos, labirintos e o jogo da velha não foram muito fáceis nem muito difíceis.
26	Senti dificuldade com os comandos do jogo.
27	Senti que estava tendo progresso durante o desenrolar do jogo.
28	Senti que estava colaborando com outros colegas.
29	A colaboração no jogo ajuda a aprendizagem.
30	O jogo suporta a interação social entre os jogadores.
31	Eu jogaria este jogo novamente
32	Algumas coisas do jogo me irritaram.
33	Fiquei torcendo para o jogo acabar logo.Achei o jogo meio parado.

Tabela 6.3 – Tabela da Estrutura da aprendizagem.

Conhecimento	
34	Depois do jogo consigo lembrar de mais informações relacionada as tarefas apresentas pelo jogo.
35	Depois do jogo consigo compreender melhor as tarefas apresentadas no jogo
36	Depois do jogo sinto que consigo aplicar melhor as tarefas relacionadas com o jogo.

Segundo a escala de concordância adotada (discordo fortemente = 1 ponto, discordo parcialmente = 2 pontos, não concordo nem discordo = 3 pontos, concordo parcialmente = 4 pontos e concordo fortemente = 5 pontos), foi realizada uma distribuição em percentagem do número de indivíduos correspondente a escala escolhida.

Tabela 6.4 – Distribuição de concordância entre as afirmativas do questionário.

	Escala de Concordância				
Questão	1	2	3	4	5
1	0,00%	0,00%	0,00%	33,33%	66,67%
2	0,00%	0,00%	0,00%	50,00%	50,00%
3	0,00%	0,00%	0,00%	16,67%	83,33%
4	0,00%	0,00%	0,00%	83,33%	16,67%
5	0,00%	0,00%	33,33%	16,67%	50,00%
6	0,00%	0,00%	0,00%	33,33%	66,67%
7	0,00%	0,00%	0,00%	33,33%	66,67%
8	0,00%	0,00%	0,00%	0,00%	100,00%
9	33,33%	33,33%	33,33%	0,00%	0,00%
10	66,67%	16,67%	16,67%	0,00%	0,00%
11	66,67%	33,33%	0,00%	0,00%	0,00%
12	66,67%	33,33%	0,00%	0,00%	0,00%
13	100,00%	0,00%	0,00%	0,00%	0,00%
14	0,00%	0,00%	16,67%	66,67%	16,67%
15	0,00%	0,00%	0,00%	66,67%	33,33%
16	0,00%	0,00%	0,00%	66,67%	33,33%
17	0,00%	0,00%	66,67%	33,33%	0,00%
18	0,00%	100,00%	0,00%	0,00%	0,00%
19	0,00%	0,00%	66,67%	33,33%	0,00%
20	0,00%	0,00%	33,33%	33,33%	33,33%
21	0,00%	0,00%	33,33%	33,33%	33,33%
22	0,00%	0,00%	33,33%	66,67%	0,00%
23	0,00%	0,00%	66,67%	33,33%	0,00%
24	0,00%	0,00%	66,67%	0,00%	33,33%
25	0,00%	0,00%	0,00%	100,00%	0,00%
26	66,67%	33,33%	0,00%	0,00%	0,00%
27	0,00%	33,33%	33,33%	33,33%	0,00%
28	66,67%	0,00%	33,33%	0,00%	0,00%
29	0,00%	66,67%	33,33%	0,00%	33,33%
30	66,67%	0,00%	0,00%	0,00%	33,33%
31	0,00%	0,00%	16,67%	16,67%	66,67%

32	33,33%	33,33%	0,00%	33,33%	0,00%
33	16,67%	0,00%	83,33%	0,00%	0,00%
34	0,00%	0,00%	66,67%	33,33%	0,00%
35	0,00%	0,00%	33,33%	66,67%	0,00%
36	0,00%	0,00%	16,67%	50,00%	33,33%

Mediante apresentação da tabela 6.4, pode-se afirmar:

- As questões de 1 a 16 são responsáveis pela **avaliação da motivação** que o jogo proporciona. Este aspecto obteve plena concordância entre os especialistas, levando a acreditar que o jogo possui o caráter motivacional elevado.
- As questões de 17 a 33 são responsáveis pela **experiência do usuário**. De acordo com resultado anterior (tabela 6.4), o jogo não proporciona uma imersão elevada, já que o mesmo é voltado para crianças na etapa de alfabetização e não utiliza artifícios de realidade virtual. Além disso, a interação social não existe, uma vez que o jogo não suporta categoria *multiplayer*. Entretanto, o jogo oferece bons desafios e, consegue distinguir de forma clara as tarefas de ensino e tarefas do jogo, pois os comandos dos dispositivos de entrada (teclado e controle) estão bem claros, não apresentando dificuldades ao usuário. Em nível de divertimento, o jogo está adequado para público infantil.
- As questões de 34 a 36 são responsáveis pelo **conhecimento**. Mediante aos resultados, o jogo conseguiu atingir um ponto fundamental, promove de forma eficaz o processo de aprendizado de leitura e escrita, já que aplica as tarefas de ensino em contextos diferentes e de forma lúdica, facilitando o processo de aprendizagem do usuário.

Pela análise de dados e pelo *feedback* após preenchimento dos questionários, o jogo mostra-se eficaz no que ele se propõem a realizar, pois especialistas possuem conhecimento suficiente para este julgamento, uma vez que os mesmos trabalham diariamente com sessões de ensino informatizada, cuja finalidade é remediar as dificuldades no processo de leitura e escrita.

7. Considerações Finais

Segundo os trabalhos relacionados e os resultados obtidos pelos especialistas, o jogo ALERV possui a capacidade de despertar curiosidade, motivação, iniciativa em crianças, devido ao seu caráter motivacional. O jogo ALERV parte de situações lúdicas para proporcionar ensinamentos sobre escrita, leitura e reconhecimento de objetos do cotidiano, necessários aos ciclos iniciais de aprendizagem, o que o diferencia dos trabalhos pesquisados em que cada jogo apresenta uma única forma aprendizagem e sem a utilização de algum processo que ajude no procedimento de leitura.

Os próximos passos do projeto que envolve o desenvolvimento do ALERV é submetê-lo a testes que serão realizados em crianças que possuem os problemas citados neste trabalho, para que o jogo consiga suprimir ou diminuir tais problemas. Em questões tecnológicas, realizar a integração com tecnologias de realidade virtual para aumentar a imersão do jogo, como o Kinect do console do XBOX 360.

8. Referências

Amante, C. F., Oliveira, D. A. H. e Slaets, F. F. A. (2003) Jogos Computadorizados para Aquisição da Linguagem Escrita na educação especial. Conferência IADIS Ibero-Americana WWW/Internet.

Azevedo, M. A. e Marques, M. L. (2001) Alfabetização hoje. 4. ed. São Paulo: Cortez.

Clua, E. W. G. e Bittencourt, J. R., 2005. In: XXIV Jornadas de Atualização em Informática do XXV Congresso da Sociedade Brasileira de Computação, Capítulo 3: Desenvolvimento de Jogos 3D: Concepção, Design e Programação. Sociedade Brasileira de Computação.

de Rose, J. C. C., de Souza, D. G., Rossito, A. L. e de Rose, T. M. S. (1989). Aquisição de leitura após história de fracasso escolar: equivalência de estimulos e generalizaçäo. Psicologia: Teoria e Pesquisa, 5(3), p. 325-346.

de Rose, J. C., de Souza, D. G. e Hanna, E. S. (1996). Teaching reading and spelling: Exclusion and stimulus equivalence. Journal of Applied Behavior Analysis, 29, p. 451-469.

de Souza, D. G. e de Rose, J. C. C. (2006) Desenvolvendo programas individualizados para o ensino de leitura e escrita. Acta Comportamentalia, 14, p. 77-98.

Gee, J. P. (2007) “What video games have to teach us about learning and literacy”, http://doi.acm.org/10.1145/950566.950595, Maio.

IEEE. (1990) Starndar Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries; IEEE; New York, NY.

Klautau, A., Batista, P., Silva, C. e Neto, N. (2010) Um Reconhecedor de Voz Livre para Português Brasileiro com Interface de Programacão. UFPA: Laps.

Liberty, J. e Xie, D. (2008) Programming C# 3.0. 5 th Edition. O’Reilly Media, Inc.

Likert, R., (1932) "A Technique for the Measurement of Attitudes", Archives of Psychology 140: pp. 1-55.

MappyWin. (2010) “Map Developer”, http://www.tilemap.co.uk/mappy, Julho.

Marques, L. B. (2010) Variáveis Motivacionais no Ensino de Leitura: O jogo como recurso complementar. Tese de Doutorado. UFSCAR. São Paulo: São Carlos.

Orlando, A. F., Capobianco, D., Bela, R. E., de Souza, D. G. e Teixeira, C. A. C. (2009). GEIC (Gerenciador de Ensino Individualizado por Computador). Software sem patente. Web site: http://geic.dc.ufscar.br:8080/GEICsite/.

Reiner’s TileSets. (2010) “Free GameGraphics”, http://reinerstileset.4players.de/englisch.html, Julho.

Reis, T. S., de Souza, D. G. e de Rose, J. C. (2009) Avaliação de um programa para o ensino de leitura e escrita. Estudos em Avaliação Educacional*, 20* (44), p. 425-450.

Rosa filho, A. B. *et al*. (1998) Aprendendo a ler e escrever em pequenos passos. (Software para pesquisa).

Santos, J. A. Criança e literatura - desenvolvimento da compreensão e do gosto pela leitura. 2001. Tese (Doutorado em Educação). Universidade Federal de São Carlos.

Savi, R., Wangenheim, C. G. V. e Ulbricht, V. (2010) Proposta de um Modelo de Avaliação de Jogos Educacionais. Revista Novas Tecnologias na Educação. V. 8, n. 3

Scardovelli, T. A., Amante, F. C., Kubatamaia, C. I., Oliveira, H. A. e Frère, A. F. (2004). Jogo computadorizado para auxiliar no letramento de crianças portadoras de deficiências motoras. VII Congresso Iberoamericano de informática Educativa.

Skinner, B. F. (1972). Tecnologia do ensino. São Paulo: Herder, Edusp.

Tori, R. e Kirner, C. (2006) Fundamentos de Realidade Virtual. Symposium on Virtual and Augmented Reality - SVR.

Wazlawick, R. S. (2008) Metodologia de Pesquisa para Ciência da Computação. 4. ed. Elservier: Rio de Janeiro.

XNA. (2010) “Microsoft XNA GS”, http://www.microsoft.com/XNA, Maio.

Extração de Termos de Manuais Técnicos de Produtos Tecnológicos: uma Aplicação em Sistemas de Adaptação Textual

Fernando A. M. Muniz[1], Willian M. Watanabe[2], Carolina E. Scarton[1], Sandra M. Aluisio[1]

[1]NILC - ICMC - Universidade de São Paulo São Carlos - SP, Brasil

[2]Intermídia - ICMC - Universidade de São Paulo São Carlos - SP, Brasil

{fernando.muniz,watinha,carol.scarton}@gmail.com, sandra@icmc.usp.br

Abstract. *In tasks that require the use of technical documentation, the quality of the texts is critical. If the documentation is imprecise, incomplete or too complex, it increases the cost of realizing the tasks and might lead to higher accident rates. This work reports on how the procedural relationships (Goldman's procedural relations of Generation and Enablement) are represented in portuguese technical manuals and can be used to generate an automatic term extraction method, specifically in the context of technical manuals. We also present the adaptation of a simplified texts authoring system to assist the elaboration of technical manuals. The adaptation of the authoring system uses the list of candidate terms extracted to: (a) help identify words that should not be simplified by lexical simplification methods, and (b) elaborate the technical terms, to improve the understanding of texts.*

Resumo. *Em tarefas que exigem o uso de documentação técnica, a qualidade da documentação é um ponto crítico, pois caso ela seja imprecisa, incompleta ou muito complexa, o custo da tarefa ou até mesmo o risco de acidentes aumenta. Este trabalho estudou como as relações procedimentais entre ações gera e habilita, propostas por Goldman, são realizadas em manuais em português, para a criação de um método de extração automática de termos dedicado ao gênero procedimental, especificamente, manuais técnicos. Também propôs uma adaptação de um sistema de autoria de textos simplificados para atender este gênero textual. Esta adaptação usa a lista de candidatos a termos extraídos para: (a) auxiliar na identificação de palavras que não devem ser simplificadas pelo método de simplificação léxica, e (b) elaborar tais termos, facilitando o entendimento dos textos.*

1. Introdução

Tarefas operacionais, procedimentos de manutenção e diagnósticos de falhas em sistemas técnicos complexos requerem o uso de documentação técnica. Se a documentação está imprecisa, incompleta ou difícil de entender, o custo e o tempo da operação de reparo poderá aumentar muito, desta forma a qualidade dessa documentação é um ponto crítico. Até mesmo prejuízos a equipamentos caros ou acidentes com vítimas humanas podem ocorrer devido ao mau entendimento da documentação técnica (Eijk, 1997).

Textos de manuais são do gênero procedimental (ou instrucional). Estes textos deveriam conter uma sequência de instruções concebidas com certa precisão a fim de alcançar um objetivo. O leitor deveria ser capaz de seguir passo a passo cuidadosamente as instruções fornecidas pelo manual a fim de alcançar o objetivo (Fontan and Saint-Dizier, 2008).

O filósofo Alvin Goldman identificou duas relações procedimentais básicas, *generation* (gera) e *enablement* (habilita) (Goldman, 1970). A relação gera aparece entre duas ações e passa o sentido de que após a realização da ação "A", a ação "B" ocorrerá automaticamente, ou seja, "A" gera "B". No português, expressões linguísticas da relação gera geralmente envolvem o conectivo "para", primeiramente seguido por um infinitivo e, em ocasiões raras, seguido por um sintagma nominal. O seguinte trecho de um manual de instruções, em português, de uma serra elétrica exemplifica essa relação (Delin, 1994): *Para colocar a serra na posição de corte oblíquo, solte a porca borboleta e incline a sapata para o ângulo desejado* (Black&Decker).

A relação habilita ocorre quando a realização de uma ação "A" não resulta na realização automática da ação "B". Apesar do conectivo "para" também ser usado para a relação habilita, ele não foi encontrado em (Delin, 1994). Ao contrário, neste estudo, as relações habilita foram encontradas através de sinais de ordem temporal nas ações envolvidas, em orações consecutivas ou ligadas pela conjunção "e": *Desligue a serra da tomada antes de fazer qualquer ajuste* (Black&Decker).

Em Paris *et al.* (1995) uma análise de requisitos para uma ferramenta de suporte à escrita de documentos técnicos multilíngue confirmou que uma ferramenta de auxílio a escrita é mais útil do que uma ferramenta de geração automática que mantém o escritor longe do texto produzido. Paris *et al.* (1994) mostra que os manuais de instruções podem ter diferentes estilos; nem todas as instruções usam uma sequência de imperativos, como seria mais natural de se esperar, e que diferentes partes do manual frequentemente usam diferentes estilos. Aouladomar (2005) faz uma análise da estrutura de manuais e de perguntas relacionadas a textos procedimentais (por exemplo: "Como?" e "Por quê?") e mostra que perguntas e fragmentos de textos procedimentais podem ser combinados a fim de produzirem respostas para máquinas de busca.

O desenvolvimento contínuo de novas tecnologias e produtos, combinados com o fato de que grande parte da população (68%) tem um nível básico e rudimentar de letramento (INAF, 2009) torna clara a importância da adaptação de manuais técnicos para atender as necessidades de grande parte da população. Além disso, é um assunto interessante a ser estudado.

O projeto PorSimples[1] (Aluísio and Gasperin, 2010) aplicou as duas abordagens de Adaptação Textual, a Simplificação e a Elaboração, para ajudar leitores com baixo nível de alfabetização e letramento a compreender documentos disponíveis na Web em português do Brasil, principalmente textos jornalísticos. O trabalho apresentado nesse artigo também se dedicou à elaboração textual, mas o foco foi o gênero de textos instrucionais. Foram estudadas as realizações das relações procedimentais entre ações gera e habilita em manuais de instruções em português, para dar base para a criação de um método de extração de termos dedicado ao gênero de textos instrucionais, especificamente, aos manuais técnicos. Além disso, também propomos uma adaptação

[1] http://caravelas.icmc.usp.br/wiki/

do sistema de autoria de textos simplificados criado no projeto PorSimples – o SIMPLIFICA – para atender a escrita de manuais. O SIMPLIFICA adaptado usa a lista de candidatos a termo gerada pelo método proposto para executar duas funções: (a) auxiliar na identificação de palavras que não devem ser simplificadas pelo método de simplificação léxica baseado em sinônimos, e (b) receber uma elaboração léxica para facilitar o seu entendimento.

Vale destacar que este trabalho tem como objetivo facilitar o entendimento e compreensão de textos instrucionais por pessoas com baixo nível de alfabetização, utilizando técnicas computacionais de processamento de língua natural. Dessa forma, a proposta contribui para o processo de adquirir conhecimentos pelos usuários, sem discriminá-los pela sua condição social ou nível de alfabetização, atuando junto ao tema do SEMISH 2011 - "Computação para todos: No caminho da Evolução Social", mais especificamente no quarto Grande Desafio de Pesquisa da Computação de "Acesso participativo e universal do cidadão brasileiro ao conhecimento".

Na próxima seção, descrevemos trabalhos relacionados ao nosso nas áreas de extração automática de termos e adaptação textual. Na Seção 3, o projeto NorMan – Normalização de Manuais – que compreende o método de extração de termos dedicado a manuais técnicos e a aplicação do resultado da extração em sistemas de adaptação textual, é apresentado. A Seção 4 detalha os experimentos de avaliação intrínseca e extrínseca realizados e, na Seção 5, apresentamos nossas conclusões e trabalhos futuros.

2. Trabalhos Relacionados

2.1. Extração Automática de Termos para o Português

Teline (2004) fez uma avaliação de três abordagens de extração automática de termos: linguística, estatística e híbrida. Na abordagem estatística foram usadas as medidas de Frequência para unigramas, Frequência, Informação Mútua, Log-Likelihood e Coeficiente Dice para bigramas e para trigramas foram usadas Frequência, Informação Mútua, Log-Likelihood. Em seguida foi feita uma análise manual da lista de unigramas e bigramas candidatos a termos. Esta intervenção foi feita com o intuito de eliminar palavras e siglas da língua geral, marcas publicitárias, nomes próprios e símbolos especiais. Neste caso, o método estatístico com intervenção humana é considerado um método semiautomático. Na abordagem lingüística, foi feito um pré-processamento no corpus para permitir a realização de consultas sobre o mesmo. O primeiro tipo de consulta realizada no corpus é a busca por expressões e indicadores estruturais, que são expressões lingüísticas, que geralmente vêm acompanhados de definições, descrições e outros tipos de orações que concentram termos. Em seguida é feita uma busca padrões morfossintáticos (Ex. substantivo + adjetivo). Além disso, nesta abordagem é feito o uso de uma *stoplist*, que é uma lista contendo palavras da língua geral. Na abordagem híbrida, foi feito o processamento do corpus e a buscas por expressões e indicadores estruturais, em seguida foram aplicados métodos estatísticos (cálculo de frequência para unigramas, bigramas e trigramas e informação mútua para bigramas). Após essas etapas, foi feita uma intersecção com a lista de padrões morfossintáticos. O método implementado que teve a melhor precisão foi o estatístico com interação humana. A melhor revocação foi obtida pelo método linguístico e a medida F com maior valor também foi obtida pelo método estatístico com interação humana.

Ribeiro Jr (2008) trabalhou com construção de ontologias e fez o uso de um método de extração automática de termos, utilizando uma abordagem híbrida. Os conhecimentos linguísticos utilizados foram a classe gramatical, sintagmas nominais e padrões morfossintáticos. Em seguida, foram aplicados cálculos de relevância de frequência, tf-idf (Medida que considera relevantes os termos que possuem alta frequência de ocorrência em numero limitado de documentos) e NC-Value, que estão descritos em (Ribeiro Jr, 2008). Na extração de unigramas, o melhor resultado obtido foi com o uso das classes gramaticais e o núcleo do sintagma nominal combinado com o cálculo NC-Value usando tf-idf como parâmetro de frequência que obtiveram os melhores índices de precisão (14,7%), cobertura (49,96%) e medida F (22,39%). Para extração de bigramas, o melhor método foi o uso de padrões morfossintáticos com o cálculo de relevância de frequência, obtendo o índice de precisão de 5,7%, cobertura de 41,91% e medida F de 10,04%. Na extração de trigramas, a melhor estratégia foi o uso de padrões morfossintáticos com o cálculo de relevância tf-idf, obtendo índice de precisão de 2,9%, cobertura de 46,77% e medida F de 5,46%.

Lopes *et al*. (2010) apresentou uma avaliação de termos compostos para um corpus do domínio de pediatria. A extração de termos foi feita usando três métodos diferentes e sua performance foi avaliada através do cálculo da precisão, revocação e medida F, com base numa lista de referência de termos compostos feita a mão. A extração automática de termos foi realizada em três etapas: anotação linguística do corpus, utilizando o parser PALAVRAS (Bick, 2000); extração de termos usando a ferramenta OntoLP (Ribeiro Jr, 2008) e a seleção de termos pelo ponto de corte. A extração de termos simples foi feita através da classe gramatical. Na extração de termos compostos, três métodos foram usados. O primeiro método foi puramente estatístico. O segundo método utilizou padrões morfossintáticos e o terceiro método utilizou sintagmas nominais para tentar identificar termos que poderiam estar presentes em seu núcleo. Após a extração, a lista de candidatos a termos foi ordenada através do calculo de relevância pela frequência. Os experimentos realizados usaram corte absolutos e cortes relativos. Os resultados do experimento que usou corte absoluto mostram que na extração de bigramas o método de padrões morfossintático foi o melhor (medida F = 0.5684 em 6E-6), seguido pelo método estatístico (medida F = 0.5478 em 6E-6) e pelo método de sintagmas nominais (medida F = 0.4437 em 4E-6). Para trigramas, o melhor método foi o estatístico (medida F = 0.5211 em 6E-6), seguido pelo método de padrões morfossintáticos (medida F = 0.4612 em 6E-6) e pelo método de sintagmas nominais (medida F = 0.3671 em 4E-6). Os resultados do experimento que usou ponto de corte relativo mostraram que na extração de bigramas, o método de padrões morfossintáticos foi o melhor (medida F = 0.5547 em 10%), seguido pelo método estatístico (medida F = 0.5233 em 10%) e pelo método de sintagmas nominais (medida F = 0.4393 em 10%). Para trigramas, o método estatístico obteve o melhor resultado (medida F = 0.4831 em 5%), seguido pelo método de padrões morfossintáticos (medida F = 0.4539 em 5%) e pelo método de sintagmas nominais (medida F = 0.3510 em 10%)

2.2. Adaptação Textual

Adaptação textual (AT) é uma atividade comum de professores para facilitar a compreensão de conteúdos e também usada no cenário do ensino de novas línguas (Burstein, 2009). Seus benefícios são vários, por exemplo, ajudar aprendizes de língua e crianças aprendendo a ler diferentes gêneros textuais e também audiências com

necessidades especiais. Tais audiências podem ser leitores com baixo nível de letramento, adultos em fase de alfabetização, pessoas engajadas em cursos de Educação à Distância, para os quais a compreensão é de grande importância. Surdos que se comunicam com a língua de sinais e desejam aprender línguas como português ou inglês também se beneficiam da AT (Aluísio and Gasperin, 2010).

Os estudos na área de AT tentam responder duas questões: *O que é modificado?* e *Como é modificado?*. Para responder à primeira questão as pesquisas investigam modificações nos diferentes níveis linguísticos: fonológico, léxico, sintático e discursivo. Já para a segunda, existem duas grandes abordagens (ou tipos) de adaptações: Simplificação Textual (ST) e Elaboração Textual (ET) (Young, 1999; Urano, 2000). A primeira pode ser definida como qualquer tarefa que reduza a complexidade de um texto (por exemplo, no nível léxico ou sintático), enquanto tenta preservar o significado e informação (Siddharthan, 2003). A Elaboração Textual, por sua vez, é definida como um conjunto de técnicas para acrescentar material redundante em textos, sendo tradicionalmente usadas a adição de definições, sinônimos, antônimos, ou qualquer informação externa com o objetivo de auxiliar na compreensão do texto por meio dessa informação complementar (Rahimi, 2011). É importante dizer que a simplificação e a elaboração são fortemente relacionadas; enquanto a simplificação aumenta a inteligibilidade de um texto (torna ele mais fácil de ser lido), a elaboração melhora a compreensão do texto, isto é, facilita o entendimento de conceitos nos textos. Abaixo, relatamos os trabalhos sobre simplificação léxica próximos ao nosso, pois é este o nível da língua tratado no projeto NorMan.

Elhadad (2006) estudou como melhorar o acesso de pessoas sob tratamento ou cuidados médicos à literatura médica, com foco na terminologia da área médica. Um método para a predição em um texto de quais termos um leitor comum pode não entender foi proposto, usando definições tomadas da Web para melhorar a compreensão de tais termos não familiares. A predição é realizada via contagem de frequência de termos em corpora de textos médicos dedicado a pessoas comuns, para achar o corte na frequência ideal para separar palavras comuns de não familiares. Os autores classificam as abreviações como palavras complexas.

Devlin and Unthank (2006) apresentam o projeto HAPPI (Helping Aphasic People Process Information) em que usam um método para simplificar palavras que os leitores não entendem por palavras mais comuns com o mesmo significado e, além da simplificação léxica, imagens e áudio com versões faladas dos conceitos são apresentados aos usuários.

Existem vários estudos sobre elaboração textual para o inglês e um trabalho recente para o português, apresentados abaixo.

A ferramenta ATA (The Automated Text Adaptation Tool) (Burstein *et. al.*, 2007) é uma aplicação da área de Processamento de Língua Natural (PLN) para propósitos educativos, que é usada por aprendizes do inglês em aulas para o ensino fundamental nas quais os professores a usam para gerar adaptações de textos similares àquelas feitas manualmente. São usados sinônimos para palavras de baixa frequência, antônimos usando a WordNet de Princeton[2] e listas de falsos cognatos (homógrafos com significados diferentes), além de notas marginais (tipo de resumo) traduzidas para o espanhol.

[2] http://wordnet.princeton.edu/

Urano (2000) investigou os efeitos da simplificação e elaboração léxicas para compreensão e aquisição incidental de vocabulário por falantes japoneses aprendendo o inglês como segunda língua. As modificações realizadas foram a substituição de palavras de baixa frequência por sinônimos de alta frequência, e a adição de sinônimos seguindo as palavras desconhecidas. Os resultados deste estudo sugerem que ambas as abordagens ajudam a melhorar a compreensão no nível sentencial, entretanto, a elaboração léxica resulta em aquisição de vocabulário incidental, enquanto que a simplificação não.

Watanabe *et .al.* (2010) aplica seus estudos em pessoas com baixo letramento que navegam na Web em busca de informação. Foram usadas elaboração léxica via sinônimos mais simples, além de definições curtas da Wikipédia para definir entidades nomeadas que aparecem nos textos, além de enfatizarem estas. O conjunto de entidades nomeadas foi o proposto na avaliação conjunta de sistemas de reconhecimento de entidades nomeadas do português (HAREM[3]) (i.e., pessoa, organização, local, artefatos, cosias, eventos, entidades abstratas, quantidades, tempo, coisas produzidas por pessoas). Além disso, figuras para as entidades nomeadas da classe pessoa foram usadas.

3. O Projeto NorMan

3.1. O Método de Extração de Termos Dedicado a Manuais Técnicos

Durante a tarefa de simplificação léxica de manuais técnicos, é preciso tomar um cuidado especial com relação aos termos técnicos presentes no texto. Manuais técnicos contêm termos que não podem ser excluídos ou trocados por possíveis sinônimos da língua geral. Caso um termo técnico seja erroneamente suprimido do texto ou trocado por um mais simples durante o processo de simplificação léxica, o sentido final da sentença poderá ficar seriamente comprometido, prejudicando o entendimento do leitor. Para evitarmos esse tipo de situação, é proposto um novo método de extração de termos, sensível ao gênero de instruções. Este método foi baseado nas relações gera e habilita, que são comumente encontradas em manuais. Essas relações são de suma importância, pois elas demonstram explicitamente o que o leitor deve fazer para alcançar seu objetivo com segurança e livre de erros. Sabendo que as relações gera e habilita contêm as instruções propriamente ditas, conclui-se que as sentenças que possuem tais relações tem maior chance de conter candidatos a termos.

O método de extração de termos sensível ao gênero de instruções é dividido em cinco etapas, detalhadas a seguir. A Figura 1 ilustra todo o processo de extração.

ETAPA 1: A primeira etapa do método de extração é o pré-processamento dos textos[4] pelo parser PALAVRAS (Bick, 2000) para a extração do conhecimento morfossintático dos textos. O conhecimento extraído do parser é utilizado para a identificação das relações gera e habilita.

ETAPA 2: O arquivo resultante do pré-processamento pelo parser PALAVRAS é então usado como entrada pela etapa de extração na qual este arquivo é percorrido e são reconhecidas as sentenças de acordo com a marcação feita pelo parser PALAVRAS.

3 http://www.linguateca.pt/HAREM/

4 Neste trabalho, utilizamos as seções de instalação e instrução, pois nelas a frequência das relações gera e habilita é maior do que nas outras seções.

Assim que uma sentença é reconhecida, a mesma é submetida a uma busca por uma das formas gramaticais descritas por Moura (2008) e Delin *et. al*. (1994). As formas gramaticais usadas para a relação gera são as seguintes: "**Para + Infinitivo**", "**Se + Subjuntivo**" e "**Para + Frases**". Para a relação habilita, foram usadas as seguintes formas gramaticais: "**Sequencia**", "**condição Antes**" e "**condição Depois**".

ETAPA 3: Assim que uma relação gera ou habilita é identificada, a sentença então é filtrada por uma lista de padrões morfossintáticos, gerando a lista final de candidatos a termos. Os padrões morfossintáticos para extração de termos compostos são os mesmos usados por Baségio (Baségio, 2006 apud Ribeiro Jr, 2008), por exemplo, "substantivo adjetivo" e "substantivo preposição substantivo". Para o método de extração implementado neste projeto, o maior tamanho de termo composto aceito é três, isto é, são gerados uni, bi e trigramas. Para termos simples, são utilizados os mesmos padrões de Teline (2004) que extrai substantivos, adjetivos e verbos.

ETAPA 4: Nesta etapa, foram removidos da lista final de candidatos a termos todos aqueles n-gramas que estão presentes na *stoplist* que foi compilada para o projeto e-Termos[5] com algumas adaptações para atender o gênero de manuais de instruções.

ETAPA 5: Nesta última etapa, a lista de candidatos a termos é ranqueada pelo C-value (Frantzy and Ananiadou, 1997). O C-value é uma medida estatística usada para extração de termos multi-palavras. De posse dessa medida, o sistema faz o raqueamento dos candidatos a termos.

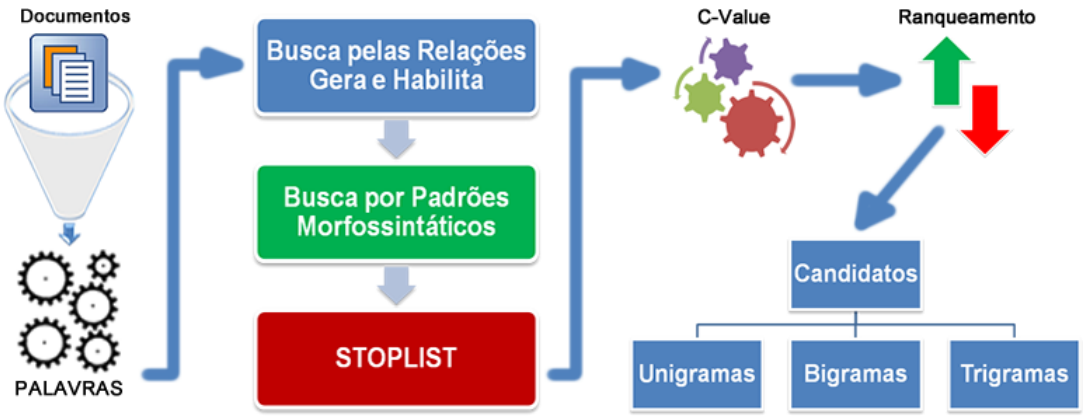


Figura 1. Processo de Extração de Termos para Textos Instrucionais

3.2. Aplicação da Extração de Termos em Sistemas de Adaptação Textual

O SIMPLIFICA é um sistema de autoria para apoiar a produção de textos simplificados, em que textos originais recebem simplificação automática com possível pós-edição pelos autores. Foi customizado para simplificar cada fenômeno sintático descrito em (Specia *et al.*, 2008) e mais o tratamento de adjuntos adverbiais longos, para atender as necessidades de alfabetizados de nível rudimentar e básico (Gasperin *et. al.*, 2010). Possui também o módulo de Simplificação Léxica e um avaliador automático da complexidade de textos (Aluisio *et. al*., 2010), baseado nas classes de alfabetização do INAF (2009), isto é, classifica um texto de acordo com três classes: adequado a

5 http://www.etermos.cnptia.embrapa.br/

alfabetizados em nível pleno, básico e rudimentar e pode indicar a necessidade de mais simplificação léxica ou sintática, fornecendo a possibilidade de simplificar via um processo cíclico que pode ser avaliado a cada momento que o usuário autor desejar.

A primeira etapa da simplificação léxica é a marcação das palavras consideradas complexas. Para isso, são usados três dicionários que foram desenvolvidos para o Projeto PorSimples. Um dicionário é composto por palavras comuns para crianças, outro composto por palavras frequentes e outro composto por palavras concretas. O sistema percorre o texto e verifica cada palavra. Se a palavra não está no dicionário de palavras simples e também não é um nome próprio, então ela é considerada uma palavra complexa. Para fazer a busca da palavra no dicionário de palavras simples, a mesma precisa estar lematizada e para isso foi utilizado o dicionário Unitex-PB[6] para descobrir o lema de cada palavra.

A busca por um lema em um dicionário pode se tornar um problema quando processamos palavras ambíguas, como por exemplo, a palavra "canto", que possui dois significados: o ponto ou área em que linhas e superfícies se encontram e formam um ângulo, ou então pode ser o verbo "cantar" conjugado na primeira pessoa do singular do presente do Indicativo. Para tratar essas ambiguidades eventualmente encontradas nos textos, o sistema utiliza o etiquetador gramatical MXPOST POS tagger (Ratnaparkhi, 1996) treinado com o conjunto de etiquetas NILC tagset[7], que tem como objetivo etiquetar o texto identificando a categoria gramatical de cada palavra automaticamente.

Após a etiquetação das palavras, todas aquelas que não são substantivos, preposições ou numerais são selecionadas e então a busca no dicionário de palavras simples é feita utilizando a informação da categoria gramatical de cada palavra. Como se trata de uma operação automática, ela não é totalmente precisa e algumas palavras podem não constar no dicionário. Além disso, se o sistema não foi capaz de categorizar gramaticalmente a palavra, a busca no dicionário é feita utilizando apenas o lema e mesmo que a palavra não seja encontrada ela é marcada como complexa. Em seguida, todas as palavras que foram marcadas são associadas com sinônimos mais simples. Para isso, dois recursos são utilizados: o tesauro TeP 2.0[8] e o PAPEL[9]. Esse procedimento é executado quando o usuário clica na palavra marcada, disparando uma consulta por sinônimos no dicionário de palavras simples.

Sabendo-se que em manuais de instrução os termos técnicos não podem sofrer substituição, pois a substituição de determinado termo pode alterar a interpretação de determinada instrução no manual e, em razão disso, causar acidentes ou prejuízos, foram implementadas algumas adaptações no método de simplificação léxica para atender o gênero de manuais de instruções.

Sabendo-se dos resultados positivos que a utilização de elaborações textuais traz para a compreensão do texto (Watanabe *et. al.*, 2010), decidiu-se aplicar esta estratégia para elaborar os termos técnicos durante a simplificação léxica de manuais de instrução. Desta forma, estes receberam uma definição adicional, com o uso da primeira oração da Wikipédia que relata o termo.

---

6 http://www.nilc.icmc.usp.br/nilc/projects/unitexpb/web/dicionarios.html
7 http://www.nilc.icmc.usp.br/nilc/TagSet/ManualEtiquetagem.htm
8 http://www.nilc.icmc.usp.br/tep2/
9 http://www.linguateca.pt/PAPEL/

O primeiro passo na adaptação da simplificação léxica foi a geração de um dicionário a partir da lista de termos extraídos e seu respectivo C-Value pelo NorMan. Este dicionário foi então inserido como um novo dicionário de palavras simples no sistema de simplificação léxica. Dessa forma, o sistema passa a considerar os termos técnicos como palavras simples, evitando que os mesmos sejam marcados como palavras complexas, sendo estes associados com definições da Wikipédia, sempre que houver uma, que podem ser usadas como explicações acompanhando os termos.

Tendo em vista que o método de extração de termos sensível ao gênero de manuais de instruções extrai unigramas, bigramas e trigramas, o sistema foi adaptado para percorrer o texto em três etapas e não apenas uma vez. Na primeira etapa, o sistema percorre o texto formando trigramas e verificando cada um deles no dicionário gerado pelo NorMan. Caso haja intersecção entre dois trigramas, aquele com o maior C-Value é escolhido para ser elaborado por uma definição da Wikipédia. Na segunda etapa, o sistema percorre o texto novamente formando bigramas e verificando cada um deles. Caso haja intersecção entre dois bigramas, vence aquele com maior C-Value, assim como ocorre com os trigramas, porém caso haja intersecção com trigramas, o trigrama tem preferência. Por último, o texto é percorrido da forma tradicional, buscando cada palavra no dicionário e caso haja intersecção com bigramas ou trigramas, os bigramas e trigramas têm prioridade sobre unigramas.

Após identificar todos os termos técnicos do dicionário do NorMan no texto, tais termos são marcados para receber informações adicionais. Quando o usuário clica em um termo técnico marcado, uma consulta é feita na Wikipédia e é apresentado para o usuário um link como uma inf**o**rmação adicional (Figura 2), contendo as duas primeiras orações da Wikipédia para aquele termo técnico.



Figura 2. Trecho de manual com termos técnicos marcados para receber informações adicionais da Wikipédia

4. Avaliação Intrínseca e Extrínseca

A avaliação intrínseca foi realizada através da comparação dos resultados da extração de termos feita pelo NorMan com outros quatros métodos que foram descritos no trabalho de (Teline, 2004): o método linguístico, estatístico e dois métodos híbridos, sendo que o Híbrido-A utilizou indicadores estruturais no texto para ajudar na identificação de termos e o método Híbrido-B não utilizou estes indicadores. Essa avaliação teve como objetivo verificar se o método NorMan trazia resultados diferentes dos demais métodos, o que significaria que o método trouxe novidades. O corpus utilizado nesta avaliação

continha 111.164 palavras e era composto pelas seções de "instruções" e "instalação" de 50 manuais de produtos tecnológicos. A comparação entre os métodos foi feita através do cálculo da intersecção dos candidatos a termos extraídos pelos métodos devido a falta de uma lista de referência para o cálculo de precisão, cobertura e medida F. Também foi calculado a similaridade entre os métodos através do cálculo do coeficiente Jaccard (J), que é uma estatística usada para comparar a similaridade e diversidade de conjuntos de amostra.

Para unigramas, a intersecção entre o NorMan e os quatros métodos foram de 71,6% para método linguístico (J=0,44), 77,3% para o método estatístico (J=0,52), 42,4% para o método Híbrido-A (J=0,37) e 51,8% para o Híbrido-B (J=0,44). Na extração de bigramas, o método NorMan teve uma intersecção de 42,5% com o método Linguístico (J=0,15), 36,3% com o método estatístico (J=0,03), 10,8% com o método Híbrido-A (J=0,09) e 12,6% com o Híbrido B (J=0,10). Na extração de trigramas, o método NorMan teve uma intersecção de 46,9% com o método Linguístico (J=0,17), 22,2% com o método estatístico (J=0,03), 10,6% com o método Híbrido-A (J=0,09) e 15,1% com o Híbrido-B (J=0,13). Estes resultados demonstram que, embora na extração de unigramas tenha havido intersecção de até 77,3%, para bigramas e trigramas a intersecção foi bem mais baixa, demonstrando que o método trouxe novidades, já que foi capaz de recuperar termos diferentes dos métodos de extração usados com textos científicos. O cálculo do coeficiente de similaridade (J) também demonstrou que apesar de a similaridade entre os métodos de unigramas ter variado entre 0,37 e 0,44, para bigramas e trigramas a similaridade entre os métodos foi bem mais baixa, variando de 0,03 a 0,15, reforçando que o método trouxe novidades.

Também foi feita uma avaliação do método através do cálculo da estatística Kappa, que é comumente utilizada para medir a concordância entre diferentes anotadores em tarefas de PLN. Nesta avaliação, o corpus foi limitado a somente as seções de "instalação" e "instruções" dos manuais de televisores. Essa escolha se deu pelo fato de o produto ser extremamente popular e, portanto, facilitaria a tarefa dos avaliadores. Não houve recursos financeiros disponíveis para a contratação de técnico especialista em TVs, portanto os avaliadores foram membros do próprio laboratório onde a pesquisa foi realizada. Foram selecionados os 150 primeiros termos extraídos da lista de uni, bi e trigramas. A lista de unigramas foi avaliada por três avaliadores diferentes, sendo que o avaliador 1 identificou 104 (69,33%) termos entre os 150 candidatos da lista de unigramas. O avaliador 2 identificou 126 (84%) e o avaliador 3 identificou 105 (70%). O valor da medida Kappa resultante dessa avaliação foi de 0.507. Este valor representa que houve um índice moderado de concordância entre os avaliadores. A lista de bigramas foi avaliada por dois avaliadores diferentes, sendo que o Avaliador 4 identificou 91 (60,66%) termos entre 150 candidatos da lista. O avaliador 5 identificou 75 (50%) termos. O valor da medida Kappa resultante dessa avaliação foi de 0.069. Este valor representa um índice de concordância pequeno/fraco. A lista de trigramas foi avaliada por outros dois avaliadores, sendo que o avaliador 6 identificou 99 (66) termos entre 150 candidatos da lista. O avaliador 7 identificou 116 (77,3%) termos. O valor da medida Kappa resultante dessa avaliação foi de 0.061. Este valor representa um índice de concordância pequeno/fraco. Os valores de concordância encontrados nesta etapa da avaliação podem ser um reflexo de que os avaliadores não são especialistas e, além disso, evidencia a dificuldade da avaliação manual da tarefa de extração automática de termos neste gênero de textos.

A avaliação extrínseca tinha como objetivo verificar se adaptação da simplificação léxica para o gênero de manuais de instrução ajudaria pessoas que trabalham em funções técnicas a entender melhor um manual de instrução. Esta avaliação foi feita através da submissão de um trecho de manual de instrução de televisão no sistema SIMPLIFICA já adaptado para simplificação léxica de manuais de instruções. O trecho escolhido para avaliação continha 16 termos distintos marcados como termos para elaboração léxica. Dentre estes 16 termos, 12 foram reconhecidos pelo sistema como termo técnico, pois pertencia às listas de candidatos a termo geradas pelo método NorMan e os quatro restantes, como palavras complexas. 15 voluntários participaram da avaliação, que consistia na leitura do trecho do manual e sua interação com os termos marcados. Os voluntários eram funcionários da Prefeitura do Campus da USP em São Carlos, com faixa etária entre 33 e 68 anos, escolaridade entre ensino fundamental incompleto e pós-graduação (um único funcionário) e com funções que podem ser divididas entre auxiliares e técnicos. Após a leitura, os voluntários eram questionados sobre o conhecimento prévio de cada termo marcado e a utilidade das informações extras fornecidas pela Wikipédia. Por fim, cada voluntário respondeu 16 questões de múltipla escolha, referente a cada termo destacado no trecho. A contabilização das respostas (Figura 3) mostrou que os voluntários responderam que já conheciam 67,7% dos termos técnicos e 100% das palavras complexas. Os voluntários consideraram 62,19% das informações da Wikipédia como sendo úteis contra apenas 6,6% das palavras complexas. Este valor de apenas 6,6% de utilidade nas palavras complexas é reflexo da qualidade do recurso léxico disponível, pois os sinônimos apresentados como opções de substituição fugiam do contexto do texto e, portanto, não foram considerados úteis. A média de acerto no questionário foi de 86,6% tanto para termos técnicos como para palavras complexas. O fato de os voluntários responderem que tinham conhecimento prévio sobre 67,7% dos termos técnicos e mesmo assim considerarem 62,19% das informações da Wikipédia como sendo úteis demonstra que o sistema, de fato, atingiu seu objetivo, que era auxiliar os voluntários a compreenderem melhor o manual de instrução.

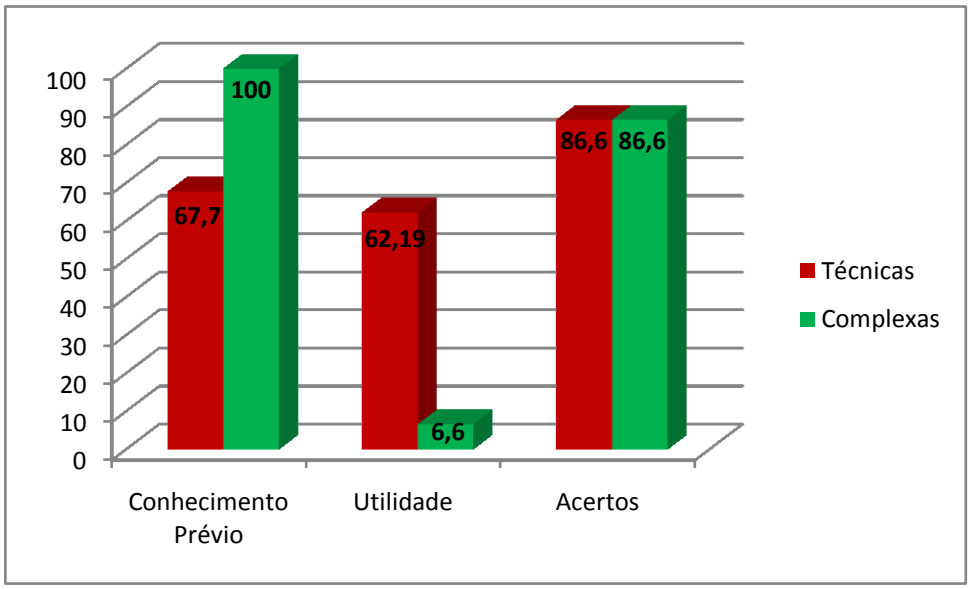


Figura 3. Comparação de conhecimento prévio, utilidade e acertos entre as palavras complexas e os termos técnicos

Foram feitas análises de correlações para verificar a existência de relações entre duas ou mais variáveis através do cálculo estatístico qui-quadrado. Estes cálculos mostraram que o cargo exercido voluntário teve influência no índice de utilidade das informações trazidas pela Wikipédia e dicionário de sinônimos (qui-quadrado = 9.3548, 0.01>P>0.001, Grau Liberdade (G.L.) = 1). Além disso, o conhecimento prévio dos termos técnicos estava correlacionado ao nível de acerto dos termos técnicos sinônimos (qui-quadrado = 4.0074, 0.05>P>0.02, G.L. = 1). Também foi verificado que a idade (qui-quadrado = 6.4071, 0.02>P>0.01, G.L. = 1) e o nível de escolaridade (qui-quadrado = 25.3215, P<0.001, G.L. = 1) influenciaram o nível de acerto e a idade também influenciou a utilidade (qui-quadrado = 4.0625, 0.05>P>0.02, G.L. = 1).

5. Considerações Finais

O objetivo principal do projeto NorMan foi o desenvolvimento de um método de extração de termos de manuais de instruções de produtos tecnológicos que pudesse extrair candidatos a partir de um conjunto pequeno de textos. Este cenário difere da aplicação da extração de termos de textos científicos na qual o tamanho do corpus é bem grande, dada a disponibilidade deste gênero textual. Como estudo de caso da aplicação do método, foi adaptado o sistema de autoria de textos simplificados SIMPLIFICA, para realizar a extração de termos de manuais técnicos e apresentar uma definição extraída da Wikipédia para os termos. As avaliações intrínsecas reportaram que o sistema produz resultados distintos dos métodos apresentados na literatura científica, recuperando termos específicos para o contexto, que seriam negligenciados pelos métodos propostos para o gênero de textos científico.

Pela avaliação extrínseca pode ser observado o aumento de reconhecimento de utilidade dos mecanismos de elaboração textual apresentado para os termos técnicos em comparação com as palavras complexas, ou seja, a informação textual elaborada (extraída da Wikipédia) apresentou maior utilidade pelos usuários. No teste realizado com os usuários também foi reportado um aumento de entendimento adquirido a respeito dos termos técnicos, depois de se interagir com os mecanismos que apresentavam a definição extraída da Wikipédia. Esses resultados destacam que a extração de termos, combinada com a elaboração textual, impactou positivamente na compreensão dos textos pelos usuários do experimento. Como trabalhos futuros, indicamos a adaptação sintática para o mesmo contexto do sistema (manuais técnicos), criação de listas de referências de termos do gênero de manuais de instruções para produtos tecnológicos e aprofundamento do estudo de uso de recursos de linguísticos, como a Wikipédia, disponíveis livremente para criação de outros métodos de elaboração textual.

Referências

Aluisio, S. M. and Gasperin, C. (2010). Fostering Digital Inclusion and Accessibility: The PorSimples project for Simplification of Portuguese Texts. In the Proceedings of the NAACL HLT 2010 Young Investigators Workshop on Computational Approaches to Languages of the Americas, June, 2010, Los Angeles, California, Association for Computational Linguistics, p. 46-53.

Aluísio, S. M.; Specia, L.; Gasperin, C.; Scarton, C. E. (2010). Readability Assessment for Text Simplification. In: NAACL 5th Workshop on Innovative Use of NLP for Building Educational Applications (BEA-2010), 2010, Los Angeles.

Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications. New York : ACL, 2010. v. 1. p. 1-9.

Aouladomar, F. (2005). Towards Answering Procedural Questions, Workshop KRAQ05, IJCAI 05, Edinburgh, p. 21-30. Disponível em: http://www.irit.fr/recherches/ILPL/kraq05V1.pdf

Bick, E. (2000). The Parsing System "Palavras". Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework. Tese (Doutorado) – Arhus University, 2000.

Burstein, J., Shore, J., Sabatini, J., Lee, Y.W., Ventura, M. (2007). The automated text adaptation tool. In NAACL '07: Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations on XX, p. 3-4.

Burstein, J. (2009). Opportunities for Natural Language Processing Research in Education. In the Proceedings of CICLing, p. 6-27.

Delin, J.; Hartley, A.; Paris, C., Scott, D.; Vander Linden, K. (1994). Expressing Procedural Relationships in Multilingual Instructions, Proceedings of the Seventh International Workshop on Natural Language Generation, p. 61-70.

Devlin, S. and Unthank, G. (2006). Helping aphasic people process online information. Proceedings of ASSETS, 2006, p. 225-226.

van der Eijk, P. (1997). Controlled languages in technical documentation, Proceedings of the Computational Linguistics in the Netherlands, 1997.

Elhadad, N. (2006). Comprehending technical texts: predicting and defining unfamiliar terms. AMIA Annu Symp Proc. 2006:239-43.

Fontan, L.; Saint-Dizier, P. (2008). Analyzing the explanation structure of procedural texts: dealing with Advices and Warnings. In: International Symposium on Text Semantics (STEP 2008), Venise, Johan Bos (Ed.), Association for Computational Linguistics (ACL) p. 115-127.

Frantzy, K. T.; Ananiadou, S. (1997). Automatic Term Recognition using Contextual Cues. Manchester Metropolitan University. Third Delos Workshop Cross-Language Information Retrieval Zurich, 5-7 March 1997. Disponível em: http://www.ercim.eu/publication/ws-proceedings/DELOS3/Frantzi.pdf

Gasperin, C. Maziero, E. and Aluísio, S.M. (2010). Challenging Choices for Text Simplification, In: Proceedings of PROPOR 2010, p. 40-50, António Branco, Aldebaro Klautau, Renata Vieira, Vera Lúcia Strube de Lima (Eds.): Computational Processing of the Portuguese Language, Springer 2010, v. 6001. p. 40-50.

Goldman, A. (1970). A Theory of Human Action, Prentice-Hall, 230 pp.

INAF (2009). Instituto P. Montenegro e Ação Educativa. INAF Brasil - Indicador de Alfabetismo Funcional - 2009. Disponível em: http://www. ibope. com.br/ipm/relatorios/relatorio_inaf_2009.pdf

Lopes, L.; Vieira, R.; Finatto, M. J.; Martins, D. (2010). Extracting compound terms from domain corpora. Journal of the Brazilian Computer Society (Impresso), p. 1-13, 2010.

Moura, A. B. N. (2008). Tipologia Textual e Ativação de Terminologia: um Estudo em Manuais Técnicos de Produtos Tecnológicos. Tese de Doutorado em Letras - Universidade Federal do Rio Grande do Sul, p. 287.

Paris, C.; Scott, D. (1994). Stylistic variation in multilingual instructions. *In Proceedings of the Seventh International Workshop on Natural Language Generation*, Kennebunkport, MN, 21--24 June 1994, pages 45--52.

Paris, C.; Vander Linden, K., Fischer, M.; Hartley, A.; Pemberton, L.; Power, R.; Scott, D. (1995). A support tool for writing multilingual instructions. *In Proceedings of the 14th International Joint Conference on Artificial Intelligence*, p. 1398--1404, Montreal, Canada, 1995.

Ratnaparkhi, A. (1996). A maximum entropy model for part-of-speech tagging. In Brill, E. & Church, K., editors, *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, p. 133–142. Nova Jérsei, EUA.

Ribeiro Jr., L. C. (2008). OntoLP : Construção Semi-Automática de Ontologias a partir de Textos da Língua Portuguesa. Dissertação (Mestrado) - Programa de Pós-Graduação em Computação Aplicada, Universidade do Vale do Rio dos Sinos, 2008.

Siddharthan, A. (2003): Syntactic Simplification and Text Cohesion. PhD thesis, University of Cambridge (2003).

Specia, L.; Aluísio, S. M.; Pardo, T. A. S. (2008). Manual de Simplificação Sintática para o Português. Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional (NILC-TR-08-06), 27 p., Junho 2008, São Carlos-SP. Disponível em: http://caravelas.icmc.usp.br/wiki/index.php/Publications.

Teline, M. F. (2004). Avaliação de métodos para extração automática de terminologia de textos em português. ICMC-USP, São Carlos, 2004. Dissertação de Mestrado.

Urano, K. (2000): Lexical Simplification and Elaboration: Sentence comprehension and incidental vocabulary acquisition. Unpublished master's thesis, University of Hawai'i at Manoa, Honolulu (2000). Disponível em: http://www.urano-ken.com/research/thesis.pdf

Watanabe, W.M., Candido Jr, A., Amancio, M.A., Oliveira, M., Pardo, T.A.S., Fortes, R.P.M., Aluísio, S.M. (2010). Adapting web content for low-literacy readers by using lexical elaboration and named entities labeling. In the Proceedings of the W4A-7th International Cross-Disciplinary Conference on Web Accessibility 2010, (2010). Nova York: ACM Press, v. 1, 1--9 (2010)

Young, D.N. (1999). Linguistic simplification of SL reading material: Effective Instructional Practice? The Modern Language Journal, 83(3), 350--366 (1999)

Rahimi, M. Y. (2011). Use of Syntactic Elaboration Techniques to Enhance Comprehensibility of EST Texts. English Language Teaching, Vol. 4, No. 1, 11-17.

Desenvolvimento do Modelo WSIVM para Aperfeiçoar a Segurança em SOA e Serviços Web

Rafael Bosse Brinhosa, Carla Merkle Westphall, Carlos Becker Westphall

Departamento de Informática e Estatística (INE)
Programa de Pós-Graduação em Ciência da Computação (PPGCC)
Universidade Federal de Santa Catarina (UFSC) – Florianópolis, SC – Brazil

{brinhosa, carlamw, westphal}@inf.ufsc.br

Abstract. *The SOA architecture primarily based on Web Services is having a steady adoption, despite this growth is lower than expected when it was established, that´s mainly because of difficulties defining security related aspects. Web Services inherited a lot of well-known security problems of web applications and brought new ones. The majority of attacks today are consequences of bad input validation in the application-level. This paper presents how to implement an input validation model for Web Services, which can be used to prevent Cross-site Scripting and SQL injection through using predefined models´ specification of valid inputs. The proposed WSIVM (Web Services Input Validation Model) consists of a XML Schema, a XML Specification and a module for performing input validation according to the schema. A case study showing the performance results is also presented.*

Resumo. *O uso da arquitetura SOA baseado principalmente na utilização de serviços web tem obtido larga adoção, porém, devido às dificuldades encontradas quanto aos aspectos de segurança, este crescimento tem sido menor do que o esperado. O uso de serviços web herdou muitos problemas de segurança conhecidos em aplicações web e trouxe outros novos. Atualmente a maioria dos ataques são conseqüência da má validação de entradas de dados no nível das aplicações. Este trabalho propõe um modelo para validação de entradas de dados para serviços web que pode ser utilizado para impedir ataques como Cross-site Scripting e SQL injection através da especificação de modelos pré-definidos de entradas válidas. O modelo proposto denominado WSIVM (Web Services Input Validation Model) possui um XML Schema, uma especificação XML e um módulo que faz a validação das entradas de acordo com a especificação. Um estudo de caso demonstrando resultados de desempenho também é apresentado.*

1. Introdução

Com o advento de diferentes tecnologias de colaboração e compartilhamento de informações, novos modos de interações estão evoluindo e gerando novos requisitos para o desenvolvimento de aplicações distribuídas. Empresas estão testemunhando o aumento de colaboração e compartilhamento de informações e uma maior necessidade do uso de recursos distribuídos e de computação [Belapurkar et al., 2009].

O paradigma de arquitetura orientada a serviços (SOA - *Services Oriented Architecture*) transformou a Internet de um repositório de dados para um repositório de

serviços [Saleem et al. 2010]. No estilo SOA uma aplicação é composta por serviços reusáveis que são integrados através de suas interfaces padronizadas.

A tecnologia de serviços web (*Web Services*), fundamentada no uso de padrões abertos, facilita a troca de informações, a interoperabilidade e o reuso de software e, por isso, é considerada um dos principais componentes da arquitetura orientada a serviços. Serviços web são componentes de software descritos que podem ser descobertos e usados para implementar aplicações. Serviços web são adequados para integrar sistemas heterogêneos, pois fazem uso extensivo da linguagem XML (*Extensible Markup Language*) [Nordbotten, 2009]. A interface de um serviço web, por exemplo, é descrita usando uma linguagem baseada em XML, a WSDL (*Web Services Description Language*). Além disso, a comunicação entre as partes de uma aplicação distribuída é executada usando mensagens SOAP (*Simple Object Access Protocol*) baseadas em XML.

Há um uso prático grande de serviços web prontos e disponíveis, que podem ser usados para se obter informações sobre o clima, sobre ações da bolsa de valores, sobre o código postal (CEP) [Cep, 2011] ou ainda para disponibilizar informações organizacionais do governo federal [Siorg, 2009].

Apesar do paradigma SOA apresentar diminuição de custos com a eliminação de esforços redundantes por reusar software, a segurança é uma das principais preocupações, conforme o pesquisador do Instituto Gartner [Feiman, 2010].

Para a implementação de segurança em serviços web, diversos padrões e especificações foram criados. No entanto, apenas o uso dos padrões corretos não garante que a segurança desejada seja obtida [Viega e Epstein, 2006] [Jensen et al. 2009] [Lakshminarayanan, 2010].

O relatório do instituto SANS (*SysAdmin, Audit, Network, Security*) [Sans, 2009] que lista os maiores riscos à segurança cibernética e a comunidade OWASP (*Open Web Application Security Project*) [Owasp, 2011] descrevem que a validação de entradas de dados pode ser um dos controles mais efetivos para a segurança de aplicações web.

A validação de entradas de dados [Bertino et al., 2009] [Scholte, 2011] é um conjunto de controles que uma aplicação deve fazer nas suas entradas de dados nos aspectos léxicos, sintáticos e de checagem de tipos, integridade e origem. A falta desses controles tornou-se um dos maiores problemas dos softwares que fazem interface com a Internet devido à facilidade de sofrer ataques e a necessidade de validar as entradas de dados dos usuários de forma a permitir apenas entradas confiáveis.

Desta forma, no ambiente de SOA e serviços web, usados por várias empresas, o aperfeiçoamento da segurança com o uso de mecanismos mais robustos de validação de dados passou a ser fundamental [Owasp, 2011] [Scholte, 2011].

Este trabalho propõe um modelo para validação de entradas de dados em serviços web, fornecendo proteção contra ataques baseados na entrada de dados maliciosos. O modelo proposto é chamado WSIVM (*Web Services Input Validation Model*) e contém um mecanismo para validação, um esquema XML, uma especificação XML e um módulo que faz a validação.

O texto do artigo está organizado da seguinte maneira: na seção 2 são apresentados os trabalhos relacionados; a seção 3 descreve os problemas de segurança em serviços web; na seção 4 o modelo WSIVM proposto é apresentado; a implementação é descrita na seção 5; um estudo de caso é mostrado na seção 6; e a seção 7 finaliza o artigo com as conclusões.

2. Trabalhos Relacionados

A falta de validação de entradas de dados é a maior causa de ataques a aplicações Web [Cenzic, 2009] [Owasp, 2011] [Scholte, 2011], sejam estas aplicações desenvolvidas com serviços web ou com outras tecnologias. Isso acontece porque a falta de validação de entradas de dados possibilita a ocorrência de vários dos ataques listados em [Sans, 2011] [Owasp, 2010]. Dentre os ataques que podem ser citados estão a injeção de códigos maliciosos usando SQL (*Structured Query Language*) e *Cross-Site Scripting* que permite a execução de códigos (*scripts*) no browser do lado do cliente com o objetivo de realizar uma ação maliciosa [Owasp, 2010].

Muitos trabalhos têm sido realizados no intuito de garantir a validação de entradas em aplicações web como a biblioteca *Microsoft Anti-Cross Site Scripting Library* [Microsoft, 2010] e o uso de soluções Open-source em PHP. No entanto, existem poucos mecanismos específicos para serviços web.

No que tange a mecanismos de implementação de segurança para serviços web não seguros, o MIT (*Massachusetts Institute of Technology*) possui uma implementação denominada WS-Security Wrapper [Ssa, 2007] que é um intermediador entre o serviço web e o cliente fazendo a validação de certos aspectos. No entanto, este trabalho foi desenvolvido para possuir compatibilidade apenas com serviços web desenvolvidos em plataforma Microsoft.Net versão 1.1 e não inclui características como a validação de entradas de dados pré-definidas.

Um mecanismo reusável e independente para entrada de dados é muito importante no processo de criação de um serviço web seguro. O mecanismo proposto neste trabalho, o WSIVM, ajuda nesta tarefa de forma diferenciada dos demais trabalhos analisados. Primeiramente, porque foca principalmente no aspecto de tratamento de entrada de dados, diferente do trabalho IAPF (*Integrated Application and Protocol Framework*) [Sidharth e Liu, 2007] que busca abordar todos os aspectos de segurança relacionados com serviços web. Além disso, outros trabalhos [Sun e Li, 2008] [Nordbotten, 2009] tem seu foco no uso de tecnologias existentes como a cifragem XML para garantir a segurança de serviços web, mas não mencionam a validação de entradas de dados.

Com relação aos aspectos de validação de entradas em aplicações web, existem alguns trabalhos como [Lin e Chen, 2009] que desenvolveram uma ferramenta que insere automaticamente a validação de entradas no lado do servidor eliminando vulnerabilidades de inserções maliciosas. No entanto, esta abordagem possui uma desvantagem que é a grande obtenção de falsos-positivos, isto é, o validador pode falhar considerando uma mensagem inválida, no entanto, a mensagem é válida.

Além dos trabalhos já relacionados existe ainda outra categoria de trabalhos, focados no desenvolvimento de *firewalls* como [Bebawy et al. 2005], que trata a proteção contra ataques de negação de serviço e de estouro de pilha. Os firewalls de XML, apresentado em [Blyth, 2009], aborda validação da estrutura do conteúdo XML,

mas não do conteúdo propriamente dito. O trabalho de [Loh et al., 2006] menciona a proteção contra injeção de SQL através do *XML Schema* e de uma lista negra pré-compilada de comandos SQL - essa abordagem costuma apresentar muitos falsos-positivos, no entanto, detalhes sobre a eficácia deste trabalho com testes mais extensos não foram apresentados.

Dentre os trabalhos relacionados observa-se que existe uma falta de trabalhos abordando a validação de entradas especificamente para serviços web.

3. Problemas de Segurança em Serviços Web

Serviços web criam novos riscos à segurança das organizações porque antigos métodos de proteção como *firewalls* e antivírus não são capazes de protegê-los. Por serem desenvolvidos para fazerem uso do protocolo HTTP (*Hypertext Transfer Protocol*) utilizando a porta 80, os *firewalls* comuns que atuam na camada de rede permitem o fluxo normal das requisições HTTP sem bloqueá-lo.

Além disso, suas funcionalidades são expostas através de arquivos WSDL, sendo que através das descrições dos métodos e variáveis do arquivo WSDL é possível obter importantes informações para um ataque conhecido como *WSDL scanning* [Belapurkar et al., 2009].

Existem ataques diretamente relacionados com a manipulação de dados: *Cross-Site Scripting* (ou XSS) e injeção de SQL. O trabalho de [Kieyzun et al., 2009] classifica os ataques XSS em dois tipos: ataques XSS de primeira ordem e de segunda ordem.

No ataque XSS de primeira ordem, a vulnerabilidade resulta de a aplicação inserir parte da entrada do usuário na própria página. O usuário malicioso usa engenharia social para convencer a vítima a clicar em uma URL que contenha código malicioso HTML/JavaScript. O navegador web do usuário mostra a página HTML e executa o JavaScript que fazia parte da URL maliciosa recebida, resultando no roubo de *cookies* de sessão ou outros dados sigilosos do usuário. Este tipo de ataque dificilmente pode ser elaborado contra serviços web.

Nos ataques XSS de segunda ordem a vulnerabilidade resulta do armazenamento das entradas maliciosas do usuário no banco de dados da aplicação, então, quando a página HTML é acessada, o código é executado e mostrado para as vítimas (por exemplo, em páginas de redes sociais). Os ataques de segunda ordem são mais difíceis de serem evitados porque a aplicação necessita validar ou sanear entradas de dados que possam conter código de *script* executável. Aplicado a serviços web é possível sofrer ataques ao apresentar dados não validados diretamente ao usuário. Por exemplo, fazendo-se uso de AJAX (***A**synchronous **J**avascript **a**nd **XML***), podem-se obter dados fornecidos por serviços web de terceiros que podem estar contaminados. Usando, por exemplo, o comando `document.write(xmlhttp.responseText);` caso a resposta desta chamada AJAX feita para um serviço web contenha dados HTML e JavaScript, estes dados serão interpretados e executados oferecendo risco ao usuário.

O ataque de injeção de código SQL (*SQL injection*) funciona através de entradas de dados maliciosos visando a execução de comandos SQL no banco de dados [Clarke, 2009] [Kieyzun et al., 2009] [Owasp, 2010].

Em serviços web que não possuem o devido tratamento de exceção, a mensagem de erro pode conter dados valiosos para o atacante utilizar. Assim, através de tentativa e erro, o atacante pode descobrir que tecnologia de banco de dados está sendo utilizada, que tabelas existem que podem ser exploradas e todas as informações necessárias para efetuar um ataque. Ataques de injeção de SQL podem elevar privilégios, sendo possível executar comandos em modo de administrador no servidor comprometido. É possível testar se um serviço web é vulnerável através do envio de requisições SOAP com os parâmetros devidamente manipulados. Por exemplo, enviando “"1=1 –“ como parâmetro para um determinado serviço, é possível que se tenha como retorno as saídas das figuras 1 e 2. Podem-se observar nessas figuras, dois exemplos de respostas de saídas de erro do banco de dados que foram retornadas pelo servidor. Essas respostas podem ser utilizadas para descobrir detalhes de banco de dados e servir para enviar novas solicitações ao banco, obtendo-se mais detalhes, até efetuar comandos mais complexos no banco de dados, que podem resultar na elevação de privilégios, injeção de arquivos, roubo ou destruição do banco de dados.

Através da resposta de saída da Figura 1 pode-se identificar que se trata de um banco de dados MySQL e que a injeção do comando SQL foi efetuada, pelo tipo de erro retornado. Na resposta de saída da Figura 2, conclui-se que o nome de uma das colunas do banco de dados é `ItemId`, pela sintaxe do comando `select` do SQL e o banco de dados pode ser deduzido como Microsoft SQL Server, pelo uso da sintaxe de *stored procedure* “`dbo.`”.

```
ERROR: The query was not accomplished. Description: 1064 - You have an
error in your SQL syntax; check the manual that corresponds to your
MySQL server version for the right syntax to use near '1=1'' at line 1
```

Figura 1. Exemplo de resposta de saída 1.

```
Line 11: Incorrect syntax near ''')) or  ItemId in (select ItemId
from dbo.GetItemParents('4''. Unclosed quotation mark before the
character string ')) ) ) > 0 '.
```

Figura 2. Exemplo de resposta de saída 2.

O ataque do tipo *Blind SQL injection* trata-se de um tipo de injeção de SQL no qual os resultados não são exibidos para o atacante. É muito comumente utilizado contra serviços web, pois muitos servidores evitam que as mensagens de erro geradas pelo serviço cheguem ao usuário. Em serviços web geralmente o erro HTTP 500 é retornado quando uma tentativa de *Blind SQL injection* é efetuada, no entanto, existem técnicas como a medição dos tempos de resposta do servidor que podem ser utilizadas para determinar os parâmetros necessários para efetuar o ataque com sucesso.

Apesar de ser difícil conseguir informações confiáveis sobre incidentes de segurança ou vazamentos de dados como relatado no livro [Shostack e Stewart, 2008], em [Databreaches, 2009] é possível observar que os maiores casos de ataques em que ocorreram roubos de dados tiveram relação com a injeção de dados maliciosos. Na tabela existente em [Databreaches, 2009], por exemplo, consta o caso da entidade Heartland Payment Systems no qual foram perdidos 130.000.000 de dados e pode-se ler o motivo do incidente: injeção de código SQL.

4. WSIVM – Modelo de Validação de Entradas de Dados

Nesta seção é descrito o modelo WSIVM (*Web Services Input Validation Model*) que propõe validar as entradas de dados para fornecer segurança para serviços web. Inicialmente é descrito o funcionamento de serviços web sem o uso do modelo. Na seqüência é feita a explicação do funcionamento de serviços web usando o modelo proposto.

4.1. Funcionamento de Serviços Web sem o WSIVM

De maneira tradicional, um cliente encontra o serviço web que necessita. Este encontro ou descoberta é feita através de pesquisas em repositórios de serviços web. Os repositórios armazenam as referências dos serviços web no formato UDDI (*Universal Description, Discovery and Integration*). UDDI é um protocolo padrão que especifica um método para publicar e descobrir diretórios de serviços em uma arquitetura orientada a serviços. Depois da descoberta o cliente realiza uma requisição ao serviço web (Figura 3). O serviço web retorna a resposta do cliente contendo a resposta de sua requisição (Figura 4).

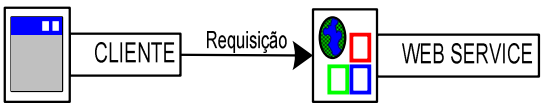


Figura 3. Requisição ao serviço web.

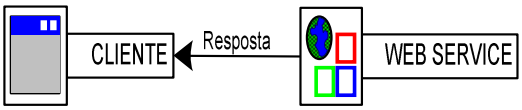


Figura 4. Resposta do serviço web.

Neste processo tradicional, a requisição do cliente é feita diretamente ao serviço web, que processa as entradas e retorna o resultado. O padrão utilizado para esta troca de mensagens é o formato SOAP baseado em XML.

Supondo que o serviço web executa a entrada do usuário sem fazer nenhum tipo de validação e que uma entrada de usuário faça parte da consulta SQL descrita em (1).

```
SELECT nome, idade FROM clientes WHERE nome=entrada;          (1)
```

Se o usuário fornecer como entrada o nome “Paulo”, que é uma entrada válida para o nome, o serviço web retornará como resposta o nome e a idade do cliente “Paulo”. Entretanto, se o usuário fornecer a seguinte *string* como entrada maliciosa “`Paulo' UNION SELECT nome, senha FROM clientes; --`“, a consulta SQL ficaria representada em (2).

```
SELECT nome FROM clientes WHERE
nome='Paulo' UNION SELECT nome, senha FROM clientes; --   (2)
```

Como o serviço web não realiza validação desta entrada representada em (2), retornará um valor sigiloso como a “senha do cliente” para o usuário que enviou a requisição maliciosa para o serviço web. Para o WSDL, a requisição é válida, pois se trata de uma *string* conforme especificado na descrição do serviço. No entanto, a confiança na entrada do usuário e a falta de validação da mesma acarretaram numa vulnerabilidade que divulgou dados sigilosos.

4.2. Funcionamento de Serviços Web com o WSIVM

O modelo WSIVM (*Web Services Input Validation Model*) propõe validar as entradas de dados para fornecer segurança para serviços web.

O modelo proposto traz algumas vantagens em comparação com a validação de entradas feita normalmente em uma aplicação, já que: (a) evita o desperdício de processamento do servidor com mensagens inválidas; (b) diminui a possibilidade de ataques de negação de serviço utilizando o conteúdo das mensagens; e, (c) é independente da tecnologia utilizada para o desenvolvimento interno dos serviços.

Com o WSIVM, o exemplo descrito na seção 4.1 é executado da seguinte forma: o usuário faz a requisição de forma usual conforme demonstrado na Figura 3, no entanto, esta requisição é validada pelo WSIVM (Figura 5).

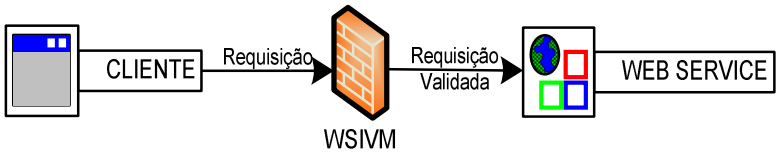


Figura 5. Requisição SOAP com WSIVM.

Caso seja enviada uma requisição maliciosa, como demonstrado no exemplo da entrada representada em (2), ao invés de enviar a senha, o mecanismo de validação do WSIVM faz a validação e retorna um erro genérico para o usuário. Desta forma o serviço web não recebe a solicitação mal-intencionada (Figura 6).

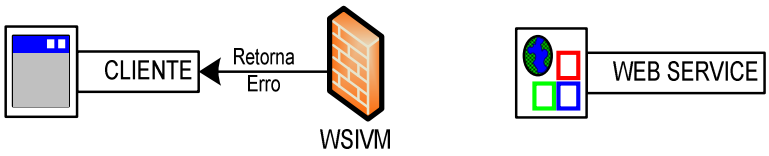


Figura 6. WSIVM bloqueia solicitação maliciosa.

De forma mais detalhada, o que ocorre quando uma mensagem chega para ser validada no WSIVM, é que a entrada enviada pelo usuário é validada através de um módulo que foi desenvolvido com a especificação XML que se encontra no servidor, conforme o algoritmo da Figura 7.

```
1. Recebe requisição SOAP
2. Processa mensagem XML
3. Checa parâmetros
4. Se checagem tiver falha
   Então Retorna mensagem de erro
   Senão Serviço web processa a requisição
```

Figura 7. Algoritmo da Validação das Mensagens.

Este processo evita o consumo desnecessário de recursos do servidor, assim, considerando o exemplo, o serviço web não executaria a consulta SQL caso fosse enviada uma requisição maliciosa.

A interação dos componentes do modelo WSIVM é representada na Figura 8.

O *WSIVMModule* é um módulo responsável por chamar os outros componentes.

O *WSIVMValidator* mapeia a mensagem SOAP obtendo os campos do corpo da mensagem e envia para o *WSIVMXMLLoader*.

O *WSIVMXMLLoader* carrega os elementos e as regras especificadas na especificação XML e verifica com o *WSIVMVerifier* a validade ou não da resposta.

O *WSIVMVerifier* contém todas as regras pré-definidas para validação das entradas e é responsável por validá-las.

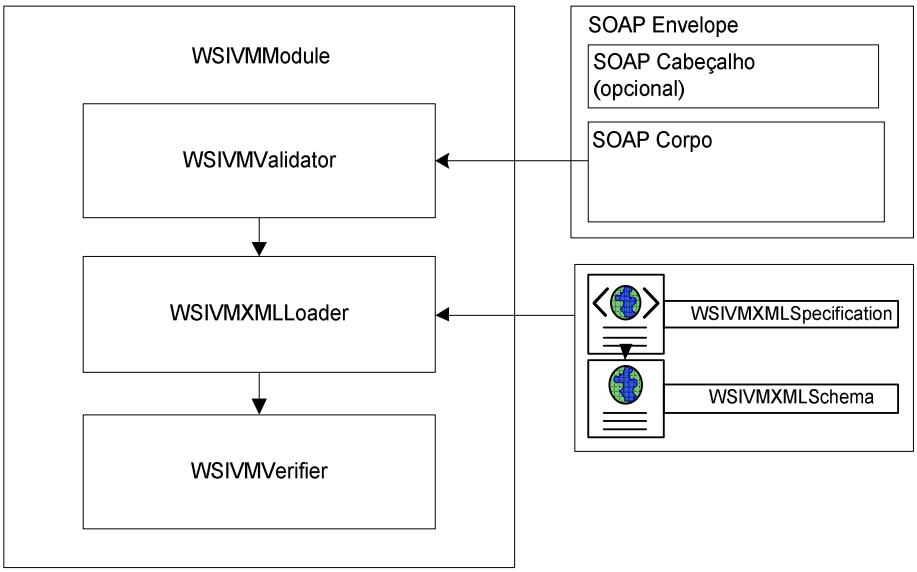


Figura 8. Funcionamento do WSIVM.

5. Desenvolvimento da Implementação

A implementação do modelo foi desenvolvida usando o servidor web Apache Tomcat e o framework Apache Axis2 para mensagens SOAP [Apache, 2010] (Figura 9). O Apache Axis2 foi escolhido para a implementação deste trabalho por sua capacidade de extensão através de módulos e a facilidade de interceptar mensagens SOAP através dos módulos.

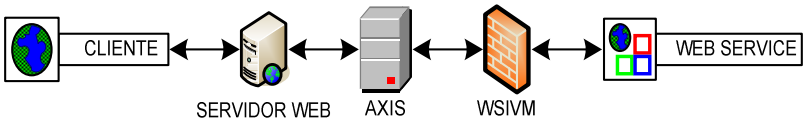


Figura 9. WSIVM.

Para a implementação do módulo de validação para o Apache Axis2 foi utilizado o módulo Rampart, que é o modo de extensão do Apache Axis2.

A fase de interceptação pode ser especificada no arquivo `Module.xml`. Foi escolhido fazer a interceptação da mensagem na fase de `PreDispatch`, que é a fase imediatamente anterior ao envio da mensagem para processamento no serviço web.

O funcionamento da implementação ocorre da seguinte forma: o cliente que pode ser uma aplicação, uma página web ou qualquer mecanismo capaz de se comunicar com um serviço web envia uma mensagem ao serviço web. Esta mensagem passa pelo servidor web que no caso utilizado é o Apache Tomcat. O servidor Web transmite essa mensagem SOAP para o Apache Axis. O Apache Axis envia a mensagem para ser analisada pelo WSIVM. Após a mensagem ser analisada, caso seja considerada inválida, é enviado um erro ao cliente pelo WSIVM. Caso seja considerada válida, é

transmitida para ser processada normalmente pelo serviço web retornando o resultado ao cliente (Figura 9).

Segue a descrição detalhada dos componentes do modelo WSIVM: WSIVMXMLSchema, WSIVMXMLSpecification e WSIVM Rampart module.

O **WSIVMXMLSchema** é a especificação do esquema de validação das entradas. Define o formato da especificação XML bem como os atributos válidos.

O **WSIVMXMLSpecification** é a especificação da validação das entradas. Especifica os parâmetros válidos de acordo com um conjunto de atributos pré-definidos. É utilizada para a validação das entradas do usuário. Dentre os parâmetros possíveis nas entradas estão:

- **OperationName:** nome da operação ou função exposta do serviço web a que se refere a validação
- **SanityzeOperation:** define se os parâmetros desta operação ou função podem ser reformulados se preciso para a remoção de caracteres não aceitos
- **ParamName:** nome do parâmetro ou campo a que se refere a validação
- **Allowed:** tipo de campo permitido, sendo válidos (text, html, html+java-script, email, number e all)
- **Length:** especifica o tamanho exato do campo
- **Maxsize:** especifica o tamanho máximo do campo
- **Minsize:** especifica o tamanho mínimo do campo
- **Nillable:** determina se é possível que o campo seja nulo ou não (true ou false)
- **regEx:** permite especificar uma expressão regular para validação

O **WSIVM Rampart module** é o principal componente do mecanismo implementado. Trata-se de um módulo do Apache Axis 2 que recebe os dados do cliente e faz a validação de acordo com a especificação XML chamando classes Java para realizar as validações. Este módulo é composto por um arquivo `wsivm.mar` que possui os seguintes componentes compactados:

- module.xml: contém a descrição do módulo, a classe que fará a validação e a fase em que a validação ocorrerá;
- MANIFEST.MF: arquivo de manifesto Java; e
- classes Java referentes à validação de entradas: WSIVMModule, WSIVMValidator, WSIVMXMLLoader e WSIVMVerifier.

6. Estudo de Caso

Como estudo de caso foi desenvolvido um sistema hipotético de cadastro de alunos para uma universidade denominado GerenciadorUniversidade. O sistema é composto por uma aplicação cliente denominada ClienteGerenciador e um serviço web servidor denominado Gerenciador.

Para o desenvolvimento do serviço web foi utilizada a linguagem Java e os testes de desempenho foram realizados utilizando o programa soapUI [Soapui, 2010].

Para o desenvolvimento do serviço web GerenciadorUniversidade criou-se uma classe com as operações `buscaAluno` e `cadastraAluno` e uma classe para lidar com as operações do banco de dados. Este serviço web foi desenvolvido sem nenhuma validação de entradas nas operações das classes Java propositalmente, deixando para o WSIVM a validação das mesmas.

A operação `buscaAluno` recebe um número de matrícula (Id) que deve ser um número inteiro maior que zero e com valor máximo de 10000 e retorna o cadastro do aluno contendo uma *String* com suas informações. A operação `cadastraAluno` recebe as informações do aluno que não devem conter código HTML ou Javascript e cadastra no banco de dados MYSQL. No banco de dados foi criada uma tabela `alunos` com os campos id (identificador auto-incremental), nome, idade, email, comentario, site e dataniver (data de aniversário).

Após a criação do serviço web e do banco, foi criada uma classe Java denominada `testaGerenciador` para testá-lo localmente.

Para o funcionamento do WSIVM foi especificado o WSIVMXMLSpecification – GerenciadorUniversidade (Figura 10), onde estão descritos os parâmetros para validação das entradas.

```
<?xml version="1.0" encoding="UTF-8"?>
<valid_inputs_specification xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
WebServiceID="GerenciadorUniversidade" xsi:noNamespaceSchemaLocation="valid_inputs_specification.xsd">
        <operation name="cadastraAluno">
        <input name="nome" type="String" min="5" max="20" accept="text" sanityze="true"/>
        <input name="idade" type="Integer" min="0" max="150" accept="number" sanityze="true"/>
        <input name="email" type="String" min="0" max="200" accept="email" sanityze="true"/>
        <input name="comentario" type="String" min="0" max="200" accept="text" sanityze="true"/>
        <input name="site" type="String" min="0" max="300" accept="url" sanityze="true"/>
        <input name="data" type="String" min="0" max="200" accept="regex" regexpattern= "(\\d{4})-(\\d{2})-
(\\d{2})" sanityze="true"/>
        </operation>
        <operation name="buscaAluno">
        <input name="id" type="Integer" min="0" max="10000" accept="number" sanityze="true"/>
        </operation>
</valid_inputs_specification>
```

Figura 10. WSIVMXMLSpecification – GerenciadorUniversidade.

A WSIVMXMLSpecification – GerenciadorUniversidade foi especificada de acordo com o WSIVMXMLSchema padrão do modelo e ambos os arquivos foram colocados na pasta C:\WSIVM do Windows.

Foi criado então o arquivo `Services.xml`, requerido para o Apache Axis 2 (Figura 11).

Foi gerado o pacote denominado `Gerenciador.aar` contendo a classe Gerenciador e MySQL e o descritor MANIFEST.MF e o arquivo Services.xml na pasta META-INF. Este pacote pode ser gerado como um pacote .jar, renomeado como .aar e colocado na pasta Services do Apache Axis2. Foi então copiado o arquivo do módulo (`wsivm.mar`) para dentro do diretório ..\Tomcat6.0\webapps\axis2\WEB-INF\modules.

Neste experimento foram realizados dois testes, um usando o modelo de validação de entradas WSIVM e outro não. O seguinte cenário foi configurado para a realização dos testes: 150 usuários são inicializados gradativamente seguindo a ordem

de 1 usuário sendo inicializado a cada 2 segundos. O teste é executado por 300 segundos que equivalem a 5 minutos. A base de dados é limpa para poder analisar o número de operações de cadastro de alunos efetuadas com sucesso.

```
<service>
        <parameter name="ServiceClass"
locked="false">exemplo.wsivm.universidade.Gerenciador</parameter>
        <operation name="cadastraAluno">
        <messageReceiver
class="org.apache.axis2.rpc.receivers.RPCMessageReceiver"/>
        </operation>
        <operation name="buscaAluno">
        <messageReceiver
class="org.apache.axis2.rpc.receivers.RPCMessageReceiver"/>
        </operation>
            <module ref="wsivm"/>
    <parameter
name="validationXML">file:///C:/WSIVM/valid_inputs_specification.xml</param
eter>
</service>
```

Figura 11. Services.xml – GerenciadorUniversidade.

O soapUI oferece uma interface amigável para a realização dos testes. Os testes são realizados fazendo chamadas diretas ao serviço web. A mensagem SOAP é enviada conforme o exemplo da Figura 12.

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:univ="http://universidade.wsivm.exemplo">
   <soap:Header/>
   <soap:Body>
      <univ:cadastraAluno>
         <univ:nome>John</univ:nome>
         <univ:idade>12</univ:idade>
         <univ:email>john@hsj.com</univ:email>
         <univ:comentario>Passou</univ:comentario>
         <univ:site>http://www.gol.com</univ:site>
         <univ:dataniver>1980-09-12</univ:dataniver>
      </univ:cadastraAluno>
   </soap:Body>
</soap:Envelope>
```

Figura 12. Exemplo de mensagem SOAP enviada pelo soapUI.

A Figura 13 apresenta o gráfico de resultados com os tempos de resposta dos testes com e sem validação de entradas. No eixo X é mostrado o tempo percorrido do teste e no eixo Y é mostrado o valor em milissegundos do tempo de resposta.

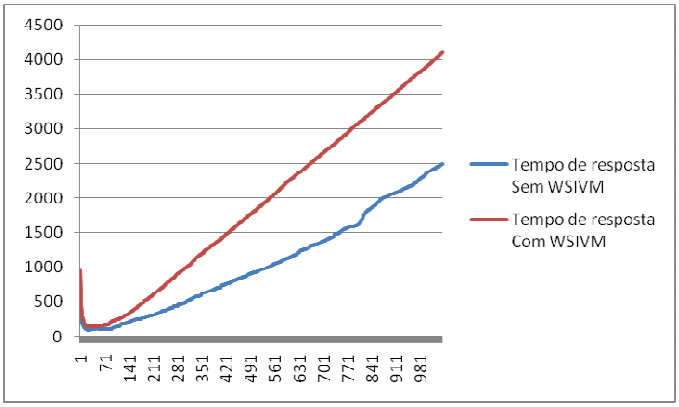


Figura 13. Tempos de resposta com e sem o uso do WSIVM.

A Figura 14 apresenta o gráfico de resultados com o *throughput* dos testes com e sem a validação de entradas do WSIVM. No eixo X é mostrado o tempo percorrido do teste e no eixo Y é mostrado o número de bytes por segundo (B/s).

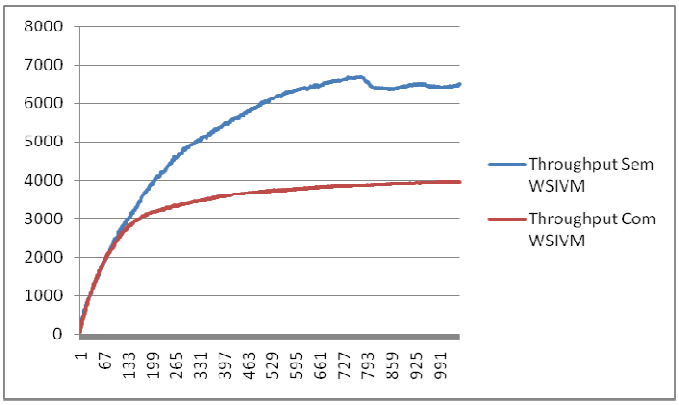


Figura 14. Throughput com e sem validação WSIVM.

Pode-se observar que a taxa de transferência de bytes por segundo (B/s) ou *throughput* cai bastante com o uso do WSIVM.

Tabela 1. Resultados consolidados do Estudo de caso.

Comparação	Tempo Min	Tempo Max	Tempo Médio	Bytes Transferidos	Bytes por segundo (*throughput*)	Inserções no Banco de dados
Sem WSIVM	35 ms	27848 ms	2494,85 ms	1974195 B	6506 B/s	10078
Com WSIVM	64 ms	13346 ms	4541,24 ms	1236330 B	4012 B/s	5134
Total	83%	-52%	82%	-37%	-38%	-49%

Na tabela 1, os cálculos que aparecem na linha Total em cada uma das colunas foram feitos da seguinte forma: diminui-se o resultado "Com WSIVM" do "Sem WSIVM" da coluna e este valor é dividido pelo valor "Sem WSIVM" da coluna.

Pode-se observar um aumento significativo na média dos tempos de resposta com o uso do WSIVM, com um aumento de 82%. O *throughput* total diminuiu em 38%. E o resultado do teste que é o cadastramento de alunos no banco de dados diminuiu em 49%, de 10078 alunos cadastrados para 5134.

O tempo gasto no processamento das mensagens XML teve impacto no desempenho das transações que são validadas uma a uma e comparadas com as regras especificadas de entradas consideradas válidas. A interpretação das mensagens é uma tarefa custosa que demanda grande quantidade de processamento e de memória para que a validação seja feita antes do processamento pelo serviço web e o retorno da resposta.

Também houve uma diminuição no número total de bytes transferidos, pois, devido ao maior tempo de processamento da mensagem, houve um menor número de mensagens processadas e conseqüentemente um menor número de respostas.

No caso apresentado, devido ao tempo gasto para a validação de cada mensagem, a aplicação pôde processar menos mensagens no mesmo período de tempo, resultando num menor número de inserções de alunos no banco de dados.

Uma diminuição no desempenho era esperada devido ao tempo para interpretar e percorrer as árvores XML para validação dos campos o que costuma ser custoso do ponto de vista de processamento. No entanto, a validação dos campos não permitindo que dados inválidos fossem inseridos pode compensar a perda de desempenho. Esta perda de desempenho poderá ser abordada em trabalhos futuros: testes utilizando outros mecanismos de interpretação de arquivos XML poderão ser efetuados, bem como o uso de um serviço web que demande mais processamento, comprovando o ganho com menor desperdício de processamento com mensagens inválidas. Mesmo assim, a proteção dos serviços obtida com o uso do modelo é uma vantagem que deve ser considerada.

Nos testes feitos nenhuma entrada indevida foi processada já que o ambiente estava devidamente configurado para filtrar as entradas inválidas.

7. Conclusões

Como as entradas de dados não validadas é o maior desafio para qualquer equipe de desenvolvimento de aplicações no ambiente Web e é a origem dos problemas de segurança de muitas aplicações [Owasp, 2010] [Owasp, 2011], um mecanismo reusável e independente para validação da entrada de dados, como o WSIMV proposto neste trabalho, é uma contribuição importante para a segurança de serviços web.

O WSIVM tem seu foco na validação das entradas de dados, permitindo a aceitação apenas de entradas válidas, já que é baseado na abordagem de lista branca, na qual apenas valores pré-definidos são aceitos e os demais são considerados inválidos.

Este modelo é particularmente interessante para o caso de serviços web que demandem grande quantidade de processamento das entradas de dados, pois, garantindo-se que apenas entradas válidas sejam aceitas, evita-se o desperdício de processamento pela aplicação.

Fazer a validação de entradas através do modelo apresentado é uma solução para aplicações legadas que não foram concebidas com validação de entradas de dados, pois se fazendo a validação nos pontos de entrada do serviço web, diminui-se a necessidade de fazer um número maior de alterações na aplicação já existente, reduzindo custos de desenvolvimento.

Além disso, conforme [Tsipenyuk et al., 2005] a abordagem de lista branca é mais confiável que a de lista negra. Na abordagem de lista negra todos os valores são considerados válidos menos os explicitamente especificados. Este tipo de abordagem tem alguns problemas, por exemplo, caso seja desejado validar um campo para que não contenha código HTML e seja criada uma lista negra com base na versão atual do HTML, caso surjam novas versões, esta lista pode deixar de ser considerada válida.

A abordagem de lista branca utilizada no WSIVM resulta na diminuição de falsos-positivos e num meio mais confiável de validação de entradas de dados. Já o trabalho de [Lin e Chen, 2009] possui como desvantagem a grande obtenção de falsos-positivos, isto é, o validador pode falhar considerando uma mensagem inválida, no entanto, a mensagem é válida.

Este trabalho buscou uma solução para impedir ataques de injeção de dados em serviços web, fornecendo proteção com um mecanismo reusável, que economiza o

processamento de chamadas maliciosas e é capaz de fornecer a validação de entradas de dados independentemente da implementação do serviço web que faz seu uso.

No estudo de caso foi possível observar que o aperfeiçoamento de segurança teve impacto negativo sobre o desempenho do serviço web implementado, o que é algo bastante comum nas pesquisas em segurança. Entretanto, a validação das entradas diminui a possibilidade de inserção de dados inválidos e, portanto, impede ataques ou a parada da execução correta do serviço web, compensando a queda do desempenho.

Existem diferentes aspectos que podem ser abordados em trabalhos futuros: (a) otimizar a implementação para melhorar o desempenho do modelo proposto (b) desenvolver um gerador semi-automático de especificações de segurança a partir do WSDL; (c) verificar mensagens SOAP e caminhos no formato XPath; (d) usar inteligência artificial ou um sistema de detecção de anomalias; e (e) fazer um esquema de retroalimentação do filtro de validação de entradas inválidas.

Referências

Apache. (2010) "Welcome to Apache Axis2/Java". Disponível em: http://axis.apache.org/axis2/java/core/

Bebawy, R. et al. (2005) "Nedgty: Web services firewall", In: Proc. of the IEEE International Conference on Web Services - ICWS, Orlando, IEEE, p.597-601.

Belapurkar, A. et al. (2009), Distributed Systems Security Issues, Processes and Solutions, Hoboken, NJ: John Wiley e Sons.

Bertino, E. et al. (2009), Security for Web Services and Service-Oriented Architectures, Springer-Verlag New York Inc.

Blyth, A. (2009) "An Architecture for an XML Enabled Firewall. International Journal of Network Security", v. 8, n. 1, p. 31-36, ISSN 1816-3548.

Cenzic. (2009) "Web Application Security Trends Report". Disponível em: <http://www.cenzic.com/downloads/Cenzic_AppSecTrends_Q1-Q2-2009.pdf>.

Cep. (2011) "CEPWebService". Disponível em: http://www.i-stream.com.br/webservices/cep.asmx

Clarke, J. (2009), SQL Injection Attacks and Defense. Syngress Media Inc.

Databreaches. (2009) "Top 10 Worst Data Losses or Breaches, updated". Disponível em: <http://www.databreaches.net/?p=7691>.

Feiman, Joseph. (2010) "Security in the SOA World: Methodologies and Practices", Enterprise Integration Summit, April 13-14, Sao Paulo, Brazil. Disponível em: http://www.gartner.com.br/tecnologias_empresariais/pdfs/brl37l_a4.pdf

Jensen, M.; Gruschka, N.; Herkenhöner, R. (2009). A survey of attacks on web services. Computer Science-Research and Development, v. 24, n. 4, p. 185-197.

Kieyzun, A. et al. (2009) "Automatic creation of SQL Injection and cross-site scripting attacks", In: *Proceedings of the 31st International Conference on Software Engineering* (ICSE '09). IEEE Computer Society, Washington, DC, USA, 199-209.

Lakshminarayanan, Sitaraman. (2010) "Interoperable Security Standards for Web Services", IT Professional, vol. 12, no. 5, pp. 42-47, Sep./Oct.

Lin, J.; Chen, J. (2009) "An Automated Mechanism for Secure Input Handling", Journal of Computers, v. 4, n. 9, pp. 837-844, Academy Publisher.

Loh, Y.; Yau, W.; Wong, C.; Ho, W. (2006) "Design and Implementation of an XML Firewall", In: Proceedings of the *2006 International Conference on Computational Intelligence and Security*, vol. 2, pp.1147-1150, 3-6 Nov.

Microsoft. (2010) "Microsoft Anti-Cross Site Scripting Library V 4.0". Disponível em: http://www.microsoft.com/downloads/en/details.aspx?FamilyID=f4cd231b-7e06-445b-bec7-343e5884e651.

Nordbotten, N.A. (2009) "XML and Web Services Security Standards", *Communications Surveys & Tutorials, IEEE*, vol.11, no.3, pp.4-21.

Owasp. (2010) "OWASP Top 10 Web Application Security Risks". Disponível em: http://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

Owasp. (2011) "OWASP Code Review Guide. Codereview-Input Validation". Disponível em: http://www.owasp.org/index.php/Codereview-Input_Validation

Saleem, M.Q.; Jaafar, J.; Hassan, M.F. (2010) "Model driven security frameworks for addressing security problems of Service Oriented Architecture", In: Proc. of the *2010 Int. Symp. Information Technology (ITSim),* vol.3, pp.1341-1346, 15-17 June.

Sans. (2009) "Os Maiores Riscos de Segurança Cibernética". Disponível em: http://www.sans.org/top-cyber-security-risks/pdfs/portuguese.pdf

Scholte, T.; Balzarotti , D.; Kirda, E. (2011) "Quo Vadis? A Study of the Evolution of Input Validation Vulnerabilities in Web Applications", In: Proc. of the Int. Conference on Financial Cryptography and Data Security '11, St. Lucia.

Shostack, A.; Stewart, A. (2008), The new school of information security. Boston: Addison-Wesley.

Sidharth, N.; Liu, J. (2007) "A Framework for Enhancing Web Services Security", In: Proc. of the *Computer Software and Applications Conference, 2007, COMPSAC 2007, 31st Annual International*, vol.1, no., pp.23-30, 24-27 July.

Siorg. (2009) "Sistema de Informações Organizacionais do Governo Federal – SIORG – Descrição do Web Service SIORG versão 2.0", Outubro. Disponível em: http://catalogo.governoeletronico.gov.br/pasta_servico/web-service-do-siorg

Soapui. (2010) "The Web Service, SOA and SOAP Testing Tool – soapUI". Disponível em: http://www.soapui.org/

Ssa. (2007) "WS-Security Wrapper", Sosnoski Software Associates Ltd. Disponível em: http://wsswrapper.sourceforge.net/

Sun, L.; Li, Y. (2008) "XML and web services security", In: Proceedings of the *12th International Conference on Computer Supported Cooperative Work in Design*, *CSCWD 2008*, pp.765-770, 16-18 April.

Tsipenyuk, K.; Chess, B.; McGraw, G. (2005) "Seven pernicious kingdoms: a taxonomy of software security errors", Security & Privacy, IEEE, v. 3, n. 6, p. 81-84.

Viega, J.; Epstein, J. (2006) "Why Applying Standards to Web Services Is Not Enough", IEEE security & privacy, New York, NY, v. 4, n. 4, p. 25-31.

RIST: Um *Middleware* para Replicação e Interoperabilidade de Sistemas Transacionais

Aldelir Fernando Luiz[1], Lau Cheuk Lung[1,2], Miguel Correia[3]

[1]Departamento de Automação e Sistemas - PGEAS - UFSC

[2]Departamento de Informática e Estatística - PPGCC - UFSC

[3]Instituto Superior Técnico / INESC-ID - Portugal

aldelir@das.ufsc.br, lau.lung@inf.ufsc.br, miguel.p.correia@ist.utl.pt

Resumo. *Neste trabalho apresentamos o middleware RIST, uma solução para replicação de bases de dados transacionais, que além de encapsular os requisitos necessários para a interoperabilidade em ambientes heterogêneos é tolerante a faltas Bizantinas e totalmente distribuído. O RIST omite toda a complexidade do cliente, de forma que é apresentada ao mesmo uma única base de dados consistente. Além disso, a aplicação cliente não necessita de modificações, já que o RIST segue as especificações do JDBC e pode operar sobre sistemas de gerência de base de dados off-the-shelf proprietários e/ou de código aberto.*

Abstract. *This paper presents the middleware RIST, a solution for transactional database replication, which encapsulates the requirements for interoperability in heterogeneous environments. RIST is designed for Byzantine Fault Tolerance of a distributed database system. RIST omits all complexities from clients, in a transparent way where the database can be handled in a centralized fashion. Furthermore, the client application do not need any modification because RIST is compatible to the JDBC and can operates over owner or open source off-the-shelf database management systems.*

1. Introdução

O presente trabalho se insere no desafio 5 **(Desenvolvimento tecnológico de qualidade: sistemas disponíveis, corretos, seguros, escaláveis, persistentes e ubíquos)**, pois visa à busca de novas soluções tecnológicas, de baixo custo, em termos de modelos e arquiteturas, para um sistema de processamento de transações que dê suporte ao desenvolvimento de aplicações com alta disponibilidade, corretas e seguras mesmo em presença de ataques de segurança e intrusões [Lung 2009][1]. Os sistemas de processamento de transações podem ser considerados o cerne dos sistemas de informação que dão suporte as diversas áreas do âmbito socioeconômico. Em todas as áreas de conhecimento onde sistemas de informação são usados como ferramentas de apoio ao negócio (sistemas de gerenciamento de dados, sistemas de telecomunicação, sistemas de controle industrial, comércio eletrônico e outros) existe uma grande dependência quanto à confiabilidade e tempestividade dos dados manipulados por estes sistemas. Assim, é importante que os sistemas de gerenciamento de dados (e transações) sejam capazes de prover meios que permitam sua operação de forma contínua e consistente, a despeito de fatores externos (p.ex. ataques de segurança) que possam interferir no correto funcionamento do sistema. Desta forma, a replicação de dados [Gray et al. 1996]

[1]http://www.gd2.ufam.edu.br/

torna-se peça fundamental dentro do contexto de sistemas de gerência de bases de dados (SGBD).

Apesar dos comentários tecidos por [Gray et al. 1996] a respeito dos perigos encontrados na replicação de dados, nos últimos anos muitos trabalhos foram propostos no intuito de desmistificar os conceitos previamente definidos. A razão para isto decorre do fato de que a replicação de dados é um meio bastante factível para melhorar o desempenho e prover maior disponibilidade em sistemas de processamento de transações (ex. bases de dados). Por outro lado, quando os dados são residentes em mais de um sítio (seja replicado ou distribuído), o processo de confirmação da transação se torna mais custoso, pois uma única fase de confirmação é insuficiente para assegurar as propriedades **ACID** (**A**tomicidade, **C**onsistência, **I**solamento e **D**urabilidade) da transação [Bernstein et al. 1987]. Um método bastante conhecido para prover a confirmação de transações distribuídas e/ou replicadas é o *Two-Phase Commit* [Skeen 1981], em que o processo de confirmação é feito em duas etapas, que são a preparação e a confirmação.

A replicação de dados representa ainda, um desafio em se tratando de integração, interoperabilidade e consistência para os sistemas de gerenciamento de bases de dados. Assim, novas abordagens para replicação de bases de dados têm sido propostas na literatura, e algumas destas têm sido construídas sobre abstrações de comunicação em grupo [Schiper and Raynal 1996], de forma a prover soluções de cunho mais prático. Além disso, os protocolos e algoritmos que compõem estas soluções têm sido capazes de obter um elevado grau de consistência [Pedone et al. 2003, Cecchet et al. 2004, Vandiver et al. 2007] sem despender grande sobrecarga adicional, como observado em soluções de replicação de bases de dados usuais [Bernstein et al. 1987, Gray et al. 1996]. Neste cenário, primitivas de comunicação em grupo [Schiper and Raynal 1996] permitem especificar e implementar soluções para replicação mais eficientes, além de simplificar a adaptação de técnicas e mecanismos de tolerância a faltas [Birman and van Renesse 2005], visando tornar a base de dados resistente a alguns tipos de falhas (ex. parada, omissão, arbitraria).

Uma questão pertinente em termos de replicação de dados discorre a respeito da robustez das aplicações. Neste sentido, a abordagem de replicação ativa (ou de máquina de estados) [Schneider 1990] tem se mostrado como um elemento essencial para prover este quesito. Todavia, quando aplicada no contexto de sistemas de processamento de transações, é requerido, *a priori*, que as réplicas acordem acerca de uma ordem total para a execução das transações, para que o critério de serialização seja atendido. Este problema é um fator impeditivo, por razões de desempenho, para que a abordagem de replicação ativa pura seja de viável adoção em sistemas de processamento de transações. Em face deste fato, o trabalho de [Pedone et al. 2003] concretizou a adaptação da técnica de replicação ativa para o contexto de sistemas transacionais, o que deu origem a técnica conhecida na literatura como **Replicação de Máquina de Estados para Bases de Dados** (*Database State Machine Replication - DSMR*).

Neste trabalho apresentamos o projeto e implementação do RIST, uma solução de *middleware* para replicação de bases de dados transacionais, que consiste na concretização da técnica DSMR sob o modelo de faltas Bizantinas. Além disso, o RIST encapsula os requisitos necessários para permitir a interoperabilidade entre diferentes bases de dados, em um único ambiente replicado (e heterogêneo). O RIST mantém o controle das transações totalmente distribuído e omite toda a complexidade em termos de programação do cliente, de forma que é apresentada ao mesmo uma única base de dados consistente. Ademais, a

aplicação cliente não necessita de modificações, já que o *middleware* segue as especificações do **JDBC** [Fisher et al. 2003], provendo assim, o acesso ao ambiente confiável de forma totalmente transparente para a aplicação. O RIST pode operar sobre sistemas de gerência de base de dados *off-the-shelf* proprietário e/ou de código aberto, desde que eles atendam a alguns requisitos necessários ao funcionamento do protocolo (p.ex. consistência serializável), e, sobretudo, sem implicar em modificações nos sistemas de gerência de base de dados.

2. Conceitos em Replicação de Dados

Conforme mencionado, a replicação de dados é um aspecto de fundamental importância no contexto de sistemas transacionais (SGBDs). Esta importância advém do fato de que a replicação permite manter a disponibilidade do sistema em caso de falhas, e, sobretudo, oferece maior escalabilidade em face ao fato de possibilitar a execução em paralelo de transações em diferentes réplicas. Entretanto, o principal problema encontrado na replicação de dados está relacionado ao processo de convergência das réplicas, isto é, a forma pela qual as réplicas convergem para o mesmo estado, após a confirmação de uma transação, mais precisamente na forma como as operações devem ser propagadas às réplicas do sistema.

Em [Gray et al. 1996] é apresentada uma taxonomia para protocolos de replicação de dados, realizada por meio de dois parâmetros: (*i*) a forma pela qual as alterações são propagadas às réplicas; (*ii*) quem tem o privilégio de efetuar a propagação. Na abordagem de replicação passiva [Budhiraja et al. 1993], uma das réplicas do grupo é eleita líder e passa a ser a responsável por receber as requisições dos clientes e enviar as respostas das operações, além de ser a encarregada de propagar as atualizações às demais réplicas. Por sua vez, a abordagem de replicação ativa [Schneider 1990] é uma técnica descentralizada onde todas as réplicas recebem, executam e respondem as requisições, tendo como único requisito o determinismo. Para tanto, as réplicas partem de um mesmo estado inicial e executam a mesma sequência de operações, assim, é possível produzir os mesmos resultados.

Além das técnicas de replicação mencionadas, outra abordagem conhecida como baseada em certificação [Kung and Robinson 1981] é bastante aceita como forma de replicação de dados, pois permite que as transações sejam executadas de forma otimista em uma única réplica, onde a convergência é realizada apenas no momento da confirmação da transação através de um procedimento de coordenação e certificação, que é executado a fim de convergir o sistema a um estado global consistente. Assim, quando a aplicação solicita a confirmação da transação, é necessário que as operações da transação, bem como RS e WS (*Read-Set* e *Write-Set*)[2] sejam propagados através de um protocolo de difusão com ordem total [Défago et al. 2004]. Como resultado, quando a mensagem de propagação é entregue, todas as réplicas obtém a mesma visão do histórico de transações entregues e confirmadas, e com isso podem detectar possíveis conflitos com transações concorrentes da mesma maneira. Este processo de coordenação e certificação é necessário para que possa ser estabelecida uma ordem global para a confirmação das transações concorrentes, e assim, manter o sistema consistente.

3. Trabalhos Relacionados

Uma verificação na literatura nos mostra alguns trabalhos que tratam de replicação de bases de dados em ambientes heterogêneos [Cecchet et al. 2004, Vandiver et al. 2007] e outros que tratam da especificação de modelos de replicação para ambientes de bases

[2]RS e WS são os conjuntos de itens de dados afetados pelas operações de leitura e escrita da transação, respectivamente

de dados tolerantes e faltas Bizantinas [Pedone et al. 2011, Garcia et al. 2011]. Dentre eles, os mais próximos de nossa proposta são o C-JDBC [Cecchet et al. 2004], o HRDB [Vandiver et al. 2007], o Byzantium [Garcia et al. 2011] e o BFT-*Deferred Update* [Pedone et al. 2011], apesar do C-JDBC [Cecchet et al. 2004] não tolerar faltas Bizantinas [Lamport et al. 1982]. O C-JDBC define uma solução de *middleware* completa para replicação de bases de dados heterogêneas, e permite a criação de *clusters* de bases de dados para o uso em aplicações baseadas no JDBC. Apesar da capacidade de operar sobre múltiplas plataformas de bases de dados, o C-JDBC não provê suporte a ambientes sujeitos a faltas Bizantinas, mas apenas de parada (*crash*). As únicas soluções para replicação de bases de dados aptas a operar em ambientes mais severos tais como os sujeitos a faltas Bizantinas são o HRDB, o Byzantium e o BFT-*Deferred Update*. Todavia, apesar de o HRDB prover os requisitos de interoperabilidade para ambientes heterogêneos, ele é limitado no sentido de que o protocolo depende de um elemento central (que considera confiável) para conduzir o sistema [Vandiver et al. 2007], e assim, este coordenador não pode falhar durante as janelas de vulnerabilidade do sistema.

Por outro lado, Byzantium [Garcia et al. 2011] é uma solução que assegura o critério de consistência *snapshot isolation* em ambientes sujeitos a faltas Bizantinas, o qual é mais fraco do que o modelo de consistência ideal para o modelo de transação, o serializável. Além do *snapshot isolation* não garantir a consistência forte dos dados, neste modelo as transações observam apenas o instante da base de dados correspondente ao da última transação confirmada no sistema. Ademais, no *snapshot isolation* as transações concorrentes são capazes de ser confirmadas apenas se elas não modificaram os mesmos itens de dados, o que não ocorre no modelo serializável, modelo assegurado em nosso *middleware*. Por fim, a solução BFT-*Deferred Update* [Pedone et al. 2011] especifica um protocolo de replicação para bases de dados, que é tolerante a faltas Bizantinas, mas assume apenas réplicas maliciosas e não clientes. Assim, o protocolo considera os clientes "confiáveis", simplificando, portanto, o processo de confirmação das transações. O protocolo de replicação especificado para nosso *middleware* apresenta algumas similaridades em relação ao BFT-*Deferred Update*, no sentido de que adotamos o mesmo modelo de propagação de operações, que é o clássico *deferred update* [Bernstein et al. 1987]. Todavia, é lícito mencionar que nosso protocolo tolera faltas Bizantinas oriundas tanto das réplicas como dos clientes.

4. Arquitetura RIST - Replicação e Interoperabilidade em Sistemas Transacionais

O RIST consiste em uma arquitetura baseada em *middleware*, que provê um suporte para replicação de bases de dados heterogêneas em ambientes sujeitos a faltas Bizantinas. A concretização do RIST se dá através da implementação de um protocolo de replicação tolerante a faltas Bizantinas, a partir das especificações da API JDBC™, API esta que permite o acesso virtual a qualquer tipo de dados em formato tabular [Fisher et al. 2003]. Assim, o RIST é um *middleware* baseado no JDBC, que permite a construção de um ambiente de banco de dados confiável e seguro, a partir do uso de bases de dados heterogêneas e, sobretudo, sem a necessidade de efetuar modificações na aplicação, nem tampouco nos sistemas de banco de dados.

O protocolo de base do RIST é construído a partir da abstração de difusão com ordem total, que é usado como principal mecanismo para prover a consistência e integridade sistema. Para tanto, a concretização desta abstração se dá através do algoritmo de consenso denominado PAXOS Bizantino [Zielinski 2004], com as otimizações previstas para prover

a terminação do consenso com menos passos de comunicação [Martin and Alvisi 2005]. O RIST é constituído de dois componentes básicos: um driver genérico JDBC, que é usado como parte da aplicação cliente; e um controlador descentralizado que é o responsável por prover as funcionalidades de segurança e tolerância a faltas da arquitetura. Para a aplicação cliente, o driver JDBC específico é substituído pelo driver JDBC genérico do RIST, que oferece a mesma interface e transparência de acesso ao sistema de banco de dados. Por outro lado, o controlador do RIST é um componente replicado em $3f + 1$ servidores, dos quais até $f \leq \lfloor \frac{n-1}{3} \rfloor$ podem falhar de maneira Bizantina. Este controlador atua como uma espécie de *Proxy* entre o driver genérico JDBC e os sistemas de bases de dados, e expõe aos clientes uma visão equivalente a um ambiente centralizado e, sobretudo, prezando pela consistência de dados.

O modelo de transações admitido no contexto do RIST segue as premissas tradicionais e clássicas de transações. Assim, uma transação consiste de uma sequência de operações/comandos, seguidos por um comando COMMIT ou ABORT. Os comandos podem ler, atualizar, incluir e excluir entradas no banco de dados. No contexto de uma transação, o sistema assegura que as operações são executadas na base de dados, exatamente na mesma ordem em que elas foram enviadas pelo cliente. Além disso, nosso modelo estipula que o cliente aguarda a resposta de uma operação enviada, antes de submeter a(s) próxima(s). Conforme preconiza o modelo tradicional de transações, para a obtenção da semântica de transações **ACID** [Bernstein et al. 1987] é requerido que o sistema de banco de dados implemente mecanismos de controle de concorrência e de recuperação. O sistema RIST assegura o critério de consistência serializável, isto é, a execução concorrente de um grupo de transações toma o mesmo efeito que a execução sequencial destas transações, em alguma ordem.

4.1. Descrição do Protocolo de Replicação

Soluções para replicação de bases de dados podem assegurar diferentes critérios de consistência de dados. Para a concretização e correção deste protocolo consideramos o critério ***one-copy serializability*** (1-SR) [Bernstein et al. 1987]. É importante ressaltar que, em razão da primitiva de difusão com ordem total ser baseada no algoritmo de consenso PAXOS Bizantino, e do modelo clássico de transações não ter terminação garantida em ambientes assíncronos, o modelo de interação requerido para assegurar o progresso do sistema (*liveness*) é o de sincronismo terminal (*eventually synchronous system*) [Dwork et al. 1988]. Ademais, como premissa para o funcionamento e obtenção das garantias do protocolo, é necessário que o sistema de banco de dados implemente o controle de concorrência baseado no bloqueio de duas fases (*two-phase locking*) e critério de consistência baseado em serialização (*serializable*).

Conforme ilustrado na Figura 1, o princípio de funcionamento do protocolo de replicação da arquitetura RIST é o seguinte:

Início: o início de uma transação ocorre a partir do momento em que o cliente, por intermédio do componente de *middleware* abre uma conexão no sistema de banco de dados. Neste momento, o subsistema do cliente envia de forma transparente uma mensagem **BEGIN** por meio da primitiva de difusão com ordem total ao conjunto de réplicas e, ao entregarem esta mensagem as réplicas definem de forma determinista qual delas será a líder desta transação, que terá a tarefa de executar as operações da transação. Ao final deste processo as réplicas enviam uma mensagem de confirmação ao cliente, para atestar que a transação foi iniciada com êxito (passos 1 e 2 da Figura 1).

Execução: quando o cliente recebe a confirmação de início da transação, a mensagem de confirmação traz consigo a indicação da réplica líder da transação. A partir deste momento até o final da transação, o cliente interage apenas com a réplica líder, isto é, ele envia as operações que fazem parte da transação apenas à réplica líder, caracterizando o que é conhecido como execução especulativa [Lampson 2006] ou execução otimista. As demais réplicas passam a ter parte no contexto da transação a partir do momento em que a aplicação solicita a confirmação para a transação (passos 3 a 6 da Figura 1).

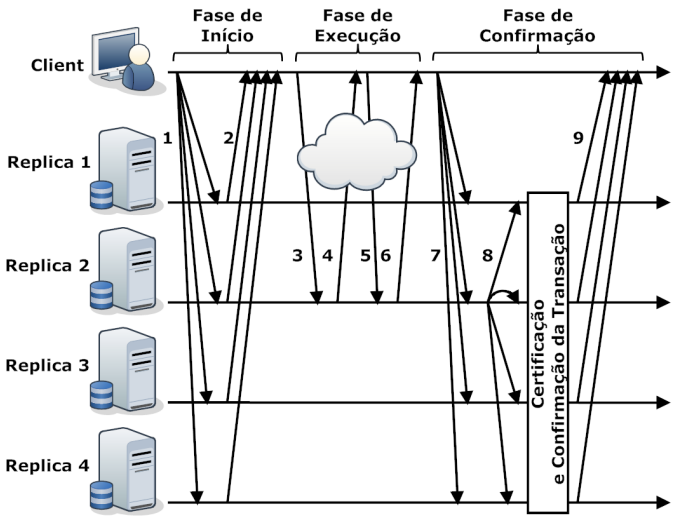


Figura 1. Funcionamento do protocolo

Confirmação: a confirmação da transação é ilustrada nos passos 7-9 da Figura 1. Ao solicitar a confirmação da transação, o componente de *middleware* da aplicação cliente envia a mensagem do pedido de confirmação por meio de difusão com ordem total, momento este onde também é realizada a propagação da transação para as demais réplicas do grupo. A mensagem do pedido de confirmação (que é assinada) traz consigo as operações executadas pela transação e o resumo criptográfico dos respectivos resultados (para fins de comparação). Assim, quando a réplica líder daquela transação recebe o pedido de confirmação, ela repassa através de difusão com ordem total o mesmo pedido de confirmação, adicionando os RS e WS produzidos pela transação, de modo a evitar que um cliente malicioso possa forjar um pedido de confirmação, ou ainda, que possa enviar pedidos de confirmação espúrios. A propagação através da difusão com ordem total assegura que todas as réplicas corretas recebam a confirmação da transação e a concretizem na mesma ordem, visando preservar o critério de serialização. A partir do recebimento da confirmação enviada pela réplica líder, as demais réplicas primeiramente verificam a consistência das mensagens enviadas (assinaturas, resumos, autenticação, etc.), e se é detectada qualquer anomalia no processo, a transação é descartada, ou do contrário, é executada. Ao iniciar a execução local da transação, as réplicas primeiramente adquirem os bloqueios de leitura e de escrita para os itens de dados afetados pela transação, e checam se RS e WS está em consonância com as operações enviadas, pois, no caso de um líder malicioso, o mesmo poderia repassar dados inválidos para as demais réplicas. Concluídas as verificações, as réplicas bloqueiam os dados e executam as operações sobre os mesmos. Após a execução, as réplicas submetem a transação a um teste de certificação, que irá verificar se não houve a violação das propriedades de serialização, isto é, se não houve conflitos da transação em questão, com outras transações concorrentes[3]. Em não havendo conflitos, as réplicas verificam se os resultados obtidos na

[3]Uma transação T_i é considerada concorrente em relação a T_j, se ela não precede T_j ($T_i \nrightarrow T_j$)

execução da transação localmente estão em consonância com aqueles enviados pela réplica líder ao cliente, cujos resumos (*hash*) dos resultados foram enviados na mensagem do pedido de confirmação, e em sendo as réplicas efetivam a transação, do contrário a transação é desfeita.

4.2. Componentes Arquiteturais

A ilustração da figura 2 apresenta os componentes do *middleware* RIST que se encontram em cada réplica do sistema. Todavia, cabe ressaltar que a arquitetura em questão é distribuída e os diversos sistemas de gerência de bases de dados (DBMS) da figura representam a capacidade do *middleware* em operar sobre diversas plataformas de SGBD. A arquitetura RIST é formada por uma série de componentes, que atuam em conjunto a fim de manter as propriedades, bem como prover as garantias propostas para o sistema. Esta seção descreve cada um dos componentes da arquitetura proposta.

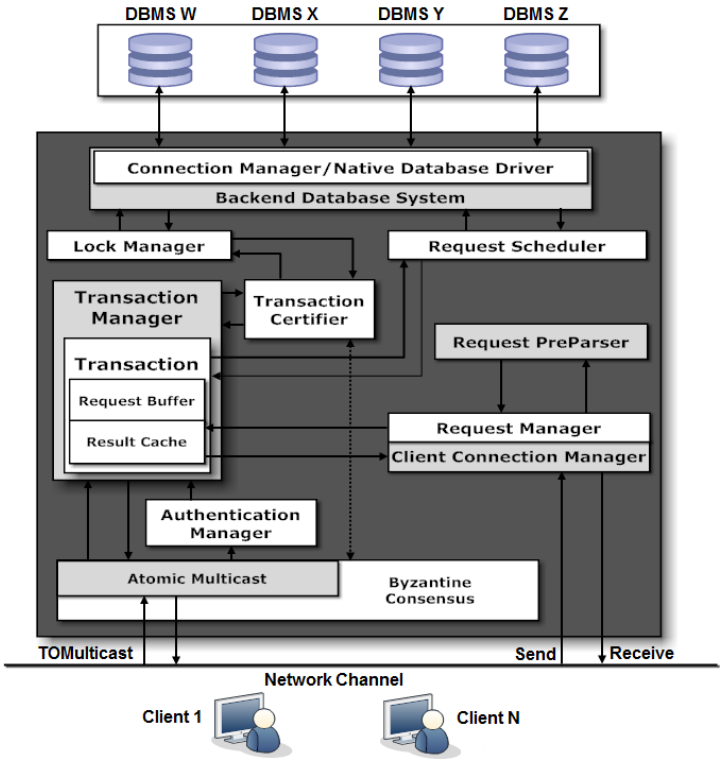


Figura 2. Componentes de uma réplica do RIST

4.2.1. Gerenciador de Conexões Cliente (*Client Connection Manager*)

O gerenciador de conexões cliente é o módulo responsável por estabelecer o canal de comunicação entre a réplica líder de uma dada transação, e o respectivo cliente emissor. Este componente opera diretamente sobre as abstrações de comunicação de médio nível, tal como *sockets* e *channels*. Assim, cabe ao módulo o recebimento das requisições/operações enviadas no contexto de uma transação e encaminhá-las ao módulo de gerenciamento de requisições para posterior tratamento e execução da operação.

4.2.2. Gerenciador de Requisições (*Request Manager*)

O módulo gerenciador de requisições é quem estabelece o vínculo da requisição ao contexto da transação a que ela pertence. A cada requisição recebida em uma transação, o módulo

primeiramente realiza a verificação acerca da sintaxe da requisição enviada para certificar se ela está bem formada, isto é, se ela segue os padrões de sintaxe definidos e aceitos para a execução pelo sistema. Uma vez que a requisição é aceita pela avaliação preliminar ela é direcionada para a transação a que pertence para posteriormente ser enviada para execução no sistema de banco de dados, pelo módulo escalonador.

4.2.3. Pré-Analisador de Requisições (*Request PreParser*)

Este componente é responsável por realizar a pré-avaliação da sintaxe do comando enviado pelo cliente, para processamento pelo sistema. Todavia, como o RIST preconiza os aspectos de interoperabilidade, as réplicas do sistema podem operar sobre sistemas de bancos de dados heterogêneos, isto é, de diferentes fornecedores. Diante disso, surgem alguns problemas que devem ser previstos e tratados pelo RIST. O primeiro deles decorre do fato dos sistemas de gerência de bancos de dados implementarem diferentes dialetos para os comandos de interação com a base de dados, no caso a linguagem SQL, apesar desta possuir uma padronização conhecida como SQL ANSI. Entretanto, estipular apenas o uso (ou o aceite) de comandos escritos segundo o padrão SQL ANSI torna-se algo restritivo. Assim, este problema é resolvido no RIST através do mapeamento do comando enviado pelo cliente para a sintaxe nativa do sistema de banco de dados da réplica em questão. E para prover melhor desempenho do processo de pré-análise de requisições, as réplicas mantém um *cache* de comandos pré-avaliados e já traduzidos para execução, para evitar o reprocessamento dos comandos usados com maior frequência. A manutenção deste *cache* é realizada de acordo com a tradicional política de LRU (*least recently used*), onde as informações são removidas do *cache* de acordo com sua obsolescência.

4.2.4. Gerenciador de Transações (*Transaction Manager*)

Quando uma mensagem de controle de transações é entregue ao *middleware* por meio do subsistema de difusão com ordem total, este encaminha a mensagem ao módulo de gerenciamento de transações que de acordo com o contexto recebido executará a ação apropriada. As mensagens de controle do protocolo de replicação do RIST são de três tipos: (*i*) mensagem de início de transação; (*ii*) mensagem de pedido de confirmação, enviado pelo cliente; (*iii*) mensagem de confirmação, enviada pela réplica líder da transação. Assim, quando o protocolo entrega alguma destas mensagens o módulo de gerenciamento de transações realiza a criação de uma transação, bem como a alteração do estado da transação para a qual a mensagem é destinada, de acordo com o tipo da mensagem. Note que, devido ao uso do protocolo de difusão com ordem total, as ações efetuadas pelo módulo gerenciador de transações ocorrem na mesma ordem em todas as réplicas do grupo. Esta é uma premissa do sistema para que o protocolo do RIST possa prover as garantias de consistência das transações perante a ocorrência de faltas Bizantinas, e assim manter o critério de serialização.

4.2.5. Certificador de Transações (*Transaction Certifier*)

O módulo de certificação é um dos componentes mais importantes do RIST. Este módulo é o responsável por realizar o teste de certificação das transações que são submetidas para confirmação. O propósito deste módulo é determinar situações de conflito entre transações concorrentes, em face a execução otimista/especulativa realizada pelo protocolo, para prover maior escalabilidade ao sistema. Deste modo, uma transação é apta a ser confirmada se, além das verificações preliminares de consistência ela passa no teste de certificação.

O teste de certificação que é baseado na certificação de [Kung and Robinson 1981] toma como subsídios as marcações efetuadas nas transações, que ocorrem quando da entrega das mensagens *REQ-COMMIT* e *COMMIT*. Mais precisamente, uma transação T_i está sujeita à certificação se é verificada a existência de alguma transação concorrente T_j, após a entrega do pedido de confirmação para T_i. Uma transação T_j é dita corrente em relação à T_i se ela não a precede ($T_j \nrightarrow T_i$). Para facilitar o entendimento da atuação do módulo de certificação, a figura 3 ilustra a execução de quatro transações concorrentes (T_1, T_2, T_3, T_4) através do protocolo.

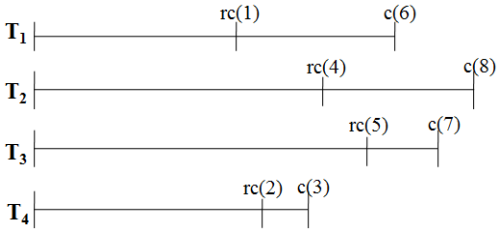


Figura 3. Execução de transações

Os eventos **rc**(n) e **c**(n) representam o momento em que ocorreu a entrega das mensagens de pedido de confirmação (*REQ-COMMIT*) e de confirmação (*COMMIT*), respectivamente. O valor n é o *timestamp* relativo à ordem de entrega das respectivas mensagens, fornecido pelo módulo de difusão com ordem total. Além da execução das transações concorrentes a figura 3 nos permite observar quais transações estão sujeitas a situações de conflitos entre si. Neste caso, em relação à transação T_4 não há transações conflitantes, pois nenhuma transação foi confirmada durante o intervalo entre o pedido de confirmação e a confirmação de fato (entre os *timestamps* 2 e 3). Por outro lado, ao analisar a transação T_1 é possível verificar que a transação T_4 é concorrente a ela, pois T_4 foi confirmada no *timestamp* 3, o que ocorreu durante o intervalo entre o pedido e a confirmação de T_1 (*timestamps* 1 e 6). Da mesma forma, T_1 é concorrente com T_3, e T_3 e T_1 são concorrentes com T_2, pelas mesmas razões.

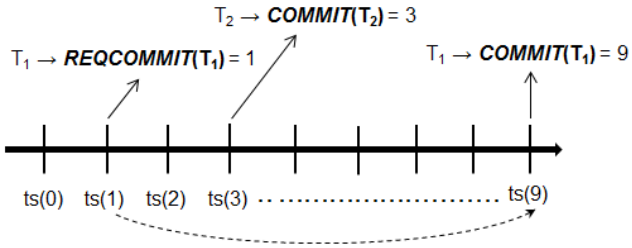


Figura 4. Relação de precedência entre transações

A figura 4 ilustra de forma mais simplificada a relação de precedência de transações, bem como quais transações devem ser consideradas pelo teste de certificação quando uma determinada transação pede confirmação. No exemplo é ilustrada a execução de duas transações identificadas por T_1 e T_2. A relação de precedência é verificada da seguinte forma: no caso de T_1, todas as transações que a precedem são aquelas que tiveram sua confirmação efetuada em algum *timestamp* anterior ao pedido de confirmação de T_1, neste caso, anterior a $ts(1)$. As transações precedentes, isto é, as que já foram confirmadas antes do pedido de confirmação para a transação em questão não são consideradas pelo teste de certificação, visto que elas já cumprem os critérios de serialização. Desta forma, apenas as transações potencialmente conflitantes, isto é, as que foram confirmadas após o pedido de confirmação da transação em questão são consideradas. No exemplo todas as transações

confirmadas a partir do *timestamp* $ts(1)$ até o *timestamp* $ts(9)$ são consideradas concorrentes com T_1 (neste caso, apenas T_2) e devem ser consideradas no teste de certificação para possibilitar a confirmação de T_1. Em suma, o teste de certificação efetuado por este módulo é que determinará se a confirmação da transação não implicará na violação do critério de serialização [Bernstein et al. 1987], de modo a preservar a integridade e consistência da base de dados. Uma transação que não passa no teste de certificação é cancelada pelo protocolo.

4.2.6. Gerenciador de Bloqueios (*Lock Manager*)

O módulo gerenciador de bloqueios é o mecanismo responsável por realizar a aquisição dos bloqueios de leitura e de escrita sobre os itens de dados afetados pela transação e que são entregues junto à mensagem de confirmação da transação. Assim, o gerenciador de bloqueios atua como uma espécie de escalonador de requisições de bloqueio sobre os itens de dados. Cabe ressaltar que como os pedidos de confirmação e a confirmação são propagados através da primitiva de difusão com ordem total, isto implica que eles ocorrem na mesma ordem em todas as réplicas corretas. E seguindo esta premissa, os pedidos de bloqueio também ocorrem na mesma ordem, de acordo com a confirmação das transações do sistema.

4.2.7. Escalonador de Requisições (*Request Scheduler*)

Quando uma nova requisição é recebida, após a execução de todo o processo de validação e associação com a respectiva transação, ela é disponibilizada ao escalonador para execução pelo sistema de banco de dados local. O escalonador é responsável por efetuar o controle de concorrência da réplica (em conjunto com o sistema de banco de dados local), bem como coordenar a execução das requisições de uma transação, de acordo com a ordem de envio das mesmas. Todas as operações executadas são síncronas com o respectivo cliente, isto é, o escalonador aguarda o recebimento da resposta da réplica líder para a requisição enviada, antes de retornar o resultado ao cliente. Neste caso, se a resposta para uma requisição não é recebida dentro de um intervalo de tempo aceitável, o escalonador retorna uma exceção como resposta à requisição. Neste caso, o cliente pode reenviar o comando para execução, se assim desejar.

4.2.8. Gerenciador de Autenticação (*Authentication Manager*)

O módulo gerenciador de autenticação estabelece o mapeamento entre o *login* e senha fornecidos pela aplicação cliente, e o *login* e senha a ser usado em cada uma das réplicas da base de dados. Assim, o *login* e senha enviados pela aplicação não são, de fato, os dados de acesso à base de dados local, mas sim uma ponte de acesso ao sistema. A funcionalidade incrementa maior segurança ao ambiente, pois os dados de acesso à base são exclusivos para cada uma das réplicas. Além disso, é permitido que cada uma das réplicas tenha diferentes logins de senhas para acesso a uma mesma base de dados.

5. Aspectos de Heterogeneidade e Não-Determinismo

Nesta seção apresentamos duas questões de fundamental importância, que decorrem do uso de réplicas heterogêneas. De acordo com o modelo de replicação de máquina de estados [Schneider 1990], é imperioso que cada comando enviado pelo cliente no contexto de uma transação seja processado de forma idêntica por cada uma das réplicas, de modo a preservar o determinismo de réplica.

Conforme mencionado na seção 4.2.3., o primeiro problema em face da heterogeneidade de réplicas de aplicação surge devido à implementação de diferentes dialetos da linguagem SQL por cada uma das réplicas de banco de dados, o que resulta em incompatibilidade, apesar desta linguagem ser padronizada pelo ANSI. Assim, é necessária a verificação preliminar e posterior tradução do comando enviado a uma determinada réplica para que o sistema possa assegurar a consistência das operações executadas em cada uma delas. Além disso, devido à algumas peculiaridades dos SGBDs tais como o tamanho dos blocos de dados, a disposição dos registros e outros, a execução dos comandos já em sintaxe correta podem retornar resultados em ordem e disposição distintas. E como a confirmação da transação requer a comparação dos resumos criptográficos dos resultados das operações, este aspecto de heterogeneidade mencionado torna-se um problema impeditivo para que a comparação dos resultados seja efetuada de forma plena nas réplicas.

Estes problemas são resolvidos no RIST através da implementação de dois mecanismos. O primeiro é a anexação de uma cláusula `ORDER BY` apropriada aos comandos SQL que já passaram pelas transformações sintáticas do módulo de pré-análise de requisições. Esta cláusula é peça fundamental para que a disposição e a ordem dos registros contidos no resultado de um comando seja a mesma para um comando em cada uma das réplicas. Além disso, para tornar os resultados das réplicas heterogêneas idênticos[4], é necessário efetuar um mapeamento sobre cada um dos resultados para uma representação uniforme. As réplicas do RIST efetuam o mapeamento dos resultados obtidos de um comando para uma cadeia de *bytes* de tamanho fixo e de representação uniforme, independente da representação específica da réplica de banco de dados. Este mapeamento ocorre após o processamento do comando e antes do envio do resultado ao cliente, para assegurar que este receba apenas resultados na representação válida do sistema.

6. Implementação, Avaliação e Resultados

Conforme mencionado na seção 4., o RIST consiste na concretização de um driver JDBC [Fisher et al. 2003], e desta forma, a implementação do protótipo do *middleware* foi realizada na linguagem Java, seguindo as especificações do JDBC. A primitiva de difusão com ordem total usada no RIST é oriunda da implementação do algoritmo PAXOS Bizantino [Zielinski 2004] que é parte do projeto *BFT-SMaRt* (*http://code.google.com/p/bft-smart*). Optou-se por usar esta implementação do PAXOS, em razão dela contemplar uma série de otimizações tais como realização do consenso em menos passos de comunicação em execuções favoráveis; acordo realizado por meio de resumos criptográficos de mensagens ao invés de fazê-lo por meio da mensagem completa; acordo por lote de mensagens ao invés de fazê-lo mensagem a mensagem.

A concretização dos canais de comunicação com os mecanismos requeridos para suportar as garantias do protocolo de replicação e execução especulativa de transações foram implementadas com o uso do *framework* Netty (*http://www.jboss.org/netty*), com vistas às otimizações e facilidades providas por este *framework*, no que compete ao desenvolvimento de aplicações baseada em passagem de mensagens. Para permitir o correto funcionamento do protocolo, bem como das comparações das respostas das réplicas durante o processo de confirmação, os resultados das operações das transações são mapeados para uma cadeia de *bytes* de tamanho fixo e posteriormente serializados para envio ao cliente. Para a serialização optamos por usar a *interface* java.io.Externalizable devido a ela apresentar maior eficiência em relação à *interface* padrão java.io.Serializable. Esta eficiência

[4]O que não ocorre naturalmente devido a representação interna de tipos de dados em cada uma das réplicas.

refletiu inclusive, em um decremento significativo no tamanho das mensagens que contém os resultados das operações.

Como é sabido que a avaliação de sistemas de computação distribuída pode ser realizada de forma analítica e/ou empírica [Jain 1991], decidimos por realizar a avaliação de nosso *middleware* nestes dois moldes. Os resultados para estas avaliações são reportados nas próximas subseções.

6.1. Avaliação Analítica

A primeira avaliação realizada em nosso *middleware* se dá em termos das complexidades algorítmicas e de algumas propriedades observadas nos trabalhos relacionados. Para esta avaliação são considerados apenas os protocolos do RIST, o HRDB [Vandiver et al. 2007], o Byzantium [Garcia et al. 2011] e o BFT-*Deferred Update* (BFT-DU) por serem os únicos a se basearem nos princípios de nossa proposta (p.ex. faltas Bizantinas). A tabela 1 descreve os resultados observados em cada um dos protocolos mencionados. Esta avaliação considerou apenas os protocolos de confirmação das soluções em razão desta ser o parte que despende o maior custo e complexidade no processamento da transação.

Tabela 1. Avaliação analítica dos protocolos de confirmação de transações

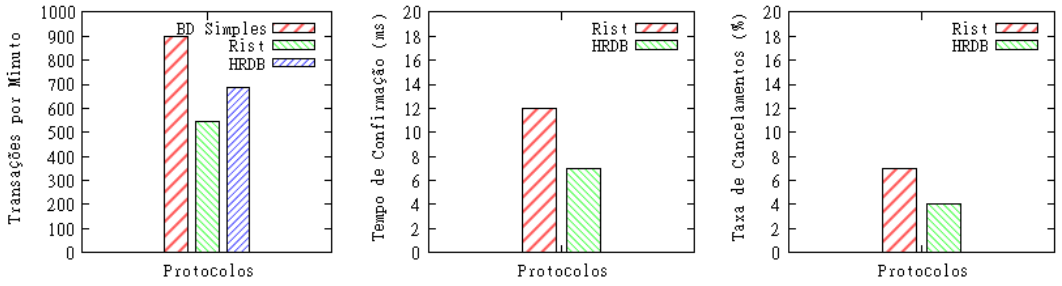
CARACTERÍSTICAS E PROPRIEDADES	PROTOCOLOS			
	HRDB	**Byzantium**	**BFT-DU**	**RIST**
Número de Réplicas	$2f + 1 + controlador$	$3f + 1$	$3f + 1$	$3f + 1$
Latência de Comunicação	4	(*TOMCast*) + 1	(*TOMCast*) + 1	2(*TOMCast*) + 1
Complexidade de Mensagens	$O(n)$	$O(n^2)$	$O(n^2)$	$O(n^2)$
Consistência	Serialização (Forte)	*Snapshot* (Fraco)	Serialização (Forte)	Serialização (Forte)
Controle Centralizado	Sim	Não	Não	Não
Entidades Bizantinas	Clientes e Réplicas	Clientes e Réplicas	Apenas Réplicas	Clientes e Réplicas

A tabela 1 nos mostra que em termos de número de réplicas o HRDB é a melhor solução, já que o mesmo mantém o controle "centralizado". O HRDB não requer um acordo bizantino tal como ocorre nos demais protocolos devido a difusão com ordem total. Em termos de latência o HRDB ainda é o melhor, o que decorre do número reduzido de mensagens do protocolo também em razão do controle centralizado. O Byzantium e o BFT-DU são equivalentes e mais eficientes do que o RIST. Isso ocorre devido ao RIST usar duas difusões com ordem total (*TOMCast - Total Order MultiCast*) para o processo a confirmação, a despeito de uma difusão usada pelos demais. Contudo, o uso das duas difusões é necessário para que seja possível obter o critério de consistência serializável em face a clientes e servidores Bizantinos, uma característica não atendida em plenitude pelas demais soluções.

Em se tratando da complexidade de mensagens, todos os protocolos das soluções em que a confirmação ocorre de forma não centralizada são equivalentes e obviamente menos eficientes em relação ao HRDB. Novamente, isto decorre do uso do coordenador centralizado e da ausência do acordo entre as réplicas. O resultado para as demais soluções em termos de complexidade é oriundo do uso de protocolos de difusão com ordem total de nível inferior equivalentes (p.ex. Paxos Bizantino e PBFT). Embora os resultados gerais apontem para a perda, em termos de eficiência de nossa solução, é lícito citar que o RIST é a única solução que provê a consistência ideal (serializável) em ambientes sujeitos a faltas Bizantinas de maneira totalmente distribuída, e sem restrições quanto ao comportamento Bizantino por parte dos processos (clientes e réplicas). Deste modo, isto pode ser considerado um grande avanço em relação aos demais protocolos avaliados, o que justifica o uso do RIST a despeito da eficiência verificada.

6.2. Avaliação Experimental

A outra avaliação realizada acerca de nossa proposta e trabalhos relacionados se dá de maneira empírica, por meio da experimentação dos protocolos avaliados em um ambiente real. Todavia, esta avaliação foi realizada apenas considerando os protocolos dos *middlewares* RIST e HRDB, devido a não termos tido acesso a implementação dos demais protocolos. A avaliação foi realizada por meio do *benchmark* TPC-C, onde utilizamos uma implementação própria do *benchmark* a partir do SQL padrão, para possibilitar a execução do mesmo em qualquer sistema de gerência de base de dados que dê suporte ao SQL padrão. Para a execução do TPC-C, as bases de dados foram carregada com 10 *warehouses* e 10 *distritos* para cada uma, visando induzir maior concorrência. Os testes foram executados a partir de 20 clientes com 200 ms de intervalo entre a submissão de transações, tendo como objetivo demonstrar a aplicabilidade dos sistemas, bem como o desempenho relativo de cada um deles.



(a) Throughput. (b) Tempo de Confirmação. (c) Cancelamentos (%).

Figura 5. Avaliação experimental das soluções de *middleware*.

O ambiente no qual o *benchmark* foi realizado contou com um conjunto de 4 máquinas Itautec InfoWay ST4262, todas com a mesma composição de *hardware* e de *software*, interligadas por meio de uma rede local (LAN), e tendo como configurações: 1 microprocessador Intel®Core™2 Duo E8400 2.67GHz, 4GB RAM e uma interface Ethernet Gigabit Intel 82567LM-3. Como sistema operacional adotamos o Debian GNU/Linux 6.0, *kernel* 2.6.32-5-686 #1 SMP. Os sistemas de gerência de bases de dados usados nos experimentos foram o Informix Innovator-C Edition 11.70-UC1 e o MySQL 5.5.8, sendo duas (2) instâncias de cada um para totalizar quatro (4) réplicas (ex. $3f + 1$ réplicas para suportar 1 falta Bizantina). No experimento da figura 5(a) o valor para a **base de dados simples** é oriundo da média verificada para os resultados de uma réplica com o Informix e uma com o MySQL.

A figura 5 apresenta os resultados do TPC-C, os quais foram obtidos a partir do média de 25 execuções para cada um dos experimentos. Conforme podemos observar na figura 5(a), nosso protocolo é o que apresenta menor vazão em relação aos demais. O resultado reportado para uma única base de dados é apenas um referencial, visto que protocolos de replicação têm um custo adicional e por esta razão são menos escaláveis que bases de dados não replicadas. Em nosso caso, isso decorre do uso das duas difusões com ordem total pelo processo de confirmação do RIST. O HRDB apresenta maior vazão em virtude de ele manter o controle centralizado e não requerer o uso de difusões com ordem total e outras verificações adicionais decorrentes do acordo Bizantino (p. ex. assinaturas). Todavia, como o HRDB depende de um coordenador centralizado para conduzi-lo, isso implica na existência de um ponto de falha que não é encontrado no RIST. Além disso, como adotamos a técnica de execução otimista para as transações, o processo de confirmação do RIST despende um tempo maior em relação ao HRDB, conforme observado na figura 5(b). Isto

já era esperado devido ao modelo de propagação adotado por nosso protocolo, pois, no momento do pedido de confirmação toda a transação é propagada às demais réplicas para fins de convergência do estado das mesmas. Como no HRDB a propagação é realizada operação a operação, quando a confirmação da transação é efetuada, a réplica primária apenas compara os resultados obtidos nela e nas réplicas secundárias, para cada operação, não havendo nenhuma propagação no processo. De outro lado, há um aspecto inerente à abordagem especulativa/otimista de transações, que é o cancelamento de parcela de transações, quando em situações de concorrência se verifica a existência de conflitos de leitura e de escrita [Kung and Robinson 1981, Lampson 2006] sobre um mesmo item de dados. Cabe ressaltar que este cancelamento é necessário para que não seja violado o critério de consistência de serialização das transações que se encontram em confirmação sobre as já confirmadas. Os resultados nos mostram que o HRDB proporciona parcela menor de cancelamentos devido ao protocolo atrasar a execução de transações em que se verifica a situação de conflitos, ao contrário do que ocorre na abordagem otimista/especulativa usada no RIST.

Cabe salientar que nos resultados das figuras 5(a) e 5(b) os valores para a **base de dados simples** não foram reportados por dois motivos, o primeiro decorre do tempo de confirmação ser ínfimo em bases de dados não replicadas, e segundo, porquê o cancelamento de parcela das transações é uma implicação do uso de algumas técnicas empregadas nos protocolos de replicação, e portanto, não são verificados em bases de dados não replicadas. No entanto, apesar dos resultados apontarem para a aparente ineficiência de nosso *middleware*, cremos que há uma competitividade do mesmo em relação aos demais, principalmente devido as garantias que ele oferece. Além disso, há uma série de vantagens e desvantagens que levam a melhores ou piores resultados em termos de indicadores considerados e, portanto, cabe ao projetista do sistema decidir qual é a solução mais adequado para uma aplicação específica.

7. Conclusões

Este trabalho apresentou a proposta de um *middleware* para replicação em sistemas transacionais tolerante a faltas Bizantinas e como prova de conceito analisou de forma analítica e empírica a solução proposta e as demais existentes que se baseiam no mesmo princípio de funcionamento. Por meio destas avaliações, verificamos que a proposta é factível de uso prático, além de ser uma solução flexível para replicação de bases de dados, que visa prover escalabilidade, disponibilidade, heterogeneidade e tolerância a faltas. Além disso, devido ao uso das interfaces do JDBC, o RIST opera independente do sistema de gerência de base de dados e, sobretudo, é transparente para aplicações baseadas nesta interface.

Um aspecto importante e que deve ser relevado, é que a despeito do desempenho verificado e de algumas vantagens e custos mensurados para algumas das outras soluções em determinados aspectos, nenhuma delas contempla uma solução para replicação em sistemas transacionais completa, já que elas são limitadas em termos de consistência de dados e capacidade de tolerar faltas por parte das entidades do sistema. Diante disso, se verifica que nossa solução é a única que provê o critério de consistência ideal para sistemas de transações replicadas e, sobretudo, suporta faltas Bizantinas oriundas de clientes e réplicas a partir de um protocolo totalmente distribuído.

Referências

Bernstein, P. A., Hadzilacos, V., and Goodman, N. (1987). *Concurrency Control and Recovery in Database Systems*. Addison-Wesley.

Birman, K. and van Renesse, R. (2005). *Reliable Distributed Systems: Technologies, Web Services, and Applications*. Springer.

Budhiraja, N., Marzullo, K., Schneider, F. B., and Toueg, S. (1993). The primary-backup approach. *Distributed systems (2nd Ed.)*, pages 199–216.

Cecchet, E., Julie, M., and Zwaenepoel, W. (2004). C-JDBC: Flexible Database Clustering Middleware. In *USENIX Annual Technical Conference*.

Défago, X., Schiper, A., and Urbán, P. (2004). Total order broadcast and multicast algorithms: Taxonomy and survey. *ACM Computing Surveys*, 36(4):372–421.

Dwork, C., Lynch, N. A., and Stockmeyer, L. (1988). Consensus in the presence of partial synchrony. *Journal of ACM*, 35(2):288–322.

Fisher, M., Ellis, J., and Bruce, J. (2003). *JDBC^{TM}API Tutorial and Reference, Third Edition*. Addison Wesley, 3 edition.

Garcia, R., Rodrigues, R., and Preguiça, N. (2011). Efficient middleware for byzantine fault-tolerant database replication. In *Proceedings of the 6th European conference on Computer Systems - EuroSys'11*. ACM.

Gray, J., Helland, P., O'Neil, P., and Shasha, D. (1996). The dangers of replication and a solution. In *SIGMOD '96: Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pages 173–182, New York, NY, USA. ACM.

Jain, R. K. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. John Wiley & Sons.

Kung, H. T. and Robinson, J. T. (1981). On optimistic methods for concurrency control. *ACM Transactions on Database Systems*, 6:213–226.

Lamport, L., Shostak, R., and Pease, M. (1982). The Byzantine generals problem. *ACM Transactions on Programing Languages and Systems*, 4(3):382–401.

Lampson, B. W. (2006). Lazy and speculative execution in computer systems. In Shvartsman, A. A., editor, *OPODIS'06: Proceedings of the 10th International Conference on Principles of Distributed Systems*, volume 4305 of *LNCS*, pages 1–2. Springer-Verlag.

Lung, L. C. (2009). Tolerância a intrusões em sistemas de computação distribuída. In *II Seminário sobre Grandes Desafios da Computação no Brasil*, pages 13–16.

Martin, J.-P. and Alvisi, L. (2005). Fast Byzantine consensus. In *Proceedings of the Dependable Systems and Networks - DSN 2005*, pages 402–411.

Pedone, F., Guerraoui, R., and Schiper, A. (2003). The database state machine approach. *Distributed and Parallel Databases*, 14(1):71–98.

Pedone, F., Schiper, N., and Armendáriz-Iñigo, J. (2011). Byzantine fault-tolerant deferred update replication. In *Proceedings of the 5th Latin-American Symposium on Dependable Computing - LADC'11*. SBC.

Schiper, A. and Raynal, M. (1996). From group communication to transactions in distributed systems. *Communications of the ACM*, 39:84–87.

Schneider, F. B. (1990). Implementing fault-tolerant service using the state machine approach: A tutorial. *ACM Computing Surveys*, 22(4):299–319.

Skeen, D. (1981). Nonblocking commit protocols. In *SIGMOD '81: Proceedings of the 1981 ACM SIGMOD international conference on Management of data*, pages 133–142, New York, NY, USA. ACM.

Vandiver, B., Balakrishnan, H., Liskov, B., and Madden, S. (2007). Tolerating byzantine faults in transaction processing systems using commit barrier scheduling. In *SOSP'07: Proceedings of 21st ACM Symposium on Operating Systems Principles*.

Zielinski, P. (2004). Paxos at war. Technical Report UCAM-CL-TR-593, University of Cambridge Computer Laboratory, Cambridge, UK.

Um Mecanismo de Segurança com Adaptação Dinâmica em Tempo de Execução para Dispositivos Móveis

Alexandre Correia Cirqueira*, Rossana M. C. Andrade, Miguel F. de Castro

Universidade Federal do Ceará (UFC)
Mestrado e Doutorado em Ciência da Computação (MDCC)
GREat – Grupo de Redes de Computadores, Engenharia de Software e Sistemas
Caixa Postal 6001 – 60455-760 – Fortaleza – CE

alexandrecorreia@great.ufc.br, {rossana, miguel}@ufc.br

Abstract. *The increasing use of mobile devices and their applications exchanging information in diverse environments highlights the importance of ensuring data security. Additionally, the trend in the use of sustainable practices advocated by green computing imposes the need for designing applications flexible enough to deal with them. This paper then proposes an adaptive security mechanism, focusing on confidentiality of information exchanged among applications for mobile devices, able to adapt their security degree according to the context and provide the efficient use of resources.*

Resumo. *A crescente utilização de dispositivos móveis e suas aplicações, trafegando informações nos mais variados ambientes, evidenciam a importância da garantia de segurança destes dados. Adicionalmente, a tendência no uso de práticas sustentáveis, defendidas pela computação verde, impõe a necessidade de concepção de aplicações flexíveis o suficiente para lidar com estes desafios. Este trabalho propõe então um mecanismo de segurança adaptativa, com foco na confidencialidade de informações trafegadas por aplicações de dispositivos móveis, capaz de adaptar seu grau de segurança de acordo com o contexto e proporcionar o emprego eficiente dos recursos.*

1. Introdução

A possibilidade de captura de informações trafegadas por aplicações de dispositivos móveis (DMs), com um dispositivo receptor preparado para este fim, aponta para a necessidade de um tratamento eficaz de confidencialidade destas informações [Bringel 2005]. Um mecanismo de segurança que preserva a confidencialidade é a criptografia, a qual atua de forma a tornar os dados compreensíveis apenas para quem possuir sua chave de decifragem [BISHOP, 2003]. Geralmente é empregada em protocolos que oferecem segurança na camada de enlace. Entretanto, como os DMs passeiam por ambientes diversos, nem sempre estes protocolos estão disponíveis ou são adequados para determinadas aplicações que requerem maior segurança. Assim, para garantir o grau de confidencialidade destas informações, [Bringel 2005] sugere que a criptografia deve ser implantada na camada de aplicação e se adequar aos ambientes de uso, pois desta forma, o agente malicioso em posse de algum dado trafegado não encontrará facilidade na tentativa de acesso à informação ali contida.

* Bolsista de mestrado (MDCC/UFC) financiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior CAPES/PROPAG

Por um lado, a manutenção do grau de confidencialidade adequado de uma aplicação, em seus diversos ambientes de uso, possibilita maior segurança em ambientes críticos [Carvalho 2008] e a avaliação de desempenho de algoritmos criptográficos possibilita a construção de mecanismos de segurança eficientes. Por outro lado, o uso eficiente de recursos é um dos princípios fundamentais da computação verde, definida como o estudo e a prática da utilização eficiente dos recursos computacionais, primando pela redução do uso de materiais maléficos à natureza, maximizando a eficiência energética e a vida útil do produto, promovendo sua reciclagem e biodegradabilidade e evitando o aumento dos resíduos de fabricação [Wilbanks 2008].

Sendo assim, neste trabalho defendemos que uma adaptação dinâmica em tempo de execução, alocando adequadamente um algoritmo criptográfico que proporcione o grau de confidencialidade requerido por uma aplicação em cada ambiente de uso, pode garantir a manutenção da segurança e o emprego eficiente dos recursos do DM.

Considerando a construção das aplicações para DMs com a adoção de confidencialidade na forma tradicional, onde um único algoritmo criptográfico é usado de forma estática, o fator de avaliação para sua escolha é um grande desafio, pois este critério de decisão, normalmente baseado no desempenho, é proporcionalmente desfavorável à garantia de confidencialidade proporcionada [Hamad 2009]. Pesquisas de novas técnicas para lidar com o *trade-off* entre o uso de mecanismos de segurança e o consumo de recursos computacionais são destacadas como essenciais para a computação [Maliki 2010], [Taddeo 2010].

Especificamente, o *trade-off* entre prover confidencialidade com o uso adequado de criptografia e diminuir o consumo de recursos dos DMs é a motivação principal deste trabalho. O desenvolvimento de uma solução para engenheiros de software com o intuito de facilitar a adoção da adaptação dinâmica da confidencialidade, em tempo de execução e baseado em informações de contexto, é outra motivação.

A constatação de que uma aplicação nem sempre requer o maior grau de confidencialidade desperta a busca por uma adaptação de acordo com suas necessidades. Para a identificação destas necessidades e indicação da melhor adaptação é necessária a exploração de informações de contexto, o qual é definido como qualquer informação que pode ser usada para caracterizar uma pessoa, lugar ou objeto, que seja relevante na interação do usuário com a aplicação, incluindo ambos [Dey 2001]. A análise do contexto propicia a captura de informações que possibilitam a adaptação eficiente da criptografia, exemplificando, pode-se ter uma aplicação utilizando uma rede classificada como segura, onde ela adapta a criptografia mudando o algoritmo por um que consome menos recursos, poupando o hardware e aumentando sua vida útil.

Aplicações que originalmente requeriam determinado grau de segurança, mas que são construídas com um grau menor (por decisões de negócio), poderiam usar a adaptação dinâmica dos algoritmos de criptografia de acordo com o contexto e restringir a sobrecarga do hardware, ocasionada pelos algoritmos complexos, apenas aos ambientes que necessitam do maior grau de confidencialidade.

Este trabalho evidencia a necessidade de se poupar recursos do DM quando determinadas aplicações permitirem a adaptação de algoritmos criptográficos menos

complexos e oferece alto grau de confidencialidade ao adaptar os algoritmos mais complexos em ambientes de alto risco para aplicações não protegidas adequadamente.

Esta manutenção no nível de confidencialidade aumenta a segurança no tráfego das informações dos usuários de DMs, cada vez mais necessária devido à absorção desta tecnologia no cotidiano dos cidadãos comuns, e também está alinhado com a *green computing* e sua preocupação com o gerenciamento dos recursos.

O restante deste artigo está dividido como segue. A seção 2 apresenta considerações sobre o tratamento de segurança adaptativa em DMs com foco em redução de consumo. Nesta seção também são discutidos os trabalhos relacionados. Na seção 3, a visão geral, os algoritmos e a arquitetura do mecanismo proposto neste trabalho são discutidos. A adoção do mecanismo é tratada na seção 4, onde as atividades necessárias e colaborativas são detalhadas. Na seção 6 são apresentadas as conclusões.

2. Segurança Adaptativa e Eficiência de Consumo

Sistemas de segurança desenvolvidos utilizando os métodos existentes de desenvolvimento de software adaptativo têm sido implementados nos níveis de hardware, do sistema operacional, de rede e no nível de aplicação, porém existe uma consciência geral de que o assunto é relevante e carece de pesquisa [Elkhodary 2007]. Um estudo de metodologias para sistemas de segurança adaptativos e dinâmicos que alteram seu comportamento em tempo de execução apresentado por Elkhodary (2007) indica a necessidade de pesquisas sobre o *trade-off* entre oferta de segurança e eficiência no uso de recursos computacionais, pois ele afirma que a busca pela garantia de segurança normalmente gera sobrecarga do hardware.

Focando nas aplicações de DMs, onde as mudanças de contexto e as necessidades de adaptação podem ser constantes, este trabalho considera as orientações para sistemas adaptativos relacionadas por Elkhodary (2007 apud McKinley 2004), as quais destacam que a segurança adaptativa deve atuar na camada de aplicação e no sistema operacional, e se adaptar em conformidade quando determinados eventos alterarem o nível de ameaça do sistema.

Quanto à definição de complexidade e relevância dos algoritmos criptográficos, há um modelo de avaliação que busca um equilíbrio entre segurança e qualidade de serviço [Chen et al. 2010]. Este modelo considera que os algoritmos criptográficos sejam avaliados pelo tamanho de sua chave (devido à complexidade na avaliação de oferta de segurança) atrelando uma variável que usa pesos de acordo com sua complexidade computacional para definir o nível de garantia de confiança proporcionada. O modelo de Chen (2010) faz uma relação com o impacto no desempenho ocasionado pelo processamento destes algoritmos em dispositivos existentes nas redes sem fio, devido à sobrecarga na alocação de seus recursos.

Este trabalho segue as recomendações citadas nesta seção para a construção do seu mecanismo e relaciona, abaixo, algumas propostas relacionadas. A maioria trata de adaptação de protocolos de segurança na camada de enlace com diferenças bem sutis.

2.1. Trabalhos Relacionados

Os trabalhos mais relevantes encontrados e relacionados com eficiência de consumo de recursos computacionais, quando se trata o quesito segurança, mais especificamente a criptografia de dados para garantia de confidencialidade, estão destacados a seguir.

Um mecanismo de adaptação dinâmica é encontrado em um trabalho [Taddeo 2010] voltado para redes de sensores sem fio (RSSF), o qual adapta a segurança de forma gradual pela seleção de algoritmos criptográficos. Segundo os autores, normalmente se utiliza uma abordagem onde, ou se usa um algoritmo de segurança ou não se usa nenhum, a qual se estende para os ambientes de computação móvel, contrariando as exigências de requisitos para estes dispositivos, que incluem mobilidade, flexibilidade, configuração em tempo real e dinamicidade de ambientes, pois o mesmo algoritmo é usado em todos os ambientes percorridos. O trabalho destaca também que um dos tópicos mais relevantes na atualidade, e também o problema principal e complexo neste tipo de adaptação, é a manutenção do nível adequado de proteção, principalmente em tempo de execução. Os autores propõem então um mecanismo que lida com adaptação de segurança de acordo com os requisitos de segurança da aplicação e com as restrições de energia de forma dinâmica. Baseiam-se no princípio da proteção adequada, o qual prega que uma informação deve ser protegida em escalas consistentes com este valor, sendo assim aplicada adequadamente de acordo com o contexto [Taddeo apud Pfleeger 2006]. Seu foco principal é satisfazer os requisitos de consumo de energia, atuando na adaptação do nível do sinal propagado, encerrando algumas aplicações. Ele provê um algoritmo para todas as aplicações em uso enquanto o MeSAD mantém o algoritmo adequado para cada aplicação e ainda provê uma avaliação para melhor prover esta adaptação baseada no contexto, mas sem encerrá-las como a abordagem de Taddeo (2010) o faz.

Outro trabalho utiliza a adaptação com base em estudos de possíveis cenários para decisões em tempo de projeto [Hamad et al. 2009], considerando sua adoção em dispositivos com limitações de recursos. Ele investiga alguns algoritmos, considerados pelos autores como os mais populares quanto às suas complexidades e utilização de recursos. Sua adaptação está centrada em oferecer aos usuários a opção de realizarem escolhas do melhor esquema de segurança a adotar de acordo com essas informações. Difere da proposta aqui tratada porque trata as adaptações em tempo de projeto, de forma não dinâmica e não age em tempo de execução.

Um serviço de criptografia adaptativo é proposto por Izquierdo et al. (2007), levando em conta as limitações de capacidade de processamento e requisitos de segurança. É uma arquitetura de criptografia de dados que usa mais de um algoritmo para criptografar vários blocos de informação. O serviço usa a criptografia de apenas alguns blocos da informação, ou seja, dentre o total dos blocos trafegados alguns são enviados sem proteção, também pode usar outros algoritmos em paralelo para criptografar diferentes blocos. A diferença do presente trabalho deve-se ao tratamento de segurança a ser aplicado no MeSAD considerar a criptografia da informação por completo sem dividí-la em blocos, e também por não deixar trechos da informação trafegar sem proteção, o que Izquierdo (2007) afirma que acontece.

Com foco em desempenho, o trabalho de Rocha et al. (2007) é um middleware adaptativo de seleção dinâmica de protocolos de segurança na camada de enlace, que

usa variações de parâmetros da rede sem fio, recursos disponíveis do sistema em uso e níveis de segurança. Trata diferentes tipos de dados com configurações específicas de segurança. Difere deste trabalho por focar na camada de enlace, enquanto o mecanismo proposto neste artigo, apesar de focar na camada de aplicação, atua transversalmente, adaptando seu grau de confidencialidade.

O Prometheus [Pirmez 2009] é definido como um serviço de segurança ciente de contexto capaz de adicionar controles de segurança dinamicamente. Não oferece algumas características destacadas como importantes pelo próprio trabalho, como considerações sobre medições de consumo de energia e o devido tratamento para redução de consumo. Sua adoção está focada em nichos de aplicações onde as políticas de utilização devem estar bem definidas, com regras de execução entre si e com definições de prioridades de execução, diferentemente deste trabalho, que é caracterizado pela necessidade de independência das aplicações em execução e não limitação de seu uso, e que tampouco encerra qualquer aplicação em uso.

O SARM (Security Adaptation Reference Monitor) [Maliki 2010] foi proposto para lidar com ambientes dinâmicos das redes sem fio e também defende a abordagem de formas flexíveis para lidar com a segurança de maneira dinâmica, em tempo de execução e com uso de informações de contexto. Propõe um monitor genérico de adaptação de segurança considerando o desempenho em relação ao consumo de energia. Também defende o uso de adaptação de segurança em tempo de execução na rede sem fio para atenuar as conseqüências do número substancial de ameaças quando não se pode eliminá-las por completo, pois não há consciência de qual mecanismo de segurança se deve utilizar nas aplicações e que sejam tão dinâmicos na proteção quanto são as ameaças. Para cada aplicação, o usuário deve preencher uma avaliação técnica detalhando suas preferências de uso e é baseado nestas avaliações que o SARM realiza as adaptações. Restringe-se a avaliar ou não o uso de segurança e, com esta decisão, considera que está economizando energia. Diferencia-se deste trabalho pela maneira como lida com as informações de contexto, por não tratar escalas de segurança, não levantar informações de consumo de energia e concentrar as avaliações de todas as aplicações no monitor a partir do kernel.

Tabela 1. Principais características dos trabalhos relacionados e o MeSAD

Abordagens propostas	Usa graus ou escalas de segurança	Sensível ao Contexto	Trata economia de recursos	Mantém sempre um nível de segurança	Adaptação dinâmica em tempo de execução	Atua na camada de aplicação	Mantém independência das aplicações
Taddeo	✓	✓	✓		✓		
Hamad	✓		✓	✓			
Izquierdo		✓	✓		✓		
Rocha	✓	✓	✓		✓		
Pirmez		✓		✓	✓	✓	
Maliki		✓	✓		✓		
MeSAD	✓	✓	✓	✓	✓	✓	✓

Na Tabela 1 pode ser verificado que os outros trabalhos deixam de fora pelo menos um dentre os aspectos que são considerados no MeSAD, o qual respeita os requisitos de segurança e de execução de cada aplicação, usando a sensibilidade ao contexto na adaptação dinâmica e trata a eficiência na utilização dos recursos do DM.

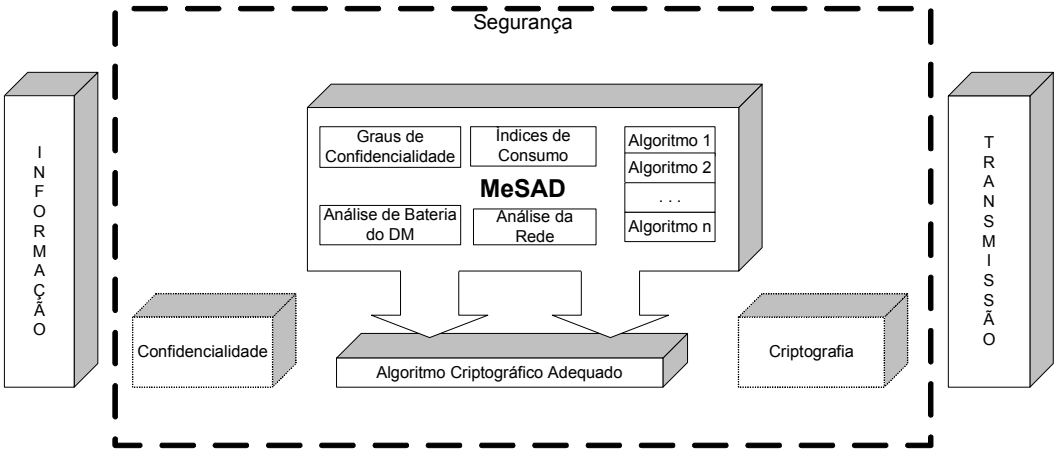


Figura 1. Visão Geral do MeSAD

3. Mecanismo de Segurança com Adaptação Dinâmica

A confidencialidade tratada na camada de aplicação tem a vantagem da exploração das propriedades específicas das aplicações para possibilitar o emprego mais eficiente dos mecanismos de segurança. Assim, o Mecanismo de Segurança com Adaptação Dinâmica (MeSAD) proposto neste trabalho, visa explorar as características individuais das aplicações para melhor adaptar o tipo de criptografia necessária, diminuindo a carga imposta pelo custo computacional desta operação e preservando os recursos do DM.

3.1. Visão Geral

Uma visão geral do funcionamento do MeSAD pode ser visto na Figura 1, onde uma aplicação que precisa garantir confidencialidade da Informação em tempo de execução, implanta aspectos de Segurança para realizar a Transmissão de seus dados. O requisito de segurança considerado é o de Confidencialidade que utiliza a Criptografia dos dados através de um Algoritmo Criptográfico. Este último é definido pelo MeSAD que utiliza informações de contexto para avaliar qual é o algoritmo mais adequado.

O ambiente no qual são utilizados os DMs apresenta uma série de particularidades, dentre elas a necessidade da manutenção de diferentes níveis de confidencialidade para os diversos graus de riscos de utilização deste dispositivo. O MeSAD foi projetado para resolver este problema mantendo o nível de segurança adequado, de maneira dinâmica, à medida que o DM passeia pelos diversos ambientes.

Neste tipo de cenário, um aspecto crítico de sistemas preocupados com contexto é a capacidade de adaptar seu comportamento de acordo com o contexto atual e, para isso, informações devem ser adquiridas de fontes externas (e.g., sensoriamento). O MeSAD utiliza informações de contexto capturadas em tempo de execução para definir a criptografia mais adequada a fim de garantir o grau adequado de confidencialidade da aplicação. Desta forma, ele possibilita que diferentes aplicações possam utilizar algoritmos de criptografia distintos de acordo com seu contexto e classificação de grau de confidencialidade, proporcionando o emprego mais eficiente dos recursos e evitando a carga proporcionada pelas soluções de segurança generalizadas.

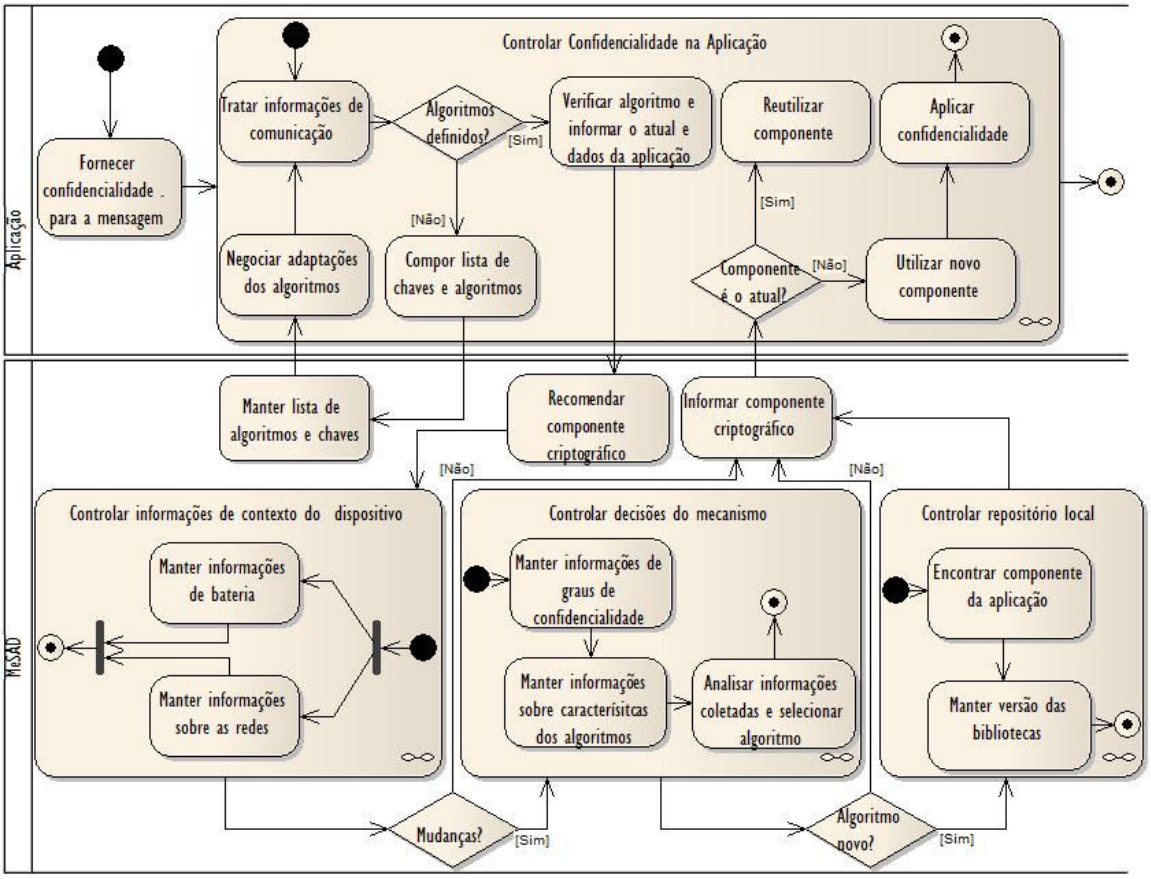


Figura 2. Os algoritmos do MeSAD dispostos na aplicação e no DM

Na Figura 2 estão evidenciados os algoritmos do mecanismo proposto, onde o módulo criptográfico que estará embutido na aplicação se encontra detalhado na raia **Aplicação** e o algoritmo do MeSAD em funcionamento no DM está detalhado na raia **MeSAD** da figura. A figura mostra a atividade de fornecimento de confidencialidade para a mensagem, iniciada pela aplicação e a partir dali as atividades do MeSAD interagem para controlar a escolha mais acertada do algoritmo e a utilização do componente nesta oferta de segurança criptográfica para a mensagem.

Para proporcionar segurança individualizada por aplicação e adaptativa, o MeSAD faz uso de informações de configuração da aplicação, informações de contexto capturadas em tempo de execução, histórico sobre redes utilizadas pelo usuário e classificações configuradas em tempo de projeto. As informações de contexto obtidas em tempo de execução, juntamente com as informações geradas em tempo de projeto irão influenciar a tomada de decisão.

O funcionamento do MeSAD é demonstrado na Figura 3, onde um repositório local conterá todos os componentes necessários para as aplicações instaladas no DM e todos os componentes implementam as interfaces que possibilitam que as aplicações os utilizem. A aplicação irá utilizar o módulo MeSAD UsaCripto, existente em seu Módulo Criptográfico local e responsável por tratar a criptografia e descriptografia das suas informações. Ele possui um componente que implementa a interface da biblioteca de

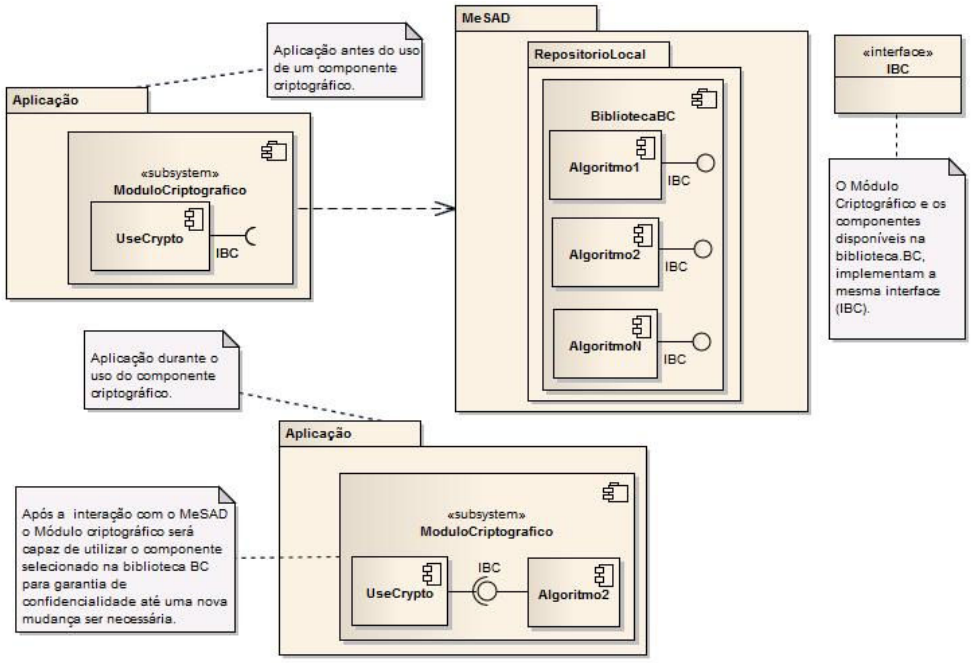


Figura 3. Uma aplicação sendo provida por um componente de criptografia

segurança adotada representada por IBC. Quando necessitar utilizar criptografia, ele irá consultar o módulo principal MeSAD existente no DM, passando informações da aplicação. O mecanismo analisará as informações de contexto e de configuração, aplicará suas regras de decisão e acessará seu Repositório Local para localizar a biblioteca do componente identificado e o fornecerá para que UseCrypto o utilize.

No caso representado pela Figura 3, o componente implementa a interface da biblioteca BC e o MeSAD está escolhendo um componente desta biblioteca, a qual implementa a interface IBC que provê compatibilidade com UseCrypto. Estes componentes poderão, em tempo de execução, serem substituídos de acordo com a necessidade da aplicação e com o contexto identificado no momento do uso.

3.2. Arquitetura do MeSAD

O MeSAD possui uma arquitetura focada no uso de componentes para prover adaptação dinâmica, pois eles podem ser adicionados, substituídos e reconfigurados em tempo de execução, permitindo adaptação às novas necessidades [Rocha 2007 apud McKinley 2004]. O MeSAD é instalado no DM e os módulos criptográficos (subsistemas) das aplicações que o adotam realizam a interação, sendo assim é dividido em duas partes: a solicitante e a fornecedora.

Na Figura 4, pode-se observar a existência de um Módulo Criptográfico na representação da Aplicação. Ele é adicionado na implementação das aplicações que utilizam o mecanismo e é responsável por acionar o MeSAD, provendo informações sobre versão e configuração da aplicação, e por tratar a confidencialidade. O pacote MeSAD representa a maior parte do mecanismo instalado no DM e é responsável por atender às requisições de decisão e fornecer os componentes para as aplicações. Ele aciona o Módulo de Decisão ao receber as informações sobre a aplicação que requer

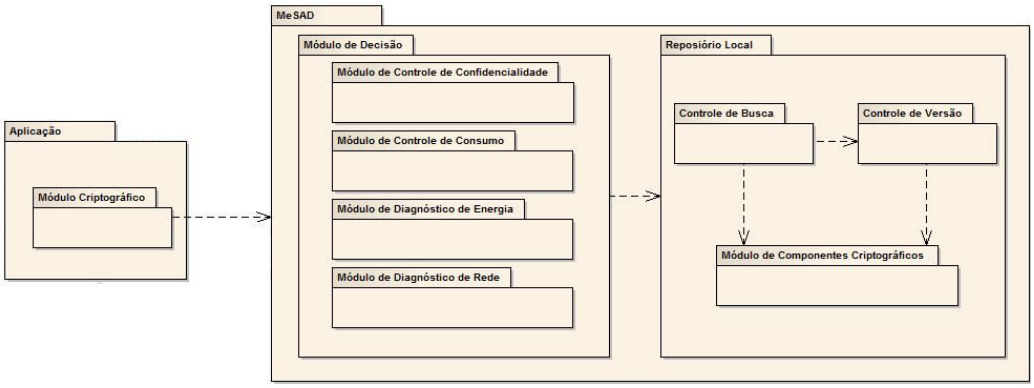


Figura 4. Diagrama de pacotes do MeSAD

criptografia. Este módulo utiliza seus mecanismos (representados por seus módulos internos) para capturar as informações necessárias para a tomada de decisão sobre qual algoritmo de criptografia é o mais adequado para a situação verificada e, em seguida, consulta o Repositório Local para encontrar o componente que o representa. Este último analisa a informação e verifica a versão e o local de busca para suprir a aplicação.

A seleção dos algoritmos candidatos será possível com o uso dos graus de confidencialidade, a análise de confiança das redes, o nível de bateria do DM e o índice de consumo de energia dos algoritmos.

Com o algoritmo criptográfico definido, ele será utilizado pela aplicação até que o mecanismo identifique a necessidade de uma nova adaptação. Esta mudança ocorrerá caso o nível de bateria entre em estado crítico, a rede utilizada mude ou o mecanismo solicite a mudança divulgando as novas configurações pelo protocolo da aplicação.

Todo o processo de uso e substituição dos algoritmos está ilustrado na Figura 5, onde estão representados CtrlCripto e UseCripto, que implementam as interfaces do MeSAD e são partes do Módulo Criptográfico presente em cada aplicação. O CtrlCripto controla o acesso ao MeSAD para obter o componente de criptografia mais adequado para a aplicação, de acordo com o contexto, e o fornece ao UseCripto, que, por sua vez, coloca em prática a cifragem e a decifragem das informações. Ele também verifica se o componente precisa ser modificado e, em caso positivo, fornece o novo componente para UseCripto, estando representado na figura pelo laço e a condição de mudança do componente. Pode-se notar a interação do Módulo de Decisão com seus módulos internos que representam as atividades do mecanismo (graus de confidencialidade, índices de consumo, análise de bateria e análise de rede) e também o acesso ao Repositório Local onde será identificado o componente de acordo com as necessidades e configurações da aplicação.

A parte do MeSAD que permanece na aplicação e trata a confidencialidade das mensagens está descrita no diagrama da Figura 6, o qual mostra a interação do Módulo Criptográfico da aplicação com os outros módulos. A aplicação interage com CtrlCrypto passando o texto para que ele promova o tratamento (cifragem/decifragem). Informações de configuração desta aplicação são colhidas e enviadas ao MeSAD do DM para fornecimento do componente de criptografia adequado. Em seguida, informações como chaves, algoritmo inicial e parâmetros para mudanças posteriores são protegidas

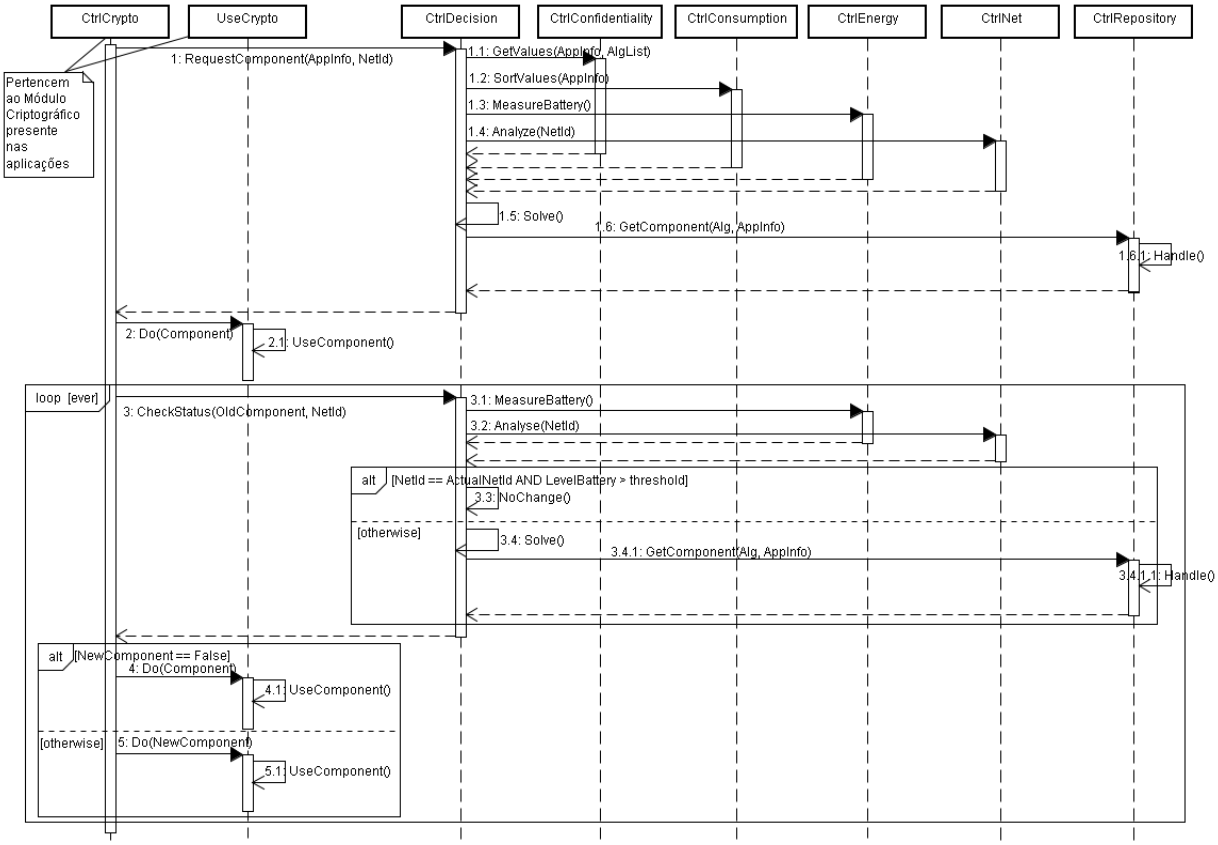


Figura 5. Diagrama de Seqüência Simplificado do MeSAD

com criptografia assimétrica e um protocolo de handshake inicia a comunicação para que seja estabelecido o mesmo algoritmo entre as partes (CtrlProtocolo). Após esta definição a mensagem é criptografada com o algoritmo simétrico estabelecido e enviada. Quando for necessário nova adaptação, esta informação será adicionada na última mensagem criptografada com o algoritmo anteriormente definido para divulgação do próximo algoritmo a ser utilizado nas próximas mensagens trocadas.

Quanto à decifragem de uma mensagem recebida, a Figura 6 mostra a interação do protocolo para verificação da lista de chaves e a verificação do componente em uso na criptografia da mensagem recebida. Caso ele já esteja em uso não é necessário novo fornecimento, caso contrário o MeSAD fornecerá o novo componente para uso.

4. Implantação do MeSAD

A adoção do MeSAD em um projeto de software adiciona atividades que vão desde a fase de concepção até a manutenção da aplicação, seguindo as orientações de Elkhodary (2007). Incorpora a preocupação com segurança nos processos de criação de softwares, tão necessária e cada vez mais requisitada, porém tão pouco utilizada [Chen et al. 2010].

Em cada fase do ciclo de vida do software, há uma maneira de atuar para a geração dos artefatos necessários para a utilização eficaz do mecanismo proposto. Porém, na fase de projeto, uma vez definidas as configurações de níveis de

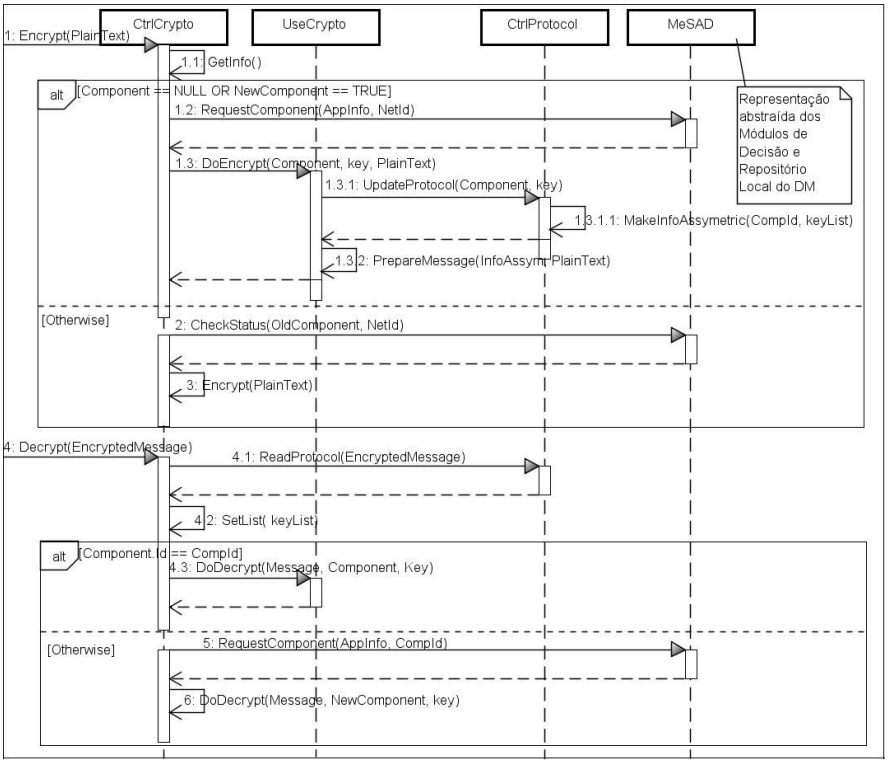


Figura 6. Diagrama de Seqüência da parte do MeSAD presente na aplicação

confidencialidade a ser praticada nas aplicações que utilizarão o MeSAD, esta etapa do processo poderá ser reutilizada para todas as novas aplicações, facilitando sua adoção.

Nas fases de análise e de projeto são definidos ou reutilizados os graus de confidencialidade pertinentes à aplicação que será criada e isto acontecerá de acordo com os requisitos e arquitetura avaliados no projeto. Também serão criados ou reutilizados, os índices de consumo de bateria para cada algoritmo criptográfico. No momento da construção do código as informações dos graus de confidencialidade serão utilizadas para a geração e introdução dos arquivos do MeSAD na aplicação. Os testes indicarão a possibilidade de refinamento dos graus definidos como entradas do mecanismo. A arquitetura do mecanismo facilita o processo de evolução ou manutenção da aplicação, pois basta definir os novos graus de confidencialidade e gerar as novas entradas do mecanismo para serem substituídas na aplicação.

4.1. Atividades Colaborativas

O funcionamento do mecanismo está atrelado às atividades que colaboram entre si, adquirindo os valores de entrada para as tomadas de decisões e execução das ações.

4.1.1. Graus de Confidencialidade

Como citado anteriormente, existem informações que deverão ser geradas em tempo de projeto e elas são fundamentais para o correto funcionamento do mecanismo. Dentre elas, a definição dos **graus de confidencialidade dos algoritmos de criptografia** contidos nas bibliotecas de segurança e a definição do **grau de confidencialidade da aplicação** que estará sendo criada de acordo com seus requisitos de segurança.

No projeto da aplicação, o engenheiro de software fará uso de uma classificação de grau de confidencialidade ofertada pelos algoritmos de criptografia. Esta classificação definida no MeSAD é exposta e justificada nos parágrafos seguintes.

O nível de garantia de segurança proporcionado pela criptografia é determinado pelo algoritmo criptográfico e pelo tamanho de chave utilizada [Chen 2010]. Assim, a classificação de graus de confidencialidade dos algoritmos se baseou neste argumento.

Analisando a possibilidade de tentativas de quebra de segurança por "força bruta" e considerando que um comprimento de chave com 8 bits ($2^8 = 256$) possui 256 possibilidades de definição de chaves, ou seja, no pior caso é preciso tentar 256 vezes até se descobrir a chave, substituindo o expoente por uma variável k de tamanho igual à chave utilizada (k_{tam}), pode se generalizar esta relação como: $2^{k_{tam}}$ **(1)**.

A expressão **(1)** representa o número de tentativas para obtenção da chave de criptografia e, assim definido, pode-se concluir que quanto maior for o k_{tam}, maior será o nível de segurança proporcionado por um algoritmo. Assim, para classificar os graus de confidencialidade proporcionados, assume-se que o valor deste grau é 1 quando for utilizado o menor tamanho de chave (k_{min}). Então, o grau de confidencialidade da criptografia (l_k) pode ser definido como: $\mathbf{l_k} = 2^{k_{tam}/k_{min}} - 1$ **(2)**.

Definido este formato, deve-se agora considerar a complexidade da criptografia oferecida pelos variados algoritmos criptográficos. Como na literatura não há uma definição concisa de uma classificação de cada algoritmo de acordo com sua complexidade e importância no processo de manutenção da confidencialidade com base em seus poderes computacionais, pode-se definir as diferenças entre os algoritmos a serem considerados pela atribuição de um peso (p), com valores de 0 até 1, na equação **(2).** Desta forma, tem-se agora: $\mathbf{l_k} = (2^{k_{tam}/k_{min}} - 1) * p$ **(3)**.

A atribuição do peso (p) facilita a adoção do MeSAD por engenheiros de software que queiram modificar suas configurações originais de acordo com seus conhecimentos e necessidades, além de oferecer a facilidade de novas classificações em futuras versões do mecanismo, com a inclusão de novos algoritmos.

Com os graus de confidencialidade dos algoritmos de criptografia definidos, é necessário definir o grau de confidencialidade da aplicação que está sendo criada. Assim, o engenheiro de software deve identificar a necessidade de segurança que sua aplicação requer e classificá-la nas opções disponíveis na ferramenta de suporte ao MeSAD. Ela segue o modelo comumente adotado em aplicações que utilizam classificações de níveis de segurança e no trabalho adotado para estas definições mencionadas acima [Chen 2010], o qual também utiliza três níveis. São eles: baixo, médio e alto. Um destes níveis será atribuído à aplicação para a determinação de seu requisito de segurança de acordo com suas necessidades e para a composição dos parâmetros de configuração nos arquivos que o mecanismo produzirá.

4.1.2. Índice de Consumo do Algoritmo de Criptografia

O engenheiro de software deve produzir ou reutilizar testes para avaliar o impacto do uso de cada algoritmo criptográfico quanto ao consumo de energia da bateria dos DMs, ou seja, os dispositivos nos quais se espera que a sua aplicação esteja em uso poderão

ser utilizados em testes para produzir uma lista de índices de consumo de bateria, possibilitando a criação de uma média de consumo para determinado categoria de DMs.

Estes testes possibilitarão a criação de outra entrada do MeSAD. Estas informações serão obtidas pelo uso de uma ferramenta de apoio com este intuito. Este software está sendo criado para ser uma ferramenta de coleta e análise de informações relacionadas com a execução de diversos algoritmos criptográficos em DMs, tratando-se de um avaliador de algoritmos criptográficos com relação ao consumo de energia da bateria por eles proporcionado. Para adquirir esta informação de índices de consumo, funcionará basicamente da seguinte forma: uma aplicação é instalada nos DMs que irão ser testados, alguns parâmetros são informados e a aplicação é então executada. Os resultados dos testes são enviados para um servidor web e mantidos para a criação de entradas de configuração do mecanismo a ser utilizado.

4.1.3. Análise de Confiança

Como confiança é uma sensação que provém do indivíduo, o qual interage com os elementos que propiciam ou não tal sensação, sua participação na definição de um valor que reflita o grau de confiança nas redes em uso é fundamental. Assim sendo, o MeSAD dará suporte para o usuário realizar uma avaliação de suas redes e irá considerá-la para conceber um grau de confiança em cada rede onde utilizará criptografia.

Esta informação possui duas formas de ser identificada se complementam para estabelecer qual rede é mais ou menos confiável. Uma delas é através de uma análise que é feita no momento do uso da aplicação, onde a rede em atividade é avaliada para identificar os mecanismos que incrementam a segurança no tráfego via rádio das informações, caso existam. A segunda forma de identificar o grau de confiança da rede é a análise direta do usuário quanto à sua confiança na rede que ele está utilizando. Esta avaliação é importante porque apenas o usuário pode informar o quanto ele confia ou não em determinadas redes e de acordo com as características desta rede, este usuário pode avaliá-la e definir o seu nível de confiança, o qual será mantido para reutilização. Um exemplo de como esta informação é restrita ao conhecimento do usuário seria uma rede existente num ambiente remoto ou isolado (e.g., uma chácara com acesso à internet fornecido por um ponto de acesso de fraca segurança em seu enlace aéreo), onde o usuário sabe que não existiria um agente malicioso nas proximidades capaz de capturar a sua transmissão. Outro exemplo seria o usuário acessando uma rede a partir de um aeroporto em um ponto de acesso com tecnologia moderna de proteção, porém com chave de acesso compartilhada por várias pessoas, diminuindo sua confiança nesta rede.

4.1.4. Análise do Nível de Bateria

Como o MeSAD atua em tempo de execução para propiciar a adaptação dinâmica, de acordo com informações de contexto e sendo uma destas informações ligada diretamente com a preocupação de consumo eficaz de recursos computacionais com foco na bateria, nada mais natural que o nível de energia do DM seja monitorado para refinar a escolha do algoritmo criptográfico para uso na garantia de confidencialidade. Esta informação é capturada em determinados instantes, quando a adaptação estiver sendo considerada e pode influenciar na escolha de um algoritmo que consome menos de acordo com o nível da bateria.

5. Conclusões

Este trabalho apresentou o MeSAD, cujo objetivo é proporcionar a segurança adequada para aplicações de DMs em ambientes considerados inseguros, e diminuir a saturação no uso de recursos quando estiverem em ambientes considerados seguros, através da adaptação da confidencialidade em tempo de execução. A construção do MeSAD considerando questões como computação verde, aumento de confidencialidade em ambientes críticos, adaptação em tempo de execução de acordo com o contexto e tratamento individualizado de confidencialidade por aplicação, o coloca em sintonia com os desafios do futuro da computação.

Para facilitar a implantação do MeSAD nas aplicações criadas para DMs, uma ferramenta de apoio está em desenvolvimento e será responsável por configurar informações sobre a aplicação, avaliar o consumo de energia dos algoritmos criptográficos e gerar os arquivos necessários para a utilização do mecanismo. Diminuindo consideravelmente o trabalho do engenheiro de software que precisa tratar a criptografia das informações nas suas aplicações.

No futuro, pretende-se ainda disponibilizar o maior número possível de bibliotecas de segurança e adicionar outros requisitos de segurança, além da confidencialidade, bem como oferecer o mecanismo em diferentes plataformas.

Referências

Bishop, M., Computer Security: Art and Science, Addison-Wesley, 2003.

Bosch, Jan. Design and Use of Software Architecture, Addison-Wesley, 2000

Bragg, R., et al. (2004). Network Security: The Complete Reference. McGraw-Hill.

Bringel, J. R. M. F., Viana, W., Braga, R., Andrade, R. M. C. (2005).FRAMESEC: A Framework for the Application Development with End-to-End Security Provision in the Mobile Computing Environment. In: AICT 2005. Proc. IEEE Computer Society.

Carvalho, A. F. M. (2008). M-CODE: Um Modelo para Medição de Confidencialidade e Desempenho para Aplicações Móveis Seguras. 96p. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal do Ceará, Brasil.

Chen, J., Hu, C., Zeng, H., (2010). A Novel Model for Evaluating Optimal Parameters of Security and Quality of Service. Journal of Computers, vol 5, no. 6.

Dey, A.K. (2001). Understanding and Using Context. Personal Ubiq. Comp. v5, pp4-7

Darco, P., DE Santis, A., Feara, A. L., Massucci, B. (2010). Variations on a theme by Akl and Taylor: Security and tradeoffs. Theor. Comput. 411, 1, 213-227.

Elkhodary, A., Whittle, J. (2007). Survey of Approaches to Adaptive Application Security. ICSE Workshops SEAMS '07. pp. 16-16.

Gentry, C., Ramzan, Z. (2004). Provable cryptographic security and its applications to mobil wireless computing. Wireless Personal Communication, v. 29, pp.191-203.

Georgas, J.C., Hoek A.V.D., Taylor R.N. (2005). Architectural Runtime Configuration Management in Support of Dependable Self-Adaptive Software, WADS, v30,pp. 1-6.

Hamad, F., Smalov L., James A. (2009). Energy-aware Security in M-Commerce and the Internet of Things. IETE. 26:357-62.

Herrick, D. R., Ritschard, M. R. (2009). Greening your computing technology, the near and far perspectives. In Proceedings of the ACM SIGUCCS Fall Conference on User Services Conference. SIGUCCS '09. ACM, New York, NY, 297-304.

Izquierdo, A., Sierra, J.M., Torres, J. (2006). On the implementation of security policies with adaptative encryption. Comput. Commun. 29(15), 2750–2758.

Li, H.(2006).Multilievel Adaptive Security System.92p. Thesis(Doctor of Philosophy in Computer Engineering).New Jersey Institute of Technology(NJIT),New Jersey, USA

Maliki, T.E.; Seigner, J., (2010). A Security Adaptation Reference Monitor (SARM) for Highly Dynamic Wireless Environments. Emerging Security Information Systems and Technologies. International Conference, pp.63-68, 18-25.

Oreiz, P., et. al(2008).Runtime Software Adaptation: Frameworks, Approach and Styles. 30th International Conference on Software Engineering. ICSE'08. ACM, pp 899-910.

Pirmez, M. (2009). Prometheus: Um Serviço de Segurança Adaptativa. 115p. Dissertação (Mestrado em Informática) - Universidade Federal do Rio de Janeiro.RJ.

Rocha, B.P.S., Costa, D.N.O., Moreira,(2010).Adaptive security protocol selection for mobile computing. J. Netw. Comput. Appl. 33, 5 (September), 569-587.

Rocha, L.S., Castro, C. E. P. L., Machado, J. C., Andrade, R. M. C. (2007). Utilizando Reconfiguração Dinâmica e Notificação de Contextos para o Desenvolvimento de Software Ubíquo. In: XXI SBES. João Pessoa.

Salido, J., Lazos, L., Poovendran R. (2008). Energy and bandwidth-efficient key distribution in wireless Ad Hoc networks: A cross-layer approach. IEEE/ACM Transactions on Networking, vol. 15, No. 6, pp. 1527-1540. DEC.

Schilit, B., Adams, N., Want, R.(1994). Context-aware computing applications. In IEEE Workshop on Mobile Comp. Systems Applications. IEEE Comp. Soc. Press. p85-90.

Solyman, S. H., Omari M. (2004). An Efficient Application of a Dynamic Crypto System in Mobile Wireless Security. WCNC/IEEE C. Society, vol.2, pp. 837-842.

Stephen, Ruth. (2009). Green IT – More Than a Three Percent Solution, IEEE Internet Computing, vol. 13, no. 4, pp. 74-78, July/Aug., doi:10.1109/MIC.2009.82.

Szyperski, C. A. (1998). Component software: beyond object-oriented programming, ACM Press/Addison-Wesley Publishing Co., New York, NY.

Taddeo, A.V. et al.(2010).Gradual Adaptation of Security for Sensor Networks. In:IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks.

Viana, W., Cavalcante, P., Andrade, R. M. C. (2005).Mobile Adapter: Uma abordagem para a construção de Mobile Application Servers adaptativos utilizando as especificações CC/PP e UAProf. In: XXV Congresso da Sociedade Brasileira de Computação, 2005, Sao Leopoldo, RS-Brasil. Anais do XXXII SEMISH.

Wilbanks, L. (2008). Green, My Favorite Color. IT Pro v.10, n.6. pp.63-64. IEEE Comp Soc. Press.

Classificação de Gêneros Musicais por Texturas no Espaço de Frequência

Yandre M. G. Costa[1,2], Luiz S. Oliveira[2], Alessandro L. Koerich[3], Fabien Gouyon[4]

[1]Departamento de Informática
Universidade Estadual de Maringá (UEM)*
Av. Colombo, 5790 – 87020-900 – Maringá – PR – Brazil

[2]Departamento de Informática
Universidade Federal do Paraná (UFPR)†
Curitiba – PR – Brazil

[3]Programa de Pós-graduação em Informática
Pontifícia Universidade Católica do Paraná (PUC-PR)
Curitiba – PR – Brazil

[4]Instituto de Engenharia de Sistemas e Computadores do Porto – (INESC-Porto)
Porto – Portugal

yandre@din.uem.br, lesoliveira@inf.ufpr.br, alekoe@ppgia.pucpr.br,

fgouyon@inescporto.pt

Abstract. *This paper describes an attempt to perform automatic musical genre classification based on spectrograms extracted from segments of digital music pieces, taken from the "Latin Music Database". Feature vectors with textural characteristics were extracted from digital images of spectrograms by using gray level co-occurrence matrix. The recognition rate obtained with a Support Vector Machine classifier was 60,11%. This rate is slightly higher than other obtained with different approaches recently performed over the same database. In addition, the classifier proposed here was combined with another classifier. The results obtained show an recognition rate about 66,11% and the upper limit found combining this two classifiers is about 75%.*

Resumo. *Este artigo descreve a classificação de gêneros musicais utilizando espectrogramas extraídos de músicas. Foram gerados espectrogramas a partir do sinal de áudio das músicas, tomadas da "Latin Music Database". A partir destes espectrogramas foram extraídas características de textura com o uso de matriz de co-ocorrência. Foi utilizado Support Vector Machine para a classificação e o desempenho final do esquema de classificação proposto atingiu uma taxa média de acerto de 60,11%. Este desempenho é superior ao de outras abordagens recentemente aplicadas sobre a mesma base. O classificador aqui proposto foi ainda combinado com um outro classificador descrito na literatura. A combinação atingiu taxa de acerto de 66,11%, com limite superior de 75%.*

*Este trabalho é financiado pela Fundação Araucária (Chamada de projetos 01/2009 - protocolo 16.767), †CNPq e CAPES.

1. Introdução

Com a rápida expansão da Internet uma imensa massa de dados oriundos de diferentes fontes tem se tornado disponível *on-line*. A gestão da informação em grandes volumes de dados multimídia está arrolada entre os grandes desafios da pesquisa em computação no Brasil [Lucena et al. 2006], estabelecidos no seminário realizado em 2006 e com isso o desenvolvimento des pesquisas que busquem soluções neste contexto se mostra bastante oportuno. Estudos apontam que, em 2007, a massa de dados digitais espalhada ao redor do mundo consumia aproximadamente 281 exabytes e que, em 2011, esse volume deve se multiplicar por 10 [Gantz et al. 2008]. Porém, boa parte destas informações não segue um padrão de apresentação e não está disponível de maneira estruturada, o que torna muito difícil fazer uso adequado das mesmas.

Devido a isso, tarefas como busca, recuperação, indexação, extração e sumarização automática dessas informações se tornaram problemas importantes sobre os quais muitas pesquisas têm sido realizadas. Neste contexto uma área de pesquisa que tem crescido nos últimos anos é a de recuperação de informações multimídia, que visa criar ferramentas capazes de organizar e gerenciar essa grande quantidade de informações. No momento, a maior parte das informações sobre dados multimídia são organizadas e classificadas baseadas em meta-informações textuais que são associadas ao seu conteúdo, como é o caso dos rótulos ID3 incorporados nos arquivos de áudio no formato MP3. Apesar destas informações serem relevantes para as tarefas de indexação, busca e recuperação, elas dependem da intervenção humana para gerá-las e, posteriormente, associá-las aos arquivos multimídia.

A classificação automática de gêneros musicais pode auxiliar ou substituir o usuário humano nesse processo, assim como prover um componente importante para um sistema de recuperação de informações para músicas, podendo também tornar menos subjetivo este processo de atribuição. Estudos sobre comportamento de usuários de Sistemas de Recuperação de Informação Multimídia, também conhecidos como sistemas MIR (*Multimidia Information Retrieval*), indicam que o gênero musical é o segundo item mais fornecido nas *queries* de busca.

A idéia de classificação automática de gêneros musicais como uma tarefa de reconhecimento de padrões de sinais de músicas foi apresentada em [Tzanetakis and Cook 2002]. Neste trabalho, foi proposto um conjunto abrangente de características para representar o sinal de áudio. Os experimentos foram avaliados em uma base de dados contendo 1.000 músicas de 10 gêneros distintos, sendo 100 músicas de cada gênero. O acerto obtido inicialmente nessa base foi de cerca de 60%. É importante observar que os experimentos foram realizados utilizando apenas os primeiros 30 segundos de cada música.

Em [Li et al. 2003], foi realizado um estudo comparativo para a classificação automática de gêneros musicais baseada em conteúdo entre o conjunto de características propostas por Tzanetakis e Cook e um novo conjunto de características. Os experimentos foram realizados em duas bases de dados, a primeira foi a mesma utilizada nos experimentos de Tzanetakis e Cook e uma segunda base contendo 756 músicas de cinco gêneros. Um aspecto importante dessa segunda base de dados é que as características foram extraídas do segmento composto pelo segundo 31 ao segundo 60, em vez dos primeiros trinta segundos. As conclusões dos experimentos realizados neste trabalho mostram a

melhor taxa de classificação obtida foi de 78% com a primeira base de dados e de 74% com a segunda base.

No trabalho de [Costa et al. 2004] foi proposto um novo método para a classificação automática de gêneros musicais, baseado na extração de características de três segmentos do sinal de áudio. As características foram extraídas do início, meio e fim da música. Para cada segmento foi treinado um classificador. As saídas fornecidas por cada classificador individualmente foram combinadas utilizando a regra de votação majoritária. Os classificadores utilizados foram MLP (Redes Neurais do tipo *Multi-Layer Perceptron*) e k-NN (k *Nearest Neighbor*). Uma continuação deste trabalho foi apresentada por [Koerich and Poitevin 2005] e os resultados obtidos mostraram uma melhora na taxa de acerto em relação aos segmentos individuais. O uso de mais de um segmento passou a ser utilizado em outros trabalhos subsequentes. Uma vantagem bastante clara que se obtém com o uso desta abordagem consiste no fato de que ela permite colher uma amostragem melhor do sinal, podendo captar variações presentes ao longo do mesmo que eventualmente um único segmento não conseguiria captar. Um único segmento de uma música pode ter características mais próximas às de um gênero diferente daquele identificado em seu rótulo, utilizando mais de um segmento e fazendo a fusão dos descritores extraídos de cada um deles, o erro cometido em um segmento é diluído e o impacto que ele provocaria na classificação tende a diminuir.

Um problema correlacionado foi apresentado por [Yu and Slotine 2009]. Neste trabalho os autores apresentam uma tentativa de reconhecer instrumentos musicais através de imagens de texturas caracterizadas nos espectrogramas obtidos a partir do sinal de áudio colhido de 8 tipos diferentes de instrumentos musicais. Neste trabalho, em que não ocorre a mistura polifônica naturalmente presente no sinal de músicas, os resultados mostram um acerto em torno de 85% para o método. Este trabalho inspirou o desenvolvimento dos experimentos que serão relatados no presente artigo, na medida em que suscitou a hipótese de que hajam informações presentes nas imagens caracterizadas pelos espectrogramas gerados a partir do sinal de áudio de musicas que permitam classificá-las de acordo com seus respectivos gêneros musicais.

Em [Lopes et al. 2010] é apresentado um método para o reconhecimento de gêneros musicais, aplicado sobre gêneros latinos, baseado na seleção de instâncias de vetores com características de tempo curto e características de baixo nível do sinal de áudio de músicas. O método empregou o *"artist filter"* na formação dos *datasets*. Proposto em [Pampalk et al. 2005], o *"artist filter"* determina que os títulos musicais de um mesmo interprete sejam todos atribuídos à um mesmo conjunto quando da divisão da base de músicas. Com isto, títulos de um mesmo interprete não são comparados em momento algum, pois nunca estarão simultaneamente no conjunto de treino e de teste. Esta técnica aumenta consideravelmente a dificuldade de se alcançar boas taxas de reconhecimento, entretanto, pode conferir maior robustez ao classificador, induzindo a classificação a ser orientada efetivamente a gênero, em vez de ser orientada a interprete. A base de dados empregada foi a *Latin Music Database (LMD)*, uma base particularmente desafiadora. Ao final, obteve-se taxa de acerto de 59,6%.

Este trabalho descreve uma tentativa de automatizar o reconhecimento de gêneros musicais latinos através da análise de texturas presentes nas imagens de espectrogramas gerados a partir do sinal de áudio de músicas. Os espectrogramas podem repre-

sentar dados referentes a um sinal de áudio no domínio de tempo e frequência, e pode ser um instrumento bastante útil para discernir detalhes importantes acerca do mesmo [French and Handy 2007]. A extração de atributos descritores das imagens dos espectrogramas caracteriza um novo formato de atributo descritor. A fim de verificar a complementaridade deste formato com outros empregados em trabalhos de classificação de gêneros musicais, foi realizada a combinação do classificador aqui proposto com o classificador descrito em [Lopes et al. 2010].

A principal contribuição deste trabalho é introduzir uma investigação acerca da complementaridade de um formato de descritor, extraído de espectrogramas gerados a partir do sinal de áudio, em relação a outros formatos de descritores já utilizados para os propósitos de classificação automática de gêneros musicais. Neste sentido, serão apresentadas estratégias estabelecidas dentro deste processo, como a divisão da imagem do espectrograma em zonas, fato que permitiu discriminar informações do sinal representantes de diferentes faixas de frequência, proporcionando melhores resultados em relação a taxas de acerto. Os experimentos baseados no classificador aqui proposto atingiram taxas médias de acerto de 60,11%, com desvio padrão de 9,06. Esta taxa é comparável a de outros trabalhos recentes aplicados sobre esta mesma base, utilizando o mesmo conjunto de dados. Resultados experimentais aqui descritos ainda mostram que esse tipo de estratégia é capaz de gerar resultados complementares aos sistemas tradicionais de extração de características de áudio, possibilitando o aumento das taxas de reconhecimento através da combinação de classificadores. A combinação do classificador aqui proposto com o classificador apresentado em [Lopes et al. 2010] gerou taxa média de acerto de 66,11% e o limite superior encontrado para a combinação foi de 75%.

A organização deste trabalho encontra-se da seguinte forma: na seção 2 são descritos detalhes acerca da extração de características a partir das imagens de espectrogramas empregada neste trabalho; a seção 3 aborda o esquema de classificação aqui proposto; a seção 4 mostra os resultados experimentais; e a seção 5 apresenta as conclusões e possíveis trabalhos futuros.

2. Extração de Características

Esta seção descreve detalhes acerca da extração de características das musicas com o uso de espectrogramas. Inicialmente, será abordada a geração dos espectrogramas a partir do sinal de áudio. Em seguida, serão detalhadas as técnicas de processamento de imagens empregadas para se obter atributos descritores das características.

2.1. Fonte de Dados para Geração dos Espectrogramas

Para realizar os experimentos aqui descritos, tomou-se como fonte de dados a *Latin Music Database*, uma base particularmente desafiadora, composta por músicas latinas de 10 diferentes gêneros musicais (Axé, Bachata, Bolero, Forró, Gaúcha, Merengue, Pagode, Salsa, Sertaneja, Tango) apresentada em [Silla Jr et al. 2008]. A LMD é baseada na percepção de especialistas humanos e foi desenvolvida com propósitos também voltados à classificação automática de gêneros musicais. No total, foram tomadas 900 musicas desta base, divididas em 3 conjuntos, cada um com 300 músicas, doravante denominados *dataset1*, *dataset2* e *dataset3*. Em cada um destes conjuntos foram incluídas 30 músicas de cada gênero e foi empregado o *"artist filter"* na distribuição dos títulos entre os conjuntos. Devido ao uso do *"artist filter"* não foi possível utilizar todas as músicas da LMD

no experimento, já que existe uma diferença significativa entre as quantidades de músicas de cada intérprete presentes na base.

Adicionalmente, adotou-se a estratégia descrita em [Costa et al. 2004] na qual os autores utilizam três diferentes segmentos dispersos ao longo do sinal do áudio. Para isto, foram tomados segmentos do início, meio e final de cada música. A fim de evitar que efeitos como *"fade in"*, *"fade out"* e vibração de platéia em músicas gravadas ao vivo tornassem trechos da amostra pouco discriminantes, utilizou-se como amostra do início da música o segmento compreendido entre o segundo 11 e o segundo 20 da música, e como amostra do final da música o segmento compreendido entre o segundo n-20 e o segundo n-11, sendo n a duração da música em segundos. O segmento central foi extraído do intervalo compreendido entre o segundo m-5 e o segundo m+5, sendo m o segundo que se encontra exatamente no meio do sinal da música. A figura 1 ilustra esta estratégia.

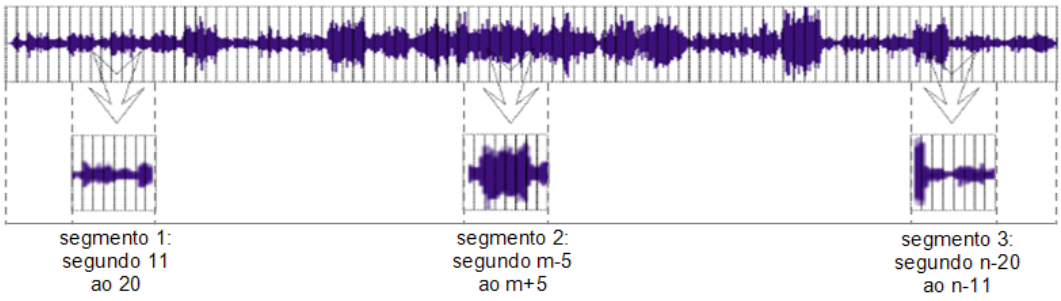


Figura 1. Extração de segmentos do sinal.

Foram gerados espectrogramas a partir de arquivos de áudio com a concatenação dos três segmentos para cada música utilizando-se o software SoX 14.3.0 (*Sound eXchange*), disponível em *http://sox.sourceforge.net*. As imagens dos espectrogramas foram geradas com valores *default* de largura (800 pixels) e altura (550 pixels).

Depois de extraídas as imagens dos espectrogramas das músicas, elas foram convertidas para níveis de cinza para melhor se adequarem aos processos subsequentes. A figura 2 mostra três exemplos de espectrograma utilizados nos experimentos. Estes espectrogramas referem-se a músicas de diferentes gêneros musicais e foram tomados aleatoriamente da base.

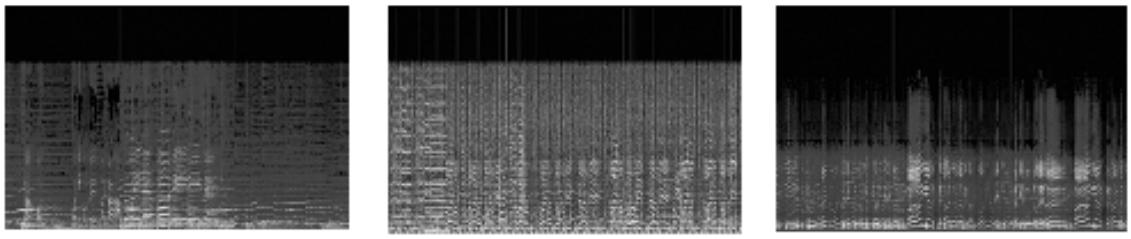


Figura 2. Exemplos de espectrograma.

2.2. Divisão das imagens dos espectrogramas em zonas

Depois de segmentadas as imagens dos espectrogramas, a fim de se remover legendas e outras áreas que não a de interesse, foi estabelecida uma divisão das imagens em zonas

para que se pudesse preservar parcialmente informações espaciais relacionadas às características a serem extraídas.

As imagens dos espectrogramas dos 3 segmentos foram divididas em 10 zonas. Esta divisão foi estabelecida após a realização de diversos testes com quantidades de zonas variando de 1 a 40. Os resultados indicaram a melhor taxa de acerto na situação em que os espectrogramas foram divididos em 10 zonas. O gráfico mostrado na figura 3 permite verificar a importância do zoneamento das imagens empregadas neste processo de classificação. O gráfico apresenta a evolução das taxas de acerto obtidas com os diferentes divisões em zonas testadas: 1 zona, 3 zonas, 5 zonas, 8 zonas, 10 zonas, 15 zonas, 20 zonas, 30 zonas e 40 zonas.

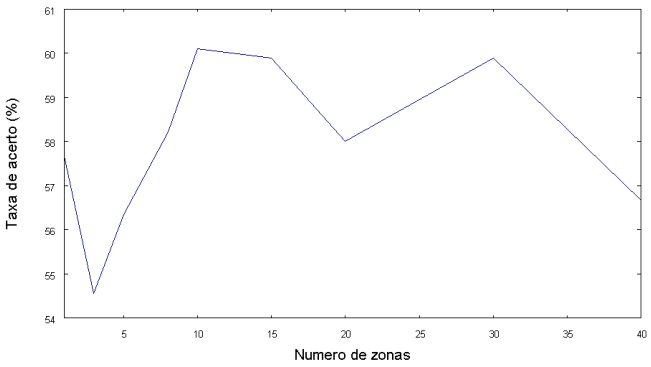


Figura 3. Evolução das taxas de acerto em função de diferentes quantidades de zonas.

Além de melhor taxa de acerto, a divisão em 10 zonas produz descritores com custo de armazenamento e tempo de processamento consideravelmente reduzidos em relação as outras divisões que apresentaram taxas de acerto comparáveis às dela. Com a divisão em zonas foi possível discriminar as características presentes em diferentes faixas de frequência representadas no espectrograma. Esta divisão está ilustrada na figura 4. De cada zona foram extraídos valores correspondentes às características descritas na próxima subseção.

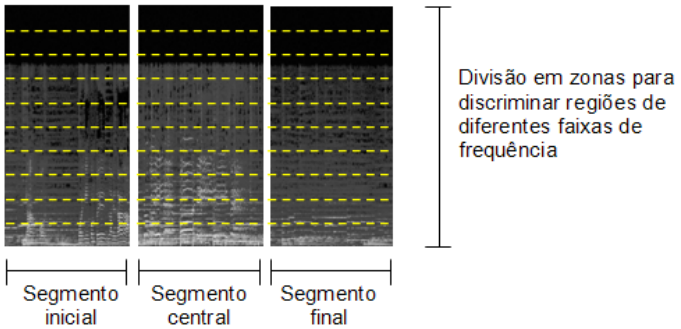


Figura 4. Esquema de divisão das imagens dos espectrogramas em zonas.

2.3. Matriz de Co-ocorrência

Partindo-se do princípio de que podem existir características texturais presentes nas imagens dos espectrogramas que sejam úteis na classificação de gêneros musicais, optou-se

por extraí-las empregando-se matrizes de co-ocorrência. Matriz de co-ocorrência (MC) é um método estatístico para descrição de texturas baseado na ocorrência repetida de níveis de cinza em uma textura, foi proposto em [Haralick et al. 1973] e ainda é empregado com sucesso em diferentes domínios de aplicação.

A partir da matriz de co-ocorrência, pode-se extrair várias medidas relacionadas a características de uma textura. Neste trabalho, para descrever as texturas encontradas nos espectrogramas, foram extraídas as medidas de contraste, energia, entropia, correlação, homogeneidade, momento de terceira ordem e probabilidade máxima. Este conjunto de características foi escolhido depois de alguns testes envolvendo algumas diferentes combinações de características, ao final, esta combinação apresentou o melhor desempenho.

A MC para imagens em tons de cinza armazena a probabilidade de que dois valores de intensidades de cinza estejam envolvidos por uma determinada relação espacial. Nos experimentos aqui descritos, os tons de cinza foram quantizados para 64 valores. Assim, cada MC gerada foi uma matriz de ordem 64×64. Parâmetros como a distância *d* entre os pixels e o ângulo θ caracterizado pela orientação da reta que passa pelos pixels são importantes na caracterização da relação espacial. Nos experimentos aqui descritos, os melhores resultados em termos de taxa de acerto foram obtidos com *d*=1, e foram utilizadas as orientações $0^o, 45^o, 90^o$ e 135^o para o ângulo θ. Sendo *p(i,j)* a probabilidade de ocorrência das intensidades de cinza *i* e *j* observando a distância *d*=1 e com um dado ângulo θ, as características utilizadas neste trabalho são encontradas pelas seguintes equações:

$$Contraste = \sum_{i=1}^{64}\sum_{j=1}^{64}(i-j)^2\, p(i,j) \quad (1)$$

$$Energia = \sum_{i=1}^{64}\sum_{j=1}^{64}(p(i,j))^2 \quad (2)$$

$$Entropia = -\sum_{i=1}^{64}\sum_{j=1}^{64} p(i,j)\; log\; p(i,j) \quad (3)$$

$$Homogeneidade = \sum_{i=1}^{64}\sum_{j=1}^{64}\frac{p(i,j)}{1+(i-j)^2} \quad (4)$$

$$Momento\ de\ terceira\ ordem = \sum_{i=1}^{64}\sum_{j=1}^{64} p(i,j)(i-j)^3 \quad (5)$$

$$Probabilidade\ máxima = \sum_{i=1}^{64}\sum_{j=1}^{64} max\; p(i,j) \quad (6)$$

$$\textit{Correlação} = \frac{p(i,j)-\mu_x\mu_y}{\sigma_x^2\sigma_y^2} \quad (7)$$

onde $\mu_x = \sum_{i=1}^{64} i \times p_x(i),\ p_x(i) = \sum_{j=1}^{64} p(i,j),\ \sigma_x^2 = \sum_{i=1}^{64}(i-\mu_x)^2 p_x(i)$, $\mu_y = \sum_{j=1}^{64} j \times p_y(j),\ p_y(j) = \sum_{i=1}^{64} p(i,j)$ *e* $\sigma_y^2 = \sum_{j=1}^{64}(j-\mu_y)^2 p_y(j)$.

Depois de encontrados os valores de cada uma destas características para cada uma das zonas descritas na subseção 2.2, foram finalmente formados os vetores com atributos descritores das músicas para a realização subsequente do processo de classificação. Para cada zona da imagem do espectrograma foram extraídas as 7 características em cada uma das 4 orientações previstas nesta subseção. Assim, para cada zona foi formado um vetor com um total de 28 características. A próxima seção descreve o emprego destes vetores no processo de classificação.

3. Esquema Geral da classificação Baseada em Espectrogramas

O classificador empregado neste trabalho foi o *Support Vector Machine* (SVM)[Vapnik 2000]. Para isto, foi utilizado o software LIBSVM [Chang and Lin 2001]. Para a execução das tarefas de classificação, foram tomados cuidados apropriados visando a normalização dos dados a fim de evitar que eventuais discrepâncias entre os valores de mesmas características em vetores diferentes prejudicassem o processo de classificação. A normalização foi feita de forma que os valores ficassem compreendidos em uma escala de variação entre -1 e 1. Além disso, os parâmetros C e γ do kernel Gaussiano foram otimizados utilizando uma busca gulosa com validação cruzada a fim de se alcançar melhores resultados.

Foi realizada a validação cruzada entre os três *datasets*, com 300 músicas cada, previamente descritos. Os conjuntos de teste e treinamento utilizados nos experimentos foram formados pelos vetores de características extraídos de cada zona da imagem do espectrograma e a divisão de conjuntos utilizada foi a mesma empregada em [Lopes et al. 2010]. Assim, considerando que de cada música foram extraídos 3 segmentos, de cada segmento de música foi extraído um espectrograma e que, a partir de cada espectrograma foram criadas 10 zonas, o conjunto de teste foi formado por 9.000 vetores enquanto o conjunto de treinamento foi formado por 18.000 vetores em cada execução do classificador. A figura 5 ilustra o esquema geral da classificação.

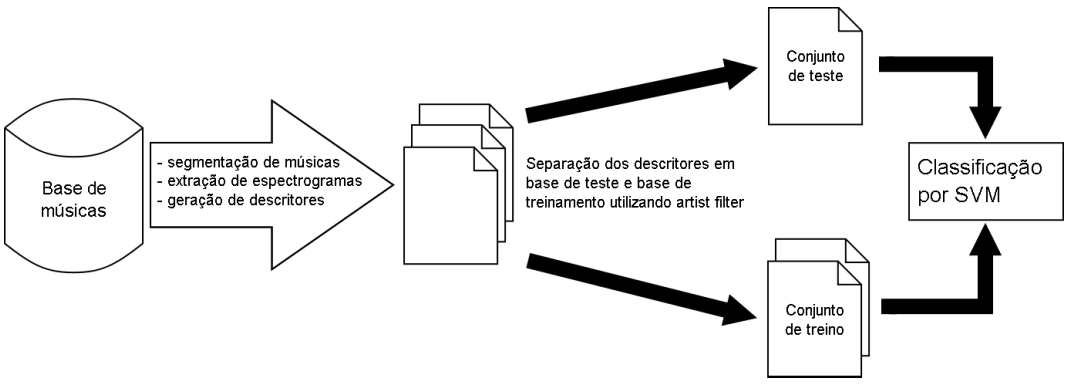


Figura 5. Esquema geral da classificação automática de gêneros musicais.

Depois de executado o classificador SVM para cada um dos cruzamentos previstos na validação cruzada, quais sejam: dataset1 $\times$ dataset2+dataset3, dataset2 $\times$ dataset1+dataset3 e dataset3 $\times$ dataset1+dataset2, obteve-se como resultado uma classe, neste caso gênero musical, atribuída ao vetor extraído de cada zona. A classificação final de cada música foi encontrada através da votação entre as classes atribuídas para as 30 zonas da mesma. Assim, o gênero para o qual foi atribuído o maior número de zonas

foi escolhido como gênero ao qual a música pertence. A figura 6 mostra o esquema de votação.

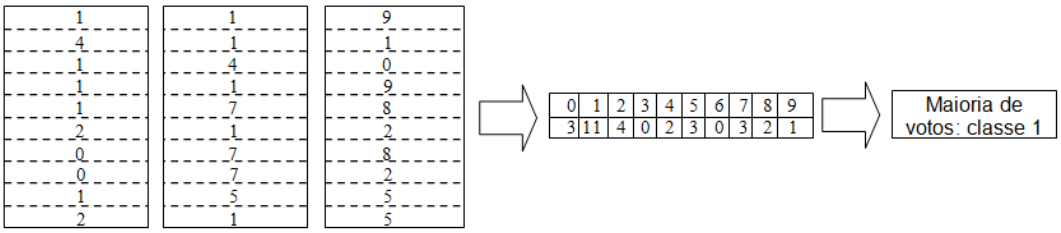


Figura 6. Votação para a decisão final.

Este esquema de votação é frequentemente empregado em trabalhos similares, que envolvem o reconhecimento de gêneros musicais. Em geral, ele oferece bons resultados, já que é bastante comum que partes de uma música sejam parecidas com um gênero ao qual ela não pertence. O esquema de votação ajuda a evitar algumas confusões que poderiam ser causadas por este motivo, já que toma a decisão com base na maioria das zonas criadas.

A próxima seção descreve detalhes acerca da combinação deste classificador com outro baseado na seleção de instâncias de vetores com catacterísticas de tempo curto e características de baixo nível do sinal de áudio de músicas.

4. Resultados Experimentais

A LMD tem como característica o fato de reunir muitos gêneros com significativa similaridade entre si no que diz respeito a instrumentalização, estrutura rítmica e conteúdo harmônico. Isto acontece porque muitos gêneros presentes na base são originários de um mesmo país ou de países com grandes semelhanças no que diz respeito a aspectos culturais. Este fato faz com que a tentativa de discriminar tais gêneros automaticamente seja particularmente desafiadora. As próximas subseções descrevem os resultados obtidos.

4.1. Resultados obtidos com o classificador proposto

A tabela 1 mostra o percentual médio de acerto obtido por gênero com a estratégia de classificação aqui apresentada, seus respectivos desvios padrão e a média geral. Considerando que este trabalho tem como objetivo verificar a complementaridade de um método de classificação baseado em características particulares em relação a outras já conhecidas, não houve preocupação voltada à realização de testes estatísticos mais apurados que permitissem uma efetiva comparação com resultados já apresentados na literatura.

A tabela 2 mostra a matriz de confusão acumulada após a execução da validação cruzada entre os 3 *datasets*. A matriz de confusão mostra o número de vezes que os títulos musicais pertencentes aos gêneros identificados nas linhas foram classificados em cada um dos gêneros identificados pelas colunas. A diagonal principal da matriz corresponde aos casos em que houve acerto. Nesta matriz pode-se perceber a presença de uma taxa de confusão especialmente acentuada entre os gêneros de origem brasileira. É importante lembrar que o número total de títulos pertencentes a cada gênero, considerando os três *datasets* envolvidos nos experimentos, é igual a 90.

Gênero	**Taxa de acerto**	σ
axé	73,33%	8,82
bachata	82,22%	15,03
bolero	64,44%	8,39
forró	65,56%	8,39
gaúcha	35,56%	5,09
merengue	80,00%	6,67
pagode	46,67%	17,64
salsa	42,22%	6,94
sertaneja	17,78%	6,94
tango	93,33%	6,67
geral	**60,11%**	**9,06**

Tabela 1. Desempenhos médios por gênero e seus respectivos desvios padrão

	axé	bac	bol	for	gaú	mer	pag	sal	ser	tan
axé	66	2	3	0	0	3	8	0	3	5
bachata	1	74	4	1	0	2	4	2	0	2
bolero	1	4	58	7	2	0	5	0	0	13
forró	0	0	8	59	5	0	7	4	4	3
gaúcha	20	2	10	17	32	1	1	3	1	3
merengue	2	2	3	1	2	72	3	3	0	2
pagode	12	0	15	6	1	0	42	5	3	6
salsa	2	6	15	8	1	11	5	38	2	2
sertaneja	31	2	21	10	3	0	3	2	16	2
tango	1	0	4	1	0	0	0	0	0	84

Tabela 2. Matriz de confusão acumulada depois da validação cruzada

De forma geral, os resultados obtidos nos experimentos mostram taxas de acerto comparáveis às do primeiro trabalho realizado com a LMD [Silla Jr et al. 2008]. É importante observar que naquele trabalho não foi empregado o *"artist filter"* durante a divisão dos conjuntos para posterior classificação. Além disso, os experimentos aqui descritos apresentaram uma taxa de acerto média ligeiramente superior quando comparados com outros trabalhos recentemente realizados sobre a mesma base de dados, como [Lopes et al. 2010]. Entretanto, houve uma redução significativa do desvio padrão relacionado a taxa média de acerto entre os diferentes gêneros envolvidos no estudo.

4.2. Resultados obtidos com a combinação de classificadores

A fim de verificar em que medida os atributos descritores extraídos com o método aqui proposto podem ser complementares com os de abordagens mais tradicionais, realizou-se a combinação das saídas deste classificador com as do proposto em [Lopes et al. 2010]. Antes de descrever maiores detalhes acerca da combinação realizada, é válido ressaltar que os *datasets* empregados nos experimentos descritos neste trabalho foram compostos exatamente pelas mesmas músicas presentes nos *datasets* utilizados nos experimentos descritos em [Lopes et al. 2010]. Assim, entende-se que os resultados podem ser comparados e combinados de forma consistente e correta.

Para realizar a combinação foi necessário associar à saída dos classificadores probabilidades que indicam o grau de confiabilidade da mesma. Para o classificador aqui proposto (classificador 1), baseado em características extraídas dos espectrogramas, a probabilidade associada à cada gênero foi encontrada através da divisão do número de zonas atribuídas à este gênero pelo número total de zonas associadas a cada música.

Em [Lopes et al. 2010], o número total de instâncias extraídas de cada música no processo de classificação proposto é igual a 646. Para o cálculo das probabilidades associadas à cada gênero neste classificador (classificador 2), tomou-se o número de instâncias atribuídas ao gênero e dividiu-se por 646.

Foram testadas duas diferentes regras de combinação para atribuir a classificação final a cada título musical, as regras MAX e SUM [Kittler et al. 2002]. Na primeira, quando houve divergência entre as classes atribuídas pelos dois classificadores, foi estabelecida como classe aquela cujo classificador apresentou maior probabilidade associada à saída. Na segunda, em caso de divergência foi estabelecida como classe aquela cuja soma das probabilidades indicadas pelos dois classificadores apresentou maior valor. A figura 7 mostra o esquema empregado na combinação dos classificadores.

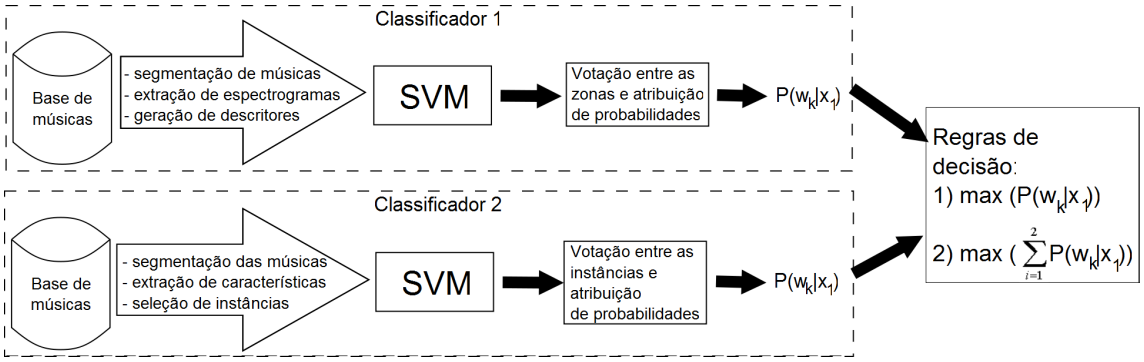


Figura 7. Esquema empregado na combinação dos classificadores.

A tabela 3 mostra as taxas médias de acerto obtidas por gênero e também a taxa geral quando foram empregadas as regras de decisão MAX e SUM.

Os resultados obtidos mostram que os atributos descritores propostos neste trabalho, baseados em características texturais extraídas de espectrogramas, não somente possuem algum poder de discriminação de gêneros musicais, como também possuem alguma complementaridade em relação a outros formatos de atributos descritores já propostos. Levando-se em consideração o fato de que a combinação dos classificadores apresentou as piores taxas médias de acerto nos gêneros brasileiros, o que já era esperado devido ao fato de que estes gêneros não possuem diferenças significativas em termos de instrumentalização, estrutura rítmica e conteúdo harmônico, acredita-se que este esquema pode proporcionar resultados ainda melhores em outras bases de músicas.

O desempenho médio geral de 66,11% de acerto, obtido com a regra de combinação SUM, é particularmente animador e com ele suscita naturalmente a hipótese de que outros esquemas de combinação ou a combinação com outros classificadores, diferentes dos aqui utilizados, pode trazer resultados ainda melhores. A tabela 4 mostra os limites superiores de acerto médios obtidos com os dois classificadores aqui empregados. Este resultado correponde à média entre as três diferetes situações em que cada

Gênero	Regra MAX		Regra SUM	
	Taxa de acerto	σ	Taxa de acerto	σ
axé	64,44%	19,53	74,44%	15,03
bachata	92,22%	6,94	90,00%	6,67
bolero	75,56%	15,40	77,78%	16,44
forró	44,44%	24,57	46,67%	26,03
gaúcha	47,78%	1,92	52,22%	3,85
merengue	85,56%	10,72	88,89%	8,39
pagode	56,67%	8,82	57,78%	1,92
salsa	47,78%	9,62	46,67%	12,02
sertaneja	36,67%	34,80	35,56%	24,11
tango	92,22%	6,94	91,11%	8,39
geral	**64,33%**	**13,93**	**66,11%**	**12,29**

Tabela 3. Desempenhos médios por gênero e geral na combinação dos classificadores com as regras MAX e SUM

um dos diferentes *datases* foi utilizado como conjunto de teste. O limite superior é encontrado considerando como corretamente classificados os itens em que pelo menos um dos classificadores envolvidos na combinação foi capaz de realizar a classificação corretamente. Com base nestes dados, que mostram que a média geral de acerto pode chegar a 75%, percebe-se que vale a pena investir em experimentos que envolvam a combinação dinâmica de classificadores.

Gênero	Taxa de acerto	σ
axé	81,11%	13,47
bachata	93,33%	6,67
bolero	81,11%	5,09
forró	66,67%	6,67
gaúcha	60,00%	12,02
merengue	88,89%	6,94
pagode	75,56%	3,85
salsa	62,22%	11,71
sertaneja	46,67%	17,64
tango	94,44%	5,09
geral	**75,00%**	**8,91**

Tabela 4. Limite superior combinando os dois classificadores

5. Conclusões e Trabalhos Futuros

O uso da LMD com o emprego de *"artist filter"* introduziu uma dificuldade adicional ao processo de classificação proposto neste trabalho. Mesmo assim, os resultados obtidos são comparáveis aos obtidos em outros trabalhos e, sob certos aspectos, ligeiramente superiores. A taxa de acerto média foi de 60,11% com desvio padrão de 9,06.

A combinação do classificador aqui proposto com um outro classificador, baseado num descritor de formato diferente, produziu resultados bastante animadores, 66,11% no

melhor caso (regra de combinação SUM). Este resultado mostra a presença de complementaridade do formato de descritor aqui proposto com outros já existentes. Existem boas perspectivas de se alcançar, em trabalhos futuros, taxas de acerto ainda melhores realizando a combinação com outros formatos de descritores ou mesmo utilizando outros esquemas de combinação com o mesmo classificador, já que o limite superior encontrado para esta combinação foi de 75%.

Trabalhos futuros envolvendo metaclasses também podem ser realizados, já que a matriz de confusão resultante dos experimentos mostra um alto índice de confusão especialmente entre músicas de gêneros brasileiros. A criação de metaclasses permitiria investir em técnicas especificamente voltadas a discriminação entre estes gêneros, buscando uma melhoria no desempenho final do classificador.

Adicionalmente, pretende-se aplicar os métodos aqui descritos em outras bases de músicas, a fim de identificar o comportamento destes métodos em outros cenários.

Referências

Chang, C.-C. and Lin, C.-J. (2001). *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Costa, C., Valle Jr, J., and Koerich, A. (2004). Automatic classification of audio data. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 562–567.

French, M. and Handy, R. (2007). Spectrograms: turning signals into pictures. *Journal of engineering technology*, 24(1):p. 32–35.

Gantz, J., Chute, C., Manfrediz, A., Minton, S., Reinsel, D., Schlichting, W., and Toncheva, A. (2008). The diverse and exploding digital universe: An updated forecast of worldwide information growth through 2011. *IDC white paper, sponsored by EMC*.

Haralick, R., Shanmugam, K., and Dinstein, I. (1973). Textural features for image classification. *IEEE Transactions on systems, man and cybernetics*, 3(6):p. 610–621.

Kittler, J., Hatef, M., Duin, R., and Matas, J. (2002). On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):p. 226–239.

Koerich, A. and Poitevin, C. (2005). Combination of Homogeneous Classifiers for Musical Genre Classification. *IEEE International Conference on Systems, Man, and Cybernetics*, pages 554–559.

Li, T., Ogihara, M., and Li, Q. (2003). A comparative study on content-based music genre classification. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 282–289. ACM.

Lopes, M., Gouyon, F., Koerich, A. L., and Oliveira, L. E. S. (2010). Selection of training instances for music genre classification. *ICPR 2010 - 20th International Conference on Pattern Recognition*.

Lucena, C., Medeiros, C., Lucchesi, C., Maldonado, J., Almeida, V., and outros (2006). Grandes Desafios da Pesquisa em Computação no Brasil - 2006-2016. *Relatório sobre o seminário Grandes Desafios da Pesquisa em Computação. São Paulo. SBC/CAPES/FAPESP.*

Pampalk, E., Flexer, A., and Widmer, G. (2005). Improvements of audio-based music similarity and genre classification. In *proc. ISMIR*, volume 5.

Silla Jr, C., Koerich, A., and Kaestner, C. (2008). The latin music database. In *Proceedings of the 9th International Conference on Music Information Retrieval*, pages 451–456.

Tzanetakis, G. and Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on speech and audio processing*, 10(5):p. 293–302.

Vapnik, V. (2000). *The nature of statistical learning theory*. Springer Verlag.

Yu, G. and Slotine, J. (2009). Audio classification from time-frequency texture. *ICASSP 2009 - International Conference on Acoustics, Speech and Signal Processing*, pages 1677–1680.

CiênciaBrasil - The Brazilian Portal of Science and Technology

Alberto H. F. Laender[1], Mirella M. Moro[1], Altigran S. Silva[2], Clodoveu A. Davis Jr.[1], Marcos André Gonçalves[1], Renata Galante[3], Allan J. C. Silva[1], Carolina A. S. Bigonha[1], Daniel Hasan Dalip[1], Eduardo M. Barbosa[1], Eduardo N. Borges[3], Eli Cortez[2], Peterson Procópio Jr.[1], Rafael Odon de Alencar[1], Thiago N. C. Cardoso[1], Thiago Salles[1]

[1]Departamento de Ciência da Computação
Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG

[2]Departamento de Ciência da Computação
Universidade Federal do Amazonas (UFAM), Manaus, AM

[3]Instituto de Informática
Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre, RS

{laender,mirella,clodoveu,mgoncalv,allan,carolb,hasan,
emb,peterson,odon,thiagon,tsalles}@dcc.ufmg.br,
{alti,eccv}@dcc.ufam.edu.br, {galante,enborges}@inf.ufrgs.br

Abstract. *Research social networks are a potentially useful resource for studying science and technology indicators from specific communities (e.g., a country). However, building and analyzing such networks beget challenges beyond those from regular social networks, since data about actors and their relationships are usually dispersed across various sources. In this paper, we present a research social network built from an individual perspective by gathering data from a Brazilian curricula vitae repository. We describe its architecture and the solutions adopted for data collection, extraction and deduplication, and for materializing and visualizing the network.*

1. Introduction

In July 2008, CNPq (the Brazilian National Council for Scientific and Technological Development) lauched a research program called National Institutes of Science and Technology (or simply INCT)[1]. The INCT program has become a powerful initiative for promoting science, technology and innovation in the country. Its main goal is to mobilize and aggregate in networks the best research groups in frontier areas, which cover a wide range of topics, including agribusiness, health, energy, nanotechnology, computing and social sciences. Among those institutes, the INWeb[2] (INCT for the Web) focuses on cutting edge research, technology transfer, education of highly qualified human resources and dissemination of science and technology on Web-related topics. One of its projects, *CiênciaBrasil* – the *Brazilian Portal of Science and Technology*[3] – consists of building and analyzing a *Research Social Network* that allows studying science and technology indicators (knowledge production, research collaboration, human resources formation, technology transfer etc.) involving Brazilian researchers within the INCT program.

[1]INCT Program http://www.cnpq.br/programas/inct/_apresentacao
[2]InWeb http://www.inweb.org.br
[3]CiênciaBrasil http://pbct.inweb.org.br

Gathering and processing information for building and analyzing research networks beget challenges that go beyond those of regular social networks, such as Facebook and MySpace[4] (see [Tang et.al 2008]). The data is usually not provided by the network members themselves (i.e., researchers) but must be collected and extracted from various sources (including the Web). Moreover, the existing relationships among these members are generally implicitly embedded in the data and must be derived by processing them (e.g., using sophisticated automatic extraction techniques). Even if the data can be correctly obtained, data duplication and (name) ambiguity may occur since such data is usually collected from several heterogeneous sources. In order to process such a network with a minimum of quality, we need to deal with those issues as well as others associated with the visualization, navigation and analysis of specific instances of the network. Examples of Web applications that exploit academic networks and face those challenges are ArnetMiner[5] and Microsoft Academic Search[6].

In Brazil, *Lattes*[7] is a Web platform made available by CNPq for storing, managing and searching for curricula vitae of researchers involved with Brazilian institutions. With Lattes, researchers (from undergrads to seniors) inform their academic achievements including education, professional history, publications, funding, academic positions, advising, awards, etc. Lattes has been recognized as one of "the cleanest researcher databases in existence" [Lane 2010]. With such organized data, Lattes became the natural source for building *CiênciaBrasil.* Indeed, the existence of a standardized platform for curricula vitae publication provides an opportunity for building a large, *individually centered* repository of scientific and educational information. From the individual perspective, it is possible to put together research groups and associations, as well as to build a fairly reliable research social network covering several knowledge fields.

However, obtaining and organizing data from Lattes is a challenge per se. When using Lattes, researchers have to fill out a form with predefined fields to include information about their scientific production. This is done in an isolated way and, when published as a Web page, there are no clear delimiters to (for example) easily extracting the desired fields from publication citations (e.g., coauthor names, title, venue). Hence, it requires sophisticated extraction strategies. Moreover, since several coauthors include the same publications in their curricula vitae, group and network statistics require a *deduplication* step in order to generate precise production figures. Deduplication is known to be a difficult problem [Carvalho et al. 2008, Geer 2008, Sarawagi and Bhamidipaty 2002], which happens not only due to possible errors in filling out forms, but also due to non-standardized inclusion of coauthor information.

In this paper we introduce *CiênciaBrasil* and provide an overview of our solutions for some of the aforementioned challenges. We also illustrate the potential of *CiênciaBrasil* as a platform for large scale academic research evaluation and analysis. Furthermore, all challenges presented in this work are related to two of SBC's Grand Challenges in Computer Science Research in Brazil[8]. Specifically, it is related to the first

[4]Facebook http://www.facebook.com; MySpace http://www.myspace.com
[5]ArnetMiner http://www.arnetminer.com
[6]Microsoft Academic Search http://academic.research.microsoft.com
[7]Lattes http://lattes.cnpq.br/english
[8]SBC http://www.sbc.org.br

challenge ("Management of information over massive volumes of distributed multimedia data") due to the big volume of data extracted and evaluated for building the networks and to the fourth challenge ("Participative and universal access to knowledge for the Brazilian citizen") since the visualization and analysis of the research networks will allow to identify and disseminate information that is related to the Brazilian research, bringing it closer to the regular (i.e., non-scientist) citizen.

The contributions of this paper can be then summarized as follows. After discussing related work (Section 2), we introduce an architecture for building academic networks from an individual perspective. In a nutshell, with such an architecture, we collect and treat data from individuals and then build the network. Also, we use specific algorithms for collecting and extracting data as well as deduplicating it. Then, we build the research network and provide different ways to visualize it (Section 3). We also present the major features of *CiênciaBrasil* and discuss examples of different visualization resources provided on its portal (Section 4).

2. Related Work

This paper presents *CiênciaBrasil*, a Web portal for analyzing and viewing information on science and technology. Specifically, *CiênciaBrasil* gathers data from Brazilian researchers, integrates them and generates distinct individual and group-based visualizations that can aid on different analysis and evaluation processes. In order to generate such views, *CiênciaBrasil* employs social networks techniques that are tailored for the research context. This section discusses some tools that are similar or that employ concepts from social networks.

ArnetMiner [Tang et.al 2008] provides online search and mining services for researcher social networks. It creates profiles for researchers based on Web data that includes bibliographic data and researcher's data from multiple sources. With the networks, it also allows to discover patterns, to find experts, conferences and papers on specific topics, and to rank people based on their research achievements.

Microsoft Academic Search is based on the concept of object-level vertical search [Nie et al. 2005] and provides different ways to explore scientific papers, conferences, journals, and authors. Specifically for authors (the focus of our work), it shows a profile page with affiliation, publications, citations, g-index, h-index, coauthors, and coauthorship graph. The data is acquired from Web sources including DBLP, ACM Digital Library, CiteSeer, among others.

Both ArnetMiner and MS Academic Search extract researchers' profiles by integrating publication information, for example from digital libraries. Then, they build academic networks and provide search services. However, extracting such data from the Web and integrating them can be error prone: ArnetMiner allows registered users to correct errors through an editing interface and Microsoft Academic Research has serious problems with name disambiguation. For example, our coauthor Clodoveu Davis Jr. has *five* entries for different profiles (in March 2011).

SciVerse Scopus[9] is one of the largest abstract and citation data source available for research literature. Among its features, it has an author search service that provides

[9]Scopus http://www.scopus.com

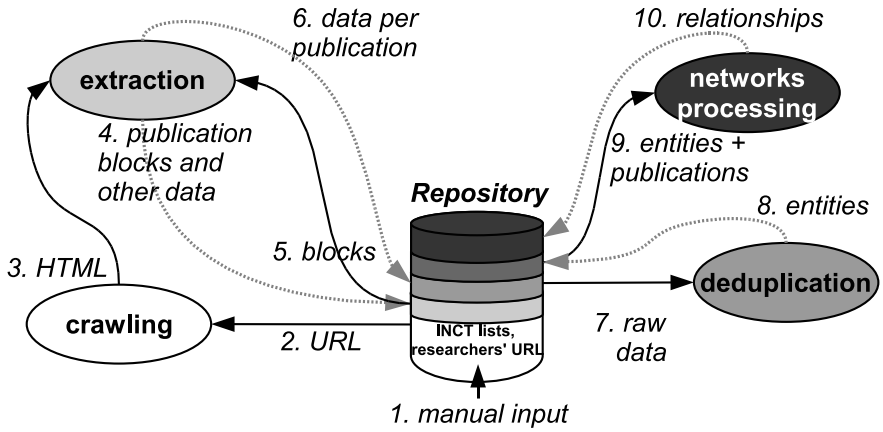


Figure 1. *CiênciaBrasil* Architecture

a full profile of authors, which includes name, affiliation, references, citations, h-index, coauthors, subject area, and history. SciVerse Scopus is similar to *CiênciaBrasil* in the sense that it builds author profiles based on its repository. However, its repository is actually a collection of external sources, and it incurs on problems such as disambiguation (two different people with same name) and deduplication (the same person with two different names). For example, our coauthor Alberto H. F. Laender has *three* entries, even though it correctly found three forms of writing his name (Alberto H. F. Laender, A. H. F. Laender and Alberto Laender). Nonetheless, SciVerse Scopus also offers other services such as the Journal Analytics that are beyond *CiênciaBrasil* scope.

3. *CiênciaBrasil* Architecture

This section introduces the general architecture employed by *CiênciaBrasil* to build a research social network from an individual perspective. In a nutshell, with this architecture, information from individuals is collected and treated to, then, build the network based on it. Specifically, the whole process starts with the manual data input of the INCTs basic data (Arrow 1 in Figure 1). The basic data is composed of URLs to the Lattes curricula vitae of the researchers. From the URLs, the curricula vitae pages are collected (Arrows 2 and 3). The process follows by extracting and organizing the necessary information (Arrows 4 to 6), which then is deduplicated (Arrows 7 and 8) and disambiguated (Arrow 9) before building the network (Arrow 10). Next, we present the technical and scientific details of each process within the architecture.

3.1. Crawling

The first process is to manually input the basic data from each INCT into the repository. The data include the name, acronym, principal investigators, home institution, other collaborating institutions, etc, and the list of Lattes URLs of its participating researchers (Arrow 1 in Figure 1). The first version of *CiênciaBrasil* covers 4,676 researchers from 123 INCTs. The input list of URLs is then fed to the crawling process (Arrow 2 in Figure 1), for collecting the respective curricula vitae from the Lattes platform.

The crawling process applies an asynchronous concurrent download strategy to increase throughput (e.g., due to the high latency observed when establishing/negotiating HTTP connections, handling several connections simultaneously ultimately maximizes

bandwidth usage and increases the number of crawled curricula vitae's per time unit). It also keeps track of download errors to avoid any possible data loss [Heydon and Najork 1999]. After crawling, the gathered data is available in pure HTML format (Arrow 3 in Figure 1), which will be further decomposed in finer-grained structures called *information blocks*.

3.2. Data Extraction

The data extraction process involves two steps. First, it extracts *information blocks* from the HTML pages, i.e., blocks containing relevant information for building the research social network. Then, it extracts specific finer-grain data from the information blocks, such as publication metadata (e.g., author names, title, venue, etc.).

In the first step, despite the standardized structure of the Lattes curriculum vitae (with predefined fields filled by the researchers) there are no explicit delimiters to guide the extraction process. For example, Figure 2(a) illustrates a piece of the HTML page extracted from a specific curriculum vitae. While some unrelated fields are grouped by the same HTML markup, there are freely formatted fields that need additional parsing (e.g., the researchers' address). This fact, along with the existence of optional fields, represents a major challenge for correctly obtaining the information blocks.

For addressing the optional fields, we have crafted a number of regular expressions for discovering and grouping related information into blocks. The regular expressions are dynamically generated and processed by a finite state machine. Also, the expressions are based on the tag names within the HTML file. Most of these blocks roughly correspond to sections of a curriculum vitae, such as affiliation, education, publications, professional history, etc. For handling freely formated fields, we have adopted some heuristics to extract the largest possible amount of information from them. Extracted blocks are then stored in the *CiênciaBrasil* Repository (a relational database) for further processing (Arrow 4 in Figure 1). From the 4,676 Lattes curricula vitae processed in this first step, it extracted 338,968 information blocks with publication data.

In the second step, it gets information blocks (Arrow 5 in Figure 1) and extracts detailed data (Arrow 6 in Figure 1) in order to build the research social network. We could build such a network based on advisor-advisee relationship [Wang et.al 2010], affiliation [Beauchesne 2011], coauthorship [Lopes et al. 2010] and so on. Given the wide range of areas covered by the INCTs, we decided that coauthorship would be the best relationship to build the network (since some of the researchers did their PhDs in international institutions and have changed affiliations over time). Therefore, *CiênciaBrasil* network is based on coauthorship relationships derived from publication metadata extracted from the corresponding information blocks.

As an example, consider the information block shown in Figure 2(b) which contains a publication citation previously extracted from an actual Lattes curriculum vitae. Figure 2(c) shows the result of the second step of the extraction process. This step employs ONDUX [Cortez et al. 2010], an on-demand unsupervised method for text extraction that relies on a knowledge base to perform the extraction process. We chose to use ONDUX because it achieves good extraction results without any user intervention and easily adapts to scenarios in which the input text does not have any standardization.

```
<img src="images/curriculo/ico_doi.gif"></a>Martins, Waister ;
<a href="http://lattes.cnpq.br/3457219624656691" target="blank"
class="coautor">Gon\c{c}alves, Marcos</a> ; Laender, Alberto H. F. ;
<a href="http://lattes.cnpq.br/3527197809276361" target="blank"
class="coautor">Ziviani, Nivio</a> . Assessing the quality of scientific
conferences based on bibliographic citations. Scientometrics <sup>
<img id="01389130_1" src="/buscatextual/images/curriculo/jcr.gif"
class="tipAjax" tooltip="&lt;span&gt;SCIENTOMETRICS (PRINT)
(ISSN 0138-9130) &lt;BR/&gt;Fator de Impacto (JCR 2009): 2.167
&lt;/span&gt;" style=""></sup>, v. 83, p. 133-155, 2010.
```

(a)

Martins, Waister ; Gonçalves, Marcos; Laender, Alberto H. F. ; Ziviani, Nivio . Assessing the quality of scientific conferences based on bibliographic citations. Scientometrics , v. 83, p. 133-155, 2010.

(b)

Title: Assessing the quality of scientific conferences based on bibliographic citations
Authors: Martins, Waister ; Gonçalves, Marcos ; Laender, Alberto H. F. ; Ziviani, Nivio
Venue: Scientometrics
Year: 2010

(c)

Figure 2. Publication metadata extraction

3.3. Data Deduplication

As *CiênciaBrasil* processes data coming from individual curricula vitae (Arrow 7 in Figure 1), each coauthored publication is likely to be listed in more than one Lattes curricula vitae. However, not all of these occurrences are exact duplicates. For instance, author names may appear in different order or differently abbreviated. Moreover, in some cases, even the publication title might be differently written: besides having typos, it is very common to change the paper title between acceptance and publication, so one author may add the original title while another uses the new one. Not considering such duplicates may interfere with derived statistics and further analysis carried out with the stored data.

Hence, it is necessary to carry out a *deduplication* process over all publication entries. We adopt an unsupervised heuristics-based method [Borges et al. 2011] (developed in the context of InWeb). Such a method discards duplicates in which authors' initials and year diverge, thus avoiding unnecessary comparisons. Furthermore, to perform efficiently, the data deduplication process is parallelized using a map-reduce framework[10], which allows new machines to be easily added to improve time performance.

Notice that *nearly-duplicated* publications represent a great challenge for deduplication. In most cases, they consist of publications that share most of the authors, have similar titles and were published in or close to the same year. For example, the article "Infectious bronchitis virus isolate *IBV/Brasil/200/1981* nucleocapsid protein (N) gene" and "Infectious bronchitis virus isolate *IBV/Brasil/PM1/1987* nucleocapsid protein (N) gene" have the same list of coauthors and were published in the same year. That is a simple example that characterizes nearly-duplicated publications as an open problem and on-going work in our research group.

[10]Hadoop Pig: http://pig.apache.org

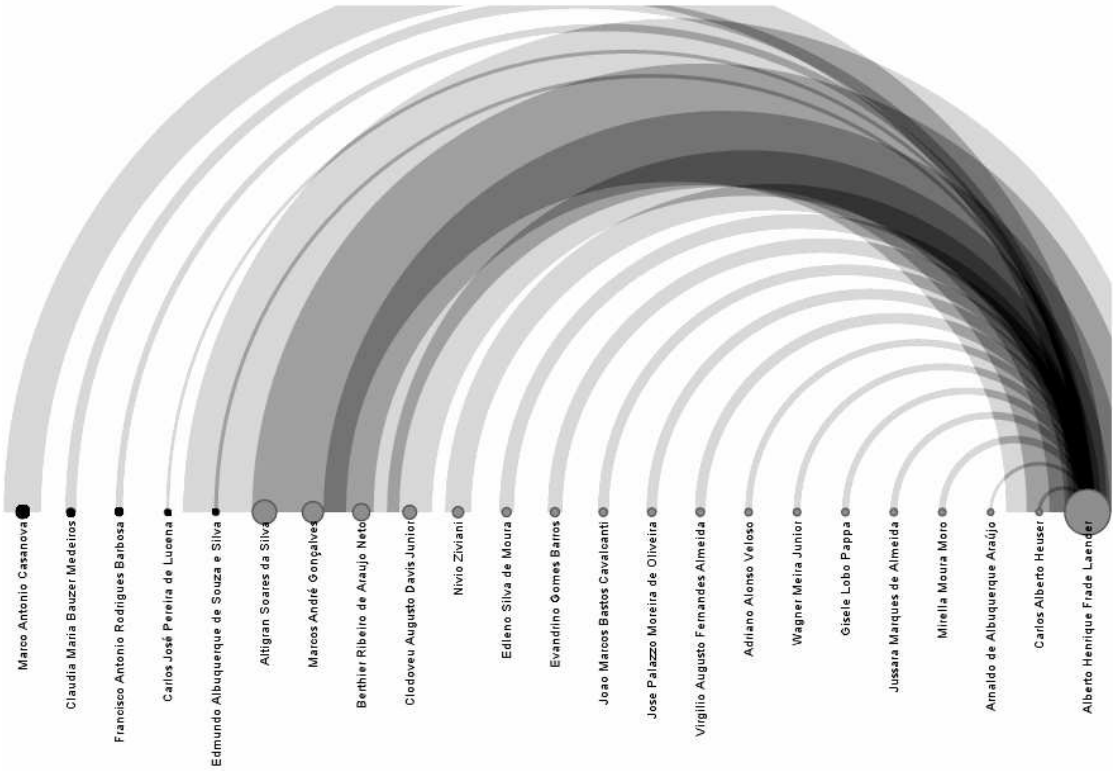


Figure 3. Author graph example

For assessing the effectiveness of the deduplication process, we have analyzed three random subsets of data with 1,507 publications, in which replicas were manually identified. This process has achieved 92% of precision (few false positives have been found) and 100% of recall (all replicas have been found). Therefore, the method from [Borges et al. 2011] does indeed offers an effective solution for handling duplicates on *CiênciaBrasil.* The data deduplication process was able to identify 288,867 unique citations of publications authored by INCT researchers. These unique citations were then stored back into the *CiênciaBrasil* Repository.

3.4. Network Materialization

After the deduplication process, the collected data is ready for materializing the network. This is done by deriving coauthorship relationships based on the existence of common publications in the researchers' curriculum vitae (Arrows 9 and 10 in Figure 1). An effective way to visualize such a network is through graphs. In particular, two different graphs are built: one on authors and one on collaborations within an INCT. Author graphs are built with the perspective of an *individual* author, as illustrated in Figure 3. This way, each researcher that has collaborated with such an author is represented by a colored circle and positioned in a horizontal line (darker circles represent coautho from a different INCT). An arc represents a collaboration between different authors and its thickness represents the strength of this collaboration (number of publications with both of them as coauthors). It is very important to notice that, for building this graph, disambiguation is not a problem since each connection in the graph relies on the existence of a same publication on the curricula vitae of the two related researchers.

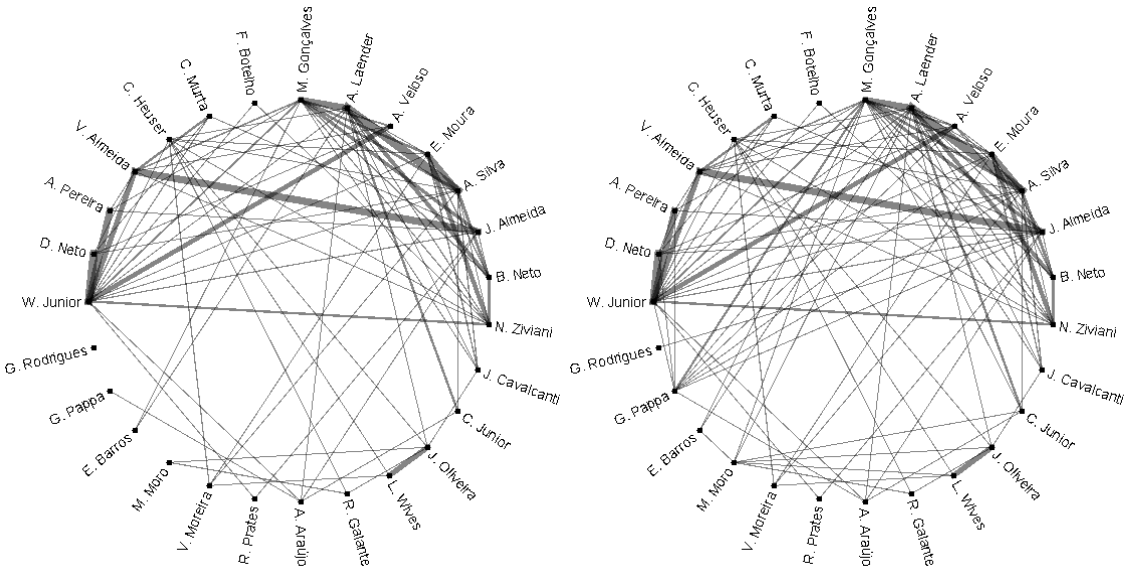


Figure 4. INCT collaboration graph evolution from 2008 (left) to 2010 (right)

The INCT graph shows the collaboration between researchers from the same INCT, as illustrated in Figure 4. Similar to the author graph, one arc represents the collaboration between two authors and its thickness the strength of this collaboration. In this kind of graph, the researchers are presented as points arranged in a wheel format. This network visualization provides an interesting view of coauthorship evolution through time as we can see in Figure 4 by comparing the density of the graphs from 2008 and 2010. On the website, those two graphs are shown as a single one in which blue edges indicate intensified collaborations and yellow edges new collaborations in the period. We are currently working on presenting yearly graphs on the website. Also, the wheel graph style is the same used in an InWeb related work [Lopes et al. 2010].

4. *CiênciaBrasil* Features

The CiênciaBrasil front-end was developed as a web application and written in Django, a Python framework for agile web development[11]. The basic list of features includes the initial webpage, browsing and visualization functions for exploring researchers and INCTs, and research keyword search box. Specifically, the *initial webpage* has explanatory texts about the project's goal, members, context and so on, as illustrated in Figure 5. This page also presents the date when data was last collected as well as the size of the current repository (i.e., number of INCTs, researchers and publications covered).

The user can search for a researcher's name and view details of her curriculum vitae as long as summarized data about her academic production. For example, Figure 6 illustrates the types of publication that a given researcher has in two granularities: by year and total. Specifically, the top part has a line chart for the annual production separated by type (i.e., event papers, book chapters, journal articles, magazine articles and others). The bottom part has a pie chart for the whole production. The researcher page has also a tag cloud based on the titles of her production (note that the clouds are further explained

[11]Django http://www.djangoproject.com



Figure 5. *CiênciaBrasil* main interface

in the next paragraphs). It is very important to notice that the charts are all interactive, i.e. the user can emphasize different parts of the graphs by clicking on them. All charts were created using Highcharts[12], a charting library written in pure JavaScript that allows to add interactive charts in a web application.

The user can also search for an INCT's name and view its detailed information, which includes general information (i.e., INCT coordinator and associated institutions), some statistics and tag clouds. The statistics include a graphic on the percentage of publication types per year. For each year, it sums up the publications of all researchers (within the INCT) grouped by category (or type). The *y* axis shows the percentage and the *x* axis the year. The goal of such statistics is to synthesize the amount and type of the researchers' publications, which makes easier to analyze and evaluate their production.

With such visualizations, it is also possible to compare not only the INCT's productions against themselves but also against areas. For example, Figure 7 shows the accumulated publications grouped by types for two different INCTs: one in software engineering and another in social studies. Notice that with such visualizations, we

[12]Highcharts http://www.highcharts.com/

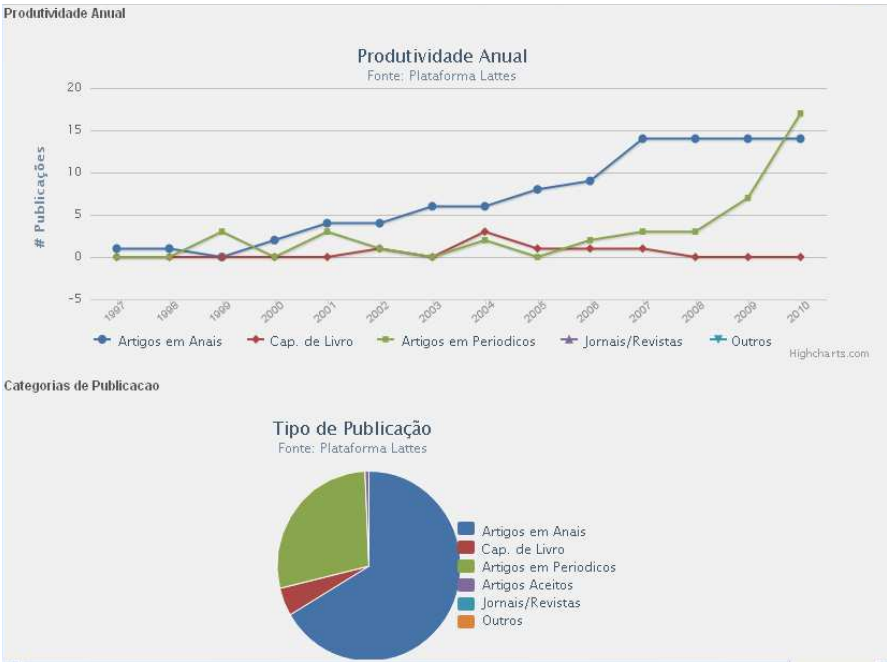


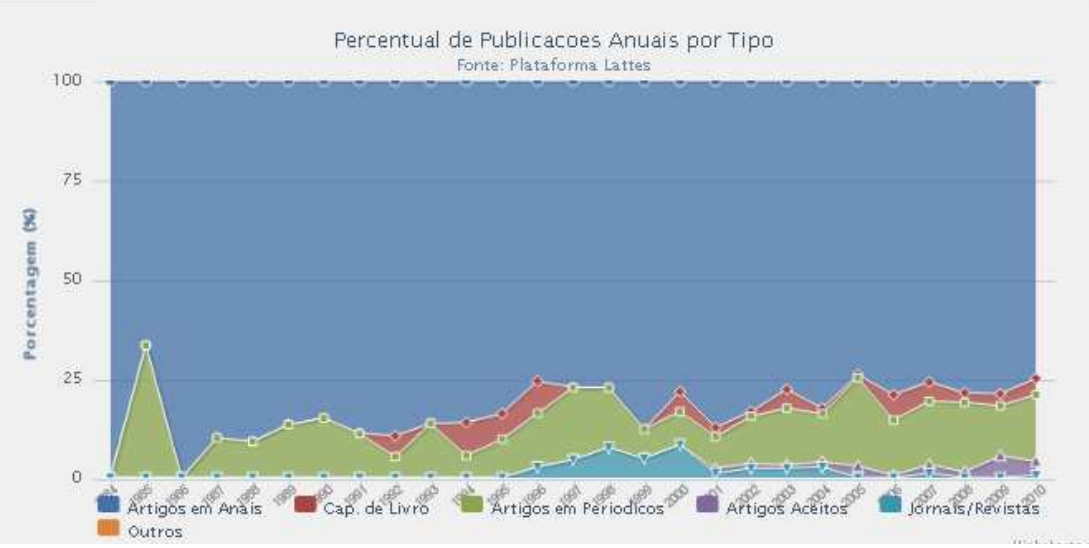
Figure 6. *CiênciaBrasil* detailed statistics on a researcher

can (clearly) see that most of the software engineering publications are on conferences, while the social ones are on journals and book chapters. This once more corroborates the findings from [Laender et.al 2008, Menezes et al. 2009], which shows (using other techniques) that papers on conferences and similar events dominates Computer Science publications.

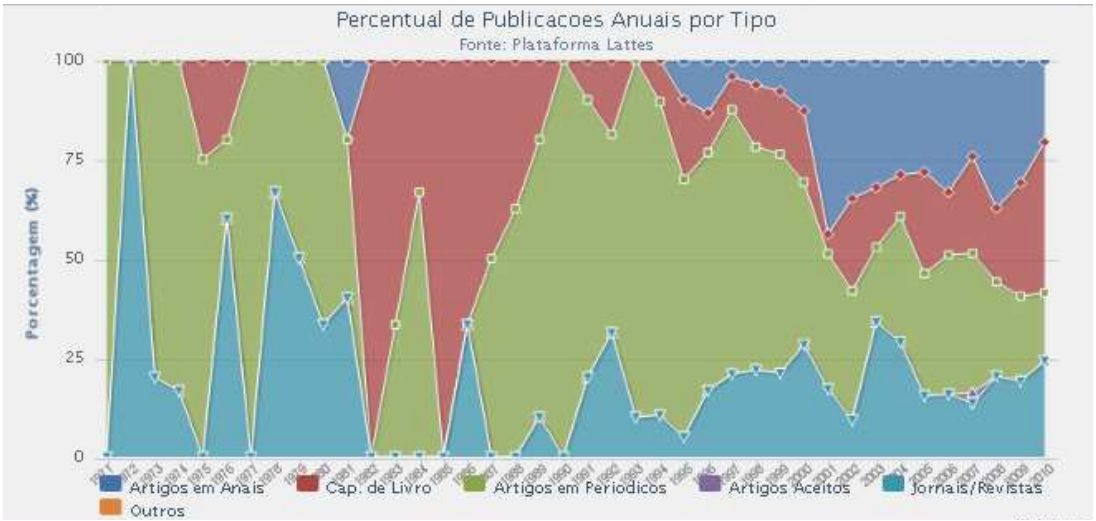
A different perspective about the publications of INCTs and researchers is provided by tag clouds. A tag cloud depicts important terms that characterize a document (often a web page). In general, a tag is usually a single word and its importance to a document defines the size or the color in which it appears in the cloud. However, in scientific documents, it is very common to have expressions besides single words. For example, in a document, you may have the single words “software” and “engineering”, as well as the expression “software engineering”, which has a totally different meaning from its single word components. Needless to say, such expressions are not found only in Computer Science, but in practically all fields. Take for example “homo sapiens”, “sweat equity”, “CT scanning”, “habeas corpus”, “relativity theory” and many others.

In our case, we extract tags from titles of scientific production by applying a number of well known feature-selection heuristics. Such heuristics are based not only on the frequency of an expression but also on it representativeness for a particular researcher or INCT [Koutrika et al. 2009]. In order to allow identifying multi-word expressions we also extract noun phrases and n-grams.

For example, Figure 8 illustrates the tag cloud for the titles of publications authored by members of the INCT on Integrated Studies of the Amazon Biodiversity. The cloud shows words for cities and states covered by the Amazon rainforest (e.g., Amazonas, Brazilian Amazon, estado do Amazonas, Rio Branco, Roraima), species, order, genus, class and family of fauna and flora that are studied over there (e.g., anura,



(a) INCT for Software Engineering



(b) INCT for Studies about the USA

Figure 7. *CiênciaBrasil* statistics on the publications of two different INCTs

apidae, arachnida, bactris gasipaes, Solanum sessiliflorum Dunal) as well as other specific expressions such as “floresta de terra firme” e “abelhas com ferrão”.

Finally, a general keyword search box is available at the top of every page. Its result is a list of both INCTs and researchers whose names approximately match the given keyword. This feature is powered by Xapian[13], an open source search engine library written in C++ with binding for many programming languages. In order to integrate Xapian with *CiênciaBrasil*, we used a modular search extension for Django called Haystack[14], which also was used for indexing. Such an index is necessary in order to speed searches up. Specifically, all INCTs and researchers are described by text documents containing their names, and those documents are used as the target data for

[13]Xapian http://xapian.org
[14]Haystack Search http://haystacksearch.org

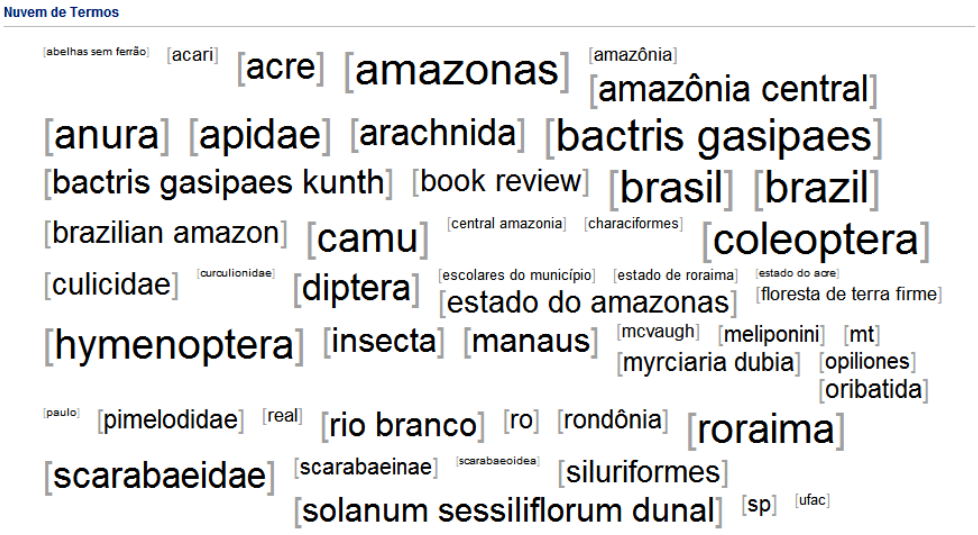


Figure 8. Tag cloud for an INCT on Integrated Studies of the Amazon Biodiversity

indexing. As the index is created just once, we need to perform a re-indexing every time we load new data.

5. Concluding Remarks

Summary. This paper presented an overview of *CiênciaBrasil*, a Web portal for viewing and analyzing a research social network formed by researchers within the Brazilian INCT program. *CiênciaBrasil* has been built from the individual perspective of each INCT researcher by using the Lattes platform as data source. Although the Lattes platform provides a unified and very clean data source for building such a research social network, a number of challenging problems still must be faced. Some of those problems and their solutions are summarized in Table 1.

Future Work. Although *CiênciaBrasil* has been developed as a test platform for experimenting with some of the methods and tools we have developed for web crawling, data extraction and data deduplication within INWeb, we expect that it might also be used as a platform for analyzing and evaluating the results and evolution of the INCT program. To provide more flexibility for such an important task, a major challenge remains: to keep *CiênciaBrasil* up-to-date with Lattes since it is a very dynamic data source. Every day, hundreds of researchers update or create their curricula vitae on Lattes. At the moment, updating the *CiênciaBrasil* Repository requires redoing all the steps described in Figure 1, which is very time consuming. Thus, a more efficient strategy must be devised to maintain *CiênciaBrasil* synchronized as much as possible with the Lattes content.

There are many different ways to improve, explore and visualize the information given by *CiênciaBrasil*. For improving it, we could also consider information about the publications that is available on other publication-oriented websites, such as Google Scholar[15] and CiteSeer[16]. For example, we could extend the publication analysis to

[15]Google Scholar http://scholar.google.com/
[16]Citeseer http://citeseer.ist.psu.edu/

Table 1. Summary of challenges and their solutions

Challenge	Solution
Gather reliable data from individuals for building the research network	Lattes platform was employed as data source
Collecting the specific Lattes curricula (from the set of researchers within INCTs)	Manual input of curriculum vitae's URL followed by a crawling process
Extracting data from specific Lattes fields (since there are no clear delimiters among them)	Regular expressions and ONDUX [Cortez et al. 2010], an on-demand unsupervised method for text extraction
Deduplication of citation records extracted from the individual curricula vitae	An unsupervised heuristics-based method proposed in [Borges et al. 2011]
Searching for groups and individuals on the portal	Xapian, an open source search engine library, and Haystack, a modular search extension for Django
Providing ways of the user to interact with the information	Highcharts, a charting library for interactive charts in a web application
Visualizing the research social network	Building researchers-oriented graphs based on the researchers coauthorships
Visualizing aggregated information on individual and group production	Grouping data for statistics on interactive charts
Building and visualizing tag clouds for word expressions	We extract tags from titles by applying well known feature-selection heuristics.

consider the number of citations of each paper. In the exploration front, we are currently working on adding new social network metrics for providing different analysis. For example, the work on [Lopes et al. 2010] (also developed within InWeb) defines new metrics that could bring some light on different facets of coauthorship relationships. In the visualization front, we are currently working on expanding the views one can get from the network. For example, we plan to add both temporal and geographical dimensions to the author and INCT graphs.

Acknowledgements

This work is partially supported by INWeb (MCT/CNPq grant 57.3871/2008-6) and by the authors' individual grants and scholarships from CNPq, CAPES and FAPEMIG.

References

Beauchesne, O. H. (2011). Map of scientific collaboration between researchers. http://olihb.com/2011/01/23/map-of-scientific-collaboration-between-researchers.

Borges, E. N., Carvalho, M. G., Galante, R., Gonçalves, M. A., and Laender, A. H. F. (2011). An Unsupervised Heuristic-based Approach for Bibliographic Metadata Deduplication. *Information Processing & Management*, Accepted for publication.

Carvalho, M. G., Laender, A. H. F., Gonçalves, M. A., and da Silva, A. S. (2008). Replica identification using genetic programming. In *Procs. of SAC - ACM Symposium on Applied Computing*, pages 1801–1806, Fortaleza, Brazil.

Cortez, E., da Silva, A. S., Gonçalves, M. A., and de Moura, E. S. (2010). ONDUX: on-demand unsupervised learning for information extraction. In *Procs. of SIGMOD Conference*, pages 807–818, Indianapolis, USA.

Geer, D. (2008). Reducing the Storage Burden via Data Deduplication. *Computer*, 41(12):15 –17.

Heydon, A. and Najork, M. (1999). Mercator: A Scalable, Extensible Web Crawler. *World Wide Web*, 2(4):219–229.

Koutrika, G., Zadeh, Z. M., and Garcia-Molina, H. (2009). Data clouds: summarizing keyword search results over structured data. In *Procs. of EDBT - Intl. Conf. on Extending Database Technology*, pages 391–402, Saint-Petersburg, Russia.

Laender et.al, A. H. F. (2008). Assessing the research and education quality of the top Brazilian Computer Science graduate programs. *SIGCSE Bulletin*, 40(2):135–145.

Lane, J. (2010). Let's make science metrics more scientific. *Nature*, 464(7288):488–489.

Lopes, G. R., Moro, M. M., Wives, L. K., and de Oliveira, J. P. M. (2010). Cooperative Authorship Social Network. In *Procs. of AMW - Alberto Mendelzon Workshop on Foundations of Databases*, Buenos Aires, Argentina.

Menezes, G. V., Ziviani, N., Laender, A. H. F., and Almeida, V. A. F. (2009). A Geographical Analysis of Knowledge Production in Computer Science. In *Procs. of WWW - International World Wide Web Conference*, pages 1041–1050, Madrid, Spain.

Nie, Z., Zhang, Y., Wen, J.-R., and Ma, W.-Y. (2005). Object-level ranking: bringing order to Web objects. In *Procs. of WWW - International World Wide Web Conference*, pages 567–574, Chiba, Japan.

Sarawagi, S. and Bhamidipaty, A. (2002). Interactive deduplication using active learning. In *Procs. of KDD - ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 269–278, Edmonton, Canada.

Tang et.al, J. (2008). ArnetMiner: Extraction and Mining of Academic Social Networks. In *Procs. of KDD - ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 990–998, Las Vegas, USA.

Wang et.al, C. (2010). Mining advisor-advisee relationships from research publication networks. In *Procs. of KDD - ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 203–212, Washington, DC.

Identificando a Reputação de Pesquisadores Usando um Modelo de Perfil Adaptativo

Cristiano R. Cervi[1,2], Renata Galante[2], José Palazzo M. de Oliveira[2]

[1]Instituto de Ciências Exatas e Geociências – Universidade de Passo Fundo (UPF)
Caixa Postal 611 – 99.001-970 – Porto Alegre – RS – Brasil

[2]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

cervi@upf.br, {galante,palazzo}@inf.ufrgs.br

Abstract. *In this paper we address the problem of identifying the profile of researchers and measure their reputation in the area. We approached the process of specifying the profile using elements of the academic life of the researcher and the researcher classified by levels of reputation. For classification, some metrics were defined based on the balance of the trajectory of a researcher, involving the key elements of their curriculum. The elements do not only focus on production and citation of papers, but in the breadth of life science researcher's built over his career. The proposed model can be used in different contexts and different areas of knowledge, because it is extensive, flexible and adaptive.*

Resumo. *Neste trabalho abordamos o problema de identificar o perfil de pesquisadores e medir sua reputação na área de atuação. Abordamos o processo de especificação do perfil utilizando elementos da vida acadêmica do pesquisador, bem como classificamos o pesquisador por meio de níveis de reputação. Para a classificação, algumas métricas foram definidas baseadas no equilíbrio da trajetória de um pesquisador, envolvendo os principais elementos de seu currículo. Os elementos não focam apenas em produção e citação de artigos, mas sim na abrangência da vida científica do pesquisador construída ao longo de sua carreira. O modelo proposto neste trabalho pode ser utilizado em diferentes contextos e em diferentes áreas do conhecimento, pois é abrangente, flexível e adaptativo.*

1. Introdução

A área de modelagem de usuários (*User Modeling*) tem como objetivo descobrir conhecimento sobre o usuário em determinado assunto (*expert finding*) e de que forma representar esse conhecimento. Diversos sistemas e métodos (Fink e Kobsa, 2002; Middleton, Shadbolt e De Roure, 2004; Gauch et al. 2007; Maia, Almeida e Almeida, 2008) tem sido propostos para se modelar o perfil de um usuário, quer seja de forma explícita ou implícita e até mesmo da união das duas técnicas. A utilização desses sistemas e métodos se dá nas mais variadas áreas de aplicação, como por exemplo: (i) detecção de clientes em sites de vendas on-line; (ii) recomendação de produtos; (iii) recomendação de especialistas para participarem de defesas de mestrado e de doutorado; (iv) detecção de pesquisadores para formarem rede de colaboração científica; (v) organização de comitê de programa de conferência científica; (vi) organização de corpo editorial e revisores de periódicos, dentre outros.

A área de métricas científicas tem um papel importante na comunidade acadêmica, pois podem auxiliar no processo de medição da qualidade da produção científica e na identificação de especialistas em determinada área. As métricas recentes se baseiam fortemente nas citações de artigos de pesquisadores (Hirsch, 2005; Jin et al., 2007; Jin, 2007; Hirsch, 2009; Zang, 2009; Jin, 2007). Entretanto, essas formas de avaliação não consideram a trajetória do pesquisador, nem mesmo o cenário onde ele está inserido. As citações a artigos deveriam ser apenas um dos elementos a serem analisados e não o elemento exclusivo.

No contexto acadêmico, a identificação de um perfil pode ocorrer por meio de um processo de análise da trajetória de toda a carreira de um pesquisador. Tal processo envolve não somente aspectos relacionados a produção científica, como artigos e livros publicados, mas também por outros elementos inerentes a atividade de um pesquisador, como por exemplo: orientações de trabalhos de mestrado e de doutorado; participação em defesas de mestrado e de doutorado; trabalhos apresentados em conferências; participação em projetos de pesquisa, dentre outros.

Diante desse contexto, existe um campo de pesquisa em aberto, pois é necessária uma métrica que englobe toda a vida do pesquisador e que proporcione uma análise abrangente. Nesse trabalho tratamos o problema de identificar e medir a reputação de um pesquisador em sua área de atuação. Isto não é uma tarefa simples, uma vez que pode envolver visões diferentes, quando pesquisadores atuam em áreas de pesquisa diferentes, porém a principal vantagem é tornar nossa proposta adaptativa. Primeiramente é necessário definir quais dados são utilizados para avaliar a reputação de um pesquisador, como representar estes dados em um modelo (perfil) e onde encontrá-los. Após, é necessário definir uma métrica para se chegar a um índice para a reputação.

Este trabalho tem como objetivo especificar um modelo de perfil de pesquisador com dados obtidos mediante análise de sua trajetória científica e classificá-lo em um nível de reputação. O processo de modelagem do perfil envolve a identificação de quais dados são relevantes para a definição do perfil e os apresenta por meio de uma ontologia de domínio. O processo de classificação do pesquisador o enquadra em um nível de reputação definido através de um índice gerado por uma nova métrica proposta neste artigo. O modelo de perfil e a métrica foram definidos baseados no equilíbrio da trajetória de um pesquisador, envolvendo os principais elementos de seu currículo. Cabe ressaltar que tanto o perfil quanto as métricas não focam apenas em publicação de artigos ou citação, mas sim na abrangência da vida científica do pesquisador construída ao longo de sua carreira.

O restante deste artigo está dividido como segue. A seção 2 descreve os trabalhos relacionados. O perfil proposta através de uma ontologia de domínio é apresentado na seção 3. A seção 4 especifica a métrica para identificar a reputação e classificar os pesquisadores em níveis. Experimentos e análises de resultados são apresentados e discutidos na seção 5. Por fim, a seção 6 apresenta as considerações finais e perspectivas de trabalhos futuros.

2. Trabalhos Relacionados

Diversos trabalhos existentes na literatura abordam questões sobre identificação de perfis acadêmicos, produção científica de pesquisadores, identificação de especialistas baseados em dados científicos ou simplesmente análise de produção para obter estatísticas relevantes. Como nosso trabalho envolve tanto a modelagem de um perfil quanto o seu para o calculo da métrica de reputação, esta seção apresenta primeiramente os trabalhos relacionados a modelagem de usuário e em seguida os trabalhos relacionados com métricas de reputação. A

seção é finalizada com uma discussão sobre o relacionamento dos dois temas de pesquisa e as contribuições da nossa proposta.

2.1. Modelagem de usuários

Os trabalhos de Tang et al. (2008a), Tang et al. (2008b) e Tang, Zhang e Yao (2007), por exemplo, utilizam dados da produção científica de pesquisadores no contexto de redes sociais acadêmicas para analisar as relações que pesquisadores possuem entre si. Redes de colaboração científica entre pesquisadores são apresentadas e discutidas em Liu et al (2005), Chen et al. (2008), Huang et al. (2008) e Menezes et al. (2009). Basicamente, os trabalhos analisam redes de co-autoria desses pesquisadores através da definição de grafos e suas relações. Os dados são obtidos da DBLP (DBLP 2010).

Outros trabalhos utilizam dados de produção científica para encontrar especialistas em determinada área do conhecimento (ZHANG, TANG e LI, 2007; MIMNO e MCCALLUM, 2007; FU et al., 2007; LI et al., 2007). Esses trabalhos utilizam informações de páginas pessoais dos pesquisadores e de diferentes bibliotecas digitais, como DBLP (DBLP 2010) e CiteSeer (CITESEER 2010).

Em uma área de aplicação de mercado, o trabalho de Park e Chang (2009) apresenta um modelo de perfil de consumidores baseado em comportamento individual ou em grupo, construídos por meio do comportamento do usuário. Esse comportamento é identificado em sua navegação em páginas *web*, inserção de itens em carrinhos de compras (*basket insertion purchase*) e áreas de interesse.

A Microsoft, por meio do site Microsoft Academic Search[1], disponibiliza consultas sobre publicações, autores, conferências e periódicos, mediante um mecanismo de busca que coleta informações na *web* e armazena em um banco de dados. Sistemas semelhantes são utilizados em Tang et al. (2008a), Tang et al. (2008b) e na Digital Bibliography & Library Project (DBLP 2010). Tais sistemas permitem a visualização de dados de pesquisadores, bem como apresentam as redes de colaboração científica entre os pesquisadores.

2.2. Métricas Para Analisar Pesquisadores

O trabalho de Krapivin et al. (2009) afirma que a função de se utilizar métricas para avaliar a produção de pesquisadores está fundamentada em dois contextos: medir a qualidade da produção científica e medir a contribuição e a reputação relacionada a pesquisadores. Nesse sentido, os trabalhos de Small (1973), White & McCain (1998), Hirsch (2005), Jin et al. (2007) e Jin (2007), abordam temas como a citação para medir a reputação de pesquisadores.

A popularização do *PageRank* (Page, 1998) fez com que diversos trabalhos no âmbito acadêmico se utilizassem de seus algoritmos para a análise da reputação de pesquisadores. Trabalhos como os de Ding et al. (2009) e Chen et al. (2006) aplicam o *PageRank* a conjuntos de dados a fim de ordenar autores e artigos, respectivamente. O trabalho de Ding et al. (2009) ordena autores por meio de redes de co-citação. Já o trabalho de Chen et al. (2006) utiliza o conceito de *damping factor*, que se baseia no fato de que um leitor tende a procurar um caminho de ordem igual a dois links nas referências do artigo original antes de sair em busca de um artigo novo.

O algoritmo PaperRank (KRAPIVIN et al., 2009) modifica o *PageRank* original. A alteração considera, além das citações entre artigos, o *ranking* do artigo citante e a densidade de citações feitas pelo artigo citante. Outras métricas também foram desenvolvidas, como o

[1] *Disponível em*: http://academic.research.microsoft.com

h-index (HIRSCH, 2005), o AR-index (JIN, 2007) e o G-Index (TOL, 2008), todos calculando a importância individual de pesquisadores.

Dentre os trabalhos citados tanto para modelagem do perfil como para cálculo de reputação, nossa proposta se difere por especificar um modelo de perfil de pesquisadores flexível e adaptativo, que pode ser utilizado em diferentes áreas de aplicação. A adaptabilidade do modelo envolve desde a inserção e eliminação de categorias e elementos, bem como o ajuste de seus pesos. Outro ponto importante em nossa abordagem é que o modelo e a métrica se propõem a identificar a reputação de um pesquisador de forma abrangente, buscando um equilíbrio entre diferentes elementos que fazem parte da trajetória científica de um pesquisador, ao invés de levar em consideração apenas métricas como o H-index ou o G-index, que são baseados apenas em citações bibliográficas.

3. Modelo de Perfil Proposto

O objetivo desta seção é especificar o modelo de perfil do pesquisador. O modelo é baseado em elementos que constroem a trajetória de um pesquisador ao longo de sua carreira científica, facilmente identificados em seu currículo. Tais elementos não focam apenas em produção de artigos, mas sim na abrangência da vida científica do pesquisador.

O modelo de perfil proposto nesse trabalho foi definido através de uma ontologia que visa representar o comportamento do pesquisador, identificado por meio de um conjunto de elementos e de seus atributos. As próximas seções apresentam, respectivamente, as categorias definidas no modelo, o conjunto de elementos de cada categoria e suas representações na ontologia (Figura 1).

3.1. Categorias e Elementos

Em nosso estudo, identificamos diversos elementos importantes para a definição do perfil de um pesquisador. Alguns destes elementos são utilizados amplamente nas comunidades científicas das diversas áreas do conhecimento, conforme pode ser observado em trabalhos como os de Chen et al. (2007), Chen et al. (2008), Ding et al. (2009), Fu et al. (2007), Krapivin, Marchese e Cassati (2009), Li et al. (2007), Menezes et al. (2009), Tang, Zhang e Yao (2007), Tang et al. (2008a), Tang et al. (2008b) e Zhang, Tang e Li (2007). A Tabela 1 apresenta as categorias definidas, seus elementos e a sigla de cada um deles.

Category	Elements	Acronym
Identification (ID)	Education Degree	ED
Advisory (ADV)	Master Dissertation Advisor	MDA
	Phd Thesis Advisor	PTA
	Postdoctoral Advisor	PA
Examining Board (EB)	Participation in Examination Boards Master Dissertation	PEBMD
	Participation in Examination Boards Phd Thesis	PEBPT
Membership (MS)	Conference Committee Coordinator	CCC
	Conference Committee Member	CCM
	Editorial Board Member	EBM
	Reviewer of Journals	RJ
Production (PROD)	Articles in Scientific Journals	ASJ
	Book Chapter Published	BCP
	Books Published	BP
	Complete Work Published in Conference Proceedings	CWPCP
	H-Index	HI
	Network Co-autorship	NC
	Research Projects	RP
	Software With Register	SWR
	Software Without Register	SWOR

Tabela 1. Categorias, elementos e siglas que representam o perfil do pesquisador.

3.2. Ontologia

A ontologia para representar o perfil do pesquisador é dividida em cinco categorias: *Advisory*, *Examining Board*, *Identification*, *Membership* and *Production*. A divisão foi necessária para melhor visualizar o conjunto de elementos da trajetória do pesquisador.

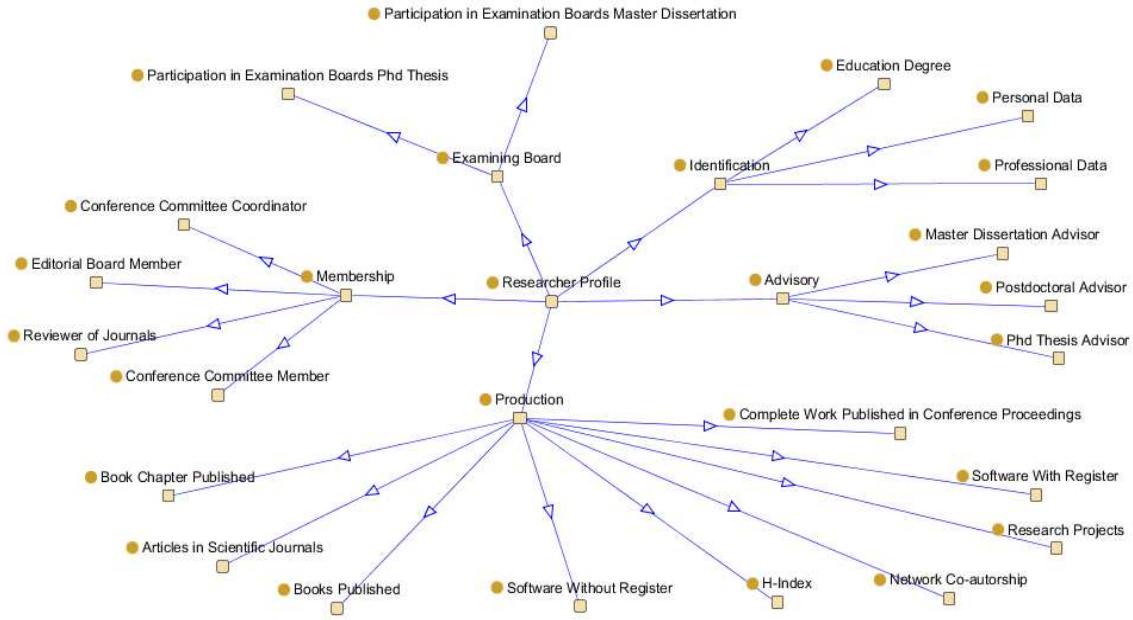


Figura. 1. Representação gráfica da ontologia.

O objetivo da ontologia apresentada na Figura 1 é aumentar a expressividade na representação, tornando mais clara a apresentação do modelo. A ontologia representa as cinco categorias e todos os seus elementos. Para uma melhor compreensão do modelo, apresentamos uma descrição de todos os elementos utilizados:

- *Identification*: por meio do elemento Education Degree, apresenta a titulação do pesquisador. As opções deste elemento são mestrado, doutorado ou pós-doutorado;
- *Advisory*: representa as orientações do pesquisador. `Master Dissertation Advisor` representa as orientações de mestrado. `Phd Thesis Advisor` representa as orientações de doutorado. Por fim, as orientações de pós doutorado são representadas pelo elemento `Postdoctoral Advisor`;
- *Examining Board*: especifica a participação em bancas do pesquisador. O elemento `Participation in Examination Boards Master Dissertation` representa a participação em defesas de mestrado. As participações em defesas de doutorado são representadas pelo elemento `Participation in Examination Boards Phd Thesis`;
- *Membership*: apresenta as conferências onde o pesquisador foi coordenador ou membro de comitê de programa, bem como se foi ou ainda é revisor de periódico ou membro de corpo editorial. O elemento `Conference Committee Coordinator` representa as conferências onde o pesquisador foi coordenador de comitê de programa. `Conference Committee Member` representa as conferências onde o pesquisador teve participação como revisor de trabalhos. O elemento `Editorial Board Member` diz respeito à atuação do pesquisador como membro de corpo editorial de periódico. Por fim, o elemento `Reviewer of Journals` apresenta os periódicos onde o pesquisador é ou foi revisor;

- *Production*: especifica a produção científica e outros elementos relacionados a produção científica do pesquisador. `Articles in Scientific Journals` representa os artigos publicados em periódicos. `Book Chapter Published` e `Books Published` representam, respectivamente, os capítulos de livros e os livros publicados pelo pesquisador. Os artigos publicados em conferências são representados pelo elemento `Complete Work Published in Conference Proceedings`. O elemento `H-Index` diz respeito às citações dos artigos do pesquisador por outros pesquisadores. `Network Co-autorship` está associado ao número de pesquisadores que possuem trabalhos publicados em conjunto com o pesquisador em questão. O elemento `Research Projects` representa o número de projetos de pesquisa do pesquisador. Os elementos `Software With Register` e `Software Without Register` representam os softwares em que o pesquisador teve participação. O primeiro refere-se aos softwares com registro e o segundo aos softwares sem registro.

4. Métrica Proposta Para Identificar a Reputação de Pesquisadores

O objetivo da métrica proposta neste trabalho é classificar o pesquisador em um nível de reputação definido através de um índice entre 1 e 5. Esta normalização foi proposta com o intuito de não distanciar com valores numéricos pesquisadores iniciantes, intermediários e experientes, bem como tornar mais compreensível a identificação da reputação. A métrica para identificar a reputação é definida por meio de um somatório entre os vários elementos que compõem o modelo de perfil do pesquisador. Para isto, são definidos pesos para as cinco categorias especificadas no modelo. O somatório dos pesos das categorias equivale a 100. Como dentro de cada categoria existem elementos, estes também possuem pesos individuais. O somatório dos pesos dos elementos é limitado ao peso máximo de cada categoria e seu somatório também está limitado ao valor 100.

4.1. Métrica

A seguir, apresentamos as fórmulas que definem a métrica para calcular a reputação de pesquisadores. Levando-se em consideração as Fórmulas 1, 2, 3, 4, 5 e 6 apresentadas a seguir, algumas observações devem ser destacadas:

- a letra `“R”` após a abreviação da categoria significa `“Researcher”`, fazendo referência ao pesquisador que se deseja descobrir a reputação;
- a letra `“W”` em frente ao nome dos elementos do modelo significa `“Weight”, ou seja,` o peso daquele elemento;
- a sigla `“NL”` significa `“Number of Levels”` e representa o número de níveis desejados no modelo. Na seção 5.3, em nossos experimentos, definimos 5 níveis;
- a sigla `“MVR”` significa `“Maximum Value of Range”`, ou seja, o máximo valor do intervalo. Este valor representa o número mais expressivo de cada elemento no modelo. Na seção 5.3, em nossos experimentos, definimos o valor mais expressivo de cada elemento levando-se em consideração um grupo de 52 pesquisadores.

A Fórmula 1 apresenta os elementos da categoria `Identification (ID)` que fazem referência ao nível de titulação do pesquisador, sendo considerado no cálculo apenas um dos valores, ou seja, a titulação máxima do pesquisador: ou *Mestrado*, ou *Doutorado* ou *Pós-doutorado*.

$$ID_{(R)} = (ED * WED) \quad (1)$$

Os elementos da categoria Advisory (ADV) são apresentados na Fórmula 2 e correspondem a *Master Dissertation Advisor, Phd Thesis Advisor* e *Postdoctoral Advisor*.

$$ADV_{(R)} = ((MDA * (WMDA / (NL - 1))) / (MVR / (NL – 1)) + (PTA * (WPTA / (NL - 1))) / (MVR / (NL - 1)) + (PA * (WPA / (NL - 1))) / (MVR / (NL – 1))) \quad (2)$$

A Fórmula 3 apresenta os elementos da categoria Examination Board (EB), onde dois elementos são definidos: *Participation in Examination Boards Master Dissertation* and *Participation in Examination Boards Phd Thesis*.

$$EB_{(R)} = ((PEBMD * (WPEBMD / (NL - 1))) / (MVR / (NL - 1)) + (PEBPT * (WPEBPT / (NL - 1))) / (MVR / (NL – 1))) \quad (3)$$

A Fórmula 4 apresenta os elementos da categoria Membership (MS). Esta categoria contempla quatro elementos: *Conference Committee Coordinator, Conference Committee Member, Editorial Board Member* and *Reviewer of Journals*.

$$MS_{(R)} = ((RJ * (WRJ / (NL - 1))) / (MVR / (NL - 1))+ (EBM * (WEBM / (NL - 1))) / (MVR / (NL - 1)) + (CCM * (WCCM / (NL - 1))) / (MVR / (NL - 1))+ (CCC * (WCCC / (NL - 1))) / (MVR / (NL – 1))) \quad (4)$$

Os elementos da categoria Production (PROD) são apresentados na Fórmula 5. No total, são nove elementos: *Articles in Scientific Journals*, *Book Chapter Published*, *Books Published*, *Complete Work Published in Conference Proceedings*, *H-Index*, *Network Co-autorship*, *Research Projects*, *Software With Register* and *Software Without Register*.

$$PROD_{(R)} = ((HI * (WHI / (NL - 1))) / (MVR / (NL - 1)) + (CWPCP * (WCWPCP / (NL - 1))) / (MVR / (NL - 1)) + (SWIR * (WSWIR / (NL - 1))) / (MVR / (NL - 1)) + (SWOR * (WSWOR / (NL - 1))) / (MVR / (NL - 1)) + (RP * (WRP / (NL - 1))) / (MVR / (NL - 1)) + (ASJ * (WASJ / (NL - 1))) / (MVR / (NL - 1)) + (BP * (WBP / (NL - 1))) / (MVR / (NL - 1)) + (BCP * (WBCP / (NL - 1))) / (MVR / (NL - 1)) + (NC * (WNC / (NL – 1))) / (MVR / (NL - 1))) \quad (5)$$

Finalmente, a Fórmula 6 apresenta o índice de reputação do pesquisador (Rep-Index) envolvendo o somatório das cinco categorias definidas no modelo.

$$Rep\text{-}Index_{(R)} = \Sigma_{ID} + \Sigma_{MS} + \Sigma_{EB} + \Sigma_{ADV} + \Sigma_{PROD} \quad (6)$$

O cálculo final do índice de reputação do pesquisador é obtido pelo somatório das cinco categorias: Identification (ID), Membership (MS), Advisory (ADV), Examination Board (EB) e Production (PROD). Já o resultado de cada categoria é obtido por meio da multiplicação do número de ocorrências de determinado elemento pelo peso do próprio elemento dividido pelo número de intervalos menos um. O resultado desta operação é dividido pelo valor máximo do intervalo, também dividido pelo número de intervalos menos um.

4.2. Níveis de Reputação

O modelo proposto trabalha com níveis de reputação adaptativos. Dependendo do contexto ou da área de utilização, pode-se definir quantos níveis de reputação desejar. Nos experimentos, que são apresentados na seção 5, trabalhamos com a reputação do pesquisador em cinco níveis. A partir do valor máximo de cada elemento (com exceção da categoria `Identification`) os cinco níveis foram gerados. Os níveis são a classificação do pesquisador e identificam sua reputação, calculada por meio do modelo proposto.

A escala de valores do índice de reputação varia de 0 a 100, onde um valor mais próximo de 0 representa uma baixa reputação (geralmente para pesquisadores em início de carreira) e um valor próximo de 100 representa uma alta reputação (geralmente pesquisadores mais experientes, com uma longa trajetória científica).

Como o modelo é abrangente, flexível e adaptativo, ele pode ser utilizado em diferentes áreas do conhecimento e em diferentes contextos, uma vez que os critérios de avaliação ou necessidades podem ser distintos. Da mesma forma, os pesos dos elementos do modelo ou ainda os próprios elementos, também podem ser adaptados ou suprimidos, se não são importantes dentro do contexto de utilização.

5. Experimentos e Análise de Resultados

O objetivo dos experimentos é validar o modelo de perfil especificado na seção 3 por meio das métricas definidas na seção 4. Os experimentos classificam o pesquisador em níveis que identificam sua reputação. Esta seção, além de apresentar os experimentos, também discute os resultados obtidos.

5.1. Aquisição de Dados

Para a validação do modelo foram obtidos dados do currículo de 52 pesquisadores brasileiros da área de Ciência da Computação. Estes pesquisadores estão vinculados a um dos Programas de Pós-graduação em Computação que possui conceito 6 (em uma escala de 3 a 7) na avaliação realizada pela Capes[2]. A escolha por este grupo de pesquisadores foi motivada por agregar pesquisadores de níveis iniciante, intermediário e experiente, bem como representar um programa de pós-graduação de qualidade e com reconhecida inserção internacional.

Os dados foram coletados da Plataforma Lattes[3], da DBLP, do Microsoft Academic Search e do Arnetminer[4]. Destaca-se que não foi necessário fazer tratamento de desambiguação de nomes ou de publicações, uma vez que o Arnetminer, a DBLP e o Microsoft Academic Search já incorporam tal recurso. Com relação ao Lattes, a extração foi semiautomatizada e realizada individualmente para cada pesquisador. O resultado da extração individual foi a quantificação, para cada pesquisador, dos elementos do modelo que possuíam relação com o Lattes.

5.2. Métricas de Comparação

Por estarem sendo amplamente utilizados na comunidade acadêmica da área de computação e por terem sido pioneiros no processo de classificação de pesquisadores, utilizamos o H-Index e o G-Index como métricas a serem comparadas com nossa proposta.

---

[2] *Capes*: Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.

[3] *Plataforma Lattes*: Base de dados de currículos de pesquisadores brasileiros.

[4] *Disponível em*: http://arnetminer.org/

Destaca-se que o H-Index e o G-Index utilizam apenas citações de artigos. Nossa abordagem, por envolver diversos elementos da trajetória científica de um pesquisador, não pode ser comparável diretamente a estes índices. Nos resultados apresentados na seção 5.4 calculamos o valor de correlação entre eles e o nosso índice.

5.3. Metodologia

Para a realização dos experimentos foram utilizadas as categorias e elementos definidos no modelo apresentado na seção 3. Para cada categoria foi definido um peso, onde o somatório destes pesos é igual a 100. Da mesma forma, para cada elemento foi definido um peso, onde o somatório destes pesos também é igual a 100.

Os pesos das categorias e os pesos dos elementos foram definidos levando-se em consideração as particularidades da área de Ciência da Computação, como por exemplo, a valorização de trabalhos publicados em conferências. Em algumas outras áreas, tais elementos poderiam ser desconsiderados ou ainda pouco valorizados, uma vez que outros elementos podem ser considerados fundamentais.

Os valores dos intervalos foram definidos por meio da análise do currículo dos 52 pesquisadores (seção 5.1). Para cada pesquisador foi analisado todo o conjunto de elementos do modelo, identificando o valor de ocorrência de cada elemento. Tendo-se todos os valores de todos os elementos dos 52 pesquisadores, foi identificado o valor máximo de cada elemento. Este valor máximo originou o elemento `Maximum Value of Range (MVR)`. A Tabela 2 apresenta as categorias e seus pesos, os elementos e seus pesos, bem como o máximo valor de cada elemento.

Category	Weight	Elements		Weight	Maximum Value of Range
Identification (ID)	15	Education Degree (ED) [5]	Postdoctoral	15	-
			Doctorate	12	-
			Master	8	-
Advisory (ADV)	15	Postdoctoral Advisor (PA)		6	4
		Phd Thesis Advisor (PTA)		5	30
		Master Dissertation Advisor (MDA)		4	66
Examining Board (EB)	15	Participation in Examination Boards Master Dissertation (PEBMD)		5	72
		Participation in Examination Boards Phd Thesis (PEBPT)		10	26
Membership (MS)	15	Conference Committee Coordinator (CCC)		3	9
		Conference Committee Member (CCM)		2	107
		Editorial Board Member (EBM)		6	5
		Reviewer of Journals (RJ)		4	32
Production (PROD)	40	Articles in Scientific Journals (ASJ)		12	96
		Book Chapter Published (BCP)		3	35
		Books Published (BP)		5	15
		Complete Work Published in Conference Proceedings (CWPCP)		8	437
		H-Index (HI)		6	24
		Network Co-autorship (NC)		2	132
		Research Projects (RP)		1	37
		Software With Register (SWR)		2	3
		Software Without Register (SWOR)		1	18
Total	**100**	**Total**		**100**	

Tabela 2. Categorias, elementos, pesos e valor máximo do intervalo de cada elemento.

[5] O elemento `Education Degree`, da categoria `Identification`, é o único que possui opções, onde apenas uma deve ser utilizada (ou Pós-doutorado ou Doutorado ou Mestrado) e seu peso máximo é 15.

Diante do valor máximo identificado para cada elemento, 5 intervalos foram definidos. Estes intervalos têm por objetivo classificar o pesquisador em níveis de reputação. Os intervalos tem valores entre 0 e 100. Assim, pegamos o valor máximo de cada elemento e dividimos por quatro, gerando os cinco intervalos.

Após os cálculos, o Rep-Index do pesquisador é definido com base nos níveis apresentados na fórmula 7.

$$Rep\text{-}Index_{(R)} = \begin{cases} 1, & Se\ Rep\text{-}Index_{(R)} >= 0\ e\ < 20 \\ 2, & Se\ Rep\text{-}Index_{(R)} >= 20\ e\ < 40 \\ 3, & Se\ Rep\text{-}Index_{(R)} >= 40\ e\ < 60 \\ 4, & Se\ Rep\text{-}Index_{(R)} >= 60\ e\ < 80 \\ 5, & Se\ Rep\text{-}Index_{(R)} >= 80\ e\ < 100 \end{cases} \quad (7)$$

5.4. Resultados

Experimentos foram realizados com os 52 pesquisadores, sendo identificados de R_1 a R_{52}. Como o grupo de pesquisadores é heterogêneo, no que tange a trajetória científica de cada um, algumas questões devem ser esclarecidas:

- No grupo existem pesquisadores que concluíram o curso de doutorado a pouco tempo, que estão iniciando sua carreira como orientadores de mestrado e de doutorado, que possuem pouca inserção internacional;
- Também existem pesquisadores que concluíram o curso de doutorado a um bom tempo, que realizaram curso de pós-doutorado, que possuem um número de orientações de mestrado e de doutorado significativo, bem como possuem inserção internacional por meio de publicação de artigos e livros;
- Por fim, no grupo definido existem pesquisadores com uma vasta experiência em pesquisa, com muitas orientações de mestrado e de doutorado, bem como possuem vasta inserção internacional, com diversos artigos e livros publicados e projetos em cooperação com instituições internacionais.

As Figuras 2, 3 e 4 apresentam os resultados dos experimentos realizados para cada um dos 52 pesquisadores, identificando-se o valor do H-Index, do G-Index e do Rep-Index.

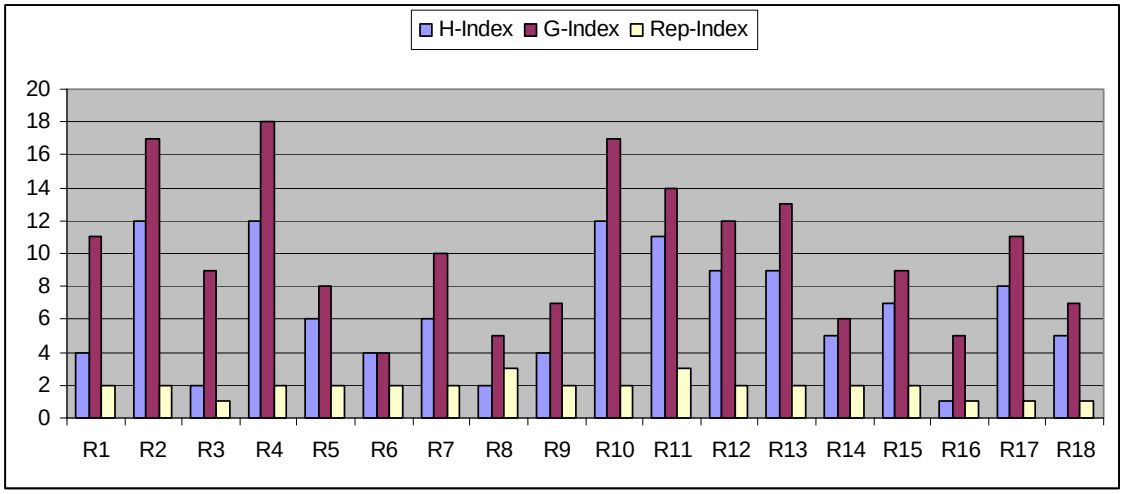


Figura 2. Dados dos pesquisadores R_1 a R_{18}

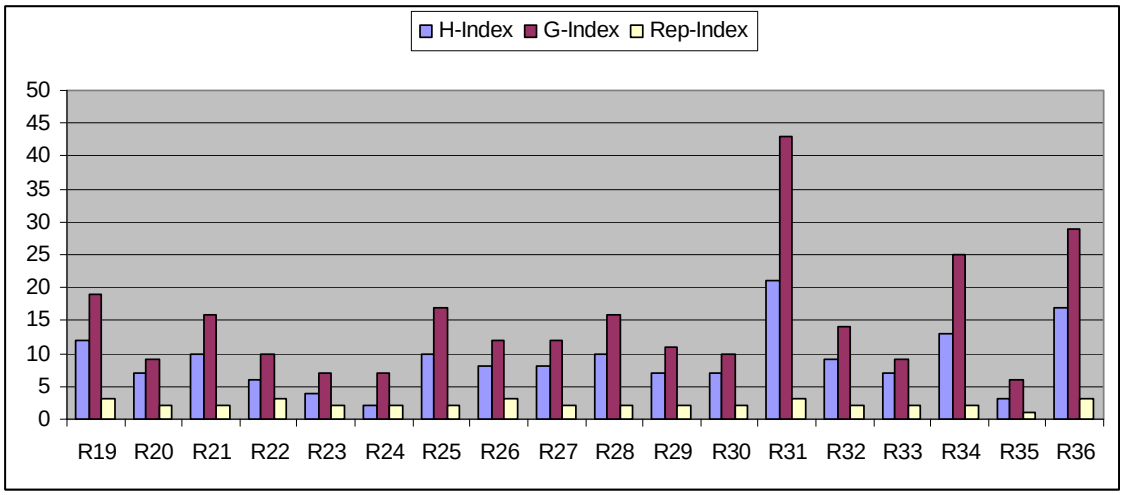


Figura 3. Dados dos pesquisadores R_{19} a R_{36}.

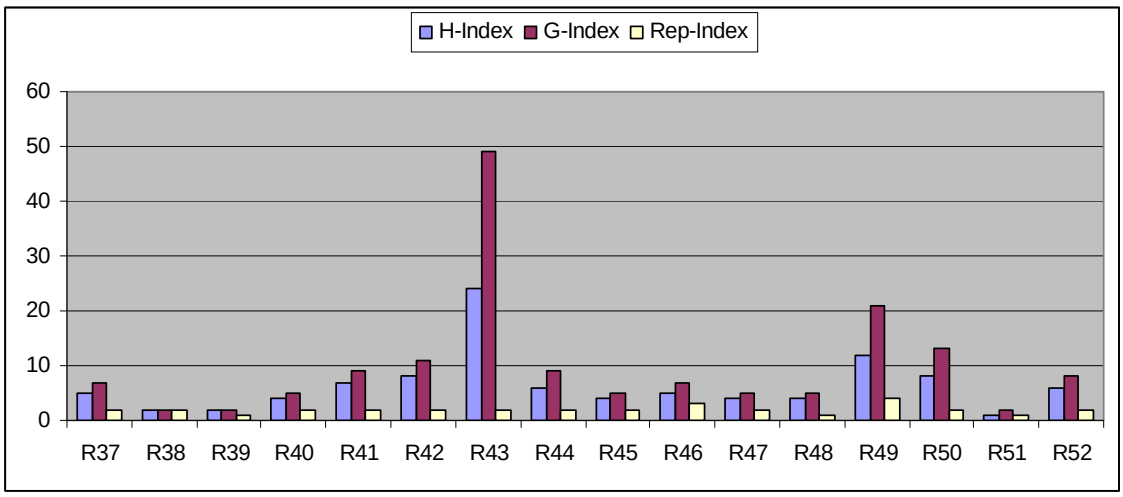


Figura 4. Dados dos pesquisadores R_{37} a R_{52}.

Para verificar a existência ou não de correlação entre os índices de cada pesquisador, foi realizado um experimento onde medimos o coeficiente de correlação[6] dos valores do H-Index com o Rep-Index e, posteriormente, do G-Index com o Rep-Index. Os resultados são apresentados a seguir:

- na relação do H-Index com o Rep-Index foi encontrado um coeficiente de correlação igual a `0,309327`, o que indica uma correlação fraca[7];
- na relação do G-Index com o Rep-Index foi encontrado um coeficiente de correlação igual a `0,258245`, o que indica uma correlação muito fraca.

[6] *Coeficiente de correlação*: indica a relação entre a produção do pesquisador e a soma dos diferentes pesos das variáveis independentes, podendo variar de -1 a +1, o que representa uma escala que vai de menos significativo até mais significativo.

[7] *Interpretação do coeficiente de correlação*: de 0.00 a 0.19 indica uma correlação muito fraca, de 0.20 a 0.39 correlação fraca, 0.40 a 0.69 correlação moderada, de 0.70 a 0.89 correlação forte e de 0.90 a 1.00 correlação muito forte.

Analisando-se os valores dos dois coeficientes de correlação identificados, pode-se perceber que nossa métrica (Rep-Index) não é comparável estatisticamente aos demais índices. A justificativa para isto é devido às outras métricas serem definidas levando-se em consideração, exclusivamente, o número de citações de artigos. Nossa proposta se difere pela abrangência do modelo, levando em consideração um grupo de elementos que fazem parte da trajetória científica de um pesquisador, construída ao longo de sua carreira acadêmica.

A figura 5 apresenta o resultado do Rep-Index dos 52 pesquisadores estudados, mostrando os percentuais em cada um dos níveis.

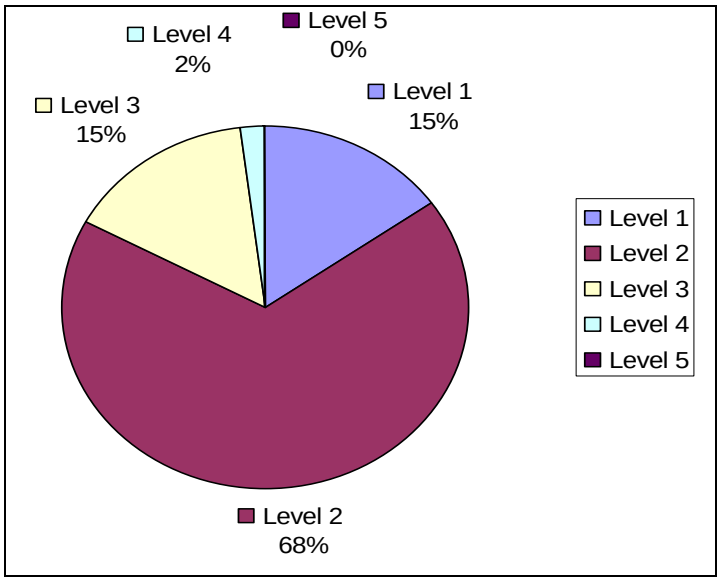


Figura 5. Resultado da classificação dos pesquisadores pelo Rep-Index.

Observa-se que o nível 2 foi o que mais teve concentração de pesquisadores em questão, chegando a 68% do total da amostra.

Um dos problemas identificados na análise dos currículos dos 52 pesquisadores é que nem todos possuem dados para todos os elementos do modelo proposto, o que acabou determinando um índice de reputação abaixo do esperado, principalmente para os pesquisadores classificados nos níveis 1 e 2.

Outra questão relevante é que como a população utilizada nos experimentos representa um dos grupos de excelência da área de computação na pesquisa brasileira, e os valores dos intervalos dos elementos do modelo são definidos pelos valores máximos de ocorrência de cada elemento, o resultado final tende a baixar. Isto se justifica pela maioria dos pesquisadores analisados não manterem um equilíbrio em sua trajetória de pesquisa, pois alguns elementos ficam bem pontuados e outros não. Isto faz com que a média final para o cálculo do índice de reputação tenda a baixar.

Com base nos resultados obtidos por meio dos experimentos conseguimos demonstrar que nossa proposta é viável, uma vez que pode ser utilizada em diferentes áreas do conhecimento, além de levar em consideração as características e peculiaridades de cada área. Levando-se em consideração a abrangência, flexibilidade e adaptabilidade do modelo, basta que ajustes sejam realizados. Isto envolve desde a alteração de pesos e valores dos elementos, bem como a supressão de elementos e até de categorias.

6. Considerações Finais e Trabalhos Futuros

Este trabalho teve como objetivo especificar um modelo de perfil de pesquisadores e apresentar métricas para avaliar a reputação de pesquisadores em sua área de atuação. O modelo levou em consideração os dados obtidos por meio da análise da trajetória científica do pesquisador, construída ao longo de sua carreira.

O processo de modelagem do perfil envolveu a identificação de quais dados foram considerados relevantes para sua definição, sendo apresentados por meio de uma ontologia de domínio. A reputação dos pesquisadores foi identificada por uma classificação, em uma escala de 1 a 5, onde 1 indica baixa reputação e 5 indica alta reputação.

Para a validação do modelo, métricas foram propostas. As métricas foram definidas baseadas no equilíbrio da trajetória do pesquisador, envolvendo os principais elementos de seu currículo. Por meio das métricas definidas, foi possível validar o modelo proposto. O modelo demonstrou equilíbrio entre os diversos elementos que formam o perfil de um pesquisador, utilizando dados quantitativos. Apesar disso, novos experimentos devem ser realizados, principalmente com uma amostra maior de pesquisadores.

Considerando a proposta apresentada neste artigo, experimentos exaustivos serão realizados com o intuito de calibrar a métrica proposta e apresentá-la por meio de formalismo. Pretende-se, ainda, realizar inferências utilizando-se da ontologia, bem como recomendar a um pesquisador a mudança de nível e evoluir na carreira; identificar pesquisadores com comportamento semelhante; cruzar o fator de impacto dos periódicos onde o pesquisador tem publicação gerando um novo elemento para o perfil e incorporar aspectos temporais no modelo.

Referências

Chen, P., Xie, H., Maslov, S., and Redner, S. (2007). Finding scientific gems with Google. Journal of Informetrics, 1, 8–15.

Chen, C., Song, I., Yuan, X., and Zhang, J. (2008). The Thematic and Citation Landscape of Data and Knowledge Engineering (1985-2007). Data & Knowledge Engineering. v.67, n.2 (Nov. 2008), p. 234-259.

CiteSeer (Scientific Literature Digital Library). Disponível em <http://citeseer.ist.psu.edu>. Acesso em julho , 2010.

DBLP (Digital Bibliography & Library Project). University of Trier. Disponível em <http://dblp.uni-trier.de>. Acesso em julho, 2010.

Ding, Y., Yan, E., Frazho, A. and Caverlee, J. (2009). PageRank for Ranking Authors in Co-citation Networks. Journal of the American Society for Information Science and Technology, 60(11), 2229-2243.

Fink, J.; Kobsa, Alfred. 2002. User Modeling for Personalized City Tours. Journal of Artificial Intelligence. *Rev.*18, 1 (September 2002), 33-74.

Fu, Y.; Xiang, R.; Liu, Y.; Zhang, M.; Ma, S. Finding Experts Using Social Network Analysis. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence. Silicon Valley, USA, 2007.

Gauch, S; Aravid, Speretta, M.; Chandramouli, A.; e Micarelli A.. 2007. User profiles for personalized information access. In *The adaptive web.* Lecture Notes In Computer Science, Vol. 4321. Springer-Verlag, Berlin, Heidelberg 54-89.

Hirsch, J. E. An Index to Quantify an Individual's Scientific Research Output. In Proceedings of the National Academy of Science. V.102, n.46, p.16569-16572, 2005.

Hope, T.; Nishimura, T.; Takeda, H. An Integrated Method for Social Network Extraction. Proceedings of the 15th International Conference on World Wide Web. Edinburgh, Scotland, 2006.

Huang, J., Zhuang, Z., Li, J., and Giles, C. L. (2008). Collaboration Over Time: Characterizing and Modeling Network Evolution. In Proceedings of the International Conference on Web Search and Web Data Mining. Palo Alto, California, USA, 2008.

Jin, B. (2007). The AR-Index: Complementing the H-Index. ISSI Newsletter, 3(1) p. 6.

Jin, B., LiMing, L., Rousseau, R. and Egghe, L. (2007). The R- and AR-indices: Complementing the H-Index. Chinese Science Bulletin. 52(6), 855– 863.

Krapivin, M., Marchese, M., Cassati, F. (2009). Exploring and Understanding Scientific Metrics in Citation Networks. Complex Sciences, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer Berlin Heidelberg. V.5, p.1550, 2009.

Li, J.; Tang, J.; Zhang, J.; Luo, Q.; Liu, Y.; Hong, M. EOS - Expertise Oriented Search Using Social Networks. In Proceedings of the 16th International Conference on World Wide Web. Banff, Canada, 2007.

Liu, X., Bollen, J., Nelson, M. L., and Van de Sompel, H. (2005). Co-authorship Networks in the Digital Library Research Community. Information Processing and Management: An International Journal. V.41, n.6 , p.1462-1480, 2005.

Maia, M.; Almeida, J.; Almeida, V. Identifying User Behavior in Online Social Networks. In Proceedings of the 1st Workshop on Social Network Systems. New York, NY, USA, 2008.

Menezes, G. V., Ziviani, N., Laender, A. H., and Almeida, V. (2009). A Geographical Analysis of Knowledge Production in Computer Science. In Proceedings of the 18th International Conference on World Wide Web. Madrid, Spain, 2009.

Middleton, S. E.; Shadbolt, N. R.; De Roure, D. C. 2004. Ontological user profiling in recommender systems. *ACM Transactions Information Systems.* 22, 1 (January 2004), 54-88.

Mimno, D.; McCallum, A. Expertise Modeling for Matching Papers With Reviewers. In Proceedings of the 13th International Conference on Knowledge Discovery and Data Mining. San Jose, USA, 2007.

Page, L.; Brin, S.; Motwani, R.; Winograd, T. (1998) The PageRank citation ranking: Bringing order to the Web. Tech. report, Stanford University.

Park, Y. and Chang, K. Individual and Group Behavior-based Customer Profile Model for Personalized Product Recommendation. Expert System. Appl. 36, 2 (Mar. 2009), 1932-1939.

Small, H. (1973). Cocitation in science literature: New measures of relationship between two documents. Journal of the American Society for Information Science, 24(4), 265–269.

Tang, J., Zhang, J., Yao, L., Li, J. Extraction and Mining of an Academic Social Network. In Proceedings of the 17th International Conference on World Wide Web. Beijing, China, 2008.

Tang, J.; Zhang, D.; Yao, L. Social Network Extraction of Academic Researchers. In Proceedings of 7th International Conference on Data Mining. Omaha, USA, 2007.

Tang, J.; Zhang, J.; Yao, L.; Li, J.; Zhang, L.; Su, Z. ArnetMiner – Extraction and Mining of Academic Social Networks. In Proceeding of the 14th International Conference on Knowledge Discovery and Data Mining. Las Vegas, USA, 2008.

Tol, R. S. J. A Rational, Successive G-Index Applied to Economics Departments in Ireland. Journal of Informetrics, V.2, n.2, p.149-155, 2008.

Trajkova, J.; Gauch, S. Improving Ontology-Based User Profiles. M.S. Thesis. EECS, University of Kansas. Lawrence, 2003.

White, H.D., McCain, K.W. (1998). Visualizing a discipline: An author cocitation analysis of information science 1972-1995. Journal of the American Society for Information Science, 49, 327–355.

Zhang, C. T. (2009) The *e*-Index, Complementing the *h*-Index for Excess Citations. PLoS ONE 4(5): e5429.

Zhang, J.; Tang, J.; Li, J. Expert Finding in a Social Network. Proceedings of 12th International Conference on Database Systems for Advanced Applications. Bangkok, Thailand, 2007.

WCAMA - III Workshop de Computação Aplicada à Gestão do Meio Ambiente e Recursos Naturais

PREFÁCIO

É com grande satisfação que damos as boas-vindas a todos os participantes do XXXI Congresso da Sociedade Brasileira de Computação e do III Workshop de Computação Aplicada à Gestão do Meio Ambiente e Recursos Naturais - WCAMA. O Congresso da Sociedade Brasileira de Computação (CSBC) é um evento realizado anualmente pela SBC, com a intenção de promover e incentivar a troca de experiências entre as comunidades científica, acadêmica e profissional na área da Computação nacional e internacional. Desde sua primeira edição, o CSBC, através de seus diversos eventos satélites, tem-se mostrado um importante local para a exposição de trabalhos científicos, palestras, cursos e debates, tanto de cunho puramente acadêmico e científico quanto social.

O CSBC 2011 está sendo organizado pelo Departamento de Informática e Matemática Aplicada (DIMAp) da Universidade Federal do Rio Grande do Norte (UFRN) e será realizado no Centro de Convenções de Natal, localizado nas proximidades dos principais hotéis e atrativos turísticos da cidade, contando com excelente acesso e infraestrutura para a realização de eventos de grande porte, como o CSBC.

O WCAMA tem como objetivo promover a integração, de forma efetiva, entre a Computação (metodologias e ferramentas) e a Ecologia (definição de políticas e gestão para o meio ambiente). A gestão de recursos naturais é uma atividade complexa e extremamente dinâmica, que exige integração entre atores das áreas sociais, políticas e tecnológicas para sua realização. A estruturação de políticas de meio ambiente nacionais e internacionais é aspecto essencial neste contexto, e a Computação pode contribuir para o detalhamento destas políticas, auxiliando em sua implementação, manutenção e monitoramento de suas consequências e efeitos. Por esse motivo, algumas das contribuições ao WCAMA se relacionam diretamente aos cinco Grandes Desafios definidos pela SBC para a Computação brasileira na presente década, em particular ao Desafio 2: modelagem computacional de sistemas complexos artificiais, naturais e socioculturais e da interação homem-natureza.

Em 2011, para sua terceira edição, o WCAMA convidou o prof. Frederico Fonseca, da Pennsylvania State University, EUA, cuja atividade de pesquisa na área de ontologias e semântica o coloca frequentemente em contato com temas que ligam a Computação ao meio ambiente, principalmente com o conjunto de recursos que precisam ser concebidos para que cientistas, cidadãos e autoridades ambientais possam interagir de forma mais efetiva. No WCAMA, o prof. Fonseca apresentará uma palestra intitulada Global Forest Information Systems: Challenges in Data, Ontology, and Information Technology. Além disso, o WCAMA terá sessões técnicas com os artigos selecionados. Este ano foi submetido um total de 22 artigos, todos de excelente qualidade e foram aceitos 8 artigos, numa média de aceitação de 36%.

Não podemos deixar de agradecer a todos os que contribuíram direta ou indiretamente para o sucesso deste evento, em especial ao palestrante convidado, Prof. Fred Fonseca, aos membros do Comitê de Programa e aos organizadores locais.

Em nome de todo o comitê de organização, desejamos a todos um ótimo evento.

Natal, Julho de 2011

Clodoveu Davis Jr
Valéria Gonçalves Soares

COMISSÕES

Coordenação Geral

Clodoveu Davis, DCC-UFMG (coordenador geral)
Valeria Gonçalves Soares, DI-UFPB (coordenadora de Comitê de Programa)
Eduardo Henrique da Silva Aranha, DIMAp-UFRN (coordenador local)

Comitê de Programa

Adenauer Yamin - UFPEL
Aline Loreto - UFPEL
Ana Carolina Salgado - UFPE
Antonio Carlos da Rocha Costa - FURG
Antonio Saraiva - USP
Clauirton Siebra – UFPB
Clodoveu Davis - UFMG
Diana Adamatti - FURG
Eliane Araújo - UFCG
Frederico Fonseca - Pennsylvania State University (EUA)
Gilberto Camara - INPE
Gracaliz Dimuro - FURG
Ivan Guilherme - UNESP
João Balsa - Faculdade de Ciências, Universidade de Lisboa (Portugal)
Joel da Silva - UFSM
Laurindo Campos - INPA
Lucídio dos Anjos Formiga Cabral - Universidade Federal da Paraíba
Luiz Satoru Ochi – UFF
Maria Isabel Sobral Escada - INPE
Marilton Aguiar - UFPEL
Natasha Lino – UFPB
Nelson Duarte Filho - FURG
Nisia Krusche - UNESP
Silvia Botelho - FURG
Valéria C. Times – UFPE
Valéria Soares – UFPB
Vinicius Ribeiro - UniRitter

ARTIGOS

Modelagem Chuva-vazão utilizando Redes Neurais Artificiais e Algoritmos Genéticos

Felippe Alex Scheidt[1], Maria Angelica de Camargo Brunetto[2]

[1, 2] Departamento de Computação – Universidade Estadual de Londrina (UEL)
Londrina – PR – Brasil

felippescheidt@gmail.com, angelica@uel.br

Abstract. *In this article it is proposed a methodology for modeling the rainfall-runoff relationship of a specific watershed, through artificial neural networks (ANN) and genetic algorithm. This model was developed in Piquiri river watershed, located at the state of Paraná, Brazil. For comparative purposes, two approaches were developed on the model: a methodology that uses only a MLP ANN and another that uses a MLP ANN with genetic algorithms. As results, the hybrid model fitted better than ANN at representing the rainfall-runoff transformation process.*

Resumo. *Este artigo apresenta uma metodologia para modelagem da relação de transformação chuva-vazão de uma bacia hidrográfica utilizando redes neurais artificiais (RNA) acopladas a um algoritmo genético (AG). Esta modelagem foi desenvolvida na bacia hidrográfica do rio Piquiri, localizada no estado do Paraná, Brasil. Para efeitos comparativos, foram desenvolvidas duas abordagens na modelagem: uma metodologia utilizando apenas uma RNA tipo MLP e outra usando uma RNA MLP com algoritmos genéticos. Observou-se que o modelo híbrido obteve melhor representação do processo de transformação chuva-vazão.*

1. Introdução

O processo de transformação chuva-vazão é um dos mais complexos fenômenos hidrológicos que ocorrem em uma bacia hidrográfica, devido a grande variação temporal e espacial dos diversos elementos do ciclo hidrológico [Sedki 2009]. Dentre tais elementos, podem ser citados precipitação, evapotranspiração e infiltração, e esta diversidade de variáveis pode aumentar a complexidade da definição de um modelo físico que descreva corretamente esse processo.

Modelos chuva-vazão descrevem a relação entre a chuva e a vazão de uma bacia hidrográfica, quantificando a transformação da chuva em vazão de um curso de água. Esta relação é descrita através de um hidrograma, que mostra a variação da vazão do curso de água no tempo, em um determinado ponto de interesse da bacia. Portanto, tem diversas finalidades práticas para a previsão de cheias, da recarga de reservatórios, da disponibilidade de água para abastecimento urbano ou irrigação, dentre outras.

O objetivo deste trabalho é testar o uso das redes neurais artificiais e algoritmos genéticos para a modelagem hidrológica da transformação chuva-vazão em um ponto intermediário da bacia do Rio Piquiri, localizada no estado do Paraná (Brasil), e cuja foz

encontra-se junto ao rio Paraná. A motivação desta pesquisa foi a construção de método de modelagem que possibilitasse trabalhar com um número reduzido de variáveis hidrometeorológicas para a representação do processo chuva-vazão. Este trabalho está organizado nas seguintes seções: a seção 2 apresenta pesquisas relacionadas à modelagem chuva-vazão; a seção 3 expõe os conceitos de redes neurais artificiais, algoritmos genéticos e modelos híbridos; a seção 4 detalha o estudo de caso, a seleção dos dados para processamento, o treinamento da RNA e a definição do algoritmo genético utilizado; a seção 5 apresenta os resultados obtidos com os modelos propostos.

2. Trabalhos Relacionados

Campolo et. al.(1999) utilizam uma rede neural para analisar e prever o comportamento do rio Tagliamento na Itália, durante períodos de fortes chuvas. O modelo proposto faz uso de diversas estações de pluviometria levando em consideração sua distribuição espacial. A área de drenagem da bacia tem um total de 1950km^2 incluindo diversas sub-bacias. No estudo, foram propostos dois modelos de previsão do nível de água: o primeiro modelo para curtos períodos de tempo (1 hora); o segundo modelo para tempos longos acima de 10 horas.

Em outro estudo proposto por [Nasseri 2008], foi aplicado RNAs para a previsão de precipitação na bacia de Upper Parramatta localizada em Sydney, Austrália, com área aproximada de 112km^2. Os autores relatam a problemática e o consumo de tempo para encontrar e selecionar uma arquitetura de rede neural adequada. Para contornar tal questão, os autores propõem a utilização de algoritmos genéticos acoplado a uma rede neural tipo MLP para melhorar o desempenho da rede durante a fase de treinamento. O uso de algoritmos genéticos é voltado nesse trabalho para determinar os pesos correspondentes das conexões entre cada neurônio artificial. Essa mesma metodologia é também investigada em [Sedki 2009] numa bacia localizada no Marrocos.

Outro estudo de caso foi conduzido por [Tokar 1999] que aplicam redes neurais para realizar previsões de vazão diária como uma função da precipitação, da temperatura e do derretimento da neve medidos diariamente, na bacia do rio Little Patuxent em Maryland, Estados Unidos. A RNA é comparada a técnicas de regressão estatística e com o modelo SCRR (*simple conceptual rainfall-runoff*), proposto com McCuen e Snyder. Concluem ao final que o modelo obtido com a RNA mostrou resultados mais sistemáticos, com necessidade de menos dados para calibração e melhor previsão.

Em relação aos artigos citados acima, esta pesquisa diferencia-se nos seguintes aspectos: são desenvolvidas duas modelagens do processo de transformação chuva-vazão, uma usando apenas redes neurais artificiais (RNA) e outra usando um modelo híbrido no qual é acoplado um algoritmo genético (AG) junto a RNA. A área total de drenagem da bacia hidrográfica do Rio Piquiri, compreende o total de 3620km^2, uma área consideravelmente maior do que os trabalhos revisados acima. Isso implica numa maior variabilidade espacial e temporal, exercendo maior influência na correlação entre os dados de precipitação e os dados de vazão. Além disso, são propostos 19 modelos de entradas de dados, para identificação de um conjunto adequado de dados de entrada. Em termos hidrológicos, este artigo apresenta uma técnica importante num cenário no qual existem poucas variáveis de monitoramento na bacia hidrográfica. Neste caso, a modelagem é possível a partir da identificação de duas variáveis tipicamente monitoradas: precipitação(P) e vazão(Q). Assim, essa metodologia tem uma

aplicabilidade maior, pois não necessita da especificação de informações adicionais sobre a bacia hidrográfica.

3. Fundamentação Teórica

A seguir são apresentados as abordagens conceituais de redes neurais artificiais, algoritmos genéticos e modelos híbridos utilizados neste trabalho.

3.1. Redes Neurais Artificiais

O diferencial da modelagem de sistemas usando RNAs é que as mesmas são desenvolvidas a partir de um conjunto de dados de entrada, com ou sem supervisão, não sendo portanto, um modelo baseado no processo físico do fenômeno. O mecanismo básico consiste em treinar esta rede com dados reais, ou seja, dados empíricos com amostras ou exemplos de padrões de entrada-saída para construir uma relação não-linear entre os exemplos e, a partir disso, criar um modelo que permita realizar previsões.

Redes neurais podem ser definidas como estruturas que contém diversas unidades de processamento (também conhecidas como neurônios artificiais) densamente interconectadas sendo capaz de desempenhar grande quantidade de processamento de dados em computação paralela e representação de conhecimento [Basheer 2000]. RNAs também são conhecidas como aproximadores universais de funções já tendo sido demonstrado que são capazes de aproximar qualquer função contínua em qualquer nível de precisão [Cybenko 1989].

3.2. Algoritmos genéticos

Algoritmo genético (AG) é uma estratégia de busca global em um espaço de estados não-linear, no qual soluções iterativas não são adequadas. De acordo com a teoria da seleção natural, na qual apenas os indivíduos mais aptos sobrevivem e geram descendentes, os algoritmos genéticos buscam simular em nível computacional o mesmo efeito através da codificação de possíveis soluções de um problema, numa população de indivíduos.

Esses indivíduos são representados por cromossomos e genes. Inicialmente, cada cromossomo é inicializado com um conjunto aleatório de genes (pesos). Esta solução (cromossomo) é avaliada por uma função de aptidão que verifica a qualidade da solução proposta. Se o cromossomo possui uma boa aptidão ele terá uma probabilidade maior de manter-se na população e gerar descendentes cada vez mais aptos [Rezende 2005].

3.3. Modelos híbridos

Em redes MLP (*Multilayer perceptron*) o algoritmo de aprendizagem *backpropagation* (BP) remete a minimizar o erro entre a função calculada pela rede e o valor esperado. Existem diversas estratégias para realizar esta minimização [Silva 1998]. Uma delas, utilizadas neste trabalho é o método do gradiente descendente, que comumente é usado no *backpropagation.* Este método pode ficar preso num mínimo local impossibilitando de encontrar o mínimo global. Um modo de superar a limitação dos algoritmos de treinamento baseados no gradiente descendente é usar AG para realizar o treinamento como uma evolução das conexões sinápticas. Assim o AG pode ser utilizado para uma

busca por um conjunto ótimo de pesos globais sem calcular o gradiente descendente, evitando a estagnação precoce do processo de busca num mínimo local.

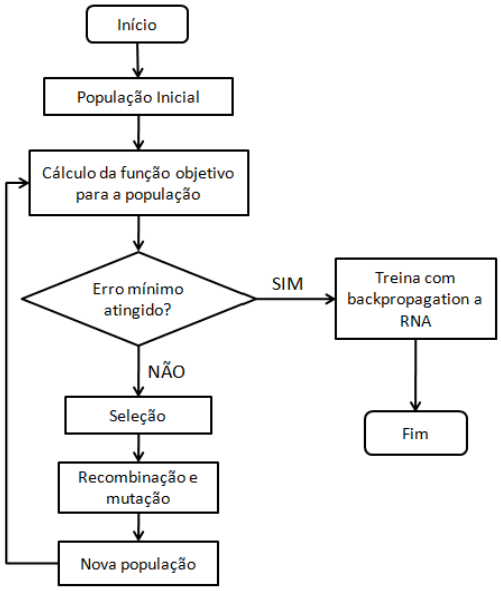


Figura 1 – Rede neural, acoplada a um AG

A abordagem evolucionária de treino de uma RNA consiste nos seguintes passos [Yao 1999]:

1. Codificar cada indivíduo da geração atual, num conjunto de pesos sinápticos e construir a RNA correspondente com os pesos.
2. Avaliar cada RNA usando RMSE (equação 3). A aptidão do indivíduo (cromossomo) é determinada pelo erro. Quanto maior o erro, menor sua aptidão.
3. Selecionar os pais para a recombinação de acordo com suas aptidões.
4. Aplicar operadores de recombinação e mutação gerando descendentes que formarão a próxima população.

A integração dos algoritmos genéticos com as RNAs (Figura 1) é vantajosa no sentido de agregar a capacidade de busca global dos AGs [Back 1997] junto com o algoritmo de treinamento e ajuste de pesos BP. O AG ajuda a superar os mínimos locais efetuando uma busca mais ampla no espaço de estados, pois ele é menos suscetível de ficar estagnado num mínimo local, pois não depende do gradiente, em comparação com os algoritmos baseados no gradiente [Yao 1999]. Após o AG terminar seu processamento, utiliza-se o BP para uma busca local usando o conjunto de pesos pré-ajustados pelo AG.

De acordo com a Figura 1, o primeiro passo do modelo híbrido consiste na formulação de uma população de cromossomos, criados aleatoriamente. Nesse caso, cada gene representa um peso sináptico da RNA. Em seguida, cada cromossomo é avaliado por uma função de aptidão (função RMSE – equação 3). É verificado na população, se algum candidato possui aptidão suficiente de acordo com um erro mínimo desejado. Caso não exista, o ciclo de execução do algoritmo genético será re-iniciado. Para isso, é selecionado cromossomos candidatos que geram novos indivíduos, que por sua vez terão alguns genes alterados de acordo com um percentual de mutação. Em

seguida esses descendentes são alocados na nova população. Esse processo repete-se até que o erro mínimo seja atingido ou um número máximo de iterações seja ultrapassado. Após a finalização do AG, o melhor cromossomo é retirado da população e uma RNA é instanciada, com os pesos equivalentes aos genes desse indivíduo. Essa RNA é passada para o BP que executará um refinamento dos pesos encontrados. Desse modo pode-se entender a fase do AG, como uma etapa de atribuição inicial dos pesos da RNA.

4. Materiais e métodos

A bacia hidrográfica utilizada como estudo de caso foi a bacia do rio Piquiri, sub-bacia da bacia do rio Paraná. O rio Piquiri é um dos principais afluentes da margem esquerda do rio Paraná. Possui 485 quilômetros de extensão aproximada. A bacia hidrográfica do rio Piquiri é a terceira maior em área física do estado do Paraná. Apresenta uma área total de 24.731km^2.

4.1. Seleção dos Dados para Processamento

Neste trabalho foram utilizadas 5 estações pluviométricas e 1 estação fluviométrica. Cada estação possui uma série histórica em escala diária de medição, com 3103 registros, na qual cada registro representa o total de chuva acumulado no período de 1 dia. No dia 05.10.2008, a vazão apresentou uma máxima de 1991,5 m^3/s sendo que no dia 24.07.2006 apresentou uma vazão mínima de 6,47 m^3/s.

4.2. Pré-processamento

Antes de serem submetidos a rede neural, as séries históricas de vazão e chuva foram normalizadas no intervalo de [0.1:0.9]. Essa etapa é importante para a adequação das variáveis em um mesmo intervalo, pois geralmente elas estão em diferentes escalas de medida como é o caso da precipitação e da vazão. A normalização é feita de acordo com a equação 1:

$$X_{norm} = 0.1 + 0.8\left(\frac{X_i}{X_{max}}\right) \qquad (1)$$

4.3. Treinamento da RNA

Para treinamento da RNA foi utilizado o método da parada antecipada [Haykin 2008], que é definido de acordo com o monitoramento do erro de treinamento e do erro de validação. Quando o erro de validação pára de diminuir, o treinamento é finalizado.

Uma das questões mais importantes e não bem definidas na literatura sobre RNAs é qual modelo de arquitetura utilizar para um determinado problema [Anders 1999]. Nesse caso, a definição da arquitetura envolve a seleção adequada do número de nodos da camada oculta e do número de neurônios da camada de entrada. Sendo assim, optou-se pela técnica apresentada por [Anders 1999] e [Kwok 2009] na qual parte-se de um ponto de vista *bottom-up*, isto é, inicia-se a rede neural, com um número mínimo de neurônios ocultos, e gradualmente aumenta-se esse número até atingir um critério de parada. A ideia consiste na definição de um modelo inicial de RNA, contendo um número *n* mínimo de neurônios ocultos, sendo inicialmente *n=1*. Ao término do treinamento, acrescenta-se um neurônio na camada oculta, repetindo o treinamento e calculando novamente o erro. Se essa nova rede contendo mais neurônios mostrar um

erro de validação menor em relação a rede anterior, então o novo modelo é aceito e parte-se para uma nova hipótese de modelo. Esse processo pára quando unidades adicionais de neurônios ocultos não conseguem reduzir o erro de validação em relação ao modelo anterior.

4.4. Definição dos pesos iniciais da RNA usando AG

Uma das metodologias testadas neste trabalho envolve o uso de AG acoplado a uma RNA. Nesse caso antes de avaliar cada modelo de RNA, foi utilizado um algoritmo genético para iniciar a matriz de pesos da RNA. A metodologia de integração de algoritmos genéticos com redes neurais inicia-se pela codificação dos pesos de cada rede em uma população de cromossomos. A codificação por valor foi utilizada para representação de cada peso sináptico, em um respectivo gene. Nessa codificação, o gene é codificado como um tipo *double*, do mesmo tipo do peso sináptico da RNA. O conjunto de genes forma um cromossomo. Cada cromossomo representa uma matriz de pesos de uma possível RNA. A recombinação entre os cromossomos utiliza dois pontos de corte, que são definidos de modo aleatórios, produzindo dois descendentes. Após essa etapa, aplica-se a mutação nos genes, que nesse caso são do tipo *double*, sendo então a mutação uma alteração do valor multiplicado por um percentual definido aleatoriamente.

Durante a fase inicial do AG, é instanciada uma população com 1000 cromossomos representando 1000 possíveis configurações de redes neurais. Para cada cromossomo, é aplicada uma função de aptidão para avaliar a qualidade da solução de cada cromossomo. Para isso, é utilizada a função RMSE (equação 3). O AG executa até atingir um erro mínimo ou um número máximo de 100 gerações, com o objetivo de escapar de um mínimo local inicial e de modo a preparar uma matriz inicial de pesos que facilite a convergência do algoritmo *backpropagation*.

4.5. Modelagem das Redes Neurais Artificiais

O objetivo dos diferentes tipos de modelos, propostos na Tabela 1, é avaliar a influência de diferentes *lags* das variáveis de entrada na RNA. O modelo mais simples, nesse caso, é o modelo 1, no qual explica a vazão *Q(t)* de acordo com a precipitação *P(t).* O modelo com maior número de *lags* de entrada é o modelo 19, no qual explica a vazão *Q(t)* em função da precipitação de *t-5* e da vazão de *t-4*, onde *t*, em escala temporal diária, representa 1 dia de medição. Para cada um dos modelos, foi instanciada uma rede neural.

Tabela 1 – Modelos da camada de entrada da RNA

Modelo	Entradas	Saída
1	*P(t)*	*Q(t)*
2	*P(t), Q(t-1)*	*Q(t)*
3	*P(t), P(t-1)*	*Q(t)*
4	*P(t), P(t-1), Q(t-1)*	*Q(t)*
5	*P(t), P(t-1), P(t-2)*	*Q(t)*
6	*P(t), P(t-1), P(t-2),Q(t-1)*	*Q(t)*
7	*P(t), P(t-1), P(t-2),Q(t-1), Q(t-2)*	*Q(t)*
8	*P(t), P(t-1), P(t-2), P(t-3)*	*Q(t)*
9	*P(t), P(t-1), P(t-2), P(t-3), Q(t-1)*	*Q(t)*

10	*P(t), P(t-1), P(t-2), P(t-3), Q(t-1), Q(t-2)*	*Q(t)*
11	*P(t), P(t-1), P(t-2), P(t-3), P(t-4)*	*Q(t)*
12	*P(t), P(t-1), P(t-2), P(t-3), P(t-4), Q(t-1)*	*Q(t)*
13	*P(t), P(t-1), P(t-2), P(t-3), P(t-4), Q(t-1), Q(t-2)*	*Q(t)*
14	*P(t), P(t-1), P(t-2), P(t-3), P(t-4), P(t-5)*	*Q(t)*
15	*P(t), P(t-1), P(t-2), P(t-3), P(t-4), P(t-5), Q(t-1)*	*Q(t)*
16	*P(t), P(t-1), P(t-2), P(t-3), P(t-4), P(t-5), Q(t-1), Q(t-2)*	*Q(t)*
17	*P(t), P(t-1), P(t-2), P(t-3), P(t-4), P(t-5), Q(t-1), Q(t-2), Q(t-3)*	*Q(t)*
18	*P(t), P(t-1), P(t-2), Q(t-1), Q(t-2), PMA7*	*Q(t)*
19	*P(t), P(t-1), P(t-2), P(t-3), P(t-4), P(t-5), Q(t-1), Q(t-2), Q(t-3), Q(t-4)*	*Q(t)*

A taxa de aprendizagem e o momentum foram definidos de forma incremental. A taxa de aprendizagem inicial foi definida em 0.05, onde a partir de 0.2 não se observou melhoria no resultado. De modo semelhante, o valor do momentum foi iniciado em 0.1 e gradualmente incrementado até atingir o valor ótimo de 0.9. A taxa de aprendizado foi investigada de modo incremental, sem a utilização de momentum. Depois de obtida a taxa de aprendizagem, foi então investigada a taxa de momentum. Mantendo a taxa de aprendizado fixo, variou-se o momentum no intervalo de [0,1..0,9].

5. Resultados e discussão

Para avaliação do desempenho de cada modelo, foram utilizados 2 critérios de avaliação: índice de eficiência de *Nash-Sutcliffe* (R^2) (equação 2) e RMSE (equação 3),

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(Q_i - \hat{Q}_i)^2}{\sum_{i=1}^{N}(Q_i - \overline{Q}_i)^2} \quad (2)$$

Onde Q_i é a vazão observada, $\hat{Q}_i$ é a vazão calculada, $\overline{Q}_i$ é a média de vazão e *N* é o número de observações. O índice de eficiência de *Nash-Sutcliffe* pode variar de -∞ *a 1*. Um índice igual a *1* corresponde a uma relação perfeita entre a vazão modelada e a vazão observada. Em síntese, quanto mais próximo de *1* o modelo atingir, mais preciso será. Para avaliação da progressão do erro entre valores observados e previstos, utilizou-se o RMSE (*root mean square error*):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Q_i - \hat{Q}_i)^2}{N}} \quad (3)$$

Onde *N* é o número total de observações, Q_i é a vazão observada e $\hat{Q}_i$ é a vazão calculada. Quanto menor o erro ou mais próximo de zero, melhor será o ajuste dos dados ao modelo obtido.

5.1 Análise preliminar

Os melhores modelos de entrada de dados selecionados para as simulações foram aqueles que utilizavam além da precipitação, também a vazão como entrada na RNA, ou seja, descartaram-se os modelos que possuíam apenas a precipitação como dado de

entrada, que nesse caso foram os modelos 1, 3, 5, 8, 11 e 14. Esses modelos apresentaram resultados inferiores a 0,5 em termos de R^2.

5.2 Fase de testes

A seguir são apresentados os resultados da avaliação da fase de teste dos melhores modelos, selecionados na etapa de treinamento. Nessa fase, é apresentado um conjunto de dados reservados, que não foram apresentados anteriormente para a RNA, e portanto, será usado nesse caso para verificação da capacidade de generalização de cada modelo.

5.2.1 Modelos AG-RNA em escala diária

Usando AG-RNA em escala diária, o modelo 9 apresentou um R^2 de 0,856 (Figura 3) e RMSE de 0,01417, apresentando em alguns pontos da aproximação, um erro principalmente nos picos de vazão, conforme Figura 2, e boa representação nos períodos de baixa vazão. O gráfico da Figura 2 mostra um erro maior da previsão do modelo, nas vazões mais altas. Nesse caso, o modelo previu vazões acima daquelas observadas.

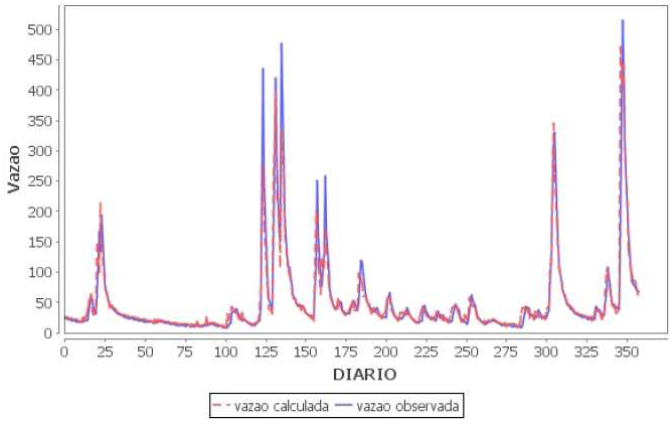


Figura 2 – Comparação entre vazão observada e calculada usando AG-RNA

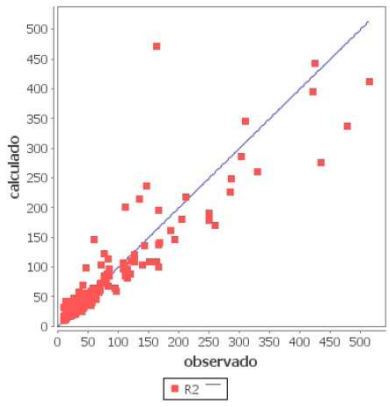


Figura 3 – Coeficiente R^2 - AG-RNA

5.2.2 Modelos RNA em escala diária

O melhor modelo RNA obtido em escala diária foi o modelo 4, que apresentou R^2 de 0,811 (Figura 5) e RMSE de 0,04724. A Figura 4 mostra um período de teste de 1 ano de previsão (vazão calculada) feita pelo modelo, em comparação com os dados observados (vazão observada). Nesse modelo houve uma superestimação maior dos picos de vazão em comparação ao modelo AG-RNA.

5.3. Resultados comparativos

Na etapa de teste da rede, o modelo híbrido AG-RNA apresentou melhores resultados de convergência quando comparado ao de uma RNA, sem AG, conforme apresentado na Tabela 2, ou seja, os modelos AG-RNA conseguiram uma melhor generalização dos dados em relação aos modelos RNA.

A principal vantagem no uso do AG-RNA está na rapidez de convergência. Em diversas simulações o modelo RNA não conseguia convergir, devido a uma má atribuição da matriz de pesos ficando preso num mínimo local. Isso exigia reinicialização da matriz de pesos, para busca num ponto diferente do espaço de estados. Em algumas situações essa matriz precisou ser reiniciada 4 vezes. O AG-RNA precisou de apenas 1 inicialização para conseguir convergência. Observa-se também que existe uma

correspondência entre o erro apresentado na fase de treino e o erro apresentado na fase de teste. Isso significa que a rede está conseguindo uma boa generalização [Atiya 1999].

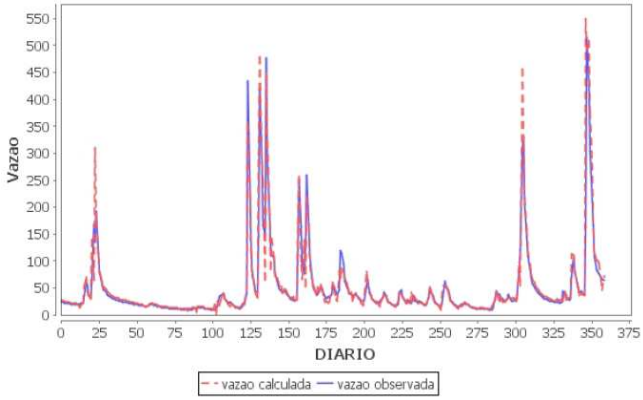


Figura 4 – Comparação entre vazão observada e vazão calculada (RNA)

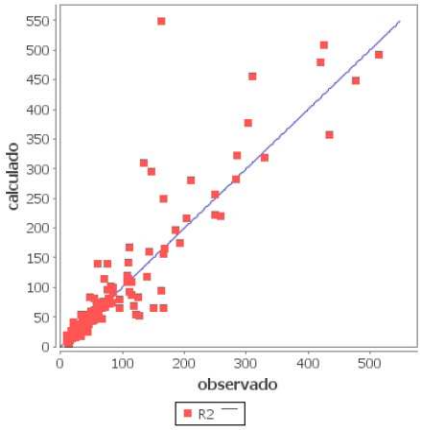


Figura 5 – Coeficiente R^2 - RNA

Tabela 2 - Comparação entre RNA e AG-RNA

Técnica	Modelo	Arquitetura	RMSE	R^2
AG-RNA	9	5-5-1	0,01417	0,856
RNA	4	3-4-1	0,04724	0,811

6. Conclusão

A vantagem do uso de RNAs na modelagem chuva-vazão está principalmente ligada à redução da dimensionalidade do fenômeno a duas variáveis. Nesse trabalho, utilizou-se apenas a precipitação e a vazão para treinamento da rede. Nesse cenário a RNA acoplada a um AG, conseguiu realizar uma boa aproximação, sendo que no melhor modelo obteve um coeficiente R^2 de 0.856 em escala diária. Esse resultado foi superior em relação a modelagem utilizando somente RNA, que apresentou um coeficiente R^2 de 0.811. Os resultados obtidos foram considerados bons quando se analisa a baixa correlação entre as duas séries temporais chuva-vazão, no caso de 0.505 e também quando se considera a área total da bacia hidrográfica utilizada como elemento de estudo de caso.

Uma desvantagem do uso de RNAs para a modelagem reside na questão da abordagem tipo caixa preta. Nesses tipos de modelos, obtemos a relação entre uma ou diversas variáveis, não sendo possível, entretanto, identificar como essa relação foi obtida, ou seja, qual o processo físico que melhor descreve o fenômeno. Um outro ponto está na questão da disponibilidade dos dados ou no tamanho da série temporal. Se houver escassez de dados, isso pode prejudicar o aprendizado da RNA, tendo impacto nesse caso, na generalização da rede para eventos onde a rede não foi ensinada. Isso pode levar a um modelo fraco.

O uso dos algoritmos genéticos acoplado a uma RNA mostrou-se interessante permitindo um melhor desempenho da modelagem, entretanto, adiciona-se mais uma etapa de processamento para iniciar a matriz de pesos. Isso deve ser considerado principalmente quando o tamanho do conjunto de dados para treinamento for muito grande, o que pode aumentar o tempo de processamento.

Agradecimentos

Agradecemos a Fundação Parque Tecnológico Itaipu (FPTI), pelo apoio financeiro concedido para desenvolvimento desta pesquisa.

Referências

Anders, Ulrich and Korn, Olaf. Model selection in neural networks. Neural Networks, 12(2):309-323, 1999.

Atiya, Amir F. e Shaheen, Samir I.. A Comparison Between Neural-Network Forecasting Techniques - Case Study: River Flow Forecasting. IEEE Transactions on Neural Networks, 10:402-409, mar 1999.

Back, T. and Hammel, U. and Schwefel H. P. Evolutionary computation: comments on the history and current state. IEEE Transactions on Evolutionary Computation, 1(1):317, April 1997.

Basheer, I.A. and Hajmeer, M. Artificial neural networks: fundamentals, computing, design, and application. Journal of Microbiological Methods, 43:3-31, dec 2000.

Campolo, Marina and Andreussi, Paolo and Soldati, Alfredo. River flood forecasting with a neural network model. Water Resources Research, 35(4):1191-1197, 1999.

Cybenko, G. Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals, and Systems, 2(4):303-314, December 1989.

Dawson, Christian W. and Wilby, Robert. An artificial neural network approach to rainfall runoff modelling. Hydrological Science, (43):47-66, August 1998.

Haykin, Simon. Redes Neurais: Princípios e prática. Bookman, second edition, 2008.

Kwok-Wing; Cheng Chun-Tian e Qiu Lin Wang, Wen-Chuan; Chau. A comparison of performance of several arti_cial intelligence methods for forecasting monthly discharge time series. Journal of Hydrology, 374(3-4):294{306, August 2009.

Nasseri, M. and Asghari, K. and Abedini, M. Optimized scenario for rainfall forecasting using genetic algorithm coupled with artificial neural network. Expert Systems with Applications, 35(3):1415-1421, October 2008.

Sedki, A. and Ouazar, D. and Elmazoudi, E. Evolving neural network using real coded genetic algorithm for daily rainfall runoff forecasting. Expert Systems with Applications, 36(3):4523-4527, April 2009.

Rezende, Solange Oliveira. Sistemas Inteligentes: Fundamentos e Aplicações. Ed. Manole, 2005.

Silva, Leandro Nunes de Castro. Análise e Síntese de Estratégias de Aprendizado para Redes Neurais Artificiais, Dissertação (Mestrado), Universidades Estadual de Campinas. 1998.

Tokar, A. Sezin and Johnson, Peggy A.. Rainfall-runoff modeling using artificial neural networks. Journal of Hydrologic Engineering, 4(3):232-239, 1999.

Yao, Xin. Evolving artificial neural networks. Proceedings of the IEEE, 87(9):1423-1447, Sept 1999.

Processamento e Visualização de Dados para a Descoberta de Conhecimento em Sistemas de Monitoramento de Qualidade de Água

Aysy Anne Andrade Duarte[1], Leonardo Bertholdo[1], Gisela de Aragão Umbuzeiro[1], Luiz Camolesi Júnior[1], Celmar Guimarães da Silva[1]

[1]Faculdade de Tecnologia – Universidade Estadual de Campinas (Unicamp)
Limeira – São Paulo – Brazil

{celmar,giselau,camolesi}@ft.unicamp.br,
{a065489,l016559}@pos.ft.unicamp.br

Abstract. *This paper proposes the use of visualization techniques and data mining in knowledge discovery in the field of data for monitoring water quality. At this stage of research, we discuss some visualization techniques to represent and identify patterns of behavior in these data. For this study, we selected, preprocessed and transformed a large set of data on measurements of water quality of rivers in the state of São Paulo. Thus, it is hoped that similar studies may be replicated and used in broader areas, regardless of the quality parameters of water involved.*

Resumo. *Este trabalho propõe o uso de técnicas de visualização e mineração de dados na descoberta de conhecimento no domínio de dados de monitoramento de qualidade de água. No atual estágio da pesquisa, são analisadas algumas técnicas de visualização para a representação e identificação de padrões de comportamento nestes dados. Para realização deste trabalho, foi selecionado, preprocessado e transformado um grande conjunto de dados relativos a medições de qualidade de água dos rios do estado de São Paulo. Com isso, espera-se que estudos semelhantes possam ser reproduzidos e empregados em áreas mais amplas, independentemente dos parâmetros de qualidade de água envolvidos.*

1. Introdução

A água é um elemento vital para a existência e a manutenção da vida em nosso planeta. No âmbito humano, além de viabilizar nossa sobrevivência, ela também torna possíveis as mais diversas atividades, como abastecimento público e industrial, irrigação agrícola, produção de energia elétrica e atividades de lazer e recreação. Além disso, trata-se de um recurso natural limitado pelo alto custo da sua obtenção nas formas menos convencionais, como as águas marinhas e subterrâneas. Não obstante a esta realidade, a expansão demográfica e industrial das últimas décadas ocasionou o comprometimento de muitos corpos d'água, como rios, lagos e reservatórios. Sendo assim, o controle da qualidade dessas águas, também conhecidas como águas doces superficiais, é de fundamental relevância para preservação deste bem [Alves et al. 2008].

Atualmente, a Companhia Ambiental do estado de São Paulo (CETESB) é responsável pelo levantamento de informações sobre a qualidade das águas dos corpos

d'água do estado de São Paulo. Para isso, conta com mais de 300 pontos de coleta de amostras, as quais são analisadas sob aspectos físicos, químicos e biológicos, formando um conjunto de dados rico em informações relativas às condições dos corpos d'água analisados [CETESB 2010]. Apesar de estes dados serem essencialmente valiosos, quando analisados de forma individual e por meio de técnicas tradicionais de consulta, a descoberta de informações não triviais em meio a estes dados torna-se bastante difícil. No entanto, existem técnicas que podem ajudar a compreender de forma mais abrangente o significado das informações levantadas.

Entre estas técnicas, a visualização de informação é uma abordagem que, por meio do mapeamento dos dados para formas gráficas e interativas, pode auxiliar na interpretação de um determinado conjunto de dados, além de proporcionar importantes descobertas. Outro enfoque que pode trazer à tona informações antes desconhecidas e úteis é a mineração de dados que, por meio da associação de diferentes áreas de conhecimento, propõe métodos para descoberta de informações implícitas em repositórios de dados.

Visando a descoberta de informações implícitas em meio aos dados de monitoramento de qualidade de água, esta pesquisa consiste na análise dos dados através do uso de técnicas de visualização que podem contribuir na descoberta de conhecimento útil neste domínio de dados. A partir disso, espera-se que este trabalho possa servir de insumo para pesquisas semelhantes em outros programas de monitoramento ambiental.

Este artigo apresenta os resultados iniciais desta pesquisa, iniciando pela Seção 2, que apresenta a caracterização do domínio de dados. Em seguida, a Seção 3 descreve o processo de descoberta de conhecimento utilizado para a preparação dos dados. Na Seção 4 é apresentada a aplicação de técnicas de visualização nos dados escolhidos e ainda o processo criado para a aplicação dessas técnicas em dados de monitoramento de qualidade de água. A Seção 5 menciona alguns trabalhos relacionados a esta pesquisa e, por fim, a Seção 6 apresenta as considerações finais referentes a este trabalho.

2. Domínio de Dados

Desde 1974, a CETESB vem coletando informações sobre a qualidade das águas doces superficiais por meio de uma ampla rede de monitoramento. Para delimitar as áreas de monitoramento, o estado de São Paulo foi dividido em 22 Unidades de Gerenciamento de Recursos Hídricos (UGRHIs). Cada uma destas UGRHIs conta com vários pontos de amostragem, de onde são coletadas as amostras de água que, posteriormente, serão analisadas em laboratório [CETESB 2010]. A Figura 1 mostra esta divisão, classificando as UGRHIs em grupos conforme suas respectivas vocações.

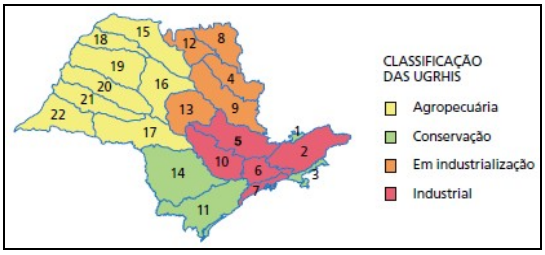


Figura 1. Classificação das 22 UGRHIs por vocação [CETESB 2010]

A análise das amostras de água feita pela CETESB contempla dezenas de parâmetros, os quais podem estar relacionados a aspectos físicos, químicos, microbiológicos, hidrobiológicos e toxicológicos. Em cada ponto de amostragem é utilizada apenas uma parte destes, sendo que o conjunto de parâmetros analisados pode variar consideravelmente de um ponto para outro. Anualmente, a CETESB disponibiliza, publicamente na Internet, as análises realizadas em cada ponto de amostragem por meio de arquivos em formato PDF. Cada arquivo contém todas as análises realizadas nos parâmetros de qualidade considerados para o ponto de amostragem associado. Estima-se que somente a rede básica, que visa essencialmente a análise da água dos rios do estado, gere um volume de dados anual de 62.000 análises [CETESB 2010], considerando que cada análise corresponde a uma medição de um parâmetro em um ponto de amostragem, realizada em uma data específica.

3. Descoberta de Conhecimento

Em decorrência do volume do conjunto de dados a ser usado no processo de descoberta de conhecimento, e do formato em que são disponibilizados, notou-se a necessidade de preparar e organizar esses dados, viabilizando assim a aplicação das técnicas de visualização e mineração de dados e, consequentemente, o levantamento de informações novas e relevantes. Mesmo essas técnicas não dispensam o papel dos especialistas no domínio tratado, cujas habilidades auxiliam na escolha do conjunto de dados a ser estudado, na definição do tipo de conhecimento a ser descoberto e como tal conhecimento pode contribuir no suporte a decisões.

3.1. Seleção, Pré-processamento e Transformação dos Dados

Devido aos formatos nativos em que os dados estavam armazenados, uma das primeiras necessidades deste trabalho foi a extração das informações dos documentos (a maioria em formato PDF). Esta tarefa consistiu primeiramente na conversão destes arquivos para arquivos-texto e, em seguida, na transformação dos dados brutos em scripts SQL para inserção em um banco de dados centralizado. A primeira parte da extração foi realizada com o auxílio de um conversor de formatos *freeware*, já a segunda parte foi executada por meio de um *script* desenvolvido em linguagem C. Paralelamente a esta tarefa, foi realizada a primeira etapa do processo de descoberta de conhecimento – a seleção do cenário de dados. Com o valioso auxílio de especialistas na área de saneamento ambiental, foi definido o seguinte escopo para esta pesquisa:

- **Pontos de amostragem:** Foram escolhidos os pontos da rede básica, a qual abrange mais de 80% dos pontos da rede de monitoramento da CETESB. Este tipo de rede visa unicamente a análise da água dos rios do estado de São Paulo, não contemplando análises de sedimentos e balneabilidade destes rios, tampouco análises oriundas de sistemas de monitoramento automático, os quais são tratados por outros tipos de rede.
- **Parâmetros de qualidade:** Foram selecionados 25 parâmetros considerados os mais significativos e com maior potencial de trazer à tona informações relevantes durantes os processos de visualização e mineração de dados. Segundo os especialistas consultados, o critério de escolha baseou-se na forte tendência de relacionamento entre estes parâmetros e as características

qualitativas dos corpos d'água. São eles: Absorbância no Ultravioleta, Alumínio Dissolvido, Cádmio Total, Chumbo Total, Chuva, Cloreto Total, Cobre Dissolvido, Condutividade, Cromo Total, Ferro Dissolvido, Manganês Total, Níquel Total, Nitrato, Nitrito, Nitrogênio Amoniacal, Oxigênio Dissolvido, pH, Potássio, Sódio, Sólidos Totais, Substância Tensoativa, Temperatura da Água, Toxicidade, Turbidez, Zinco Total.

- **Anos das análises:** Foram contempladas as análises realizadas entre os anos de 2005 a 2009. Apesar de a CETESB publicar dados de análises a partir do ano 2000, utilizamos somente os últimos cinco anos disponíveis para restringir esta pesquisa à realidade mais recente dos corpos hídricos.

4. Processo de Transformação de Dados em Visualizações

Uma vez definido o conjunto de dados a ser utilizado nesta pesquisa, foi percebido que a aplicação de técnicas de visualização de informações nesses dados não era um processo trivial, que pudesse simplesmente ser delegado ao especialista em monitoramento ambiental utilizando uma ferramenta de visualização. Porém, com o auxílio de especialistas em visualização de informação, essas ferramentas deveriam ser pré-configuradas para facilitar o processo de análise e identificação de padrões e tendências válidos, novos (antes desconhecidos), potencialmente úteis e compreensíveis para tomar decisões. Essa pré-configuração caracterizou um processo de transformação de dados em visualizações, cujas etapas são apresentadas pela Figura 2. Nas subseções seguintes serão apresentadas algumas das etapas deste processo.

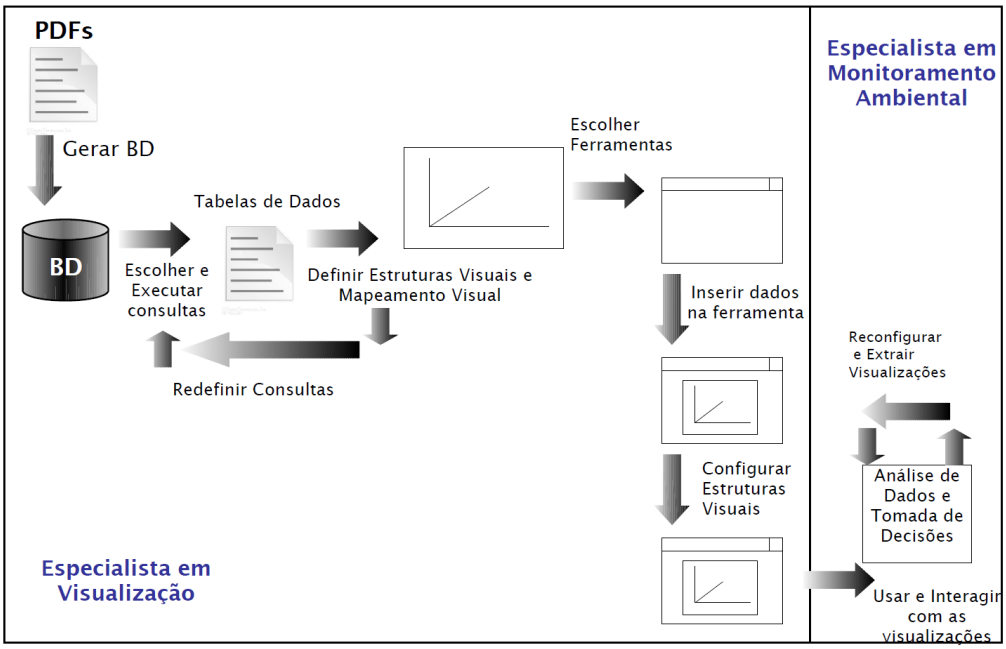


Figura 2. Processo de Transformação de dados em Estruturas Visuais

4.1 Geração do Banco de Dados

Após a seleção, pré-processamento e transformação dos dados, apresentados na Seção 3, foram efetuadas a limpeza e o pré-processamento dos dados brutos, visando eliminar erros e inconsistências que pudessem impactar nos resultados da mineração e

visualização dos dados. Para que os dados já tratados pudessem ser armazenados, foi projetada e implementada uma base de dados no sistema gerenciador de banco de dados relacional PostgreSQL. O projeto lógico desta base de dados é composto por sete tabelas, apresentadas na Figura 3, envolvendo pontos amostrais e suas regiões (UGHRIs), parâmetros e seus respectivos tipos, análises efetuadas (medidas), e padrões para classificação das medidas obtidas para cada parâmetro.

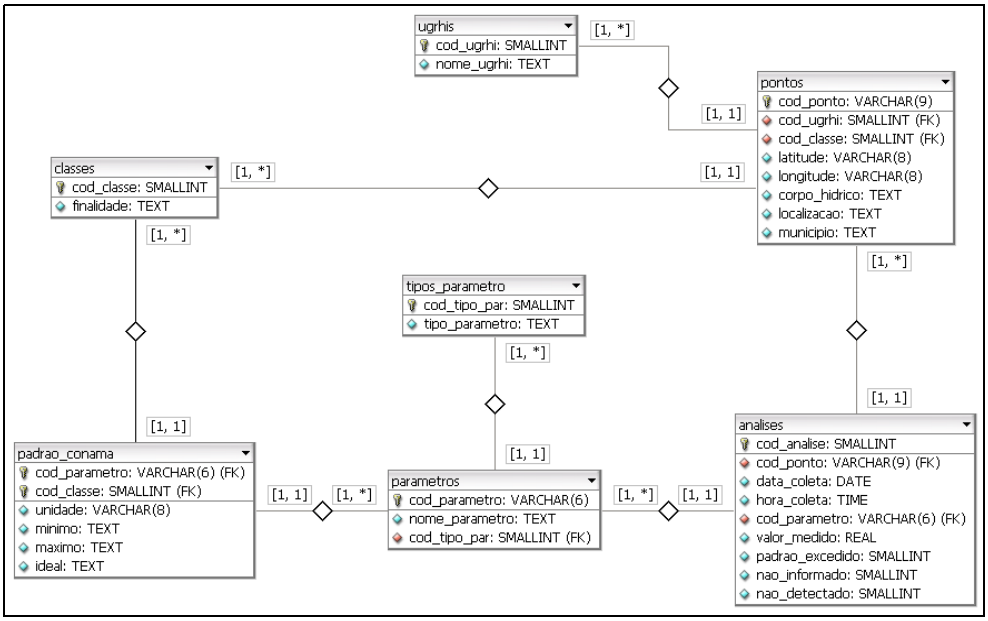


Figura 3. Projeto lógico do banco de dados

4.2 Escolha e Execução de Consultas

Uma vez projetado e implementado o banco de dados, foram feitas consultas simplificadas para identificação de quais tabelas de dados resultantes dessas consultas seriam as mais relevantes. Essas consultas eram redefinidas e executadas novamente até a obtenção dos resultados esperados. Um exemplo da tabela com os dados resultantes pode ser visto na Figura 4, contendo valores medidos em um ponto amostral em uma determinada data para alguns elementos químicos.

```
"cod_ponto";"data_coleta";"ph";"cadmio";"chumbo";"cobre";"zinco"
"TIET04170";"2006-01-11";"6,9";"0,008";"sem dados";"0,02";"0,09"
"TIET04170";"2006-03-16";"7,0";"0,0001";"0,008";"0,02";"0,1"
"TIET04170";"2006-05-02";"7,3";"0,0002";"0,005";"0,04";"0,11"
```

Figura 4: Exemplo de tabela de dados resultante das consultas

4.3 Definição de Estruturas Visuais e Mapeamento Visual

Foram realizadas experimentações com algumas estruturas visuais (a serem utilizadas para representar os dados de forma gráfica e interativa) para identificação das mais adequadas à representação do conjunto de dados. Tendo por base as variáveis Ponto Amostral (A), Tempo (T), Parâmetro (P) e Valor Medido (V), optou-se pelo uso de estruturas visuais multidimensionais, possibilitando a análise em mais de duas variáveis. A escolha dessas estruturas obrigou o refinamento das consultas da Seção 4.2, de forma a haver ainda: variáveis representando os valores medidos em um parâmetro específico

(VP_n) para um dado ponto amostral e um dado tempo; e variáveis representando valores medidos em um ponto amostral (VA_n), para um dado parâmetro e para um tempo especificado. Combinou-se essas seis variáveis em 3 conjuntos, que representam uma das tabelas de dados geradas nas consultas: {A,T,P,V}, sendo {A,T,P} chave primária; {A,T,VP_1,VP_2,VP_3...,VP_N}, sendo {A,T} chave primária; e {T,P,VA_1,VA_2,VA_3...,VA_N}, sendo {T,P} chave primária.

Com base nesses 3 conjuntos selecionados, foram realizados novos testes com diversas técnicas multidimensionais, sendo que as técnicas Heat Map, Scatterplot Matrix e Coordenadas Paralelas foram empiricamente escolhidas por terem conseguido melhor representar os dados de monitoramento ambiental.

Após a escolha das estruturas visuais, é necessário relacionar qual variável será representada por qual característica gráfica e espacial dessa estrutura. A Figura 5 mostra como exemplo um dos mapeamentos definidos para a técnica de Coordenadas Paralelas, que representa cada variável por um eixo vertical; no exemplo, a tabela de dados usada fornece 5 variáveis {A,T,VP_1,VP_2,VP_3} para a estrutura visual, evidenciando que os valores de VP3 são semelhantes independentemente dos valores das demais variáveis.

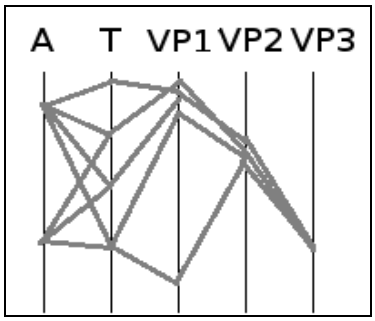


Figura 5: Um possível mapeamento visual em Coordenadas Paralelas.

4.4 Escolha de Ferramentas

Existem diversas ferramentas que representam dados multidimensionais e também contemplam pelo menos uma das técnicas escolhidas. Entre essas ferramentas estão: Spotfire (Tibco Software Inc., 2011), Tableau Public (Tableau Software, 2011), GGobi (GGobi, 2011), Parvis (Ledermann, 2011), Many Eyes (IBM, 2011), VisuLab (VisuLab, 2011) e Mondrian (Theus , 2011).

Com base na experiência prévia de alguns dos autores no uso de parte desse conjunto de ferramentas, escolheu-se as ferramentas Spotfire e Tableau Public. Estas implementam características relevantes no tocante a Visualização de Informação, especialmente as relacionadas a interação de usuários, como: filtragem, seleção direta, detalhes-sob-demanda, configuração de mapeamento visual (como definição de legendas; ou mesmo definição, via manipulação direta, de quais variáveis devem ser representadas nos eixos disponíveis na representação visual), capacidade de extração (ou seja, de salvar o gráfico em diferentes formatos, ou mesmo na Web), entre outros. Além disso, essas duas ferramentas, em conjunto, implementam as estruturas visuais definidas na seção anterior.

Alguns exemplos utilizando essas ferramentas podem ser observados na Figura 6, na qual é apresentada a visualização dos dados coletados para Condutividade, pH e

Zinco que representam os valores coletados para cada parâmetro (VP_1, VP_2, VP_3), no período de 2005 a 2009 nos pontos do Rio Paraíba do Sul. Tanto as datas (T) quanto os pontos de amostragem (A) estão mapeados como filtros na técnica Scatterplot Matrix usando a ferramenta Tableau Public. Outro exemplo pode ser visto na Figura 7 que apresenta a técnica de Coordenadas Paralelas sendo usada na ferramenta Parvis; no primeiro eixo estão representados os pontos de amostragem (A); no segundo, as datas das coletas (T); e no terceiro e quarto eixos, os dados de pH e Arsênio, respectivamente.

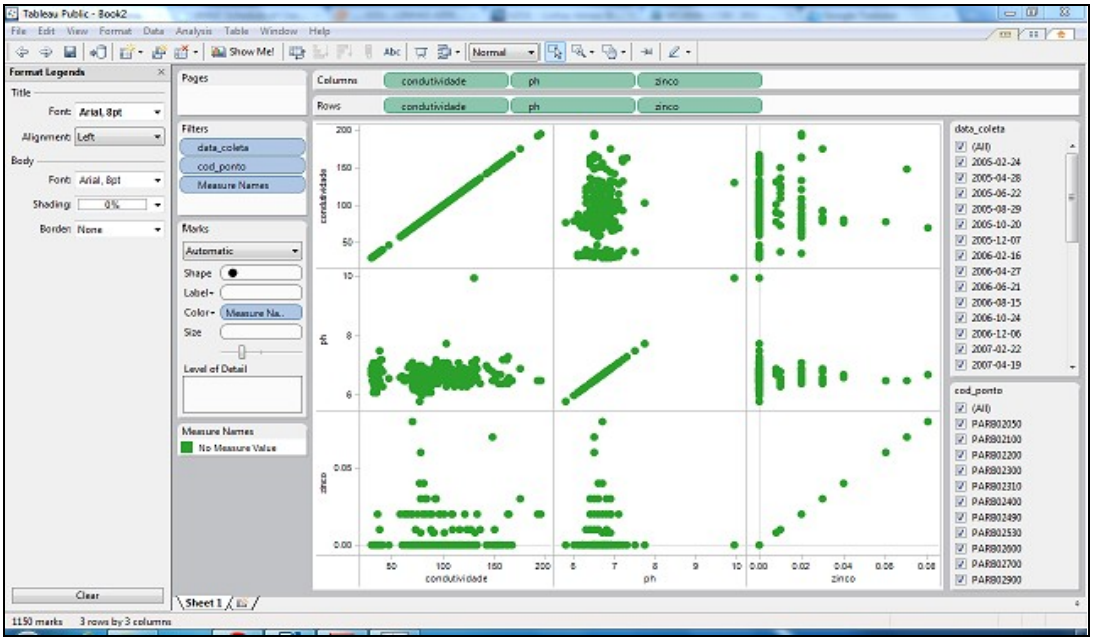


Figura 6: Ferramenta Tableau Public – Visualização usando a técnica Scatterplot Matrix

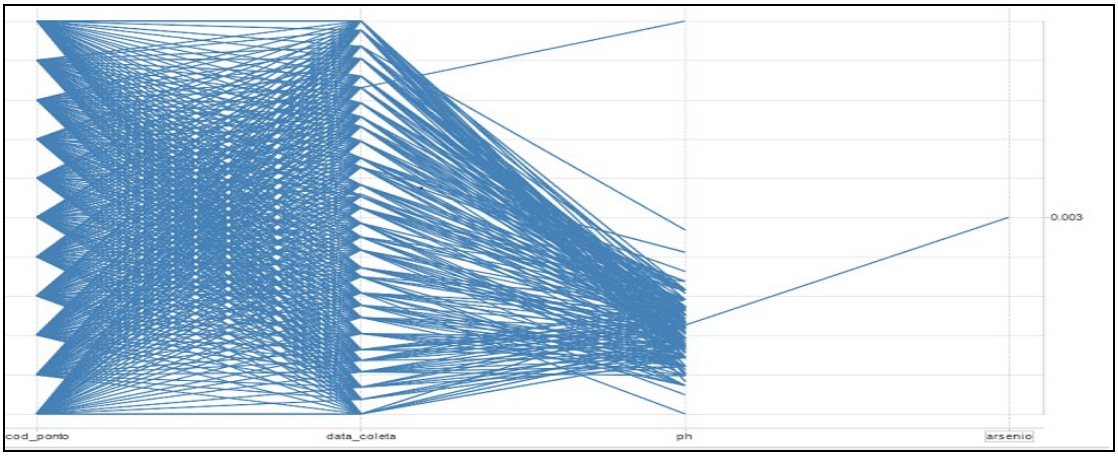


Figura 7: Visualização usando a técnica de Coordenadas Paralelas

4.5 Inserção de dados nas ferramentas e configuração de estruturas visuais

Atualmente o trabalho está na etapa de inserção dos dados nas ferramentas escolhidas usando as estruturas visuais selecionadas e configuração de cada uma delas em busca das visualizações mais adequadas às necessidades dos especialistas em monitoramento

ambiental. Após a criação dessas visualizações, pretende-se submetê-las a avaliações com usuários para comprovar se o resultado esperado já foi alcançado ou se as visualizações ainda necessitam melhores configurações para atender a análises do domínio de dados existente. A eficácia do uso de técnicas de visualização no contexto de monitoramento de qualidade da água será mensurada com base na análise de especialistas da área de monitoramento ambiental que irão avaliar os resultados obtidos a partir das visualizações.

5. Trabalhos Futuros

Em termos de processo de descoberta de conhecimento utilizando visualização de informações, ainda há a demanda pela interpretação e a avaliação das visualizações que estão sendo obtidas pelo processo desenvolvido, para descoberta de conhecimento novo e útil. Em paralelo, será feita a aplicação de mineração de dados na mesma base de dados, através da seleção de algoritmos de mineração que possam agregar conhecimento no apoio aos especialistas em controle ambiental. A mineração de dados consiste na exploração e análise de grandes quantidades de dados, visando a descoberta de padrões e regras significativas [Berry et al. 2004]. Para isso, utiliza-se algoritmos e técnicas de diferentes áreas do conhecimento como: estatística, banco de dados, reconhecimento de padrões, inteligência artificial, visualização de informação, aprendizagem de máquina, computação distribuída, entre outras.

Normalmente, as tarefas da mineração de dados são implementadas por meio de algoritmos, os quais podem ter as mais diversas finalidades e enfoques. A metodologia para escolha dos algoritmos que serão implementados neste trabalho será baseada na revisão bibliográfica de artigos sobre mineração de dados, que visará identificar quais algoritmos podem ser mais adequados para descoberta de conhecimento no domínio de monitoramento de qualidade de água. Em outras palavras, serão pesquisados algoritmos que já tenham sido utilizados em dados com características semelhantes àquelas presentes nos dados desta pesquisa. Além disso, também se pretende utilizar ferramentas já existentes de mineração de dados, que servirão para experimentar a performance e a conveniência de alguns algoritmos neste cenário de dados.

6. Trabalhos Relacionados

Existem vários modelos de visualização para aplicações específicas em meteorologia, hidrologia, e outros sistemas científicos e de engenharia. Sreevalsan-Nair et al. (2007) desenvolveram um paradigma semelhante para visualização de dados de séries temporais dos rios, reservatórios e outros componentes do sistema de água do norte da Califórnia. Essa rede de monitoramento fornece um conjunto de dados do chamado Delta Sacramento-San Joaquin, composto por reservatórios e pelos dois principais rios, Sacramento e San Joaquin. Esse paradigma desenvolvido é uma combinação de Visualização da Informação e métodos de Visualização Científica. Os dados consistem em medições diárias realizadas nos reservatórios no período de 1999 a 2006 que proveem uma variedade de conjuntos de dados de propriedades físicas e condições ambientais, chamado de "dados de qualidade da água".

Silva et al. (2010) propõem o uso de técnicas de Visualização de Informações para melhorar a análise visual de dados, visando facilitar a detecção de padrões e

tendências importantes em conjuntos de dados. Com base nessas técnicas, um protótipo foi implementado para organizar um conjunto de dados gerado em um programa de monitoramento de uma área do oceano que recebeu material dragado, o que ajudou os analistas na compreensão dos dados, obtidos a partir de amostras coletadas em nove locais de amostragem, de 29 campanhas de amostragem, onde vários produtos químicos e análises físicas foram realizadas, incluindo diversos metais e compostos orgânicos.

Belo et al. (2006) apresentam o sistema AQuA para análise e validação de parâmetros de qualidade da água. Este visa coletar os dados provenientes de leituras efetuadas pelas estações de coleta automática ou por técnicos de laboratório. Esses dados coletados são analisados e validados e, posteriormente, integrados e consolidados numa base de dados específica. O sistema oferece mecanismos de detecção de anomalias e permite a exploração dos dados, tendo o apoio de algumas funcionalidades de georreferenciamento, geração de relatórios e gráficos. Apesar de não implementar técnicas de mineração de dados, o *data warehousing* do sistema está preparado de forma a assegurar futuras tarefas de mineração de dados.

Karimipour et al. (2005) investigam a mineração de dados geoespaciais para gestão de dados ambientais e, especialmente, para gestão de qualidade de água. Um estudo de caso realizado na região entre o Azerbaijão e o Irã apresenta a correlação entre a poluição de centros industriais e indicadores de qualidade de água através de mineração de dados geoespaciais. Segundo o estudo, fica visível a relação entre o quantidade e a localização da poluição industrial e os indicadores de qualidade da água.

Com relação às pesquisas citadas, este trabalho diferencia-se pela proposta de uma análise comparativa de algumas das técnicas já conhecidas e utilizadas na área de visualização de dados. Ao fazer este paralelo, espera-se identificar as técnicas que se mostram com maior potencial de trazer à tona informações implícitas dentro do domínio específico de monitoramento de qualidade de águas superficiais.

7. Considerações Finais

Nota-se, pelo conjunto de trabalhos relacionados, o nível de relevância do tema abordado para a sua área de aplicação (Gestão Ambiental). As pesquisas em computação relacionadas à área aplicada, particularmente no Gerenciamento de Recursos Hídricos, como as apresentadas na Seção 6, têm preocupado a comunidade científica e os governos, devido às projeções de escassez futura de recursos hídricos destinados a atender às necessidades básicas da população. Esta pesquisa é resultado de um primeiro estágio de um projeto que ambiciona apoiar de forma intensa e eficaz a manutenção da qualidade de água dos rios.

Referências

Alves, E. C.; Silva, C. F.; Cossich, E. S.; Tavares, C. R. G.; Filho, E. E. S.; Carniel, A. Avaliação da qualidade da água da bacia do rio Pirapó – Maringá, Estado do Paraná, por meio de parâmetros físicos, químicos e microbiológicos. *Acta Scientiarum. Technology*, Maringá, v. 30, n. 1, p. 39-48. 2008.

Belo, O.; Lourenço, A.; Sarmento, P.; Magriço, A.; Pinho, J. L.; Lima, M.; Vieira, J. AQUA: um sistema de informação para análise e a validação de parâmetros de

qualidade da água em Alqueva. In: *Congresso da Água*, 8., 2006, 9p. Figueira da Foz, Portugal: Universidade do Minho. 2006.

Berry, M. J. A.; Linoff, G. S. (2004). *Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management*. Indianapolis:Wiley Publishing, Inc., 672 p.

CETESB. *Relatório de Qualidade das Águas Superficiais do Estado de São Paulo – 2009*. São Paulo: CETESB, 2010. Disponível em: <http://www.cetesb.sp.gov.br/agua/aguas-superficiais/35-publicacoes-/-relatorios>. Acesso em: 23 mar. 2011.

GGobi. Site do software GGobi: http://www.ggobi.org/ Acesso em: 30 mar. 2011.

IBM. Site do software Many Eyes: http://www-958.ibm.com/software/data/cognos/manyeyes/ Acesso em: 30 mar. 2011.

Karimipour, F.; Delavar, M. R.; Kinaie, M. Water Quality Management Using GIS Data Mining. *Journal of Environmental Informatics*. Canadá, v. 5, n. 2, p. 61-71. 2005.

Ledermann, F. Parvis - Parallel coordinates visualization. Site do software Parvis: http://www.mediavirus.org/parvis/ Acesso em: 30 mar. 2011.

Prass, F. S. *KDD: Processo de descoberta de conhecimento em bancos de dados*. Grupo de Interesse em Engenharia de Software, Florianópolis, v.1, p. 10-14. 2004.

Silva, C. G.; Umbuzeiro, G. A. *Enhancing visual analysis of environmental monitoring data through an Information Visualization based prototype*. WCAMA-II Workshop de Computação Aplicada à Gestão do Meio Ambiente e Recursos Naturais, 2010

Spence, R. *Information Visualization*. Addison Wesley. 2001.

Sreevalsan-Nair, J.; Nieuwenhuyseb, E. V.; Hotzc, I.; Linsend, L.; Hamanna, B., *An Interactive Visual Exploration Tool for Northern California's Water Monitoring Network*, 2007.

Tableau Software. Site do software Tableau Public. Disponível em: http://www.tableausoftware.com/public Acesso em: 30 mar. 2011.

Theus, M. Mondrian - Interactive Statistical Data Visualization in JAVA. Site do software Mondrian: http://rosuda.org/Mondrian/ Acesso em: 30 mar. 2011.

Tibco Software Inc. Site do software Spotfire. Disponível em: http://spotfire.tibco.com/ Acesso em: 30 mar. 2011.

VisuLab. Site do software Visulab: http://www.inf.ethz.ch/personal/hinterbe/Visulab/ Acesso em: 30 mar. 2011.

ISFS_Sist: Um software para Diagnosticar a Gravidade do Fenômeno da Seca no Semiárido Nordestino

Josemir A. Neves[1], Sílvio de B. Melo [2], Everardo V. S. B. Sampaio [3]

[1] Empresa de Pesquisa Agropecuária do Rio Grande do Norte
Caixa postal 188, 59158-160, Parnamirim – RN - Brasil.

Email: josemiremparn@rn.gov.br

[2] Centro de Informática (CIN) - Universidade Federal de Pernambuco (UFPE)
Av. Prof. Moraes Rego, 1235, Cidade Universitária, 50670-901, Recife – PE - Brasil.

Email: sbm@cin.ufpe.br

[3] Departamento de Energia Nuclear (DEN) – UFPE
Av. Prof. Luis Freire, 1000, Cidade Universitária, 50740-540, Recife – PE - Brasil.

Email: esampaio@ufpe.br

Abstracts – This article describes the computer application called ISFS_Sist which compute the Index of Susceptibility to Drought (ISD) developed by Neves (2010). The ISD was development following the method proposed by the JRC / OECD, using techniques for multiple imputation and data winsorization. The application was developed in the environment .NET Framework 2.0 in C#, using the R software such as statistical and graphical package. The ISFS_Sist allows easy data entry, instant calculation of the index and the generation of maps, charts and detailed reports that will provide subsidies for public managers in decision making upon the occurrence of a drought in the region. The model was calibrated and validated with data from the Rio Grande do Norte State and in this paper gives some examples of application software for the region.

Resumo - Este artigo descreve o aplicativo computacional, denominado ISFS_Sist, que implementa o Índice de Susceptibilidade ao Fenômeno da Seca (ISFS) desenvolvido por Neves (2010). O ISFS foi elaborado seguindo metodologia proposta pelo JRC/OECD, utilizando técnicas de preenchimento de dados com falhas e winsorização. O aplicativo foi desenvolvido no ambiente .NET framework 2.0, na linguagem C#, utilizando o software R como pacote estatístico e gráfico. O ISFS_Sist possibilita uma fácil entrada de dados, cálculo instantâneo do índice e a geração de mapas, gráficos e relatórios detalhados que fornecerão subsídios aos gestores públicos na tomada de decisão quando da ocorrência de uma seca na região. O modelo foi ajustado e validado com dados do Estado do Rio Grande do Norte e neste trabalho são mostrados alguns exemplos da aplicação do software para a região.

1. Introdução

O fenômeno da seca é caracterizado pela ausência parcial, total ou pela distribuição irregular das chuvas durante o período chuvoso. É cíclico e característico das regiões semiáridas espalhadas pelo mundo. Ao contrário de outros desastres naturais, tais como enchentes, terremotos e furações, a seca tem um efeito silencioso, porém não menos grave que esses fenômenos. A explicação está no seu efeito lento, na sua periodicidade, nas medidas políticas assistencialistas que mantém a população local presa à ajuda governamental e nas migrações dos trabalhadores para áreas de maior fluxo econômico, gerando diversos problemas sociais, como o aumento das favelas nas grandes cidades. Dessa forma, a gravidade de uma seca também está associada às condições sócio-econômico-culturais das populações atingidas, embora seja a ambiental, pela escassez de água, o fator ecológico limitante para plantas, animais e o próprio homem durante esses eventos [Dantas, 1961; Guerra, 1981; Duarte, 2002; Freitas, 2005].

No Brasil, a região semiárida nordestina é a mais afetada pela seca datando do século XVI o primeiro registro histórico de sua ocorrência [Dantas, 1961; Guerra, 1981; Carvalho, 1988]. Desde então, a seca tem sido registrada repetidamente na região, com efeitos danosos principalmente na agricultura, na pecuária e no suprimento hídrico das populações urbanas e rurais. O principal fenômeno meteorológico associado à ocorrência das secas no semiárido nordestino é o El Niño-Oscilação Sul (ENOS). Ele tanto causa secas no Nordeste brasileiro e em muitas regiões ao redor do mundo, como inundações em outras, exemplo das regiões Sul e Sudeste do Brasil. Durante a fase positiva do ENOS, a região semiárida nordestina experimenta uma diminuição sazonal das chuvas e consequentemente um aumento de intensidade das secas [Alves, 1998; Brunini, 2005; Mendonça; Danni-Oliveira, 2007; Araújo, 2008].

Nos últimos anos ocorreram avanços significativos no monitoramento de eventos climáticos no semiárido nordestino. Exemplo disso é o monitoramento de tempo e clima realizado pelo CPTEC/INPE, com a participação dos Centros de Meteorologia Estaduais do Nordeste, entre eles a Fundação Cearense de Meteorologia (FUNCEME). Um dos serviços resultante desse monitoramento é a previsão de ocorrência de secas. Ela possibilita o planejamento, na esfera governamental executiva, de ações de mitigação dos efeitos das secas, o que, na maioria das vezes, só ocorre quando a seca já está completamente configurada, a agropecuária em declínio e a população em estado de calamidade, com falta de água até para consumo humano.

Esse acompanhamento também é realizado através dos índices de secas, que são instrumentos que delineiam as informações relevantes através da sua agregação, quantificação e simplificação para o entendimento e compreensão do fenômeno. Em resumo, os índices de secas constituem ferramentas essenciais para um diagnóstico mais preciso da sua dimensão nas regiões atingidas, existindo para isso vários modelos de avaliação. Os índices de seca mais simples consideram somente variáveis meteorológicas e/ou hidrológicas. Os mais sofisticados associam a precipitação a parâmetros econômicos, sociais, culturais, ambientais e políticos das regiões atingidas [Byun; Whilite, 1999]. Esse é o caso do Índice FUNCEME de Severidade à Seca (IFS), desenvolvido em 2000 pela FUNCEME, e que considera além da climatologia, aspectos da física dos solos, econômicos, sociais, os riscos de perda da safra agrícola e a situação

do abastecimento de água urbano e rural dos municípios atingidos [Ensslin et al., 2000; FUNCEME, 2000; Silva; Forte, 2002], porém, possivelmente devido à complexidade da coleta de dados e à difícil operação, este não foi operacionalizado, estacionando na fase de protótipo.

A inerente complexidade do fenômeno da seca implica que nenhum índice é ideal para todas as regiões e quanto maior a gama de informações inter e intra-relacionadas, melhores resultados e respostas mais adequadas para a mitigação desses eventos serão obtidas, com estratégias de preparação e a redução dos impactos a eles associados [Smakhtin; Hugues, 2004].

Como conclusão, observa-se a necessidade do desenvolvimento de índices com dados e informações acessíveis, em softwares de fácil manuseio, que possibilitem sua implementação para uma melhor compreensão do fenômeno da seca na região semiárida nordestina. Eles podem vir a ser uma ferramenta para dar suporte aos tomadores de decisão, tanto na aplicação de recursos, como no direcionamento de ações efetivas para o enfrentamento desses eventos.

A partir desses pressupostos foi desenvolvido por Neves (2010) o Índice de Susceptibilidade ao Fenômeno da Seca (ISFS) e este artigo descreve o software ISFS_Sist que automatiza o seu cálculo.

2. Descrição do ISFS

O IFSF é constituído de sete sub-indicadores (Figura 1). Os dois primeiros, PV11 e PV12, representam as características físicas dos municípios, dadas pela aptidão agrícola dos solos, uso da terra e pela condição climatológica representada pela média pluviométrica histórica municipal. Os sub-indicadores que constituem os aspectos sócio-econômicos, PV131, PV132 e PV133, representam as características da base rural, em função da proporção da população rural do município em relação à sua população total; da participação das atividades agropecuárias na economia desses municípios; e das condições de atenuação dos efeitos das secas por conta das receitas correntes per capita municipais. Os dois últimos sub-indicadores, PV21 e PV22, constituem o quadro de riscos associados às perdas das safras agrícolas, em função da distribuição de chuvas, dada pelo Índice de Distribuição de Chuvas (IDC), e o risco de falta de água, tanto para o consumo humano quanto para o animal. Na Figura 1, os números em destaque a direita dos sub-indicadores indicam o nível de importância ou pesos destes no modelo.

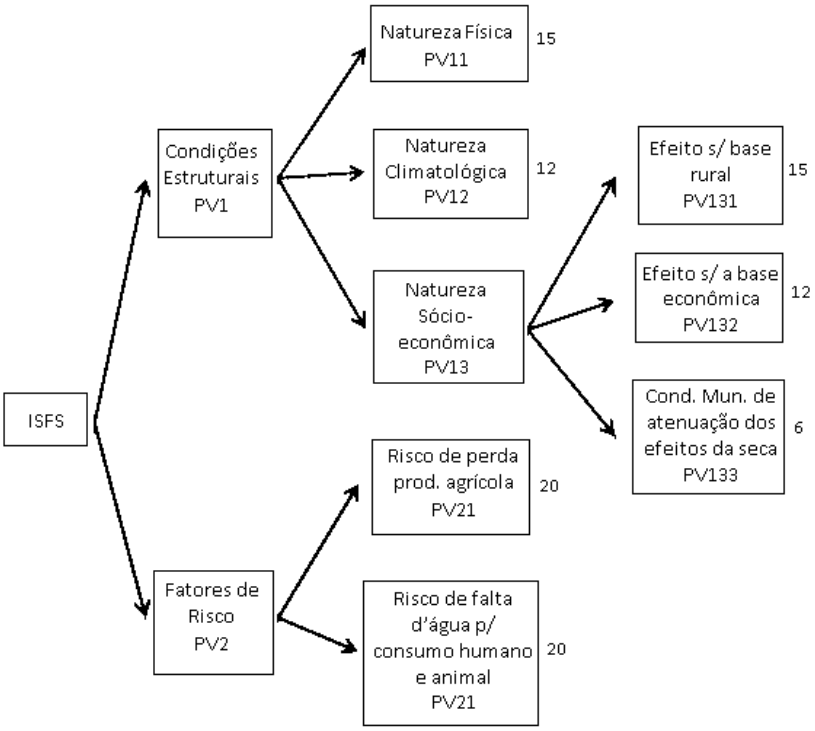


Figura 1. Arquitetura do ISFS. Fonte: Neves, 2010.

O cálculo do ISFS se dá no máximo em seis fases (Figura 2). Quando não ocorrer falhas nos dados, o procedimento se completará com o mínimo de duas fases. Na Fase um (F-1) são computadas todas as variáveis que constituem os sete sub-indicadores, conforme as equações descritas anteriormente. O sub-indicador é computado como falho quando da ausência ou falha em uma das variáveis que compõem este sub-indicador. A Fase dois (F-2) é opcional, somente ocorrerá se houver sub-indicador com assimetria maior ou igual a dois. Este procedimento é realizado com o intuito de melhorar as características da distribuição dos dados e aproximar-se da condição de normalidade multivariada exigida pelo método de preenchimento múltiplo utilizado na fase seguinte. A Fase três constitui a rotina de preenchimento múltiplo das falhas nos dados ocorridas na fase inicial. Os métodos disponíveis para o preenchimento das falhas são três: Predictive Mean Median (PMM), Bayesian Linear Regression (BLR) e Markov Chain Monte Carlo (MCMC). A fase quatro (F-4) só ocorrerá se a F-2 for executada, e a fase cinco (F-5) constitui o procedimento de winsorização, sendo permitido a utilização de dois patamares de winsorização (5 e 10%). Por fim, a fase seis (F-6) engloba as rotinas de geração do mapa final com os resultados do ISFS, a tabela com o resultado da classificação dos municípios em relação a susceptibilidade à seca e os gráficos de radar apresentando o impacto de cada um dos sub-indicadores na composição do índice.

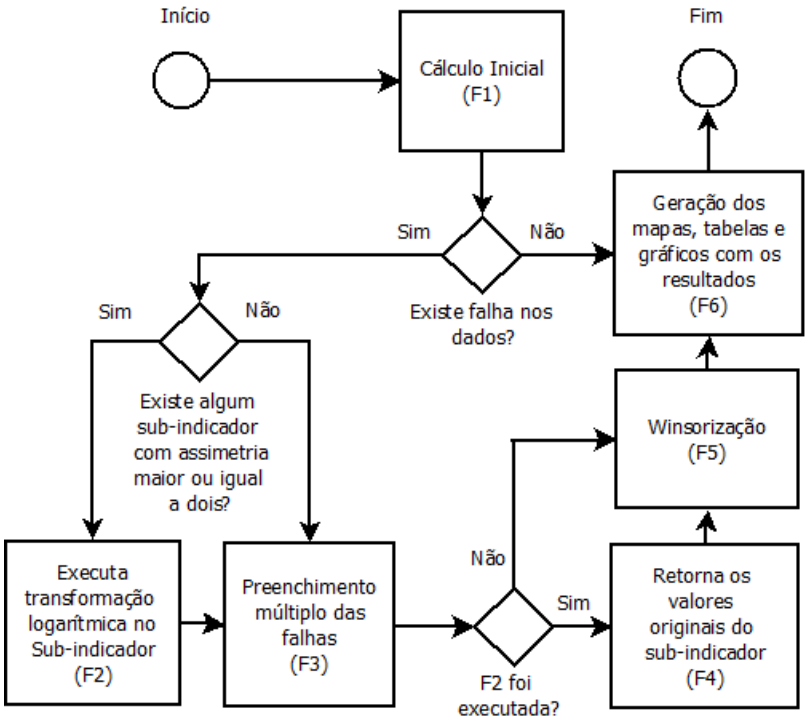


Figura 2. Estrutura do ISFS - Fluxograma de Cálculo. Fonte: Neves, 2010.

3. Características principais do *ISFS_Sist*

O *ISFS_Sist* foi desenvolvido no ambiente .NET framework 2.0, na linguagem C Sharp, utilizando o software R como pacote estatístico e gráfico, com a arquitetura descrita na Figura 3 abaixo.

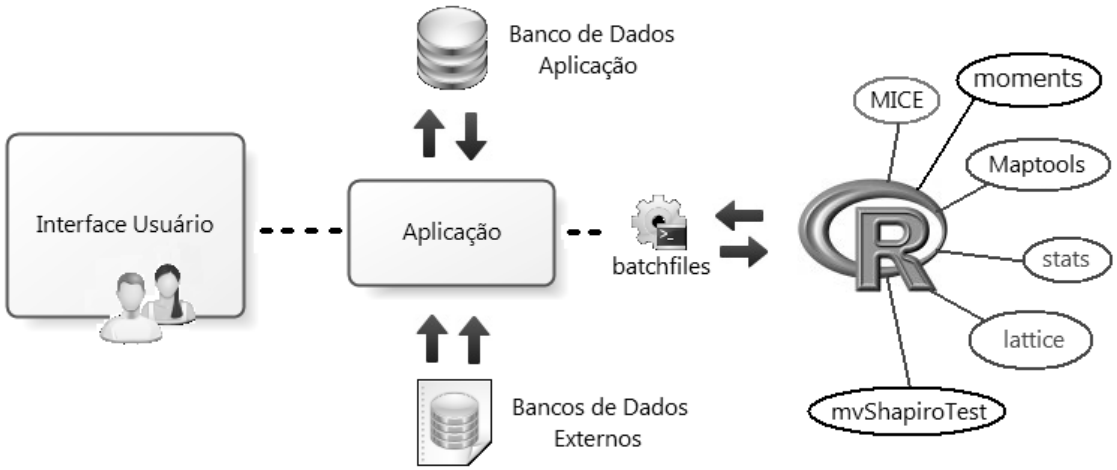


Figura 3. Diagrama esquemático dos componentes do *ISFS_Sist*. Fonte: Neves, 2010.

O software se constitui de uma aplicação principal, responsável pelas conexões com o banco de dados da aplicação, com os bancos de dados externos, com as bibliotecas estatísticas e gráficas do R e com uma interface de entrada e saída para o

usuário. A conexão com o R se dá a partir de arquivos autoexecutáveis (batchfiles) do sistema operacional windows. O banco de dados da aplicação é o MS Access. A vantagem dessa solução com relação ao banco de dados é o não requerimento de instalação de qualquer software particular, como por exemplo, o MS SQL Server. A conexão com os bancos de dados externos de dá a partir de arquivos no formato ASCII. A interface com usuário é bastante amigável, possibilitando uma fácil entrada de dados, um rápido processamento das informações e análise dos resultados a partir de relatórios, gráficos e mapas.

O R foi escolhido para integrar o ISFS_Sist por ser um software livre, de código aberto, rápido, robusto, validado pelo meio científico e que tem sido muito utilizado no meio acadêmico e também em aplicações comercias. Suas diversas bibliotecas facilitam a manipulação de dados, o cálculo de estatísticas e a visualização gráfica dos resultados, além da grande quantidade de métodos estatísticos que tem sido desenvolvidos e implementados. A sua utilização na composição do ISFS_Sist forneceu uma inegável vantagem por assegurar a qualidade dos resultados e acelerar o desenvolvimento e a produção do produto final.

As bibliotecas estatísticas e gráficas utilizadas na composição do ISFS_Sist foram:

. MICE - para o preenchimento de dados com falhas;

. Maptools - para a geração dos mapas do IDC e do ISFS;

. mvShapiroTest - para análise da normalidade multivariada dos dados;

. stats - para cálculo de estatísticas básicas e geração de números aleatórios;

. lattice - para visualização de dados multivariados;

. moments - para cálculos das assimetrias dos dados;

A biblioteca Maptools, em particular, provê um conjunto de ferramentas de manipulação e leitura de dados geográficos no formato ESRI shapefiles. Essa funcionalidade foi incorporada ao software nos módulos de geração de mapas e possibilita a utilização dos dados geográficos no formato “shp” de todo o Brasil, disponíveis na base de dados do IBGE.

Todos os relatórios do aplicativo são gerados no formato PDF. Logo, são pré-requisitos básicos para a utilização do ISFS_Sist a instalação do software R e do Adobe Acrobat Reader ou de um leitor de PDF instalado.

4. Descrição dos principais módulos do *ISFS_Sist*

O ISFS_Sist foi construído em oito módulos: Tabelas, IDC, Dados, ISFS, Relatórios/Gráficos, Parâmetros, Estatísticas/Análises e Rotinas de Administração. Na Figura 4 é apresentada a tela principal do aplicativo. As letras A, B, C e D em destaque

na figura identificam: o título do aplicativo; o menu principal; a propriedade e origem do trabalho; e, o link para o trabalho de tese que originou o aplicativo, respectivamente.



Figura 4. Tela principal do ISFS_Sist. Fonte: Neves, 2010.

São três módulos de entrada de dados: Tabelas, IDC e Dados. O módulo Tabelas contém as tabelas de dados das variáveis que constituem os sub-indicadores do ISFS, as tabelas de Estados, Microrregiões e Municípios, além de tabelas de controle: Estrutura do ISFS, Pesos, Normalização e IDC (categorias). A entrada pode ser feita manualmente, como é o caso do sub-módulo "Precipitação Diária → Editar Dados" (Figura 5), ou automaticamente a partir da importação de dados externos no formato texto (Figura 6).

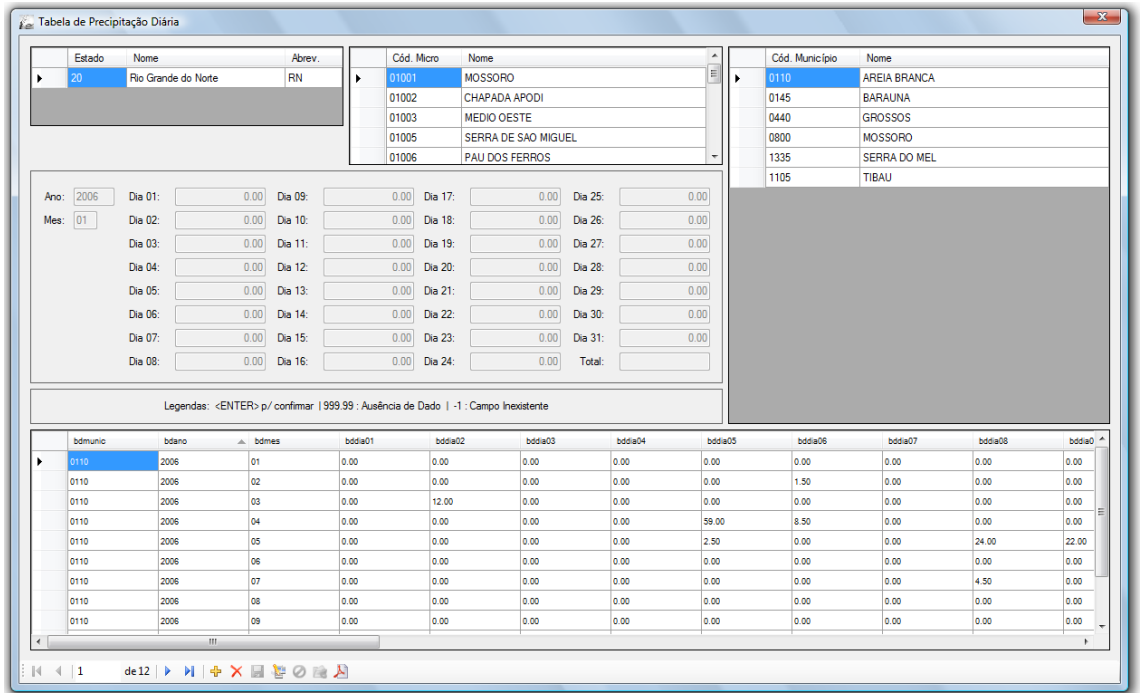


Figura 5. Tela da entrada de dados da precipitação diária dos municípios. Fonte: Neves, 2010.

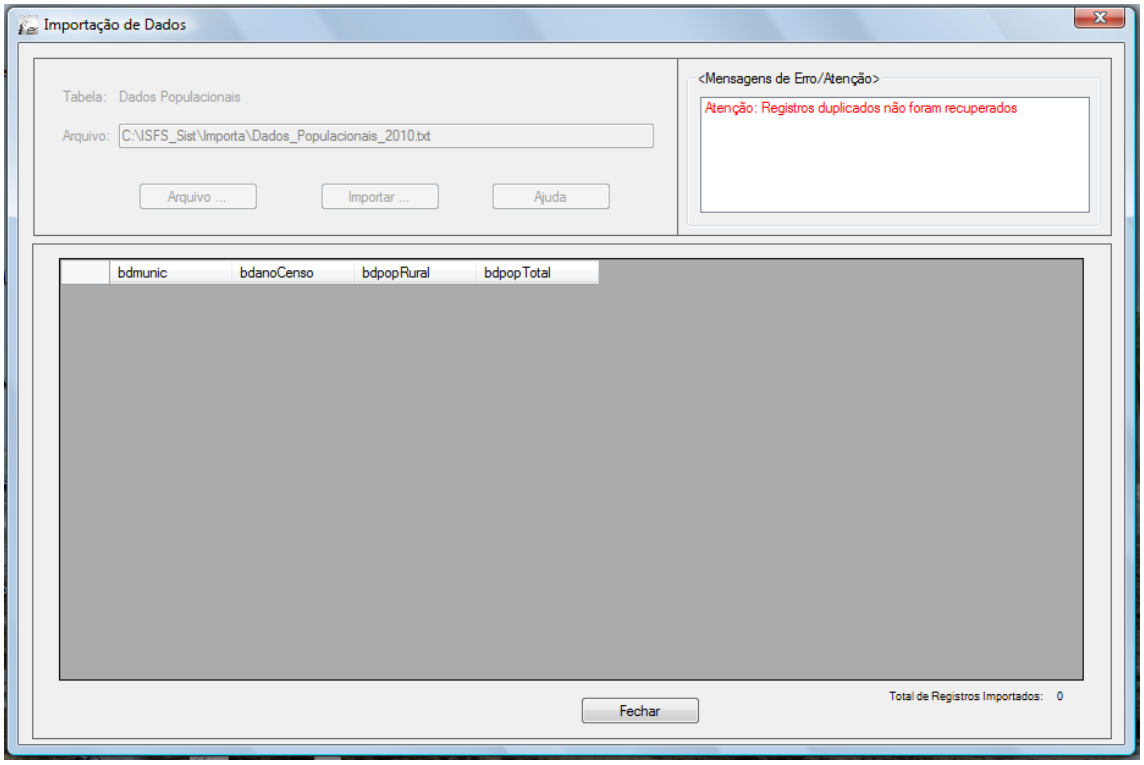


Figura 6. Tela da rotina de importação de dados dos dados populacionais municipais. Fonte: Neves, 2010.

No módulo IDC está o cálculo, consulta e a geração de mapas de IDC. E o módulo Dados disponibiliza a entrada dos dados brutos dos sub-indicadores do ISFS para o período em que se deseja calcular. Essa entrada é feita automaticamente em função da alimentação das variáveis que constituem os sub-indicadores nos módulos "Tabelas" e "IDC".

O procedimento de cálculo do índice e a geração de mapas são realizados no módulo ISFS. O usuário deverá informar o ano e a versão do cálculo e poderá acompanhar o andamento do cálculo na tela de Status da Operação.

O programa possibilita gerar tanto os mapas preliminares (antes do preenchimento múltiplo) e finais (após o preenchimento múltiplo) (Figura 7).

O módulo Relatórios/Gráficos apresenta as seguintes opções de relatórios: Classificação geral do ISFS, perfis dos municípios, avaliação conjunta das categorias do ISFS e um relatório final. A Figura 8 mostra um exemplo de relatório com a classificação do ISFS para o ano de 2006 no Estado do Rio Grande do Norte.

E o módulo Estatísticas/Análises possibilita ao usuário: a análise do padrão dos dados, informação essencial na escolha do tipo de preenchimento dos dados a ser utilizado no cálculo do índice; a realização do teste de normalidade multivariada dos dados; a análise das estimativas dos modelos de regressão utilizados no cálculo do índice, e, a análise de amplitude de variação do ISFS.

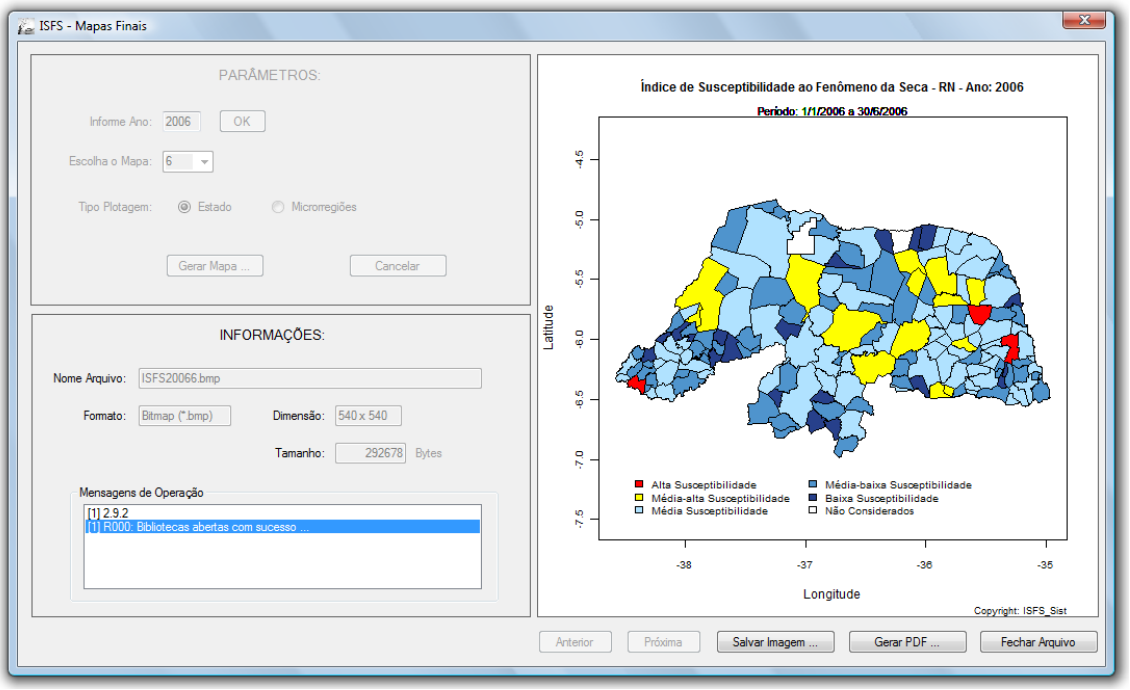


Figura 7. Módulo de geração dos mapas do ISFS para todo o Estado. Fonte: Neves, 2010.

Tuesday, May 04, 2010

ISFS - Índice de Susceptibilidade ao Fenômeno da Seca

Classificação por Categoria - ISFS - Ano: 2006

Período: 1/1/2006 a 30/4/2006

No.	Cód.Mun.	Nome	PV11	PV12	PV131	PV132	PV133	PV21	PV22	ISFS
					Alta Susceptibilidade					
1	0460	IELMO MARINHO	40.42	70.51	45.63	39.58	0	71	79.79	56.2804
2	1220	SAO JOSE DO MIPIBU	50.87	71.65	68.54	20.21	27.13	31.35	78.2	52.4771
3	0700	LUIS GOMES	50.82	78.47	16.23	7.35	37.71	67.5	79.79	52.0818
					Média-alta Susceptibilidade					
4	1290	SAO TOME	42.73	76.14	25.99	10.69	20.28	95.66	51.05	51.289
5	0790	MONTE DAS GAMELEIRAS	35.39	74.45	5.64	15.96	0	95.66	74.78	51.0948
6	0160	BENTO FERNANDES	18.81	72.98	13.32	21.2	0	95.66	79.18	51.0945
7	0020	ASSU	83.47	75	66.29	3.81	37.71	8.08	72.47	50.2965
8	0580	JOAO CAMARA	41.3	74.74	46.46	7.73	12.96	94.5	30.46	48.8326
9	0510	JANDAIRA	37.3	72.98	11.66	9.11	0	77	76.02	47.8012
10	0100	APODI	84.67	78.71	68.54	5.62	28.39	29.25	33.03	47.2649
11	0310	CURRAIS NOVOS	84.67	78.71	26.31	9.71	37.71	53	31.67	46.4585
12	1390	TAIPU	45.78	73.92	36.96	39.58	23.5	79.5	14.78	46.3005
13	1310	SENADOR ELOI DE SOUZA	38.89	73.96	14.67	35.81	7.64	95.6	23.47	45.4841
14	0960	PEDRA PRETA	37.35	73.83	10.49	8.12	2.74	83	58.29	45.4366
15	0540	JAPI	44.87	78.37	12.72	9.09	32.58	95.5	26.06	45.4034

Figura 8. Exemplo de relatório da classificação geral do ISFS. Fonte: Neves, 2010.

5. Exemplo da aplicação do ISFS_Sist

A validação do ISFS se deu com dados reais do Rio Grande do Norte para anos representando cenários secos, normais e chuvosos. Para isso foi calculado o ISFS para os anos de 1998, 2000, 2005 e 2006 dentro do aplicativo. A Figura 9 mostra o mapa do índice para o ano de 1998 e a Figura 10 mostra a tela das estimativas dos modelos de regressão utilizados no cálculo do ano de 2006.

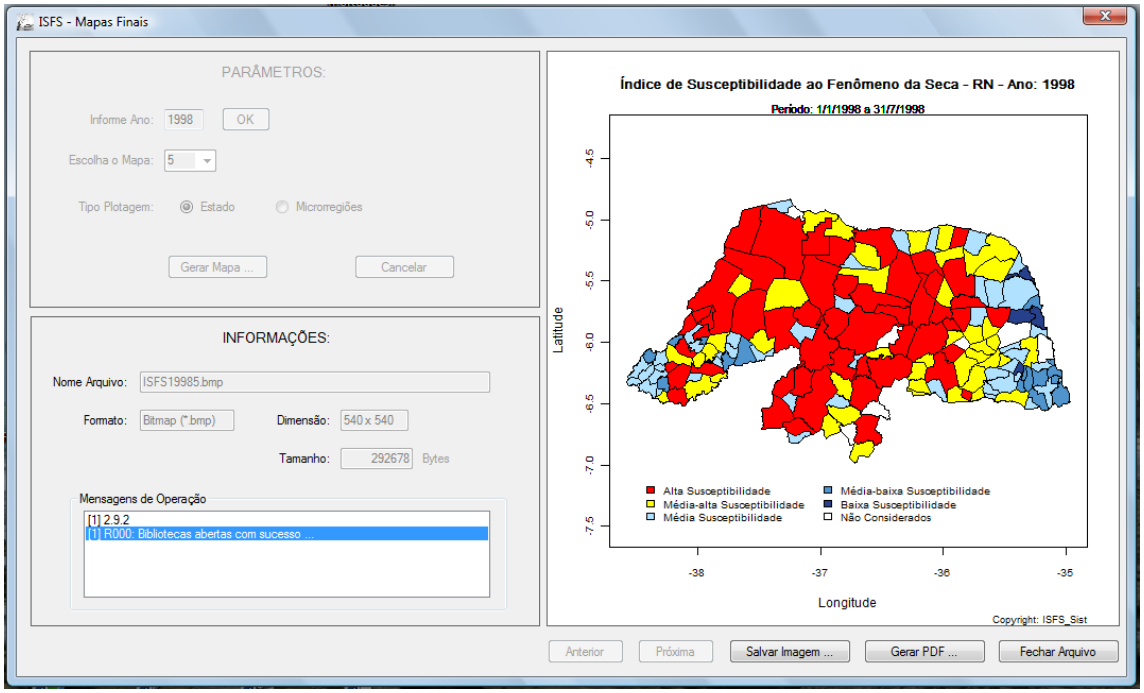


Figura 9. Mapa do ISFS para o ano de 1998.

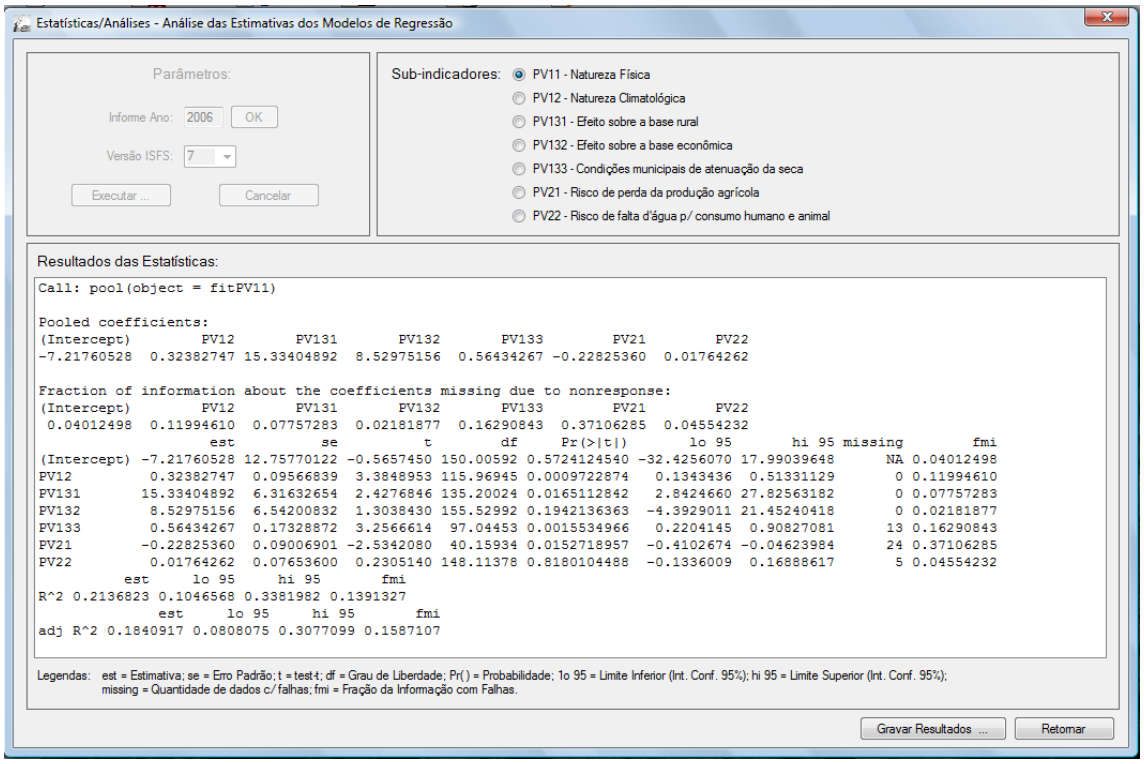


Figura 10. Estimativas dos modelos de regressão para o ano de 2006.

6. Conclusão

O aplicativo computacional ISFS_Sist, automatiza o cálculo do ISFS e se constitui em uma ferramenta flexível, que produz resultados com acurácia, rápida computação e a

geração de relatórios e gráficos detalhados. Os resultados poderão fornecer aos tomadores de decisão os subsídios necessários ao direcionamento das ações para o atendimento e assistência das populações atingidas por eventos de secas.

A constituição do ISFS e a flexibilidade em que foi desenvolvido o ISFS_Sist permite ao usuário configurar os parâmetros necessários ao melhor ajuste a região de interesse do usuário.

Não foram encontrados no mercado e na literatura pesquisada softwares similares ao ISFS_Sist. Os softwares existentes, caso do SPATIM [Smakhtin; Hugues, 2007], implementam indices de seca que utilizam somente no seu cálculo variáveis hidro-meteorológicas, portanto, não levando em consideração outras dimensões ou aspectos regionais que qualificam e quantificam melhor as nuances características de uma seca.

Por fim o ISFS_Sist foi devolvido para ser distribuído gratuitamente e encontra-se em fase de registro junto ao Instituto Nacional de Patentes Industriais (INPI), sendo que em breve será disponibilizado um link para o seu download na internet.

7. Referências Bibliográficas

Alves, J. M. B., et al. (1998) Produção agrícola de subsistência no Estado do Ceará com ênfase aos anos de ocorrência de El Niño e La Niña. Revista Brasileira de Agrometeorologia, v.6, n.2, 249-256.

Araújo, L. E. et al. (2008) Análise estatística de chuvas intensas na bacia hidrográfica do Rio Paraíba. Revista Brasileira de Meteorologia, v.23, n.2, 162-169.

Brunini, O. et al. (2005) Agricultural drought phenomenon in Latin America with focus on Brazil. In: BOKEN, V.K.; CRACKNELL, A.P., HEATCOTE, R.L. Monitoring and Predicting Agricultural Drought: A Global Study. New York, Oxford University Press. cap. 12, 156-168.

Byun, H.; Wilhite, D. A. (1999) Objective quantification of drought severity and duration. Journal of Climate, v. 12, p. 2747-2756.

Carvalho, O. (1988) A Economia Política do Nordeste (seca, irrigação e desenvolvimento). Rio de Janeiro-RJ: Editora Campus. 505 pp.

Dantas, J. A. (1961) Homens e fatos do Seridó Antigo. Garanhuns: Gráfica do "O Monitor", 168 pp.

Duarte, R. S. (2002) A seca de 1958: uma avaliação pelo ETENE. Fortaleza: Banco do Nordeste; Recife: Fundação Joaquim Nabuco. 200 pp. (Série Estudos sobre as Secas no Nordeste; v.1).

ENSSLIN, L. et al. (2000) Elaboração de um índice de severidade de seca para o Ceará utilizando-se uma metodologia de apoio à decisão (MCDA). In: XXXII SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 1, 2000, Viçosa, MG. Anais... Viçosa: Sociedade Brasileira de Pesquisa Operacional. p. 1147-1164.

Freitas, M. A. S0. (2005) Um Sistema de Suporte à Decisão para o Monitoramento de Secas Meteorológicas em Regiões Semiáridas. Revista Tecnologia, v. Suplem, 84-95.

FUNDAÇÃO CEARENSE DE METEOROLOGIA. (2000) Manual do índice FUNCEME de severidade de seca. Fortaleza. 278 p.

Guerra, P.B. (1981) A Civilização da Seca. Fortaleza, DNOCS. 324 pp.

Mendonça, F.; Danni-Oliveira, I. M. (2007) Climatologia: Noções básicas e climas do Brasil. São Paulo: Oficina de Textos. 206 pp.

SILVA, J. S. V.; FORTE, S. H. A. C. (2002) Uma contribuição nas decisões emergenciais da gestão pública em regiões semi-áridas. Revista Econômica do Nordeste. v.33, n.4.

SMAKHTIN, V. U.; HUGHES, D. A. (2004) Review, automated estimation and analyses of drought indices in South Asia. Working Paper 83. Colombo, Sri Lanka: International Water Management Institute,. 24 pp.

SMAKHTIN, V. U.; HUGHES, D. A. (2007) Automatic estimation and analyses of meteorological drought characteristics from monthly rainfall data. Environmental Modelling & Software, v. 22, p. 880-890, 2007.

SIG como apoio ao Processo de Estudos Hidrológicos para determinar pontos de vazão para outorga

Clóvis Lemos Tavares[1], Taciana de Lemos Dias [2], Daniel Rigo[3], Maria Helena Alves[4], Karina Luna[4]

[1]Centro Tecnológico – Universidade Federal do Espírito Santo
clovisbd@gmail.com

[2]Centro de Ciências Jurídicas e Econômicas - UFES
taciana.ufes@gmail.com

[3]Departamento de Engenharia Ambiental –UFES
rigo@npd.ufes.br

[4]Gerência de Meio Ambiente e Divisão de Recursos Hídricos - CESAN
helena.alves@cesan.com.br
karina.luna@cesan.com.br

Abstract. *This paper presents an application of a knowledge process used by the Hydrological Studies IEMA 2008 to generate a geographic information system for estimating runoff in the state of Espírito Santo. A new interface was developed, based on spatial analysis, flow study, isohyetal interpolation of precipitation, and identification of layers of regionalization in all 12 major river basins of Espírito Santo State. The result of the estimates can be used for planning application for the grant, identify areas for definition of points of water abstraction and effluent discharge. The paper presents results from a case study, based on the experience of the Water Company of Espírito Santo State (CESAN) developed by NIPSA and Nip do Brazil with funds from World Bank (BIRD) in the clean water program.*

Resumo. *Este artigo apresenta uma aplicação do conhecimento do processo de Estudos Hidrológico utilizado pelo IEMA 2008 na geração de um sistema de informação geográfica para estimativas de vazão no estado do Espírito Santo. Foi desenvolvida uma interface baseada em análises espaciais, estudo de fluxo, interpolação de isoietas de precipitação, identificação de camadas de regionalização em todas as 12 grandes bacias do estado do Espírito Santo. O resultado das estimativas pode ser usado para planejamento de solicitação de outorga e identificação de locais para definição dos pontos de captação de água e lançamento de efluentes. O artigo apresenta os resultados de um estudo de caso, baseado na experiência da Companhia de água do Espírito Santo (CESAN) desenvolvido pela NIPSA (Empresa multinacional espanhola de engenharia e sistemas) e NIP do Brasil (Filial da NIPSA no Brasil) com recursos do banco mundial (BIRD) no programa águas limpas.*

1. Introdução

1.1. Contextualização

Tem sido crescente a necessidade do envolvimento dos órgãos públicos no ciclo hidrológico, através da captação de água e no lançamento dos efluentes tratados. Com o objetivo de garantir o desenvolvimento sustentável, no Brasil existem órgãos reguladores que fiscalizam a utilização dos recursos hídricos federais e estaduais. No caso do Espírito Santo o órgão responsável por essa fiscalização é o Instituto Estadual de Meio Ambiente - IEMA.

O desenvolvimento sustentável, segundo [Amadora 2009], é o desenvolvimento que atende a geração atual sem prejudicar a futura, possibilitando um nível satisfatório de desenvolvimento social e econômico e de realização humana e cultural, fazendo, ao mesmo tempo, um uso razoável dos recursos da terra e preservando as espécies e os habitats naturais.

A necessidade de estimar a vazão de um ponto do recurso hídrico, e esta informação ser processada por um Sistema de Informação Geográfica (SIG) e armazenada por um banco de dados espacial, facilita o processo de estimativas sucessivas na continuidade de um ou mais corpos hídricos. Isso, com o objetivo de conhecer o melhor ponto para a necessidade de outorga, auxiliando, assim, no processo de solicitação de outorga e em uma tomada de decisão mais sustentável [Burrough 1986].

Segundo [Lanna 2000], a outorga é um instrumento discricionário através do qual o proprietário de um recurso natural estipula quem pode usá-lo com as respectivas limitações. Quando a propriedade é privada, a outorga equivale à aquiescência, pelo proprietário, de que outrem o utilize desde que se sujeite às condições por ele impostas. Quando a propriedade é pública, serve como um instrumento de gestão ambiental, a partir da atribuição de cotas entre os usuários, considerando-se a escassez do recurso e os benefícios sociais gerados.

Existem alguns incentivos governamentais para a melhoria das bases de dados ambientais que utilizam recursos do Banco Internacional para Reconstrução e Desenvolvimento - BIRD. Além dos investimentos financeiros, muitas intuições, como por exemplo, o Instituto Brasileiro de Geografia e Estatística - IBGE e o Instituto Jones dos Santos Neves - IJSN, fornecem dados gratuitos para serem utilizados, tratados e aperfeiçoados ou atualizados. Embora existam esses incentivos e várias destas informações estarem disponíveis, muitas organizações ainda não se encontram preparadas para utilizarem e gerarem novas informações a partir destas, utilizando ferramentas SIG, com o propósito de apoiar a gestão ambiental.

Segundo [Davis 2002], o grande desafio que se apresenta para os SIG na próxima década é o da simplificação do uso. É preciso tornar o acesso e a manipulação de dados geográficos mais simples e mais intuitivos para as pessoas, inclusive para reduzir um pouco a enorme necessidade de formação de pessoal que se observa hoje.

Segundo [Lima 2002], diante das facilidades crescentes de uso da microinformática, vários modelos matemáticos têm sido desenvolvidos, permitindo a realização de cálculos precisos com grande velocidade e com estrutura capaz de

absorver informações e dados relevantes, sendo possível, assim, abranger os principais aspectos de uma bacia hidrográfica. Entre as técnicas mais utilizadas estão à simulação e a otimização.

A gestão ambiental é complexa por envolver e integrar processos como licenciamento, recuperação, educação, auditoria, entre outros. Os órgãos públicos são responsáveis pela gestão dos corpos hídricos e as companhias de água são responsáveis pelo cumprimento das legislações relacionadas ao seu uso e, portanto, necessitam de solicitar outorga de acordo com condicionantes e parâmetros vigentes.

No Estado do Espírito Santo a Companhia de Água do Espírito Santo - CESAN, de acordo com [CESAN 2009], é responsável por 507.327 ligações de água e 123.172 ligações de esgoto, necessitando, portanto de utilizar o processo de solicitação de outorga de água para abastecer seus clientes distribuídos pelo Estado. Para isso, existe a necessidade da empresa de realizar estimativas de vazão como apoio no processo de solicitação de outorga e de sua manutenção.

O objetivo deste artigo é apresentar como um sistema de informação geográfica utilizando técnicas de interpolação para determinar pontos de vazão para outorga, auxilia o processo de solicitação de outorga e o processo de estudos hidrológicos como um sub-processo dentro da gestão ambiental.

Um processo é um conjunto de passos parcialmente ordenados, constituídos por atividades, métodos, práticas e transformações, usado para atingir uma meta. Esta meta geralmente está associada a um ou mais resultados concretos finais, que são produtos da execução do processo [Pádua 2003].

As aplicações de apoio a decisão de planejamento, controle e monitoramento das cidades devem possuir uma boa representação espacial, um adequado modelo da realidade representada e de seu processo de trabalho. Existe uma grande diversidade semântica para uma mesma entidade e processo e, portanto, é importante desenvolver sistemas de informação baseados em uma definição prévia dos conceitos adotados, conforme os modelos mentais de uma comunidade no domínio modelado e definido o processo a ser seguido. Esses modelos estão relacionados com as análises conceituais das aplicações e com o domínio modelado [Dias Câmara Davis Jr. 2007, Dias 2007]

Um sistema de informação deve apoiar em um processo de trabalho definido da realidade estudada e para o seu desenvolvimento, também, deve ser adotado um processo de desenvolvimento de software, os quais, podem utilizar vários modelos de ciclo de vida e ferramentas que permitem aos desenvolvedores a visualização do software através de diagramas padronizados, como, também, auxiliam na compreensão e interlocução com os usuários. O modelo de desenvolvimento evolutivo utiliza protótipos que permitem que os requisitos do software sejam definidos progressivamente e apresenta alta flexibilidade e visibilidade para os clientes e, a UML, Linguagem Unificada de Modelagem, é uma abstração gráfica para visualização, especificação, construção e documentação de artefatos de sistemas complexos de software [Pádua 2003, Pressman 2005, Booch 2000].

A finalidade de se propor um modelo de outorga baseado no conceito de sistemas de suporte à decisão, está no interesse em aplicar novas ferramentas de

tecnologia de informação e sugerir métodos para o estudo, identificação e avaliação dos potenciais usos e usuários da água [Ferraz 1996].

[McKinney, Maidment 1993] descreve a possibilidade do uso da integração das tecnologias de um Sistema Especialista de Informações Geográficas como uma ferramenta fundamental de planejamento dos sistemas de recursos hídricos. Fazem um estudo de previsão de demanda de água futura e sua distribuição geográfica. O planejamento requer tomada de decisão regional que é combinada com a técnica geográfica e os fatores institucionais.

Interpolação é o procedimento de prognóstico de valores não conhecidos de um atributo contínuo, no caso a altitude, utilizando valores conhecidos de pontos vizinhos [Aronoff 1986]. Parte desse processo envolve o conhecimento de relações de vizinhança e outra parte envolve a pressuposição de comportamento da superfície entre os pontos de valores conhecidos. A ponderação entre essas partes pode variar.

Este trabalho utiliza técnicas de interpolação para calcular valores de precipitação a partir da camada de isoietas. As isoietas são as linhas de igual precipitação (mm). Assim como em um mapa topográfico as curvas de nível representam regiões de mesma cota (Altura em relação a um referencial, que geralmente é o nível médio do mar), as isoietas são curvas que delimitam uma área com igual precipitação (quantidade de chuva que cai, medida em mm).

A simulação dos processos hidrológicos e de qualidade da água, onde equações diferenciais e relações empíricas são utilizadas para representação de vários aspectos quantitativos e qualitativos do ciclo hidrológico, é a técnica mais largamente utilizada na prática, e, apesar de não ser otimizante, proporciona meios para o tratamento detalhado e conseqüentemente a avaliação, com maior precisão, do comportamento dos sistemas [Braga, 1987].

O artigo descreve o processo de estudo hidrológico proposto para solicitação de outorga através de um estudo de caso, baseado na experiência CESAN desenvolvido pela NIPSA (Empresa Multinacional Espanhola de Engenharia e Sistemas) e sua filial NIP do Brasil. Trabalho este, financiado através dos recursos do Banco Internacional para Reconstrução e Desenvolvimento – BIRD, no programa águas limpas. Não é objeto desse artigo a demonstração das fórmulas necessárias para o cálculo de vazão, utilizadas do estudo [IEMA 2008]. Apresenta, também, o sistema de informação geográfica desenvolvido e implantado para apoiar o processo definido, através da apresentação de algumas interfaces.

2. Trabalhos Correlatos

Muitos órgãos têm trabalhos na aplicação de métodos computacionais para cálculos de estimativas de vazões. Em [Roques 2006] é apresentado à aplicação de modelos computacionais na análise de outorga para diluição de efluentes em corpos de água em fontes pontuais e difusas.

Em 2008, desenvolveu-se uma ferramenta computacional no âmbito do Instituto Estadual de Meio Ambiente e Recursos Hídricos - IEMA para análise das outorgas. A ferramenta utilizada é capaz de realizar o balanço entre a demanda e a disponibilidade. A disponibilidade hídrica é estimada a partir do estudo de regionalização de vazões,

cujos dados necessários são a área de drenagem e a precipitação média anual para o ponto de interesse. Neste caso apenas utiliza-se uma planilha eletrônica para entrada manual de dados, portanto não utiliza de técnicas de análise espacial para automatizar a entrada de dados correspondente aos diversos parâmetros das equações de estimativa de vazão.

O presente trabalho se diferencia dos mencionados pela utilização de técnicas de geoprocessamento para indicar os valores de entrada nos parâmetros das equações de estimativa de vazão elaboradas pelo estudo de regionalização de vazão do [IEMA 2008], possibilitando uma avaliação destes parâmetros pelo usuário e agilizando o processo de análises sucessivas. Como, também, apresenta o Sistema de Informação Geográfica.

3. Proposta do Processo de Estudos Hidrológicos para determinar pontos de vazão para outorga

Os estudos hidrológicos podem ser realizados através da análise de pontos específicos no corpo hídrico para a escolha do ponto a ser outorgado. Esses pontos geralmente são, inicialmente, estipulados a partir de um ponto de referência ou de um par de coordenadas.

É fundamental para o entendimento do processo a descrição semântica de ponto de outorga. Esse conceito deve ser elaborado em consenso pelos técnicos da realidade estudada, sendo assim, iniciou-se a discussão do processo através das diferentes descrições ou apresentações, pelo solicitante, do ponto solicitado para outorga, Através dessa análise observou-se que o ponto pode ser determinado por coordenada geográfica ou por uma descrição de referência. A partir da percepção das características dos dois tipos de ponto, onde os dois conceitos são fontes de dados para obter a informação da coordenada do ponto desejado. Para isso, no caso do ponto de referência é importante disponibilizar mapas com informação de referências e coordenadas geográficas para que o analista possa determinar o ponto definido de outorga.

A entidade denominada Ponto de Vazão nesta proposta contém a informação referente à estimativa de vazão em um determinado ponto especifico. O ponto que será estimado pode assumir os seguintes estados, definidos: Ponto estabelecido, Ponto referência e Ponto definido. Essa definição é de fundamental importância, pois a estimativa de vazão para o local escolhido poderá afetar outras análises de gestão ambiental e assim, a nomeação do ponto em estudo auxilia na percepção de sua semântica.

A descrição dos estados do ponto de estimativa de vazão para o estudo hidrológico deste trabalho é: Ponto estabelecido é o ponto para estimar a vazão, identificado por coordenadas, informado pelo solicitante de outorga; Ponto referência é o ponto de referência passado pelo solicitante para estimar a vazão. Essa referência é descritiva, como por exemplo, próximo a sede da Fazenda X; Ponto definido é o ponto determinado pela aplicação para solicitação de outorga. Esse ponto pode ser originado a partir do ponto referência ou do ponto estabelecido. Essa definição é baseada na combinação de aspectos relevante como distância do corpo hídrico, perfil topográfico, dentre outros.

A Figura 1 apresenta o fluxo do **processo de estudos hidrológicos para determinar pontos de vazão para outorga,** parte do processo de Solicitação de outorga relacionado ao processo de Solicitação de estimativa de vazão do corpo hídrico que contém as etapas sistematizadas pelo sistema de informação abordado neste estudo. O fluxo do processo de Solicitação de estimativa de vazão do corpo hídrico é subdividido em 2 fases, precedentes ao processo de outorga propriamente dito. Após a definição do ponto e valores de vazão calculados, inicia-se o processo de solicitação de outorga no órgão competente.

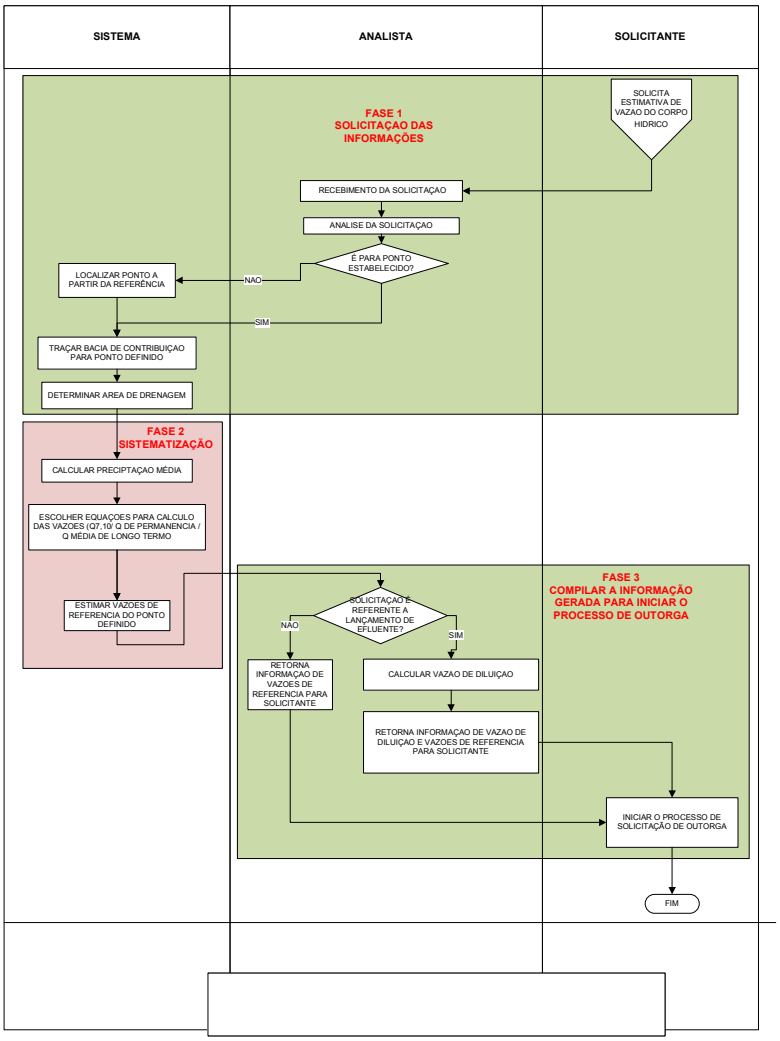


Figura 1. Fluxo do processo de estudos hidrológicos para determinar pontos de vazão para outorga.

Nas próximas seções é detalhado o processo de sistematização, apresentado através da interação do analista técnico com o sistema de informação proposto (Seção 3.1) e algumas interfaces do sistema (Seção 3.2).

3.1. Fase 2 - Sistematização

Devido ao fato do processo estar bem definido e ser de domínio de todos os técnicos, baseado nesse processo foi gerado o levantamentos de requisitos de um sistema capaz de apoiar o processo definido. Utilizando a UML para a elaboração da documentação

sistemática do processo e a prototipagem. A utilização de protótipos agilizou o desenvolvimento e facilitou o entendimento dos usuários em como o sistema contribui no processo de trabalho.

A descrição abaixo tem como objetivo apresentar a aplicação desenvolvida para estimar vazões, a partir de um ponto selecionado em um corpo hídrico, relacionado à fase 2 do fluxo de estudo hidrológico de estimar vazão do corpo hídrico.

A seguir é apresentada a interação do analista técnico, denominado usuário, com o sistema de informação proposto, que são:

- O Usuário acessa a ferramenta Estudos Hidrológico e ao clicar no ponto do curso de água abre uma janela para o usuário confirmar as informações levantadas pelo sistema.
    - Informações (Parâmetros de entrada das equações de vazão)
        - Precipitação calculada: Informação medida a partir de isoietas com técnicas de interpolação para estimar o valor da precipitação do curso hídrico selecionado. Automaticamente, carregado de acordo com o posicionamento da seleção do usuário. De acordo com [IEMA 2008] o valor deve estar entre 900 e 1.800mm.
        - Vazão Média de Longo Termo: Automaticamente carregado de acordo com o posicionamento da seleção do usuário em relação à camada de estudo de regionalização, vazão média de longo termo, do [IEMA 2008]; caso o usuário queira pode alterar o valor para realizar simulações.
        - Curva de Permanência das Vazões Médias Mensais: Automaticamente carregado de acordo com o posicionamento da seleção do usuário em relação à camada de estudo de regionalização, curva de permanência, do [IEMA 2008]; caso o usuário queira pode alterar o valor para realizar simulações.
        - Média das Vazões Mínimas Anuais das Médias de 7 dias consecutivos: Automaticamente carregado de acordo como o posicionamento da seleção do usuário em relação à camada de estudo de regionalização do [IEMA 2008]; caso o usuário queira pode alterar o valor para realizar simulações.
        - Vazão Mínima Média de 7 dias consecutivos e período de retorno T: Automaticamente carregado de acordo com o posicionamento da seleção do usuário em relação à camada de estudo de regionalização, vazão mínima média de 7 dias consecutivos por período de retorno T, do [IEMA 2008]; caso o usuário queira pode alterar o valor para realizar simulações.
        - Área de drenagem: Área à montante do ponto selecionado, que ele deseja. Automaticamente carregado de acordo com o posicionamento da seleção do usuário. O valor para a área de drenagem, segundo [IEMA 2008], deve estar entre 10 e 14.000 km².

Relacionado às variáveis de entrada acima descritas, nos mapas existem áreas em branco, que neste caso, o sistema foi programado para verificar a área mais próxima, automaticamente, perguntando ao usuário se ele deseja mantê-la ou não.

- O usuário confere as informações acima e seleciona a opção “Executar” para iniciar o processo estimativa de vazão.
- O sistema abre uma nova janela contendo o resultado das seguintes análises.
  - Vazão Média de Longo Termo (m³/s); Vazão Específica (l/s.km²); Vazão Mínima Absoluta (m³/s); Média das Vazões Mínimas Anuais das Médias de 7 dias (m³/s).
  - Q7,10 (originário do Q7,T com T igual a 10 anos) : vazão mínima em m3/s de sete dias (Q7,T) de duração e período de retorno de 10 anos; Q90 (originário da tabela de curva permanência para o valor 90%): vazão em m3/s com intervalo diário e 90% da curva de permanência; Q95 (originário da tabela de curva permanência para o valor 95%): vazão em m3/s com intervalo diário e 95% da curva de permanência.
- Caso o usuário queira gerar a tabela de permanência, basta ele clicar no botão “gerar curva de permanência”.
- Caso o usuário queira gerar a tabela de Q7,T, basta ele clicar no botão “gerar tabela de Q7,T”.

3.2 Interfaces do Sistema

A Figura 2 abaixo apresenta a interface, que mostra os resultados das etapas do sistema. Essa interface apresenta na parte superior os parâmetros da fórmula que foram calculados nas etapas anteriores ao cálculo de vazão, escolhidos pelo usuário e validados pelo sistema. A parte inferior apresenta o resultado dos cálculos de vazão descritos na fase 2 do processo de Estudos Hidrológicos.

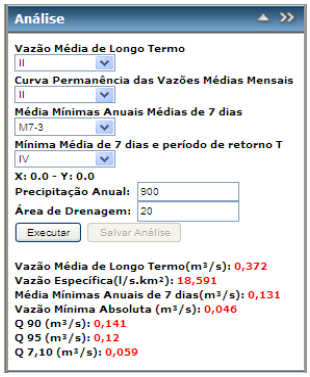


Figura 2. Interface de definição dos parâmetros de cálculo e apresentação dos resultados de estimativa de vazão

A Figura 3 apresenta em mapa um estudo de caso com um ponto selecionado e com seus valores estimados de acordo com as técnicas apresentadas nesta proposta.

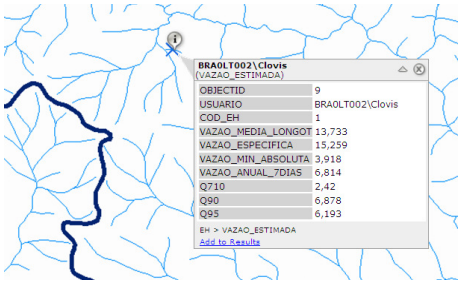


Figura 3. Interface com os resultados de um ponto de vazão do Espírito Santo.

O sistema proposto apresenta os dados do período de retorno versus as Vazões Mínimas Médias de 7 dias (Tabela 1) para cálculo da curva de permanência (Tabela 2).

Distribuição das Vazões Mínimas de 7 dias: significa vazão mínima em m3/s de sete dias de duração e período de retorno de T, que na tabela abaixo calcula de 1,5 a 100 anos.

Tabela 1. Dados do período de Retorno x Vazões Mínimas Médias de 7 dias

Vazões Mínimas de 7 dias

Período de Retorno (anos)	Vazões Mínimas Médias de 7 dias (m³/s)
1,5	0,147
2	0,118
3	0,093
5	0,074
10	0,059
15	0,053
20	0,05
25	0,048
50	0,043
100	0,04

Exportar

Curva de permanência: Uma curva de permanência de vazão, também conhecida como curva de duração é um traçado gráfico que informa com que freqüência a vazão de dada magnitude é igualada ou excedida durante o período de registro das vazões.

Tabela 2. Dados da curva de permanência

Tabela de Permanência

Permanência (%)	Vazão (m³/s)	Permanência (%)	Vazão (m³/s)
1	1,904	50	0,262
2	1,449	55	0,244
3	1,215	60	0,228
5	0,981	65	0,213
10	0,706	70	0,199
15	0,565	75	0,184
20	0,477	80	0,17
25	0,416	85	0,156
30	0,368	90	0,141
35	0,334	95	0,12
40	0,306	98	0,101
45	0,283	100	0,046

Exportar

4. Conclusão

A sistematização da proposta do processo de estudo hidrológico dentro da CESAN permitiu a agilidade em se ajustar o ponto em estudo às diversas áreas e equações do Estudo de Regionalização do IEMA 2008.

O estudo que foi descrito neste artigo permite ao leitor ter uma visão geral do processo, o seu apoio pelo sistema de informação, como também, as interfaces propostas e cálculos utilizados. Essa proposta auxilia no desenvolvimento de aplicações com o mesmo objetivo, como também, na reengenharia de processo de outorga. Além, de demonstrar mais uma aplicabilidade do Estudo de Regionalização do IEMA [2008].

A automatização do procedimento no que tange a rápida identificação do ponto em estudo, através das coordenadas, facilitou a definição dos valores dos parâmetros das equações de vazão.

Ressaltamos que a metodologia atualmente utilizada para identificação dos valores de precipitação passa pela simples interpolação entre as isoietas, não considerando a ponderação das áreas entre as mesmas delimitadas pela bacia, como sugerido no estudo (IEMA 2008). Existe, atualmente, um projeto para desenvolver um algoritmo que automatize a metodologia proposta, assim como um algoritmo que desenhe automaticamente a área de drenagem a partir de uma seção qualquer do rio.

Finalmente, a grande contribuição que o trabalho trouxe está relacionada à agilidade que o SIG forneceu, associando rapidamente o ponto em análise as diversas áreas e suas equações do Estudo de Regionalização.

7. Agradecimentos

Os colaboradores agradecem pelo apoio da CESAN ao trabalho, especialmente a Gerência de Meio Ambiente pela cessão dos dados e por ajuda mútuas bastante positivas para o resultado final. Agradecemos o apoio do NIPSA a seus projetos de pesquisa, e também ao suporte fornecido para o desenvolvimento deste trabalho.

8. Referências Bibliográficas

Amadora, C. M. A21 - Desenvolvimento Sustentável. www.cm-amadora.pt. 2009.

Aranoff, S. Geographic information systems: a management perspective. Canadá: WDL Publications, 1986.

Braga Júnior., B. P. F. Técnicas de otimização e simulação aplicadas em sistemas de recursos hídricos. In: BARTH, F. T. et al. Modelos para gerenciamento de recursos hídricos. São Paulo, Nobel/ABRH, 1987. p.427-

Booch, G; Rumbaugh, J E Jacobson, I: UML, Guia do Usuário: tradução; Fábio Freitas da Silva, Rio de Janeiro, Campus ,2000.

Burrough, P. A. Principles of Geographic Information Systems for Land Resources Assessment. Oxford, Oxford University Press, 1986.

CESAN (2009). CESAN - Abrangência. Disponível em http://www.cesan.com.br/page.php?4.

Davis Junior, C. A. Geoprocessamento: Dez Anos de Transformações. IP. Informática Pública, Belo Horizonte (MG), v. 4, n. 1, p. 17-24, 2002.

Dias, T. L. Modelagem computacional de mudanças incrementais em objetos espaciais da realidade social. Tese (Doutorado) - Instituto Nacional de Pesquisas Espaciais. São José do Campos, 2007.

Dias, T. L. Gilberto Câmara ; Davis JR., C. A. . Modelos Espaço-Temporais. In: Marco Antônio Casanova; Gilberto Câmara; Clodoveu Augusto Davis Júnior; Lúbia Vinhas; Gilberto Ribeiro de Queiroz. (Org.). Bancos de Dados Geográficos. Curitiba (PR): EspaçoGeo, 2005, v. único, p. 147-180.

Ferraz, A. R. G. Modelo decisório para a outorga de direito ao uso da água no Estado de São Paulo. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo. São Paulo, 1996. 187p.

IEMA, Instituto Estadual de Meio Ambiente e Recursos Hídricos, Estudo de Regionalização de Vazões e Metodologia para Determinação de Vazões Insignificantes para as 12 Bacias Hidrográficas do Estado do Espírito Santo, 2008.

Lanna, A. E. A inserção da gestão das águas na gestão ambiental. Em Interfaces da gestão de recursos hídricos: desafios da Lei de Águas de 1997. Org. por Héctor Raúl Muñoz. 2 ed. Brasília:Secretaria de Recursos Hídricos, 2000, p. 75-109.

Lima, G. Aplicação de Simulação Computacional na Analise dos Conflitos entre os usos Múltiplos da Água na Bacia do Rio Atibaia no Estado de São Paulo. 2002 Dissertação (Mestrado em Hidráulica e Saneamento) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2002.

Mckinney, D. C.; Maidment, D. R. Expert geographic information system for Texas water planning. Journal of Water Resource Planning and Management, v.119, n.2, p.170-183, Mar/Apr, 1993.

Pádua Filho, W. P. Engenharia de Software – fundamentos, métodos e padrões. Rio de Janeiro, LTC, 2003.

Pressman, R. S.. Software Engineering: A Practitioner's Approach, 6ªedição, Mc Graw Hill, 2005.

Roques, T. V. P. Aplicação de modelos computacionais na análise de outorga para diluição de efluentes em corpos de água – fontes pontuais e difusas. Dissertação de Mestrado. Programa de Pós Graduação em Engenharia Ambiental da Universidade Federal do Espírito Santo. Vitória, 2006.

Uso de uma rede neural artificial para previsão do volume de lodo gerado em estações de efluente têxtil: um estudo de caso

Paulo Bousfield[1], Jéssica C. S. Silva[2] Lorival Chapuis[3]

[1,2] Departamento de Engenharias, Exatas e Tecnológicas, Administração – Universidade da Região de Joinville (UNIVILLE) – [3]Instituto Superior Tupy-SOCIESC - Joinville, SC - Brasil

paulo.bousfield@univille.br, jessica.caroline@univille.br, lorival.ecp@gmail.com

Abstract. *The main objective of this study is the development of an intelligent system for estimating the quantity of mud generated by an Effluent Treatment Station (ETS) at a Textile Industry; using the artificial neural networks (ANNs). The efficiency of removing the potential ETS pollutant was analyzed through a study of the main components, verifying if the organic parameter color and material require more studies, and alternatives to reduce their concentrations, in accordance with current legislation. The creation of an ANN involved the drawing up of a database, through a series of data from 2002 up to 2007 of the parameters analyzed by the company and the mud volume (in m^3) generated after effluent treatment. Aimed at obtaining an ANN of good generalization capacity, a series of feedforward architecture were elaborated; the supervised learning and the backpropagation algorithm, for the purpose of finding the number of neurons in the concealed layer. The outcome of the tests was based on the mean quadratic error and the linear regression of the results obtained from the network prediction. It was evident that the use of ANNs for modeling effluent treatment stations, are adequate and are highly relevant in improving the process modeling techniques.*

Resumo. *Este trabalho tem como objetivo principal o uso de uma Rede Neural Artificial para prever a quantidade de lodo gerado pela Estação de Tratamento de Efluentes (ETE) de uma Indústria Têxtil da cidade de Joinville-SC, utilizando o aprendizado conexionista. O volume de água utilizado nos processos fabris da indústria têxtil culmina no grande volume de efluente. Este trabalho é importante por que visa à redução do lodo gerado na estação de tratamento de efluentes de modo a minimizar o passivo ambiental gerado pelo acondicionamento deste material nos aterros. Foi analisada a eficiência de remoção do potencial poluente da ETE mediante a análise de componentes principais (ACP), verificando-se que o parâmetro cor e matéria orgânica requerem mais estudos e alternativas de redução das suas concentrações, embora estejam de acordo com a legislação vigente. Para a confecção da Rede Neural Artificial (RNA) foi elaborado um banco de dados através das séries históricas (de 2002 a 2007) dos parâmetros que a empresa analisa diária e semanalmente e do volume de lodo (m3) gerado após o tratamento do efluente. A fim de obter uma RNA com boa capacidade de generalização foi elaborada uma série de testes com arquitetura feedforward, o aprendizado*

supervisionado e o algoritmo backpropagation, tudo com o objetivo de encontrar o número de neurônios na camada oculta. O êxito dos testes foi baseado no resultado do erro quadrático médio e a regressão linear dos resultados obtidos de predição da rede. Ficou evidente que o uso de técnicas alternativas como a do aprendizado conexionista, neste caso as RNAs para modelagem de estações de tratamento de efluente, é adequado e tem grande relevância para aprimorar as técnicas de modelagem de processos.

1. Introdução

Atualmente um dos principais problemas encontrados nas estações de tratamento é o grande volume de lodo gerado. Este volume pode ser atribuído a vários fatores como a cor, o tipo de amido usado na etapa de engomagem, as fibras de algodão dos tecidos e os produtos químicos adicionados no processo de fabricação.

Uma técnica pouco utilizada para o controle e/ou otimização das estações de tratamento de efluentes têxteis são as ferramentas da inteligência artificial, as quais têm despertado grande interesse nos últimos anos para previsão e controle de processos diversos. Entre as técnicas mais utilizadas estão as Redes Neurais Artificiais, que possuem a capacidade de aprendizado através dos dados e apresentam boa generalização (Braga et al, 2007).

Objetivou-se com este estudo desenvolver um sistema computacional inteligente para previsão do volume de lodo gerado na estação de tratamento de efluentes (ETE) de uma empresa de Joinville, SC. Atualmente o setor têxtil da indústria brasileira contribui com 5,2% da receita total da indústria de transformação nacional, aproximadamente 33 bilhões de dólares. No tocante a quantidade de empregos, este setor possui 17,3% do total de empregos da indústria de transformação aproximadamente 1,53 milhões de empregos.

De acordo com Esquerre (2002) as RNA têm sido propostas como uma ferramenta promissora para a identificação de modelos através do tratamento de dados, sendo muito úteis; dada a sua capacidade para modelar processos complexos não-lineares, mesmo em processos onde a compreensão é limitada.

As redes neurais artificiais podem ser utilizadas também em modelos de predição, desde que o processo de correlação da estrutura não se altere. Normalmente, o objetivo principal da rede neural é modelar com precisão, predizer os processos de comportamento dinâmico, a fim de controlar e melhorar o desempenho destes processos (GONTARSKI et al,2000).

A sociedade tem exigido cada vez mais das empresas públicas e privadas a implementação de políticas ambientais, o que corrobora um dos objetivos deste trabalho e também está em consonância com estudos realizados na área de tratamento de águas e efluentes domésticos e industriais, que é a minimização de efluentes e resíduos.

Em resumo, este trabalho justifica-se pela necessidade da melhoria do processo industrial no que tange ao volume de lodo gerado nas estações de tratamento de efluentes, e, culminantemente, servirá de modelo para subsidiar o desenvolvimento de modelos de sistemas de tratamento de efluentes industriais com tecnologias nacionais.

2. Materiais e Métodos

A ETE onde foi desenvolvido este trabalho é composta por processo de tratamento biológico seguido de tratamento físico-químico. É formada por um tanque de equalização, uma calha Parshall, um tanque de homogeneização, um tanque de mistura, três tanques de aeração, um decantador, um flotador, um condicionador de lodo e um filtro prensa. Entre 30 e 35% do efluente tratado é reutilizado nos processos da empresa. O restante é lançado no curso hídrico.

Para o desenvolvimento da RNA foram considerados dois pontos de coleta: P1 (entrada), localizado no tanque de equalização, e P2 (saída), localizado na saída do filtro prensa. Foram utilizadas neste estudo as análises realizadas no laboratório da empresa no período de 2002 a 2007, com os seguintes parâmetros e periodicidade de coleta: Análises diárias: temperatura (Temp), oxigênio dissolvido (OD), pH, turbidez (Turb) e cor. Análises semanais: fósforo total (PT), nitrogênio amoniacal (NAm), sólidos suspensos (Ssus), demanda bioquímica de oxigênio (DBO5), demanda química de oxigênio (DQO), sólidos sedimentáveis (SSed) e sólidos totais (ST). O Standard Methods (APHA, 1998) e equipamentos de determinação de medidas foram usados para conduzir as análises.

2.1 Desenvolvimento da Rede Neural Artificial

Para a elaboração da rede neural artificial foram considerados apenas os parâmetros: OD, Temperatura, pH, PT, NAm, SSus, DBO5, DQO, SSed e ST analisados no laboratório da empresa, bem como os valores da temperatura ambiente (TA) e da precipitação pluviométrica (PP), obtidos na estação meteorológica da UNIVILLE.

O desenvolvimento da Rede Neural Artificial segue, neste trabalho, as seguintes fases propostas por Braga et al (2007), conforme mostra a figura 1.

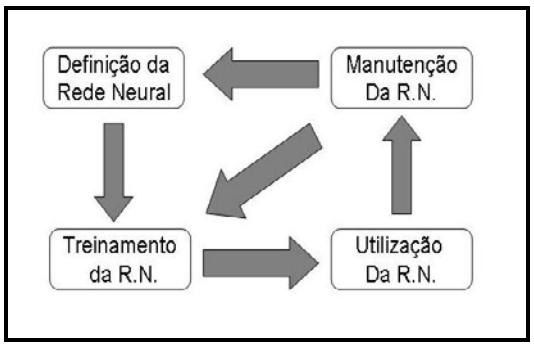


Figura 1. Fases do Desenvolvimento de RNA, adaptado de Braga (2007).

A plataforma escolhida para o desenvolvimento deste sistema foi o MATLAB® for Windows versão R2008 da MathWorks. Essa escolha se deve ao fato do MATLAB® ser uma plataforma amigável para o cálculo numérico e bastante utilizada no meio acadêmico. Para este trabalho, uma característica importante é que o elemento básico do MATLAB® é uma matriz que não requer dimensionamento, facilitando o manuseio dos dados do problema proposto.

2.2 Apresentação do Sistema

As figuras 2 e 3 mostram exemplos do sistema usado (por exemplo, os valores na tela dos parâmetros de entrada e saída).

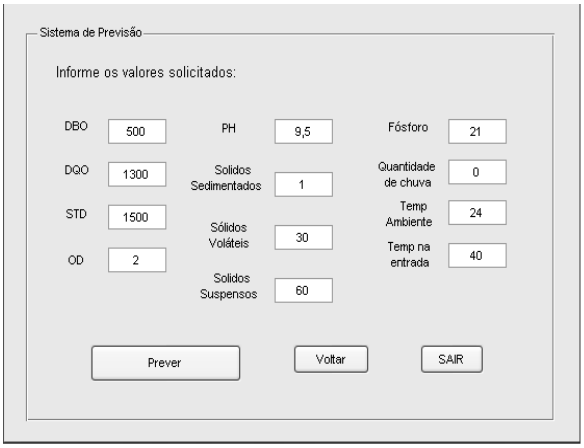


Figura 2 – Valores dos parâmetros de entrada.



Figura 3 – Sistema de valores de saída.

Ao utilizar este sistema como um simulador, pode-se notar que, em situações reais, a alteração de um parâmetro pode indicar a alteração de outros parâmetros relacionados.

3. Resultados

3.1 Eficiência da ETE

Os valores médios mensais das concentrações, os desvios padrões dos pontos de entrada (P1) e de saída (P2), as eficiências de remoção e os coeficientes de variação no período de 2002 a 2007 estão apresentados na tabela 1.

Tabela 1 – Valores médios, desvios padrão, eficiência de remoção (%) e coeficiente de variação (%) da ETE no período de 2002 a 2007.

Parâmetro	Efluente	Afluente	Eficiência (%)	CV (%)
DBO_5(mg/L)	400,13±92,26	23,42±5,83	94	23
DQO (mg/L)	1423,91±258,50	98,31±15,26	93	18
PT (mg/L)	3,95±1,70	0,55±0,13	88	43
NAm (mg/L)	11,82±5,22	7,44±4,41	37	44
OD (mg/L)	3,31±0,64	5,49±0,83	--	19
pH	9,71±0,42	6,65±0,16	--	4
SSed (mg/L)	0,62±0,55	0,13±0,10	79	89
SSus (mg/L)	775,73±255,92	174,24±37,41	78	33
ST (mg/L)	1318,76±245,59	1270,07±113	5	9
Temp(^{0}C)	39,93±2,59	31,61±2,34	--	6
Turb(NTu)	501,49±30,12	21,95±7,71	96	6
Cor (PTCO)	2690,08±164,60	118,17±39,27	96	6

Os valores de concentração de DQO do afluente (1419,27 mg/L) e da temperatura (40,07 ºC) estão condizentes com os encontrados por Cerqueira (2006) sobre efluente têxtil, onde se obteve valores de DQO e temperatura que variaram de 1226 a 2553 (mgO2/L) e de 41 a 47 (°C), respectivamente.

As concentrações médias (mínimas e máximas) mensais dos efluentes no período do estudo foram de: 8,89 a 36,82 (mg/L) de DBO5, 42,33 a 131,40 (mg/L) de DQO, 0,21 a 0,93 (mg/L) de PT, 2,31 a 25,1(mg/L) de N Am, 3,55 a 6,63 (mg/L) de OD, 6,33 a 7,06 do pH, 0,009 a 0,58(mg/L) de SSed, 30,22 a 220,23 (mg/L) de SSus, 1065 a 1504 (mg/L) de ST, 6,5 a 37,7 (NTu) de Turb; 26,9 a 36,2 ºC de temperatura e 63,3 a 178 (PTCO) de cor.

Os valores de DBO5 foram analisados de acordo com os valores de emissão constantes do decreto estadual de Santa Catarina nº 14250/81 que estabelece o valor máximo para DBO5 em 60 mg/L. Em relação aos valores da cor, a legislação atual fixa o valor máximo de lançamento de até 75 mg Pt/L. Todavia o estudo foi feito com valores dos anos de 2002 a 2007, período no qual ocorreram alterações nos valores da legislação. De 2002 a 2006 os valores máximos preconizados na legislação eram de 500mg Pt/L.

O sistema apresentou eficiência de 93% na remoção de DQO, valor superior ao encontrado Santos e Santaella (2002), (67 a 81%), com experimentos utilizando reator de lodos ativados de aeração prolongada. A remoção dos outros parâmetros foi: fósforo total, 86,3%, NAm, 36,2%, SSed, 79,3%, SSus, 77,5%, ST 3,1%, Turb, 95,6% e Cor, 95,6%. Os dois últimos parâmetros assemelham-se aos estudos de Hassemer et al (2002), que utilizaram o processo físico-químico com ozônio e coagulação/flotação para o tratamento de efluente têxtil, obtendo 98% e 97% de eficiência na remoção da cor e da turbidez, respectivamente.

3.2 Redes Neurais Artificiais

Para a definição do número de camadas, do número neurônios em cada camada e da função de transferência em cada neurônio, realizou-se uma série de testes utilizando a toolbox de Redes Neurais Artificiais, conforme pode ser visto na tabela 2.

Tabela 2 – Resultados dos testes realizados para definição da arquitetura da RNA.

	Percentual (%)			Num. Neurônios		Num	MSE - Erro Médio Quadrático			R		
Algoritmo	Trein.	Valid.	Teste	Camada Intermed.	Performance	Épocas	Trein.	Valid.	Teste	Trein.	Valid.	Teste
GDX	60%	20%	20%	2	0,0999	14	0,0915	0,1041	0,0864	0,3549	0,3053	0,3069
GDX	60%	20%	20%	2	0,0874	46	0,0876	0,1029	0,0909	0,3972	0,3360	0,2323
GDX	60%	20%	20%	4	0,0728	41	0,0738	0,0793	0,0896	0,5425	0,4291	0,4740
GDX	60%	20%	20%	4	0,0768	67	0,0788	0,0931	0,0898	0,4702	0,4248	0,4045
GDX	60%	20%	20%	6	0,0693	100	0,0693	0,0706	0,0770	0,5921	0,5824	0,4563
GDX	60%	20%	20%	6	0,0620	130	0,0674	0,0861	0,0697	0,5784	0,5222	0,5760
GDX	60%	20%	20%	8	0,0534	37	0,0508	0,0743	0,0768	0,6662	0,5513	0,5385
GDX	60%	20%	20%	8	0,0441	77	0,0475	0,0509	0,0676	0,7522	0,6800	0,5960
GDX	60%	20%	20%	10	0,2107	40	0,2115	0,1485	0,1678	0,2051	0,2630	0,2226
GDX	60%	20%	20%	10	0,0371	140	0,0371	0,0555	0,0567	0,7985	0,6923	0,7050
GDX	60%	20%	20%	12	0,0301	61	0,0310	0,0681	0,0630	0,8329	0,6657	0,6811
GDX	60%	20%	20%	12	0,0332	114	0,0345	0,0625	0,0448	0,8144	0,6681	0,7674
GDX	60%	20%	20%	15	0,0312	44	0,0348	0,0547	0,0565	0,8229	0,6748	0,6903
GDX	60%	20%	20%	15	0,0846	72	0,0859	0,1158	0,1025	0,3657	0,2548	0,2977
GDX	60%	20%	20%	20	0,0267	57	0,0286	0,0532	0,0371	0,8538	0,7210	0,7924
GDX	60%	20%	20%	20	0,0155	119	0,0196	0,0292	0,0283	0,9010	0,8584	0,8593
GDX	60%	20%	20%	25	0,0054	72	0,0065	0,0123	0,0132	0,9686	0,8886	0,9270
GDX	60%	20%	20%	25	0,0012	170	0,0012	0,0132	0,0139	0,9944	0,9391	0,9341
GDX	60%	20%	20%	30	0,0009	97	0,0011	0,0303	0,0242	0,9949	0,8561	0,8924
GDX	60%	20%	20%	30	0,0007	197	0,0007	0,0027	0,0646	0,9964	0,9882	0,9710

Foi definida uma camada oculta para esta arquitetura de rede, pois as redes diretas necessitam de pelo menos uma camada oculta para resolver problemas não linearmente separáveis, como neste trabalho, devido a complexidade dos parâmetros envolvidos no processo de tratamento. Não existe regra específica para definir a quantidade de neurônios que devem ser utilizados na camada oculta (Esquerre, 2002), como mostra a figura 4.

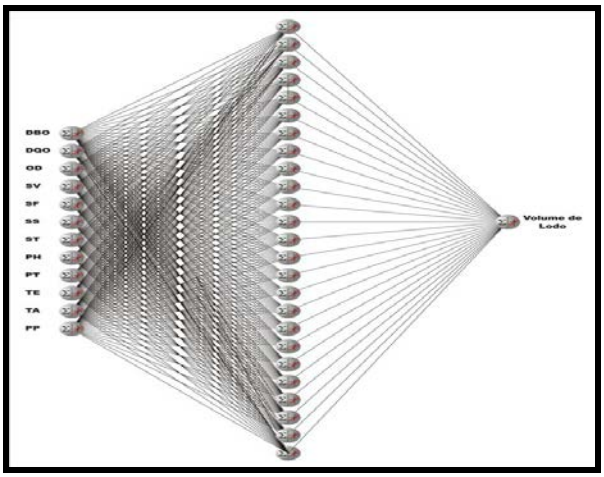


Figura 4. Arquitetura da RNA

A camada de entrada da arquitetura da rede é composta de 12 (doze) neurônios, os quais representam os parâmetros da rede, conforme segue:

Neurônio 1 - DBO - Demanda Bioquímica de Oxigênio,

Neurônio 2 - DQO - Demanda Química de Oxigênio,

Neurônio 3 - OD – Oxigênio Dissolvido,

Neurônio 4 - SV - Sólidos Voláteis,

Neurônio 5 - SF - Sólidos Suspensos,

Neurônio 6 - SS - Sólidos Sedimentados,

Neurônio 7 - ST - Sólidos Totais Dissolvidos,

Neurônio 8 - pH - Potencial de Hidrogênio,

Neurônio 9 - PT – Fósforo Total,

Neurônio 10 - TE - Temperatura do Efluente,

Neurônio 11 - TA - Temperatura Ambiente,

Neurônio 12 – PP – Precipitação Pluviométrica.

Neste estudo foram utilizados 25 neurônios na camada oculta, com base nos resultados obtidos no erro quadrático médio e na regressão linear como mostrados na tabela 2 e na figura 5. Na camada de saída tem-se apenas 1 neurônio, o volume de lodo como mostra a figura 4, e em todos os neurônios da rede, a função de transferência utilizada foi sigmóide.

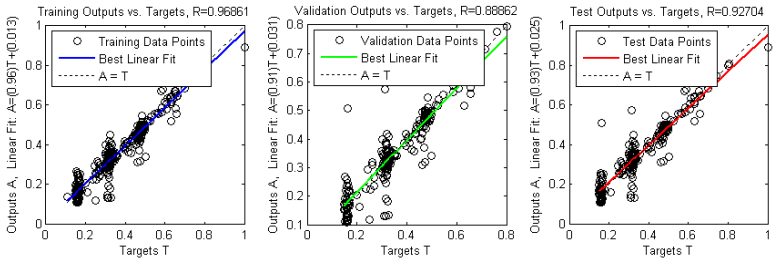


Figura 5 – Regressão Linear – Previsão da RNA e a saída desejada.

4. Conclusões

O estudo com os dados da indústria têxtil permitiu as seguintes conclusões:

A ETE obteve bons percentuais de remoção do potencial poluente, sendo para a DQO de 93%, fósforo total de 86,3%, NAm de 36,2%, SSed de 79,3%, SSus de 77,5%, ST de 3,1%, Turb de 95,6% e Cor de 95,6%. Todos os parâmetros estão com os valores de concentração de acordo com as legislações vigentes no período, para lançamento no corpo receptor que está enquadrado como rio classe III.

A arquitetura de Rede Neural Artificial para este sistema foi definida com 12 neurônios na camada de entrada, 25 neurônios na camada oculta e 1 neurônio na camada de saída, treinada com o algoritmo *Backpropagation*, obtendo resultados de Erro Médio Quadrático e Regressão Linear de acordo com os encontrados na literatura.

Este trabalho demonstra que o uso das redes neurais artificiais é uma importante ferramenta de auxílio para o gerenciamento de processos complexos, como é o caso das reações físicas, químicas e biológicas que ocorrem nas ETEs.

7. References

Alpha. Awwa Wef, (1998) “Standard methods for the examination of water and wastewater”. American Public Health Association, Washington, DC, USA.

Azevedo, F. M. et al, (2000). “Neural Networks with Applications Expert Systems”. Bookstore, Florianópolis, Brazil

Braga, A. P; CARVALHO, A. P. de L. de C.; LUDEMIR, T. B. “Redes Neurais Artificiais. Teoria e a Aplicações”. Rio de Janeiro: LTC, (2007).

Cerqueira, A. A. (2006), “Aplicação da Técnica de Eletrofloculação no Tratamento de Efluentes Têxteis”**.** Dissertação. Programa de Pós Graduação em Química. Universidade do Estado do Rio de Janeiro, Brazil UERJ, 111 p.

Esquerre, K. P. O.; MORI, M.; BRUNS, R. E., “Simulation of an industrial wastewater treatment plant using artificial neural networks and principal components analysis”. Brazilian Journal of Chemical Engineering, Vol. 19, No. 04, pp. 365 - 370, (October - December 2002)

Fiesc – Federação das Indústrias do Estado de Santa Catarina, (2007). Santa Catarina on Data. Florianópolis, Brazil

Gontarski, C. A., RODRIGUES, P.R. ; MORI, M., PRENEM, L.F. “Simulation of an industrial wastewater treatment plant using artificial neural networks”. Computers and Chemical Engineering 24 (2000) 1719-1723.

Hassemer, M.E.N.; SENS, M.L. “Tratamento do efluente de uma indústria têxtil. Processo físico-químico, ozônio e coagulação/floculação. Engenharia Sanitária e Ambiental”, v.7, n.1, (2002).

Haykin, Simon (2001) “Neural Networks: Principles and Practice”**.** 2. Ed. Bookman, Porto Alegre, Brazil.

IEMI – Instituto de Estudos de Marketing Industrial. (2008) “Textile Industry Sector Report”**.** São Paulo, Brazil.pp 103-117.

Marques, F. D et al. (2005) “Application of Time-Delay and Recurrent Neural Networks for Identification of Hingeless Helicopter Blade Falpping and Torsin Motions.” J. of the Braz. Soc. Of Mech. Sci. & Eng. April-June 2005, Vol. XXVII, No 2, Brazil. pp 97-103.

SANTOS. A.B. dos. SANTAELLA, T. S. Remoção de DQO de água residuária de indústria têxtil empregando o processo de lodos ativados em batelada. Revista Engenharia Sanitária e Ambiental, v. 7. n. 3, 2002.

Um sistema para previsão de impactos gerados pela instalação de indústrias e sua influência sobre ecossistemas costeiros no extremo sul do Brasil

Camila D. Thomasi[1] ,Gerson L. Nunes[1], Rodrigo G. A.Toledo[2], Priscila S. Teixeira[2] , Márcio M. Jugueiro[1], Diana F. Adamatti[1], Carlos R. A. Tagliani[2]

[1]Centro de Ciências Computacionais – Universidade Federal do Rio Grande (FURG)
Caixa Postal 474 – 96.201-900 – Rio Grande – RS – Brasil

[2]Laboratório de Oceanografia Geológica – Universidade Federal do Rio Grande (FURG)
Caixa Postal 474 – 96.201-900 – Rio Grande – RS – Brasil

{camilathomasi, dfrabu, paulista.oceano, marcio.juguero, dianaada}@gmail.com, pri_geografia@yahoo.com.br, crtagliani@log.furg.br

Abstract. *This paper presents an atmosferic dispersion pollutants' simulator. There will be evaluated the potential's risk contamination of the surrounded area near the industrial zone by identifying the most affected areas. For this, It was used a model of pollutants' propagation recognized by the scientific community as well as selected computational tools that fit the proposed implementation.*

Resumo. *Este artigo apresenta um simulador de dispersão de poluentes atmosféricos. Será avaliado o risco potencial de contaminação dos ambientes adjacentes à zona industrial identificando as regiões mais atingidas. Para tanto, foi utilizado um modelo de propagação de poluentes reconhecido pela comunidade científica, bem como escolhidas ferramentas computacionais que se adequassem à implantação proposta.*

1. Introdução

A emissão de poluentes na atmosfera tornou-se um grave problema para a saúde da população e para o meio ambiente devido ao crescimento tecnológico-industrial acelerado e o consequente aumento das emissões nos últimos anos. Nesse contexto, também têm aumentado o número de estudos sobre a dispersão de poluentes na atmosfera usando métodos computacionais que auxiliem na previsão de impactos sobre a saúde humana e na preservação de sistemas naturais.

Para Mattos et al. (2009) gerenciar os recursos naturais consiste em administrar o uso de plantas, terras e animais provendo qualidade de vida à população sem comprometer gerações futuras. Requerendo assim, que o governo, cidadãos e outros atores interajam para o bem comum. Neste sentido, a simulação computacional surge

como um mecanismo que reproduz sistemas próximos da realidade, além de apoiar na previsão de consequências futuras e auxiliar na tomada de decisão (Freitas Filho, 2001).

Para descrever adequadamente a relação entre as emissões de poluentes na atmosfera e a qualidade do ar, Costa (2004) considera que um modelo deve levar em conta os parâmetros físicos meteorológicos de escala local que ocasionam o transporte de poluentes de forma a reproduzir corretamente o seu campo de concentração. Permitindo, então, compreender a relação entre emissão de poluentes, variáveis meteorológicas e os parâmetros de dispersão da pluma.

Nesse sentido, o presente trabalho propõe o estudo da dispersão de poluentes atmosféricos em Rio Grande (RS) através da aplicação de um modelo numérico tridimensional. O objetivo é avaliar o raio de dispersão de partículas totais em suspensão e analisar o risco potencial de contaminação dos ecossistemas costeiros adjacentes à zona industrial da cidade, utilizando um Sistema de Informações Geográficas (SIG). O simulador permite verificar a direção e a distância que essa pluma pode alcançar e também o potencial de contaminação dos ambientes naturais adjacentes.

O artigo é dividido da seguinte forma: na Seção 2 são apresentados os conceitos básicos de dispersão de poluentes na atmosfera. A Seção 3 apresenta as ferramentas e técnicas utilizadas. A Seção 4 apresenta o protótipo do simulador desenvolvido e na seção 5 são apresentados os resultados obtidos e discussões. A seção 6 apresenta as conclusões e trabalhos futuros.

2. Dispersão de poluentes na atmosfera

A presença de substâncias sólidas, líquidas ou gasosas emitidas para a atmosfera por indústrias, veículos, usinas termelétricas, entre outras fontes, são consideradas poluentes somente quando suas concentrações atingem níveis elevados o suficiente para provocar efeitos nocivos aos seres humanos, animais, vegetação e ecossistemas como um todo.

Entre os principais poluentes estão os gases e material particulado em suspensão. O segundo pode causar efeitos adversos para a saúde humana em pessoas com doenças pulmonares como asma e bronquite, bem como afetar a vegetação, reduzir a visibilidade, contaminar solos Seinfeld (1986) e corpos hídricos.

Os estudos sobre as condições atmosféricas propícias à acumulação de poluentes têm indicado que os episódios tendem a ocorrer em situações em que se formam sistemas se alta pressão estacionários que provocam ventos fracos, em situações de alta estabilidade atmosférica. Ao contrário, em situações de ventos moderados a fortes, a pluma de poluentes tende a se dispersar.

De acordo com a avaliação dos dados da estação meteorológica de Rio Grande, Saraiva e Krusche (2001) concluíram que os eventos com alta concentração de material particulado em suspensão ocorrem quando são estabelecidas pressão alta associadas a ventos fracos, baixa temperatura e alta umidade relativa. Todos esses efeitos somados dificultam a dispersão de poluentes. Além disso, foi percebido que o vento nordeste é predominante na região.

3. Ferramentas e técnicas utilizadas

Esta seção descreve as ferramentas utilizadas para o desenvolvimento do simulador que foram escolhidas pelos seguintes motivos: disponibilidade, documentação, geração e manipulação de mapas de forma iterativa.

3.1. MATLAB

O MATLAB é um software interativo que utiliza uma linguagem de alto nível para o desenvolvimento e gerenciamento de código, arquivos e dados. Dentre as vantagens desta ferramenta podemos destacar a flexibilidade, podendo ser usado em diferentes aplicações, Mathworks (2011).

A facilidade de uso em diferentes tipos de sistemas pode ser obtida com os *toolboxes,* aplicativos específicos disponibilizados separadamente. Dentre os *toolboxes* utilizados podemos destacar o de processamento de sinais, redes neurais, sistemas de controle e lógica fuzzy, manipulação de mapas, entre outros. Sendo o aplicativo para manipulação de mapas o adotado neste trabalho.

3.2. Mapping Toolbox

Mapping toolbox é um dos aplicativos específicos fornecidos pelo MATLAB que disponibiliza ferramentas e utilitários para análise de dados geográficos, criação e manipulação de mapas.

Com essa ferramenta é possível importar tanto dados vetoriais como dados *raster*. As funcionalidades deste *toolbox* permitem desenvolver soluções personalizadas para diversos problemas geográficos, Mathworks (2011).

3.3. Sistemas de Informação Geográfica (SIG)

Os sistemas de informações são utilizados para manipular, sintetizar, pesquisar, editar e visualizar informações, geralmente armazenadas em bases de dados computacionais (Ferreira, 2006). Assim, os SIGs se tornam importantes instrumentos de analises de dados ambientais, pois permitem integrar uma série de informações, tanto dados vetoriais, como pontos, linhas e polígonos, bem como dados matriciais, como imagens *raster*, às quais estão associados bancos de dados espacializados.

Portanto, facilitam a interpretação de fenômenos de origem natural ou mesmo de origem antrópica, como é o caso da emissão atmosférica de poluentes resultantes de atividades industriais, permitindo identificar seus efeitos e potenciais impactos sobre os ecossistemas.

4. Protótipo do simulador desenvolvido

Nesta seção é apresentada toda a arquitetura do simulador desenvolvido, a forma como é feita a inserção das novas indústrias e o modelo gaussiano responsável pelo cálculo da concentração de poluentes.

4.1. Modelagem do simulador desenvolvido

O objetivo do simulador é analisar os efeitos causados pela emissão de poluentes atmosféricos ocasionados pela inserção de uma indústria em determinada região. Diante disso é possível ter uma ideia de como serão afetadas as regiões vizinhas.

O projeto encontra-se na fase da criação do protótipo, que tem como objetivo calcular a concentração de poluentes gerados pela indústria e avaliar o impacto causado pela propagação destes poluentes na região analisada.

O simulador foi subdividido em três módulos. O primeiro módulo (manipulação de mapas) tem como funções: ler, escrever, manipular e apresentar arquivos vetoriais. O segundo módulo (inserção de objetos) permite escolher a posição geográfica onde as indústrias serão inseridas e também seus parâmetros iniciais de simulação. O terceiro módulo (geração dos impactos) consiste no cálculo e apresentação da concentração de cada ponto, de acordo com sua latitude e longitude. Nesse módulo também é gerado um arquivo de saída com formato *shape* que contém a junção das camadas de: indústrias, pontos de poluição e o mapa da região.

O diagrama de fluxo de dados básicos do funcionamento do simulador é apresentado na Figura 1.

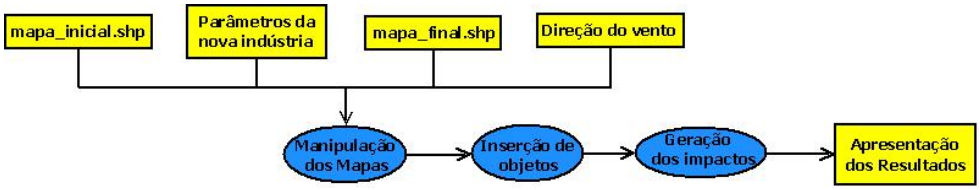


Figura 1. Diagrama de fluxo de dados do simulador

De acordo com diagrama de fluxo, as entradas básicas necessárias para iniciar a simulação são: o arquivo do mapa (*shape*) da região de interesse, os parâmetros característicos da indústria a ser inserida, o nome de um arquivo (*shape*) de saída e a direção do vento.

Na primeira etapa, os mapas são lidos, manipulados e apresentados. Após é feita a inserção da indústria na qual o usuário deve escolher os respectivos parâmetros que deseja simular. Na etapa de geração dos impactos, é feita a simulação da emissão de poluentes na atmosfera, sendo calculada a concentração de poluentes nos pontos da região de acordo com a direção dos ventos. O objetivo é mostrar para o usuário os possíveis problemas que ocorrerão num futuro próximo. Ao final do processo os gráficos de saída da simulação são salvos e o usuário pode realizar a avaliação.

4.2. Inserção de objetos

Além da camada contendo os polígonos e as informações do mapa, temos outras duas camadas que são criadas: a indústria que se deseja simular e a dos pontos de poluição gerados pelo cálculo da concentração dos poluentes.

Na camada das construções é inserida a indústria e suas características. Na outra camada são inseridos pontos com diferentes concentrações de poluição. Tanto o objeto indústria quanto o objeto ponto são definidos pelos dados da geometria e pelos dados referentes à simulação. A estrutura desses objetos é definida na figura 2.

O objeto que representa uma indústria é mostrado na Figura 3(a). Na Figura 3(b) é mostrada uma parcela do mapa dos bairros de Rio Grande com a indústria inserida. Na figura 3(c) o grid de pontos e na figura 3(d) o objeto que representa um ponto.

Atributo	Tipo	Descrição
Geometry	*string*	Geometria do polígono (‘*polygon*’ ou ‘*point*’)
BoundingBox	[2x2 *double*]	Extremos do polígono;
X	[1xN *double*]	Conjunto de coordenas x do polígono;
Y	[1xN *double*]	Conjunto de coordenas y do polígono;
POLUICAO	*double*	Nível de poluição gerado pela construção
LOCAL_X	*double*	Coordenada x da localização do centro da construção;
LOCAL_Y	*double*	Coordenada y da localização do centro da construção;

Figura 2. Estrutura dos objetos indústria e ponto

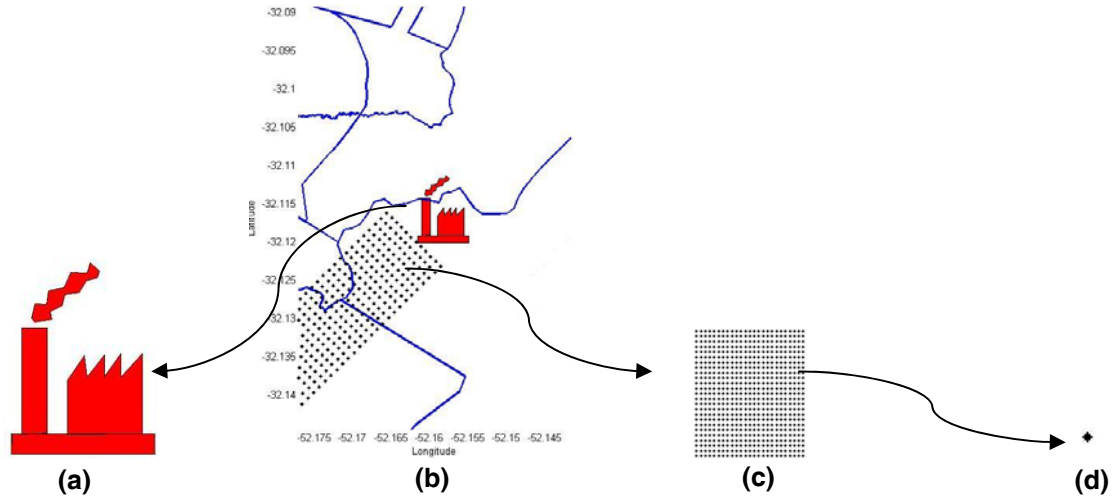


Figura 3. Objetos da simulação: (a) representação da indústria. (b) indústria inserida no mapa. (c) matriz de pontos de concentração de poluentes (d) objeto ponto.

4.3. Modelo de propagação da poluição

Os modelos analíticos não simplificam diversos componentes das equações de Navier-Stokes, Drazin e Riley (2006), fazendo com que as simulações tenham um custo computacional mais elevado. Devido a isso foi escolhido para simulação com dados reais o modelo gaussiano ISC (Industrial Source Complex) que possibilita a modelagem de emissões de uma ampla gama de fontes presentes em um complexo industrial típico.

O modelo de dispersão de poluentes foi escolhido dentre tantos por apresentar um baixo custo computacional. Ele possui uma série de simplificações climatológicas, tais como: perfil de vento, temperatura e pressão constantes ao longo do tempo. Mesmo assim podemos ter uma ideia de como serão afetadas as regiões próximas a fonte emissora. É considerado ainda que a dispersão de poluentes emitidos por uma indústria ocorre somente na troposfera a qual é dividida em 11 camadas que variam de 0 a 1 quilometro em um intervalo de 100 em 100 metros.

Segundo EPA (1995), a base do modelo é a equação linear e estacionária de pluma Gaussiana que, com algumas modificações, é utilizada para modelar emissões de fontes simples em um ponto - as chaminés. A equação 1 mostra como é calculada a concentração de poluentes em fontes pontuais.

$$x = \frac{Q.K.V.D}{2.\pi.U_s.\sigma_y.\sigma_z}\exp\left[-0.5\left(\frac{y}{\sigma_y}\right)^2\right] \quad (1)$$

A concentração x depende de diversos fatores: Q é a taxa de emissão de poluentes em gramas por segundo, K é o coeficiente de expansão utilizado para converter as concentrações calculadas para unidades desejadas, o termo vertical V representa a dispersão vertical devido à reflexão, podendo ser ou não habilitado, D é o termo de decaimento, α_y e α_z são os desvios padrões da distribuição das concentrações lateral e vertical em metros, U_s é a velocidade do vento na altura da chaminé e Y a distância do vento cruzado.

4.4. Cenário de simulação

O *shape* utilizado para a simulação corresponde aos limites políticos dos bairros da cidade de Rio Grande, disponibilizados pelo Instituto Brasileiro de Geografia e Estatística (IBGE), como mostra a Figura 4.

O arquivo *shape* é associado a um arquivo DBF (*dBase file extention*), que corresponde a uma Tabela de Atributos. Nestes registros temos um campo específico referente à concentração de poluentes que é calculado pelo modelo.



Figura 4. Bairros do município de Rio Grande.

O cenário inicial da simulação é descrito pela Figura 4. Para iniciar o processo é necessário informar alguns parâmetros: o arquivo *shape* de entrada, o nome arquivo *shape* de saída, a localização (coordenada geográfica) da indústria que será inserida, suas características e a direção do vento.

Os parâmetros passados como características da indústria são: sua taxa de emissão Q (g/s), a velocidade do vento (m/s), altura da chaminé (m), o tipo de terreno (rural ou urbano), a categoria de estabilidade (informa a medida da capacidade dispersiva na atmosfera), a temperatura ambiente (°C), a massa molar média dos poluentes (g/mol), a pressão (Pa), o intervalo amostral de pontos da região de interesse, além de informar se existe ou não reflexão ou deposição no solo.

A indústria pode ser inserida em qualquer localização, porém durante a realização das simulações foi escolhido o local onde se concentra o complexo industrial da cidade de Rio Grande, como mostrado na Figura 5. Para os testes foi considerada uma vazão de Q=275g/s, que corresponde a uma indústria de médio/grande porte, velocidade do vento de V=3m/s, temperatura atmosférica de 18ºC, e massa molar 30g/mol. A altura da chaminé é 30m, o tipo de terreno urbano, sendo F a categoria de estabilidade usada. A pressão é 1010Pa, sendo considerado que há reflexão e não há

deposição no solo. Os intervalos de terreno usados são de x = [0, 5Km] e y = [-550m, 550m] com pontos de 10 em 10 metros. Foram simulados apenas os ventos nas direções nordeste e sudoeste, predominantes na região de Rio Grande.



Figura 5. Mapa com uma indústria inserida.

5. Resultados e discussões

O simulador gerou uma matriz tridimensional georeferenciada com latitude, longitude e a concentração de poluentes em cada ponto apresentados na posição correta no mapa de acordo com a dispersão e o sentido do vento.

Os valores de concentração de poluentes são calculados e exportados para um arquivo *shape* de pontos. Este arquivo de saída pode ser carregado em um SIG, o qual possibilita a visualização dos valores da concentração de poluentes em cada ponto simulado, conforme a figura 6.

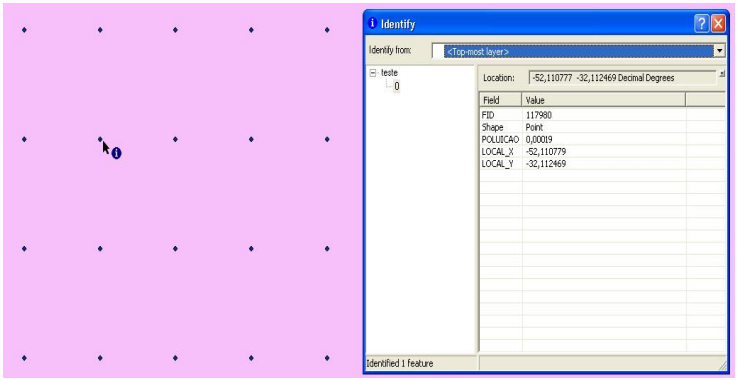


Figura 6. Informações de um ponto utilizando SIG.

O SIG gerou uma superfície, a qual foi convertida em um arquivo *raster*, cujos valores de pixel correspondem aos valores de concentração de poluentes. Deste arquivo, foram extraídas isolinhas de concentração correspondentes aos valores de 60, 80, 150 e 240µg.m^{-3} de acordo com os valores-limite de Material Particulado Total estipulados pela resolução N.º 003 de 28 de junho de 1990 do Conselho Nacional de Meio Ambiente CONAMA (1990) apresentados na figura 7.

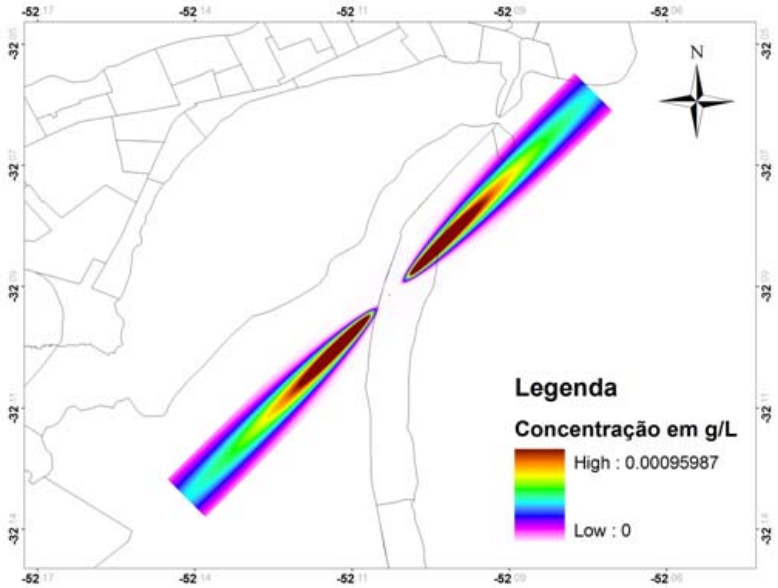


Figura 7 – Concentração de Partículas Totais em Suspensão na atmosfera em g/L simulado de 0 até 100m de altura.

Segundo CONAMA (1990), Padrões Primários de Qualidade do Ar são as concentrações de poluentes que, ultrapassadas, poderão afetar a saúde da população. Padrões Secundários de Qualidade do Ar são as concentrações de poluentes abaixo das quais se prevê o mínimo efeito adverso sobre o bem-estar da população, assim como o mínimo dano à fauna, à flora, aos materiais e ao meio ambiente em geral.

O Padrão Primário considera que a concentração média geométrica anual de partículas totais não deve ultrapassar 80µg/L e a concentração média de 24 horas deve ser de 240µg/L. O Padrão Secundário considera que a concentração média geométrica anual deve ser de 60µg/L, e a concentração média de 24 horas deve ser de 150µg/L. Ambos os padrões não podem ser excedidos mais de uma vez por ano.

Através do cálculo da distância em metros da fonte emissora até a isolinha de 150µg/m^3 se pôde estimar a distância crítica em que o ambiente no entorno está sendo afetado nesse cenário de simulação, cujo valor corresponde a aproximadamente 5400 metros. A partir do ponto de emissão, foi gerando um *buffer* múltiplo com quatro classes de distância crítica da indústria, conforme mostra a figura 8.

Essa informação é importante para integrar os Planos de Emergência para Episódios Críticos de Poluição do Ar, pois permite definir as áreas mais ameaçadas pela presença de altas concentrações de poluentes na atmosfera em curto período de tempo. Isso é resultado da ocorrência de condições meteorológicas desfavoráveis à dispersão dos poluentes. Nesses casos, a resolução supracitada define que os governos de Estado e dos Municípios, assim como entidades privadas e comunidade em geral, tomem providências com o objetivo de prevenir graves e iminentes riscos à saúde da população.

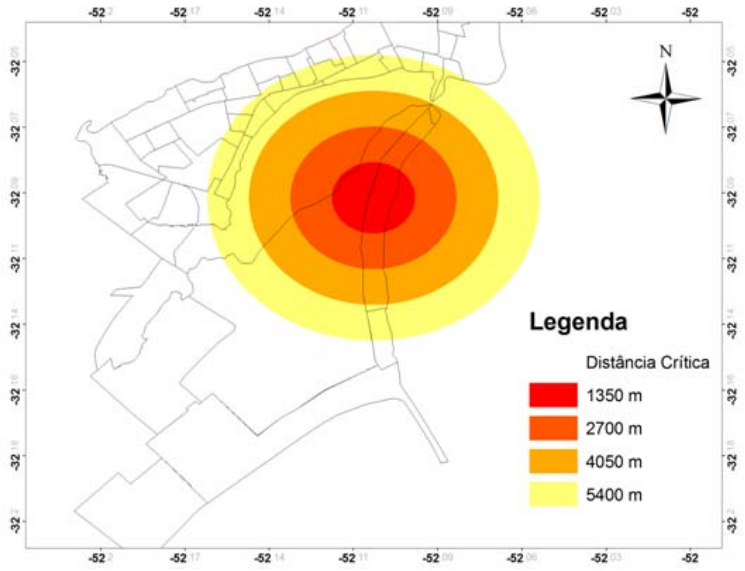


Figura 8 – Mapa mostrando o *buffer* de distância crítica da fonte emissora.

6. Conclusões e Trabalhos Futuros

O projeto "Rede Estadual de Simulação Social", apoiado pela FAPERGS/CNPq, tem como um dos eixos de pesquisa o estudo de problemas ambientais. As pesquisas realizadas objetivam a construção de um simulador para avaliação dos riscos ocasionados por grandes construções tanto em ambientes urbanos como rurais.

Com a simulação foram identificadas as zonas mais atingidas de acordo com a proximidade da fonte emissora, considerando condições meteorológicas características propícias à acumulação de poluentes. O simulador se mostrou eficaz para a análise de efeitos locais da dispersão de efluentes atmosféricos sobre os ecossistemas costeiros, indicando uma distância mínima segura de acordo com os limites máximos aceitáveis previstos pela legislação vigente.

A ideia seria estender o simulador desenvolvido, não apenas para simulação de riscos causados pela inserção de indústrias como também para outros tipos de construções. Outro ponto seria fornecer uma interface gráfica bastante usual e intuitiva ao usuário.

Por fim, deseja-se desenvolver um SAD (sistema de apoio à decisão) que auxilie as comunidades e órgãos envolvidos para correta tomada de decisão, contribuindo para o aprendizado e análise de politicas públicas relacionadas ao meio ambiente.

Agradecimentos

Agradecemos à FAPERGS (Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul) pelo financiamento e pelas bolsas de pesquisa e desenvolvimento, processo no. 10/0049-7.

Referências

Câmara, G., Casanova, M.A., Hemerly, A., Medeiros, C.M.B.; Magalhães, G. (1997), "Anatomia de Sistemas de Informação Geográfica". Curitiba, SAGRES Editora.

Costa, C. P. (2004), "Influência de efeitos não-locais na dispersão de poluentes na Camada Limite Planetária". Dissertação. Programa de Pós-Graduação em Matemática Aplicada da Universidade Federal do Rio Grande do Sul. Porto Alegre.

Conselho Nacional de Meio Ambiente (CONAMA). Resolução N.º 003 de 28 de junho de 1990.

Drazin, P., Riley, N., (2006), "The Navier-Stokes equations: a classification of flows and exact Solutions", ed. Cambridge University Press, UK.

EPA - Environmental Protection Agency (1995), User´s guide for the industrial source complex (ISC3), dispersion models volume II – Description of model algorithms. 128p.

Ferreira, N. C. (2006) "Apostila de Sistemas de Informações Geográficas". Apostila Elaborada para a Disciplina Sistema de Informações Geográficas da Coordenação da Área de Geomática do Centro Federal de Educação Tecnológica do Estado de Goiás para Aulas do Curso Superior de Tecnologia em Geoprocessamento.

Freitas Filho, P. J. (2001), "Introdução à Modelagem e Simulação de Sistemas - com Aplicações em Arena", ed. Visual Books,Florianópolis, SC, Brasil . v. 1. 322 p.

Mathworks- MATLAB and Simulink for Technical Computing, MATLAB Introduction and Key Features. Disponível em: http://www.mathworks.com/products/matlab Acesso em: 21 fev. 2011.

Mathworks- MATLAB and Simulink for Technical Computing, MATLAB Introduction and Key Features. Disponível em: http://www.mathworks.com/products/mapping, Acesso em: 21 fev. 2011.

Mattos J., Rosa Jr L., Pilla. M. (2009), "Desafios e Avanços em Computação: o estado da arte".1 ed. Editora e Gráfica Universitária, Pelotas RS :, v. 1, c. 3, pp. 67-86.

Saraiva, L. B., Krusche, N. (2001),"Condições Atmosféricas Favoráveis à concentração de Poluentes em Rio Grande, RS". Resumos do II Congresso em física da camada limite planetária e modelagem de processos de dispersão. Santa Maria, RS. v.1. p.14.

SEINFELD, J. H., (1986), "*Atmospheric Chemistry and Physics of Air Pollution"*. *Ed.* John Wilwy & Sons, Canadá. 738 p.

Yu, J.; Chen, Y; Wu, J.P (2009). "Cellular automata and GIS based landuse suitability simulation for irrigated agriculture. In: 18th World IMACS/MODSIM Congress, Cairns, Australia, p 7.

A GA for Solving a Type of Sustainable Supply Chain Design Problem

Ernesto Del R. Santibanez-Gonzalez[1], Geraldo Robson Mateus[2], Henrique Pacca Luna[3]

[1]Department of Computer Science – Federal University of Ouro Preto (UFOP)
Ouro Preto, 35400-000, MG, Brazil

[2]Department of Computer Science – Federal University of Minas Gerais
Belo Horizonte, MG, Brazil

[3]Computer Science Institute
Federal University of Alagoas (UFAL) – Maceió, AL, Brazil

santibanez.ernesto@gmail.com, mateus@dcc.ufmg.br, hpluna@gmail.com

Abstract. *We propose a hybrid genetic algorithm (GA) for solving a sustainable supply chain design problem that arises in the public sector. There is little research being done in mathematical modeling and solutions methods for these problems. The paper describes a mixed-integer 0-1 model (MIP) for this sustainable problem in which we have to determine in a network of two layers the number of facilities to be located at sites chosen from among a given set of candidate sites. Sustainable issues are integrated into the model by reducing the greenhouse gas emissions produced by the transportation and the operation of the facilities. We report computational results for instances generated from a known OR test library.*

1. Introduction

In 1987 the United Nations World Commission on Environment and Development (UNWCED) published Our Common Future Report. In this report was defined sustainable development as "development that meets the needs of the present without compromising the ability of future generations to meet their own needs". This report was a kick-off for a number of interdisciplinary studies in the field of sustainability.

By 2020 the economic impact of climate change in the world will reach 20% of the global GDP. Considering the magnitude of the impact on the global economy, the environment and the society, many agencies of the government and international institutions are taking actions to reduce and to control the emission of GHG. In this respect, this paper focuses on how governments can deploy supply chain networks for servicing people while minimizing the costs of installation, operation and transportation and, at the same time reducing the GHG emissions.

Governmental agencies and companies adopting a friendly sustainable management are facing a number of changes, from the strategy level till the operational point of view, affecting their people and impacting their business processes and their technology. In this regard, as Simchi-Levi et al. (2007) pointed out, "the strategic level deals with decisions that have a long-lasting effect on the firm. These include decisions regarding the number, location and capacities of warehouses and manufacturing plants,

or the flow of material through the logistics network". They established a clear link between facility location models and strategic decisions of supply chain management (SCM). Also, governmental agencies and companies realized that to be committed with sustainable practices could imply changes in the criteria to design and to manage supply chain. That is to say, in addition to the costs of transport, operation and installation and considerations on the level of service, the sustainable models need to consider GHG emission costs.

Supply chain design based on economic consideration has been well covered in the literature. On the other hand, the field of sustainable supply chain design and management (SSCM) is quite new (Seuring and Muller, 2008). The greatest benefits of applying SCM are obtained by an extended analysis including organizations upstream - closer to the raw materials- and downstream -closer to the consumer- of the supply chain and then back again so that the unsold products are recycled. But, by extending the focus, what this really does implies more organizations, multiplying the relation between the organizations and getting a more complex supply chain (SC) to manage. Considering that, according to Choi and Wu (2009) the focus of the supply chain management literature has been on dyadic networks (supplier units-customer units).

This paper proposes a genetic algorithm for solving a supply chain network design problem that arises in the public sector considering sustainable constraints in the form of restrictions on the dioxide carbon equivalent emissions. The authors are not conscious of any article tackling the problem of sustainability that arises in the location of such public facilities as schools or hospitals. We present a mixed-integer 0-1 network design model who allows to analyze the impact of restrictions in the GHG emissions on the fixed and transportation costs and in the location of facilities in a network of two layers.

In this paper, in Section 2 is analyzed some literature in connection with the problem. In section 3 is presented the mixed-integer 0-1 programming model. In section 4, we discuss the genetic algorithm implementation for solving the problem. In section 5 we provide some numerical results. Finally in section 6 we give some conclusions of the work.

2. Literature Review

The UNWCED report published in 1987 defined sustainable development as "development that meets the needs of the present without compromising the ability of future generations to meet their own needs".

Above and beyond other aspects, there are principally two kinds of questions related to the sustainability. One of them is the emission of GHG. For the production of a big quantity of GHG to satisfy the present needs – for example, in the processes of manufacture and distribution - we affect the climate of a dangerous way that finally is going to compromise the capacity so that future generations could satisfy their own needs. In this paper we focus on reducing the GHG emissions caused by the operation of public facilities and the transportation activities to satisfy the demand. We model the problem considering a network of two layers. We locate facilities in these layers to satisfy a fixed demand, minimizing installation and operating costs and constraining the GHG emissions.

Supply chain management (SCM) is being used to address the problem of reducing the economic impact of climate change generated by GHG emissions. For the purpose of this paper, SCM is a multidisciplinary management approach to master a set of interacting organizations. These organizations share different resources, products, services and information, with the target to obtain competitive advantages and to improve the profitability, both in the individual form and in the collective form. (Simchi-Levi et al., 2007). Given the complexity of the network, according to Choi and Wu (2009), the focus of the literature of SC has been in networks of dyadic (supplier firms – customer firms).

Disciplines integrating environmental practices into the supply chain have been called in a number of different ways. Some of them are: Sustainable Supply Chain Management (SSCM) and, Green Supply Chain Management (GSCM). Srivastava (2007) did a careful review of the literature and he showed that a wide frame of reference for GSCM has not been sufficiently developed. He defined GSCM as an integrated environment, including product design, sourcing and selection of material, manufacturing processes, delivery of the final product to the consumers, and end-of-life management of the product after its useful life. In this paper, we do not make any distinction between sustainable and green supply chain.

Mathematical modeling for designing sustainable supply chain is attracting many researchers to this field. But, according to Seuring and Muller (2008), the field of sustainable supply chain design and management (SSCM) is quite new. In this work, the proposed model focuses on two sustainable issues: economic and environmental aspects of GHG emissions. On the other side, till now much research has been done in the field of private companies' location theory. The authors are not aware of any paper addressing the sustainable problem that arises in the location of such public facilities as schools or hospitals.

Hugo and Pistikopoulos (2004) develop a multi-objective mixed-integer 0-1 model for deciding location and capacity expansion of facilities (plants), and transportation issues in a given planning horizon. They maximize profit and minimize the environmental impact of the plant operations while satisfying the market demand for products. They presented numerical results for a small problem of 3 candidate plants, 3 customers, 2 products, 2 raw materials and 5 periods. In a later work, Hugo and Pistikopoulos (2005) extend the previous model and reformulate the problem as a stochastic programming model that can address the decision-making process under uncertainty. Ramudhin et al. (2008) propose a mixed-integer 0-1 programming model for the GSC design problem. Taking into account environmental aspects, they analyze the impact of transportation, subcontracting, and production activities on the design of a supply chain network. The model is tested considering the case of a steel product manufacturer with three freight transportation modes, a product with two semi finished products that are manufactured from four parts, and at least two suppliers are competing to supply each part. The model is first solved by CPLEX Interactive Optimizer V10.0. The authors also use Goal Programming to determine the best trade-offs between two conflicting objectives: the total logistics cost and carbon emissions. In Chaabane et al. (2010) is extended the previous model considering life cycle assessment (LCA) principles in addition to the traditional material balance constraints at each node in the supply chain. They propose a multi-objective mixed-integer 0-1 model to support sustainable supply chain design over a long-term period of time. The model

distinguishes between solid and liquid wastes, as well as gaseous emissions due to various production processes and transportation modes. The model is used to evaluate the tradeoffs between economic and environmental objectives under various cost and operating strategies for an aluminum company. Finally, Diabat and Simchi-Levi (2010), use a mixed-integer 0-1 programming model including carbon emissions restrictions for designing green supply chains. The problem is to decide which plants and Distribution Centers (DCs) to open, how the DCs are allocated to the plants, and how the DCs distribute multiple types of products to satisfy retailers' demands. The objective is to minimize the total facility opening and products distribution costs subject to the total carbon emission is not more than a predetermined emission cap. They formulate the problem as a two-echelon multi-commodity facility location problem with a carbon emission constraint. They present numerical results for a 7 candidate plants, 18 candidate DCs, 63 retailers, and a single type of product. To solve the instances, the authors use ILOG CPLEX 11.0 MIP solver in the GAMS modeling language.

The problem of locating facilities and allocating customers is not new to the operations research community and covers the key aspects of supply chain design (Daskin et al., 2005). Simchi-Levi et al. (2007) establish a clear link between location models and strategic SCM. Altiparmak et al. (2006) pointed out that this problem is one of "the most comprehensive strategic decision problems that need to be optimized for long-term efficient operation of the whole supply chain". Notice that some small changes to classical facility location models are quite hard to solve (Farahani and Hekmatfar, 2009). For example, in Lai-Jun et al. (2009) a genetic algorithm was used to solve a kind of facility location problem on test networks with 10 potential facility sites and 30 demand points. In this paper we focus on the sustainable supply chain design problem that arises in governmental agencies, where you have to decide the location of schools, hospitals, police stations, fire stations, and so on, taking into account sustainable issues.

3. Problem Formulation

The sustainable supply chain network design problem consists in deciding the number and location of facilities, and the allocation of customers to these facilities, minimizing the installation and transportation costs integrated with GHG emissions constraints. We consider a public supply chain that provides products/services and it consists of two layers hierarchically related. In the general case, these supply chain consists of two layers (mid and high) of distinct types of facility. For example, health care systems may consist of local clinics and hospitals or medical centers; higher education systems may consist of technical schools and universities. For further details on related problems see the paper by Bastani and Narges Kazemzadeh (2009). Our problem is uncapacitated by nature, following most of the research on locating public facilities, i.e., we do not restrict the capacity of the facilities to service the demand. Our interest is to analyze how service costs of governmental agencies located in both layers will be affected by sustainable restrictions. We assume that GHG emissions come mainly from the operation of the facilities located in both layers and from the transportation activities involved to service a fixed demand. Notice that, the clients can be attended by only one facility of the mid layer and each mid layer facility must be allocated to one high layer facility. We suppose that GHG emissions are proportional to the demand, i.e. population to be attended, and the travel distance. Then each facility has a carbon footprint

proportional to the demand attended. We model this using a parameter to be adjusted from the IPCC recommendations.

We introduce the following inputs and sets:

J = the set of demand nodes indexed by j

I = the set of candidate facility locations at the mid layer, indexed by i

K = the set of candidate facility locations at the high layer, indexed by k

h_j = demand at customer location $j \in J$

f_i = fixed cost of locating a mid layer facility at candidate site $i \in I$

g_k = fixed cost of locating a high layer facility at candidate site k $\in K$

c_{ij} = is the unit cost of supplying demand $j \in J$ from a mid layer facility located in $i \in I$

l_{ik} = is the unit cost of supplying demand $i \in I$ from a high layer facility located in $k \in K$

M = cardinality of J

α_i = GHG emissions factor of a facility located at candidate site $i \in IUK$, in tons of CO_2e per unit demand

β_{ij} = GHG emissions factor per unit distance and per unit demand between candidate facility site $i \in I$ and customer location $j \in J$, in tons of CO_2e per km and unit demand. We also use this factor for facilities located at a higher layer ($k \in K$) that are serving mid layer facilities ($i \in I$).

We consider the following decision variables:

$$y_i = \begin{cases} 1 \text{ if we locate a facility at candidate site } i \in I \\ 0 \text{ otherwise} \end{cases}$$

$$z_k = \begin{cases} 1 \text{ if we locate a high layer facility at candidate site } k \in K \\ 0 \text{ otherwise} \end{cases}$$

$$x_{ij} = \begin{cases} 1 \text{ if the demand of } j \in J \text{ is serviced by a facility located at } i \in I \\ 0 \text{ otherwise} \end{cases}$$

$$w_{ik} = \begin{cases} 1 \text{ if the demand of } i \in I \text{ is serviced by a facility located at } k \in K \\ 0 \text{ otherwise} \end{cases}$$

The general supply chain design problem with sustainable constraints ((*GUSSCP*) is defined by:

$$v(GUSSCP) =$$

$$Minimize \sum_{i \in I} f_i y_i + \sum_{i \in I} \sum_{j \in J} h_j c_{ij} x_{ij} + \sum_{k \in K} g_k z_k + \sum_{k \in K} \sum_{i \in I} h_i l_{ik} w_{ik} \quad (1)$$

$Subject\ to$:

$$\sum_{i \in I} x_{ij} = 1 \qquad \forall j \in J \quad (2)$$

$$\sum_{j \in J} x_{ij} \leq M y_i \qquad \forall i \in I \quad (3)$$

$$\sum_{k \in K} w_{ik} \geq y_i \qquad \forall\, i \in I \qquad (4)$$

$$\sum_{i \in I} w_{ik} \leq z_k \qquad \forall\, k \in K \qquad (5)$$

$$\sum_{i \in I}\sum_{j \in J} \alpha_i\, h_j x_{ij} + \sum_{k \in K}\sum_{i \in I} \alpha_k\, h_i w_{ik} + \sum_{i \in I}\sum_{j \in J} \beta_{ij} h_j c_{ij} x_{ij} + \sum_{k \in K}\sum_{i \in I} \beta_{ik} h_i l_{ik} w_{ik} \leq GHG \qquad (6)$$

$$x_{ij}, w_{ik}, y_i, z_k \in \{0,1\} \qquad \forall\, i \in I, \forall\, j \in J, \forall\, k \in K \quad (7)$$

The objective function (1) minimizes the sum of the installation facility costs and the demand-weighted supplying costs. Constraints (2) warranty that all demand at area $j \in J$ is met by one mid layer facility. Constraints (3) warranty that a demand node $j \in J$ must be allocated to a mid layer facility $i \in I$ already opened. Constraints (4) ensure that each open mid layer facility must be allocated to one high layer facility. Constraints (5) warranty that the demand of a mid layer facility must be serviced by a high layer facility. Constraint (6) limits the total Greenhouse Gas (CO_2e) emissions to *GHG*. Constraints (7) are standard binary constraints. Regarding constraints (3), when they are replaced by constraints

$$x_{ij} \leq y_i \qquad \forall\, i \in I, \forall\, j \in J \quad (3a)$$

we got a stronger formulation for the problem, as it was also discussed in a related facility location problem (Cornuejols et al., 1977). This problem is NP-hard as generalization of well-known location problems, and therefore cannot be solved in polynomial time.

4. Genetic Algorithm Implementation

Genetic Algorithms (GAs) are a type of evolutionary algorithms (EVA) used to solve a number of combinatorial optimization problems. See for further details the papers by Goldberg (1989). According to Osman and Kelly (1996), an EVA is composed by five basic components: (a) a genetic representation of solutions to a problem; (b) a way to create an initial population of solutions; (c) an evaluation function; (d) genetic operators that alter the genetic composition of children during reproduction and (e) values for the parameters. In this section we briefly describe these components and the GA implementation for solving the *GUSSCP* problem. The basic scheme of our GA can be represented as follow:

1. Generate an initial population;
2. Compute the fitness of each individual in the population;
3. while not Finish do
    a. Compute the Fitness for all the individuals in the population;
    b. Apply selection process;
    c. Do a crossover of individuals;
    d. Generate individuals using the mutation operation;
    e. Apply the feasibility procedure
4. Endwhile;

4.1. Representation, Initial Population and Feasible Solutions

In our implementation, each solution (individual) to the problem is coded as a chromosome such that each gene corresponds to a facility location decision variable at both layers, taking value 1 if a facility is open either at the mid or high layer, and zero otherwise. We use a $(|I| + |K|)$ dimensional string to represent facilities located at both layers.

The initial population of 100 individuals is generated randomly. Then we recombine this initial population and generate randomly two sets of 100 individual each. Each gene of the chromosome is generated by a 0-1 uniform probability distribution. For each chromosome we apply a solution procedure. This procedure consists in getting a solution to *GUSSCP* disregarding sustainable constraints as follows: Given a chromosome, for every customer $j \epsilon J$ we assigned its nearest opened mid layer facility. Then for each opened mid layer facility we assigned its nearest opened high layer facility. Regarding the GHG emissions, that mechanism of generating chromosome could generate unfeasible solutions for *GUSSCP*, but our strategy was to explore the behavior of the algorithm based on an initial population composed of a number of unfeasible solutions. For our test problems, the GA implemented in this way rapidly generated a large number of unfeasible solution and we obtained poorer solutions than the next approach we will discuss it. In the second approach, after the crossover and mutation operations, we introduce a greedy-random procedure to generate to every iteration of the GA at least 50% of feasible solutions. Then our new population in every iteration has at least 50% of feasible solutions. The procedure is as follow: we generate a random individual consisting of a number of facilities opened/closed. Then based on the transportation costs we apply the solution procedure, i.e., we allocate the nearest customer to each opened mid layer facility and then for each mid layer facility we find the nearest opened high layer facility. We repeat the procedure till we get 50% of feasible solutions for the new population. We replace unfeasible solutions for the new individuals obtained through this procedure. Finally the best 100 individuals will be part of the new initial population to start the main iteration of GA. As we can see later in this paper, computational results are very good.

The fitness of a chromosome is calculated using the objective function (1). To compute the first term (installation costs) of (1) is straight forward from the chromosome. To compute the second term (transportation costs) of (1), we use the solution procedure described above: for each customer we find its nearest opened facility (minimal transportation cost) and the we do the same for each opened mid layer facility. Then we sum up both parts (installation and transportation costs) to get the objective function value for each individual of the population.

4.2. Genetic operators

We use the standard genetic operators. The crossover generates two new individual (chromosome) exchanging the genetic material of two (parental) individuals expecting that "good" solutions can generate "better" ones. We selected these individuals randomly from a two sets of individuals, each set composed of 100 individuals as described earlier. We do not limit the number of new chromosomes generated by crossover. In this work crossover probability (*cross_p*) is set to 0.7 (70%) and we perform one-point crossover. In the crossover procedure we generate a random value, if the *cross_p* value is greater than the random value then we pick one individual from

each set. We generate another random value between one and the number of potential facility sites for each layer, i.e, a cut point dividing each individual (parent) into two segments. The first child is created by combining the first segment from the first parent and the second segment from the second parent. The second child is created from the first segment of the second parent and the second segment of the first parent. The mutation operator changes the value of a chromosome with some small probability. In our case, we get this probability to 0.1 (10%) and remain constant through generations of the GA. The gene in the chromosome is selected randomly and we switch its value (0-1). We do not limit the number of new chromosomes generated by mutation. After the crossover and mutation procedures, we got a number of unfeasible solutions in the population. In the next step we correct this;

4.3. Additional GA aspects

The selection operator is based on elitist selection, favoring individuals of better fitness value to reproduce more often than the worse ones when generating the new population. In every iteration the whole population (200 individuals) is ranked in a non-decreasing order of the objective function value. As we described earlier, feasibility of constraints (6) is verified. In case there is lesser than 50% of feasible solutions in the population, a greedy-random procedure was implemented to generate new (feasible) individuals. The best (100) solutions passed to the next iteration

In our case the total size of population is 200 individuals, and 100 of new individuals are generated each iteration. We set the total number of iteration to six.

5. Computational Results

The GA solution method for this problem was coded and implemented by Scilab software. According the sustainable supply chain literature discussed in previous sections, for testing our GA implementation we generated 11 size instances of *GUSSCP*. These instances correspond to test networks up to 26 potential sites and up to 50 demand nodes taken from the ORLIB (Beasley, 1996). For every instance we use one layer. As we do not know in advance how well is going to perform the GA, in order to validate our GA solutions we used GAMS on integer linear programming model described in section 3. Every test problem was running 5 times and we present an average value in Table 1. The optimal objective values were obtained by GAMS. As we can see in Table 1, besides the few number of iteration (6) used in our GA, the GAP obtained is quite small. Both methods (GA and GAMS) quickly converge on mentioned *GUSCPS* instances and their running times are not reported. The alpha (α) and beta (β) parameters were set to one and two respectively. This was done to analyze the behavior of the algorithm and also to check how the solution change when you penalty the transportation GHG emissions. Total GHG emissions were limited to values between 3,200.00 and 10,000.00 thousands.

We notice that, when you reduce the total amount of GHG emissions permitted, and the number of facilities remain free, the number of facilities to open increase, also increasing the cost of the solution but reducing the amount of GHG emitted by the transportation component.

6. Conclusions

In this paper, we introduced a novel kind of sustainable supply chain network design problem with a GHG emission constraint. The problem addressed the design of supply network arising mainly in the public sector, where we need to satisfy the demand for services like education and health care locating a number of facilities in two layers. We limit the GHG emissions generated by the facilities located in the mid and high layer and also the transportation involved in servicing the customers. The problem was formulated as a mixed integer 0-1 linear programming problem (MIP) and solved using a genetic algorithm coded in Scilab. We conducted an experimental study on instances taken from the ORLIB. In order to validate our GA solutions we used GAMS to obtain optimal objective values on the MIP. The genetic algorithm performs very good considering we set a few number of iterations. We observed that when you reduce the total amount of GHG emissions permitted, and the number of facilities remain free, the number of facilities to open increase, also increasing the cost of the solution but reducing the amount of GHG emitted by the transportation component.

Table 1. Computational Results. First column indicates number of problem instance; Fixed Costs in thousands; z* is the optimal solution; z(GA) is the solution value provided by GA.

# Probl	Fixed Costs	Total GHG	z*	z(GA)	GAP (%)
1	25,000	10,000	1,746,347	1,775,425	1.7
2	17,500	10,000	1,727,848	1,731,842	0.2
3	12,500	10,000	1,700,236	1,700,841	<0.1
4	25,000	5,000	1,746,348	1,775,425	1.7
5	25,000	3,200	1,953,224	1,953,224	0.0
6	17,500	3,200	1,840,724	1,840,724	0.0
7	12,500	3,200	1,765,724	1,765,724	0.0
8	7,500	3,200	1,690,724	1,690,724	0.0
9	7,500	3,300	1,663,018	1,684,578	1.3
10	12,500	3,300	1,700,236	1,765,723	3.9
11	17,500	3,300	1,730,236	1,773,011	2.5

Acknowledgments

This research was partially supported by UFOP, FAPEMIG and CNPq-Brazil.

References

Altiparmak, F., Gen, M., Lin, L., and Paksoy, T. (2006) “A genetic algorithm approach for multiobjective optimization of supply chain networks”, Computers & Industrial Engineering, 51, p. 197–216.

Bastani, S., and Kazemzadeh, N. (2009) “Hierarchical Location Problem”, In: Facility Location: Concepts, Models, Algorithms and Case Studies. Contributions to Management Science, 2009, Edited by R.Z. Farahani and M. Hekmatfar, p. 219-241,Springer

Beasley, J.E. (1996) “Obtaining test problems via Internet”, Journal of Global Optimization, 8, p. 429-433.

Choi T. Y. and, Wu, Z. (2009) “Taking the leap from dyads to triads: Buyer–supplier relationships in supply networks”, Journal of Purchasing and Supply Management, 15, 4, p. 263-266.

Chaabane, A., Ramudhin, A., and Paquet, M. (2010) “Optimization and evaluation of sustainable supply chains”, In: Proceedings of the 8th. International Conference of Modelling and Simulation – MOSIM 2010, Tunisia.

Cornuejols, G., Fisher, M.L., and Nemhauser, G.L. (1977) “Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms”, Management Science, 23, p. 789–810.

Daskin, M. S., Snyder, L. V., and Berger, R. T. (2005) “Facility Location in Supply Chain Design”, In: Logistics Systems: Design and Optimization , Edited By A. Langevin and D. Riopel, Kluwer, p. 39-65.

Diabat, A., and Simchi-Levi, D. (2010) “A carbon-capped supply chain network problem”, In: Proceedings of IEEE International Conference on Industrial Engineering and Engineering Management 2009. IEEE, Piscataway, N.J., p. 523-527.

Farahani, R.Z., and Hekmatfar, M. Eds., (2009), Facility location: concepts, models, algorithms and case studies. Berlin, Germany, Springer-Verlag, 2009.

Goldberg, D. (1989), Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading.

Hugo, A and Pistikopoulos, E. (2004) “Environmentally conscious process planning under uncertainty”, In: Sixth international conference on foundations of computer-aided process design. Edited by CA. Floudas, R. Agrawal, Princeton, CACHE Corporation,.

Hugo, A and Pistikopoulos, E. (2005) “Environmentally conscious long-range planning and design of supply chain networks”, Journal of Cleaner Production: Recent advances in industrial process optimization, 13, 15, p. 1471-1491.

Lai-Jun, W., Xiao-Ling, S., and Zhongke. S. (2009) “New evolutionary Algorithm Applying to a Type of Facility Location Problem”, Information Technology Journal, 8, 4, p. 605-609.

Osman, I. H. and Kelly, J. P. (1996), MetaHeuristics: Theory & Applications, Kluwer Academic Publishers.

Ramudhin, A., Chaabane, A., Kharoune, M. and Paquet, M. (2008) “Carbon market sensitive green supply chain network design”, In: Proceedings IEEE International Conference on IEEE, Singapore, p. 1093-1097.

Seuring, S. and Muller, M. (2008) "From a literature review to a conceptual framework for sustainable supply chain management", Journal of Cleaner Production, 16, p. 1699-1710.

Simchi-Levi, D., Kaminsky, P., and Simchi-Levi, E. (2007), Designing & Managing the Supply Chain, 3rd ed. Boston, MA, USA: McGraw-Hill/Irwin.

Srivastava, S. (2007) “Green Supply-Chain Management: A State-of-the-Art Literature Review”, International Journal of Management Reviews, 9, 1, p. 53-80.

Análise e mineração de dados de sensores orbitais para acompanhamento de safras de cana-de-açúcar

Bruno F. Amaral[1], Daniel Y. Chino[1], Luciana A. S. Romani[2], Renata R. V. Gonçalves[3], Elaine P. M. de Sousa[1] , Agma J. M. Traina[1]

[1]Instituto de Ciências Matemáticas e de Computação – USP – São Carlos - Brasil

[2]Embrapa Informática Agropecuária – Campinas – SP – Brasil

[3]Cepagri – Universidade Estadual de Campinas – SP – Brasil

{brunoslash, chinodyt}@grad.icmc.usp.br, luciana@cnptia.embrapa.br, renata@cpa.unicamp.br, {parros, agma}@icmc.usp.br

Abstract. *Researches aiming greenhouse gases reduction have been motivated by the impact of extreme climate events around the world. In Brazil, sugar cane is the main source for ethanol production to replace fossil fuels. In this context, remote sensing imagery has been widely used to monitor sugar cane harvests and to support scientific research. In this paper, we propose a methodology based on data clustering to analyze NDVI time series obtained from AVHRR/NOAA satellites and monitor the growing cycles of sugar cane crops. The experiments show that our approach can identify areas with similar development patterns also considering different growing crops seasons.*

Resumo. *O impacto causado por eventos climáticos extremos em todo o mundo tem motivado pesquisas para redução de gases de efeito estufa. No Brasil, a cana-de-açúcar é a principal fonte para produção de etanol, como alternativa a combustíveis fósseis. Nesse contexto, dados de sensoriamento remoto têm sido utilizados para monitorar safras de cana-de-açúcar e apoiar pesquisas científicas. Neste trabalho, é proposta uma metodologia baseada em agrupamento de dados para analisar séries temporais de NDVI obtidas de satélites AVHRR/NOAA. Os experimentos mostram que a abordagem proposta permite identificar áreas com padrões de desenvolvimento similares, considerando também os diferentes ciclos de vida da cultura.*

1. Introdução

O aquecimento global e as mudanças climáticas são grandes desafios para os pesquisadores de diversas áreas do conhecimento. Apesar da certeza quanto à elevação da temperatura média do planeta, muitas pesquisas sobre os impactos no ecossistema terrestre e na agricultura vêm sendo desenvolvidas. No Brasil, uma das principais fontes de recursos vem da produção agrícola, como o biocombustível a partir da cana-de-açúcar, usado como fonte de energia renovável para substituir combustíveis fósseis. Desse modo, as pesquisas e estudos sobre o desenvolvimento das culturas agrícolas ao longo dos anos, de acordo com as mudanças no clima, são muito importantes. Os três cenários de mudanças climáticas (baixa, igual e alta emissão de carbono em relação aos dias atuais), analisados em função da emissão de gases de efeito estufa (GEE), têm mostrado impactos significativos na saúde, na agricultura, nos recursos hídricos, na biodiversidade e, particularmente, na incidência de eventos climáticos extremos. É

prevista a elevação nos níveis de evaporação e intensificação do ciclo hidrológico, decorrente da maior quantidade de energia disponível gerada pelo aquecimento, contribuindo para maior ocorrência de eventos extremos de precipitação, com chuvas mais intensas em determinadas áreas. O contrário também é observado, pois em algumas regiões as estiagens podem se tornar mais severas e prolongadas [IPCC 2007].

Nesse contexto, embora a cana-de-açúcar possa se beneficiar do aumento das temperaturas, um aumento do déficit hídrico em determinadas fases do desenvolvimento da cultura pode impactar sua expansão para áreas mais quentes. Portanto, estudos mais aprofundados e detalhados sobre a expansão e produtividade da cana-de-açúcar, bem como o desenvolvimento de novos cultivares, tornam-se fundamentais. Como a cana-de-açúcar é plantada em grandes extensões, é possível usar imagens de sensores remotos para auxiliar no monitoramento das plantações dessa cultura. Neste trabalho, imagens do satélite AVHRR/NOAA (*Advanced Very High Resolution Radiometer / National Oceanic and Atmospheric Administration*) são usadas para obtenção de índices, como o índice de vegetação extraído para cada pixel da imagem, gerando um grande volume de dados. Nesse cenário, o suporte computacional adequado para análise e extração de conhecimento desses dados se torna uma ferramenta muito útil.

Uma técnica importante para a análise de séries temporais de índices de vegetação visando descoberta e estudo de padrões no desenvolvimento de culturas agrícolas é a detecção de agrupamento (*clustering*) [Romani et al. 2011]. Neste trabalho, algoritmos de agrupamento são avaliados para auxiliar na visualização da expansão da cana-de-açúcar no estado de São Paulo, ao longo dos anos, de modo regional. Assim, este artigo apresenta a integração de tarefas desde extração de séries temporais a partir de índices das imagens de satélite até a identificação de agrupamento e apresentação espacial dos resultados em uma única ferramenta computacional.

2. Metodologia

Este trabalho propõe uma abordagem para analisar o desenvolvimento de culturas de cana-de-açúcar ao longo de safras por meio da utilização de técnicas de *clustering* aplicadas a séries temporais extraídas de imagens de satélite. O conjunto de dados é obtido a partir de sensores de satélites de baixa resolução, utilizando a ferramenta SatImagExplorer [Chino et al. 2010] para extração de índices vegetativos a partir de imagens de satélite. Na mesma ferramenta, foram implementados os métodos de *clustering* utilizados neste trabalho e aplicados para análise das séries temporais. As etapas principais do processo de análise dos dados são ilustradas na Figura 1.

Na primeira etapa, as imagens são obtidas por meio do sensor AVHRR do satélite NOAA. Este sensor tem sido muito usado em pesquisas relativas a ecossistemas e agricultura devido a um grande volume de dados disponíveis e por ter cobertura global, além da possibilidade de acesso gratuito aos dados. Na segunda etapa é realizada a combinação de canais desse sensor, o que fornece uma indicação da quantidade e estado da vegetação, como o NDVI (*Normalized Difference Vegetation Index*), um dos índices vegetativos utilizados no monitoramento de culturas por imagens de satélite. Um exemplo de cultura que pode ser monitorada por imagens de baixa resolução, como as do AVHRR/NOAA, é a cana-de-açúcar, por ocupar áreas vastas e bem próximas.

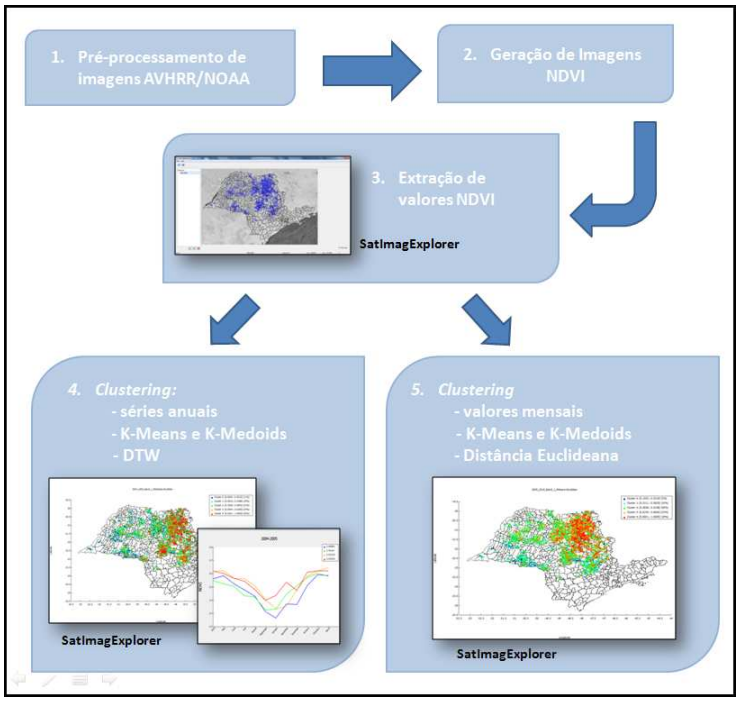


Figura 1. Metodologia utilizada neste trabalho.

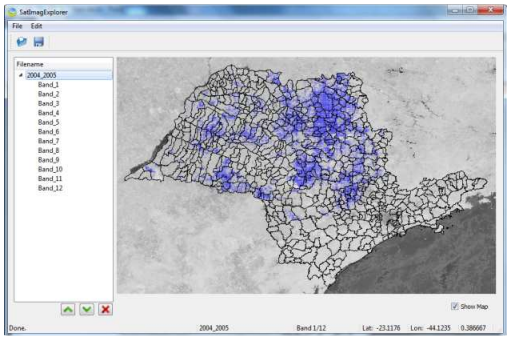


Figura 2. SatImagExplorer com imagem de São Paulo.

Na SatImagExplorer foram implementadas e incorporadas técnicas de *clustering*, incluindo suporte à descoberta de padrões nos dados extraídos de imagens NDVI. Assim, na etapa 3 a ferramenta recebe imagens de satélite como entrada e calcula índices e medidas de sequências, de modo automático, gerando séries temporais relativas a esses índices, de acordo com a área de interesse definida pelo especialista. Na Figura 2, é exibida a visualização dos valores de NDVI na SatImagExplorer. Nas etapas seguintes, o especialista pode aplicar os métodos de *clustering* que farão o agrupamento dos pontos da região do mapa, de acordo com os valores NDVI. Os métodos implementados são o *K-Means* e o *K-Medoids*, ambos descritos em [Han et al. 2001].

O *K-Means* é um algoritmo de *clustering* baseado em particionamento, em que cada objeto do conjunto de dados a ser agrupado pode pertencer a apenas um grupo (*cluster*). Inicialmente são escolhidos aleatoriamente os elementos representantes (*centroids*) dos *clusters*, e cada elemento da base é associado ao *cluster* cujo representante é o mais próximo. Utilizando o valor médio dos elementos de cada grupo, o algoritmo recalcula iterativamente os *centroids*, e redistribui todos os objetos, até que não haja mais alterações no agrupamento. Por ser baseado em valores médios, a presença de *outliers* e ruídos na base de dados pode influenciar na escolha dos *centroids*

e na formação dos *clusters*. O *K-Means* foi implementado na SatImagExplorer por ser um algoritmo rápido, simples e que demonstra bons resultados, além de ser muito conhecido e usado em diversas aplicações na área de mineração de dados.

O *K-Medoids* é um método de *clustering* semelhante ao *K-Means*, baseado em particionamento. Ele difere do anterior quanto à escolha do representante do *cluster*. No *K-Medoids*, ao invés de representar o grupo pela média de seus elementos, o algoritmo busca, como representante, o elemento mais central do *cluster*, chamado *medoid*. Dessa maneira, para cada *cluster*, todos os objetos pertencentes a ele são testados como candidatos a *medoid*, e apenas o que for o mais centralizado é escolhido. Tal característica torna o método menos sensível a ruídos e *outliers* no conjunto de dados de entrada, além de minimizar o efeito da escolha aleatória dos *medoids* iniciais. Entretanto, isso eleva significativamente seu tempo de execução.

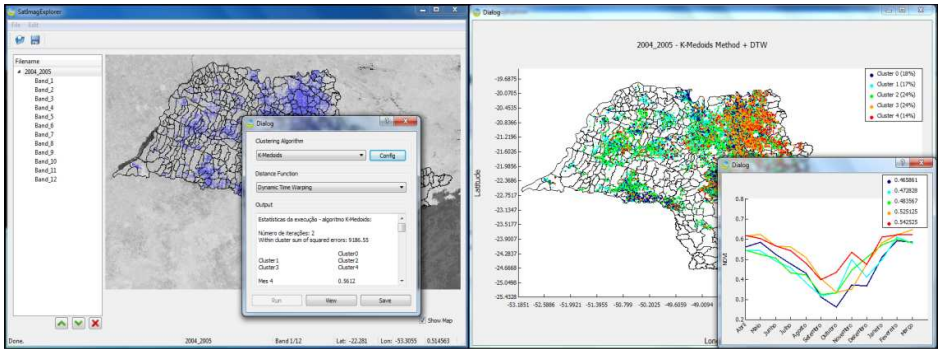


Figura 3. Aplicação dos métodos de *clustering* na SatImagExplorer.

Os métodos de *clustering* implementados utilizam métricas para calcular as distâncias entre os elementos do conjunto de dados. Neste trabalho, foram aplicadas duas abordagens na análise dos dados. Em uma delas, cada elemento do conjunto de dados é definido por apenas um valor de NDVI, referente ao índice de um mês de uma safra em uma determinada localização (pixel da imagem), visando análises mensais da região de interesse. Nesse caso, os métodos de *clustering* utilizam a distância Euclideana. A outra abordagem visa trabalhar com as séries de valores de NDVI, de modo que cada elemento do conjunto é definido por uma série de NDVI correspondente a uma ou várias safras de cana-de-açúcar. Nessa abordagem, foi utilizada a métrica DTW (*Dynamic Time Warping*) com ambas as técnicas de *clustering*, por ser uma função de distância muito aplicada na análise de séries temporais [Berndt and Clifford 1994]. A interface para o processo de *clustering* na ferramenta é ilustrada na Figura 3.

Em vários estudos [Ding et al. 2008], métricas simples como a distância Euclideana são suficientes para cálculo de similaridade em diversos domínios de dados. Entretanto, séries temporais podem possuir formatos ou perfis semelhantes, porém estar desalinhadas em relação à escala temporal. A DTW, por outro lado, realiza alinhamentos entre duas séries, reconhecendo a similaridade mesmo com deslocamentos de tempo. Na Seção 3, são descritos experimentos usando as duas abordagens.

3. Experimentos e Discussão

A metodologia descrita neste trabalho foi aplicada na análise de séries temporais de imagens de NDVI no período de 2001 a 2010 para o estado de São Paulo. Para selecionar os potenciais pixels de cana-de-açúcar foi utilizada uma máscara gerada pelo

projeto CANASAT/INPE[1] para a safra 2004/2005 como referência. Dois experimentos foram realizados:

1. Agrupamento de pixels mês a mês para cada ano-safra utilizando os algoritmos *K-Means* e *K-Medoids* com a função de distância Euclideana;
2. Agrupamento de pixels por década utilizando o algoritmo *K-Medoids* com a função de distância DTW.

Em ambos os experimentos os pixels foram separados em 5 *clusters*. No experimento 1, os pixels correspondem a um único valor mensal de NDVI e por isso são agrupados utilizando a distância Euclideana. No experimento 2, são utilizados todos os pixels da série temporal, isto é, 12 valores mensais para cada ano-safra, perfazendo um total de 120 valores de NDVI na série. Nesse experimento foi utilizada a função de distância DTW, que é mais utilizada para detecção de similaridade em séries temporais.

A fim de gerar um *baseline* para a comparação dos resultados gerados pelo modelo proposto de análise de agrupamento foram definidas, pelos especialistas, classes baseadas nos valores de NDVI considerando a safra 2004/2005, a saber:

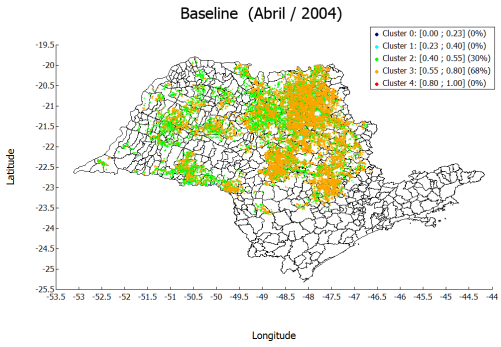
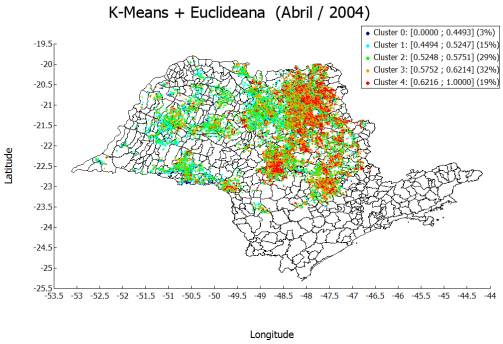
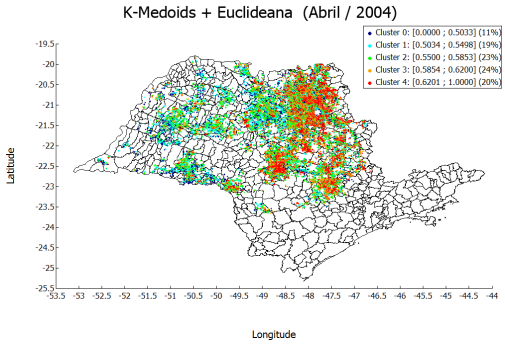
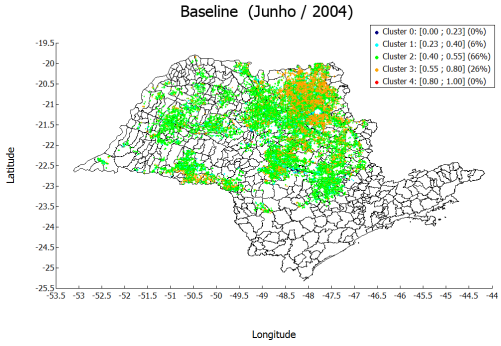
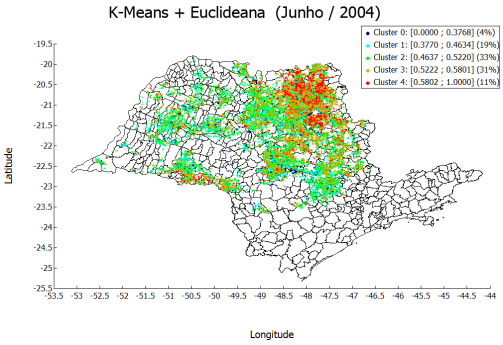
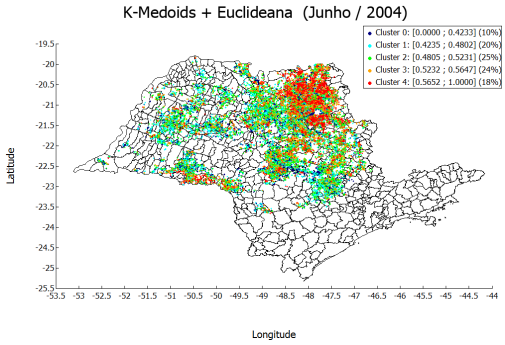
- *Cluster* 0 (azul escuro) = 0,0 a 0,23
- *Cluster* 1 (azul claro) = 0,23 a 0,40
- *Cluster* 2 (verde) = 0,40 a 0,55
- *Cluster* 3 (laranja) = 0,55 a 0,80
- *Cluster* 4 (vermelho) = 0,80 a 1,0

Foram geradas imagens mensais de acordo com esse *baseline*, como mostrado nas Figuras 4A, 4D, 4G, 4J e 4N. No primeiro experimento, ambos os métodos de agrupamento produziram resultados semelhantes, onde o *cluster* 4 (vermelho) indica os valores máximos de NDVI no mês correspondendo a áreas com maior biomassa e, consequentemente, maior produtividade. O *cluster* 0 (azul escuro) mostra os valores de NDVI mais baixos, correspondendo, provavelmente, a solo exposto.

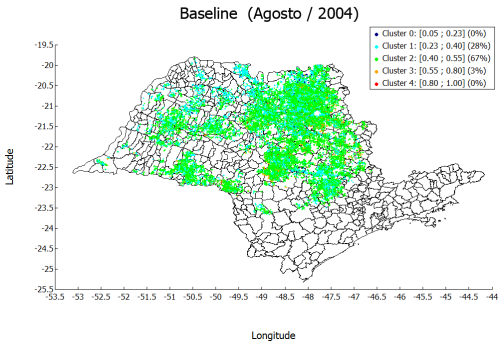
O mês de Abril corresponde ao ponto máximo do crescimento vegetativo da cana-de-açúcar. Na Figura 4A (*baseline*), os pixels aparecem em laranja com o valor máximo de NDVI, correspondendo às áreas em vermelho e laranja (acima de 0,58) geradas pelo *K-Means* e *K-Medoids*, nas Figuras 4B e 4C respectivamente. Logo os dois métodos apresentaram resultados similares à classificação proposta pelos especialistas.

Por outro lado, o mês de Outubro corresponde à época de colheita. Neste mês, os pixels em azul claro e escuro (com o valor mínimo de NDVI) na imagem *baseline* (Figura 4J) correspondem às áreas em verde, azul claro e escuro (abaixo de 0,37) geradas pelo *K-Means* e *K-Medoids*, nas Figuras 4L e 4M. Novamente, ambos os algoritmos mostraram-se similares aos resultados indicados pela classificação prévia. Análises similares podem ser realizadas para os demais meses ilustrados na Figura 4. Assim, aplicando os métodos de *clustering* na análise da safra de cana-de-açúcar (1 ano de abril a março), é possível acompanhar o ciclo da cultura ao longo dos meses, automaticamente. Com isso, demonstra-se que embora as técnicas usadas sejam bem conhecidas, a sua aplicação em séries geradas a partir de imagens de baixa resolução permite estudo da cultura em nível regional. Além disso, como não existe a predominância em todas as regiões de um ou dois agrupamentos ao longo dos meses, é possível observar que canas de ano e ano e meio são cultivadas em todo o estado.

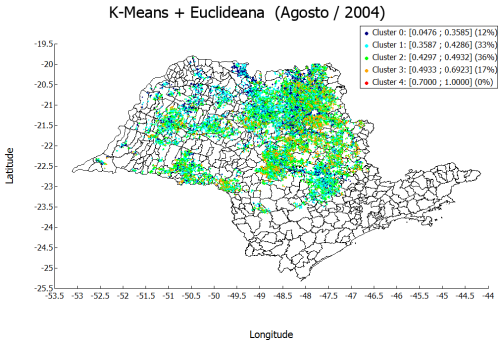
[1] www.dsr.inpe.br/laf/canasat

	***K-Means* + Distância Euclideana**	***K-Medoids* + Distância Euclideana**
Abril (2004/05)	 ***Baseline* (A)**	
	 B	 **C**
Junho (2004/05)	 ***Baseline* (D)**	
	 **E**	 **F**

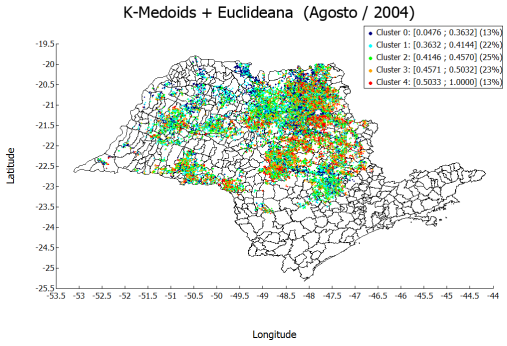
Agosto (2004/05)



***Baseline* (G)**

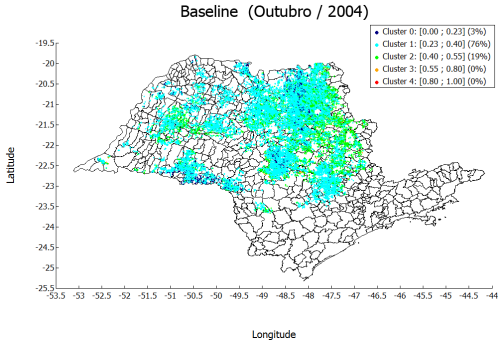


H

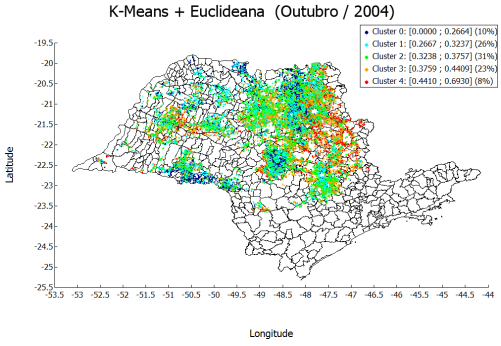


I

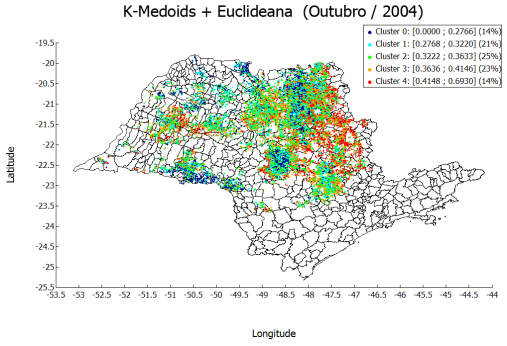
Outubro (2004/05)



***Baseline* (J)**



L



M

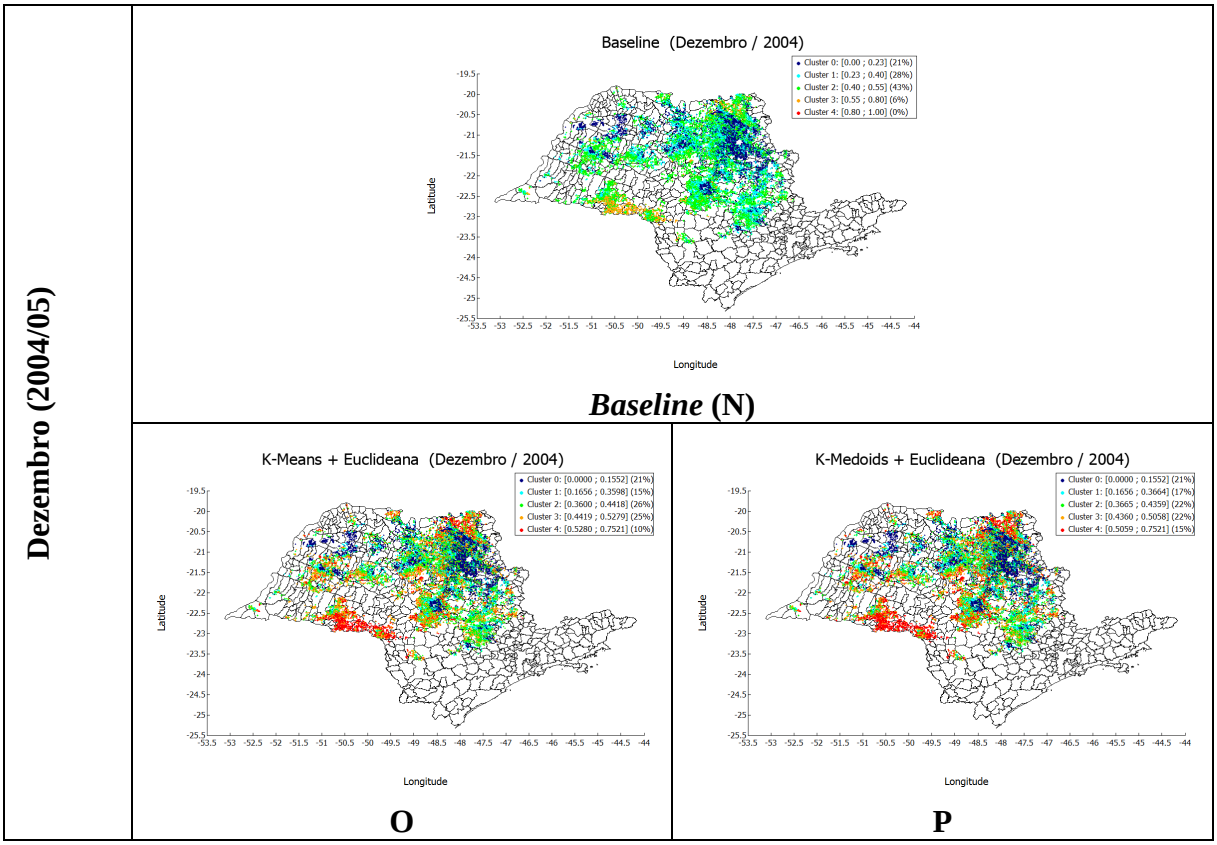


Figura 4. Mapas de São Paulo com 5 *clusters* (*baseline*, *K-Means* e *K-Medoids*).

No experimento 2, os resultados ilustrados na Figura 5 mostram que, embora tenha sido usada uma máscara para mapear a safra de cana-de-açúcar de 2004/2005, é possível diferenciar as regiões tradicionalmente produtoras (noroeste) das áreas de expansão (oeste) de cana-de-açúcar. A apresentação das diferentes áreas produtoras de cana-de-açúcar em um único mapa que representa a análise de uma década de imagens de NDVI permite que o especialista valide resultados relativos à produção de cana-de-açúcar. Além disso, tanto a técnica quanto a integração das tarefas em um único sistema tornam o trabalho do especialista mais rápido e eficiente, permitindo uma análise automática regional uma vez que são utilizadas imagens de alta resolução temporal. Na Figura 5A, os pixels do *cluster* 4 (vermelho), com 7%, apresentam um perfil diferente do padrão de NDVI da cana-de-açúcar, como pode ser visto na assinatura espectral gerada para o *medoid* do *cluster* 4 (vermelho), na Figura 5B. É possível que essa área corresponda a outro tipo de vegetação, como matas ou reflorestamento. Por outro lado, o *cluster* 3 (laranja), com 34%, representa a área de maior produção da cana-de-açúcar. O perfil de NDVI para esse *cluster*, ilustrado na Figura 5B, mostra uma tendência mais semelhante ao perfil padrão de NDVI, atingindo os maiores valores do índice.

Os *clusters* 1 e 2 (azul-claro e verde), com 22% e 26% respectivamente, indicam a área de expansão da cana-de-açúcar para o oeste do estado (Figura 5A). Os perfis de NDVI dos dois *clusters* não atingem valores muito expressivos, indicando que a produção pode não ser tão elevada quanto da região noroeste do estado. O *cluster* 0 (azul escuro), com 8%, aparece distribuído em todo o estado e de acordo com o seu perfil, corresponde a solo exposto ou mistura espectral.

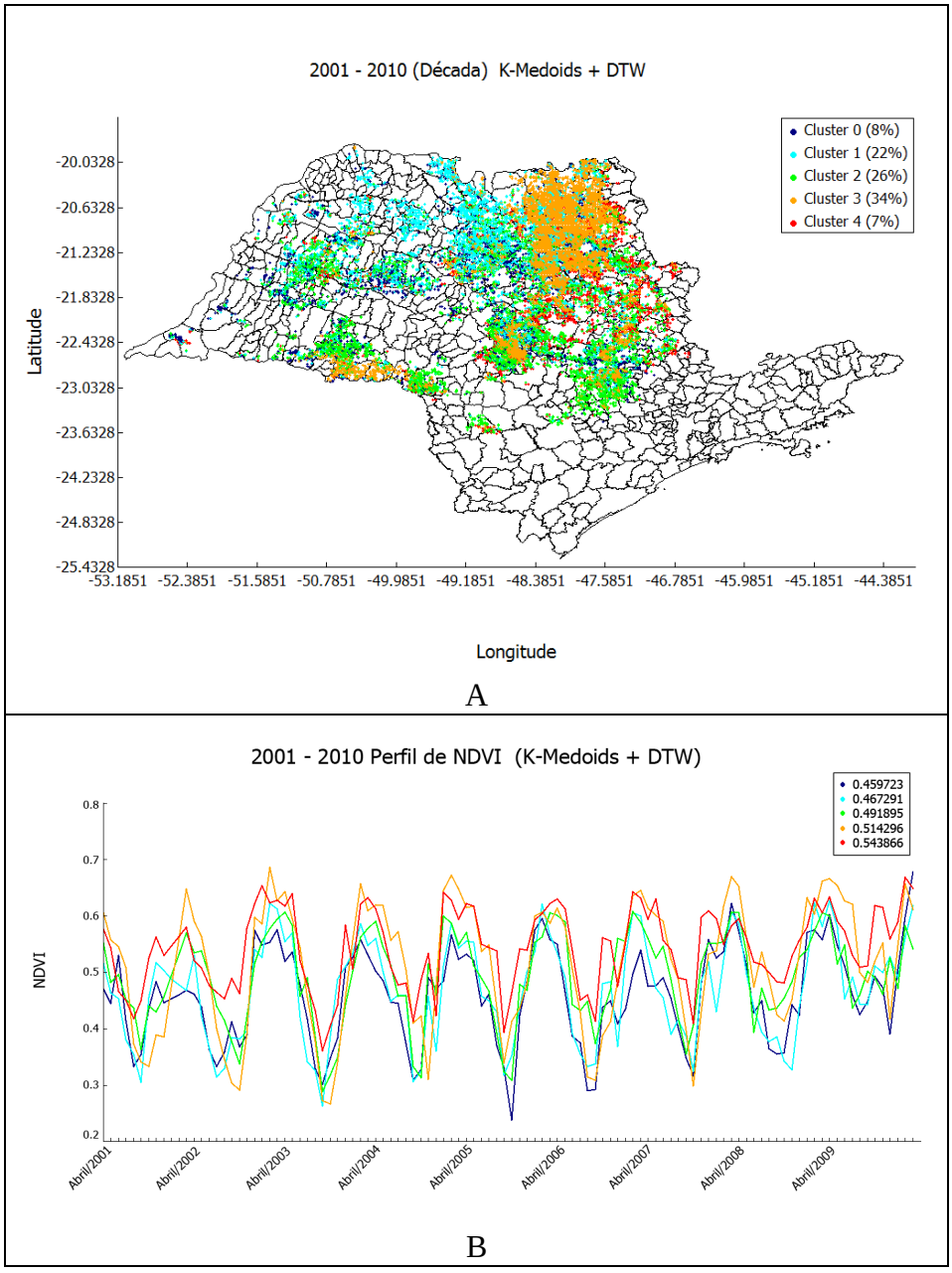


Figura 5. A. Mapa de São Paulo com 5 *clusters* (*K-Medoids* + DTW) no período de 2001 a 2010. B. Gráfico com a assinatura espectral (perfil de NDVI) dos 5 *clusters*.

A validação dos resultados foi efetuada de duas maneiras: pelo especialista e por meio da comparação com implementações disponíveis em outro software de mineração de dados, o Weka[2]. De acordo com os especialistas, os resultados são promissores, pois foi possível acompanhar a expansão da cana-de-açúcar. Os resultados da implementação no SatImagExplorer e no Weka foram similares. No entanto, o SatImagExplorer tem a vantagem de integrar todo o processo, desde a extração das séries de NDVI até a apresentação espacial do resultado da análise de agrupamento.

[2] http://www.cs.waikato.ac.nz/ml/weka/

4. Conclusões

Este trabalho propõe uma nova abordagem para avaliar a expansão da cana-de-açúcar, por meio de técnicas de mineração de dados e de séries temporais. Foram usadas técnicas de *clustering* tradicionais associadas a funções de distância para séries temporais. Todo o processo desde a extração dos pixels das imagens de satélite até a análise e visualização dos resultados foi desenvolvida em uma ferramenta (SatImagExplorer), tornando o processo mais fácil e ágil para o especialista.

Embora os métodos computacionais utilizados não sejam inovadores, a sua aplicação para apoiar o acompanhamento das safras de cana-de-açúcar, bem como averiguar sua expansão de maneira automática, é uma valiosa contribuição deste trabalho. Além disso, o potencial das técnicas utilizadas destaca-se ainda mais por terem sido aplicadas a imagens de satélite de baixa resolução espacial, o que dificulta a análise devido à possibilidade de mistura espectral. A realização das análises numa única ferramenta computacional como o SatImagExplorer, que incorpora funcionalidades desde a extração das séries a partir das imagens até a apresentação visual dos resultados, auxilia nas pesquisas sobre fontes renováveis de energia, como o etanol. O impacto deste tipo de trabalho torna-se ainda maior à medida que aumenta a necessidade por pesquisas para a redução de gases do efeito estufa, diante das recentes ocorrências de eventos extremos em diferentes localidades do planeta.

Agradecimentos

Os autores agradecem Santander, Fapesp-Microsoft Research, CNPq, Capes e Embrapa pelo apoio financeiro e Cepagri/Unicamp pelas imagens do AVHRR/NOAA.

Referências

Berndt, D. and Clifford, J. (1994) “Using dynamic time warping to find patterns in time series”. In: AAAI Workshop on Knowledge Discovery in Databases, p. 359-370, Seattle - Washington.

Chino, D. Y. T., Romani, L. A. S. e Traina, A. J. M. (2010) “Construindo séries temporais de imagens de satélite para sumarização de dados climáticos e monitoramento de safras agrícolas”. In: REIC, v. 10, p. 1-16.

Ding, H., Trajcevski, G., Scheuermann, P., Wang, X. and Keogh, E. (2008). “Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Measures”. In: VLDB Endowment, p. 1542-1552.

IPCC (2007) “Climate change 2007: Fourth assessment report (AR4)”. In: Intergovernmental Panel on Climate Change.

Han, J., Kamber, M. and Tung, A. K. H. (2001) “Spatial Clustering Methods in Data Mining: A Survey”. In: Geographic Data Mining and Knowledge Discovery, Edited by H. J. Miller and J. Han, Taylor and Francis, p. 201-230.

Romani, L. A. S., Gonçalves, R. R. V., Amaral, B. F., Zullo Jr, J., Traina Jr, C., Sousa, E. P. e Traina, A. J. M. (2011) “Acompanhamento de safras de cana-de-açúcar por meio de técnicas de agrupamento em séries temporais de NDVI”. In: XV Simpósio Brasileiro de Sensoriamento Remoto, Curitiba - PR, p. 383-390.

WEI - XIX Workshop sobre Educação em Computação

PREFÁCIO

O WEI (Workshop sobre Educação em Computação) é um evento anual promovido pela SBC (Sociedade Brasileira de Computação) e tem por objetivo debater diferentes temas relacionados ao ensino de computação e informática. O WEI constitui-se num importante fórum brasileiro para apresentações, tutoriais e debates entre docentes, discentes, coordenadores de cursos e demais interessados na melhoria do processo de ensino e de aprendizagem em Computação.

A 19a edição do Workshop sobre Educação em Computação traz mudanças significativas em relação às edições anteriores. Nesta edição, o WEI será realizado em quatro dias, a fim de aumentar o tempo dos grupos de trabalho e discussões. Nesses quatro dias de programação estão incluídas as atividades realizadas tradicionalmente no Curso de Qualidade (CQ). O objetivo principal dessa integração é fortalecer a participação no WEI dos coordenadores e demais participantes do CQ, provendo mais qualidade às discussões resultantes das atividades desenvolvidas durante o evento.

Em 2011, o tema do WEI é "Educação em Computação na era da Inclusão", o qual reflete a preocupação com o uso racional dos recursos computacionais em prol de uma formação de recursos humanos e na conscientização de como empregar os recursos computacionais para fortalecer a inclusão digital e o acesso à informação por parte de todas as camadas da sociedade. Tal tema, além de estar alinhado com o tema principal do XXXI Congresso da Sociedade Brasileira de Computação ("Computação para todos: no caminho da evolução social") recebe atenção especial por parte dos órgãos de fomento governamentais com editais de apoio a projetos de pesquisa.

Iniciativas recentes dos governantes em fornecer computadores de baixo custo a estudantes do ensino fundamental e médio, além de popularizar o acesso a Internet banda larga, também contribuem com a popularização do potencial dos computadores em atividades de ensino/aprendizagem. Discutir a formação de egressos em cursos de computação capazes de explorar esses recursos é o que se deseja nesta edição do WEI, identificando como contribuir significativamente tanto para a socialização do uso da informática, quanto no emprego de modo eficaz destes recursos em favor da cidadania.

Dentro desse contexto, foram submetidos 126 trabalhos para avaliação sendo 96 artigos completos submetidos, dos quais 20 foram aceitos (20,8% de aceitação) e 30 artigos resumidos, dos quais 6 foram selecionados (20% de aceitação). Esses 26 trabalhos estão distribuídos em seis sessões técnicas que irão ocorrer ao longo do evento e abrangem uma variedade de temas, incluindo inclusão digital, ensino de interação humano-computador, sistemas operacionais, algoritmos, estruturas de dados, dentre outros.

A coordenação do evento agradece a todos os envolvidos nesse processo e deseja um excelente workshop durante os seus quatro dias.

Auri Marcelo Rizzo Vincenzi, UFG
Marcelo Duduchi, CEETEPS
Umberto Souza da Costa, UFRN

COMISSÕES

Coordenação Geral

Auri Marcelo Rizzo Vincenzi, UFG (Coordenação Geral)
Marcelo Duduchi, CEETEPS (Coordenação Geral)
Umberto Souza da Costa, UFRN (Coordenação Local)

Colaboradores

Anna Helena Reali Costa, USP
Guilherme Horta Travassos, UFRJ
José Palazzo Moreira de Oliveira, UFRGS
Mirella M. Moro, UFMG
Noemi Rodriguez, PUC-Rio

Comitê de Programa

Alessandra Dahmer, USCS
Alexandre Cidral, UNIVILLE
Ana Paula Ambrósio, UFG
Anna Helena Reali Costa, USP
Auri Marcelo Rizzo Vincenzi, UFG (chair)
Carlos Eduardo Ferreira, USP
Carmem Hara, UFPR
Clovis Torres Fernandes, ITA
Daltro Nunes, UFRGS
Debora Paiva, UFMS
Dilermando Piva Jr., FATEC-ID
Edson Pimentel, UFABC
Eliana Almeida, UFAL
Elói Favero, UFPA
Esteban Clua, UFF
Fatima Nunes, EACH-USP
Guilherme Horta Travassos, UFRJ
Ismar Frango Silveira, Universidade Presbiteriana Mackenzie
José Maria David, UFJF
José Palazzo Moreira de Oliveira, UFRGS
Lúcia Giraffa, PUC-RS
Marcelo Duduchi, CEETEPS (co-chair)
Marcelo Augusto Santos Turine, UFMS
Marcos José Santana, ICMC-USP
Maria Augusta Vieira Nelson, PUC Minas
Marilia Macorin de Azevedo, CEETEPS
Mariza Bigonha, UFMG
Mirella M. Moro, UFMG
Morganna Diniz, UNIRIO
Nizam Omar, Universidade Presbiteriana Mackenzie

Noemi Rodriguez, PUC-Rio
Raquel Prates, UFMG
Regina Helena Carlucci Santana, ICMC-USP
Roberto Bigonha, UFMG
Ronaldo Celso Messias Correia, UNESP
Ronaldo Mello, UFSC
Rosana Braga, ICMC-USP
Sandra Fabbri, UFSCar
Simone Martins, UFF
Therezinha Souza da Costa, PUC-Rio

Revisores Externos

Abraham Lincoln Rabelo de Sousa, UFRGS
Adenilso Simao, ICMC-USP
Alex Machado, UFF
Alexandre Plastino, UFF
Anarosa Brandão, USP
André Brandão, UFF
André de Oliveira, UFSCar
Breno França, UFRJ
Claudio Toledo, USP
Denis Wolf, USP
Diego Brandão, UFF
Edson Prestes e Silva Jr, UFRGS
Eduardo do Valle Simões, USP/São Carlos
Erick Passos, UFF
Fabiano Ferrari, UFSCar
Fabiano Teixeira, USP
Fabio Costa, UFG
Gisele Pappa, UFMG
Graça Bressan, POLI-USP
Hugo do Nascimento, UFG
Isabel Cafezeiro, UFF
Jacques Brancher, UEL
Júlio Estrella, ICMC-USP
Jonh Carvalho, UFF
Leônidas Brandão, USP
Leliane Nunes de Barros, USP
Leonardo Murta, UFF
Lucia Filgueiras, POLI-USP
Lucineia Maia, UFMS
Luiz Chaimowicz, UFMG
Marcelo Ramos, USP
Marcia Jochims Kniphoff da Cruz, UNISC
Marco Aurélio Gerosa, IME/USP
Mark Joselli, UFF
Maycon Peixoto, USP

Milene Silveira, PUCRS
Neiva Coelho, UFMS
Olinto Furtado, UFSC
Paulo Andrade, UFF
Paulo Queiroz, USP
Pedro Losco Takecian, IME-USP
Plínio Leitão-Júnior, UFG
Regina Silveira, USP
Renata Araujo, UNIRIO
Renato Carmo, UFPR
Rene Pinto, USP
Ricardo Nakamura, POLI-USP
Ricardo Silveira, UFSC
Rodrigo Funabashi Jorge, UFMS
Rosângela Lima, UFF
Sarita Bruschi, ICMC-USP
Thiago Colucci, IME-USP
Thiago Santos, USP
Vania Almeida, UFSCar
Wellington Martins, UFG

ARTIGOS

Estratégias para melhorar os processos de Abstração na disciplina de Algoritmos

Dilermando Piva Jr.[1,2], Ricardo L. Freitas[2]

[1]Faculdade de Tecnologia de Indaiatuba – FATEC-ID
Rua D. Pedro I, 65, Bairro Cidade Nova, Indaiatuba, SP.

[2]Pontifícia Universidade Católica de Campinas – PUC-Campinas
Rod.Dom Pedro I, Km.136, Campinas, SP.

pivajr@gmail.com, rfreitas@puc-campinas.edu.br

Abstract. *This article continue the discussion about strategies of abstraction development as motivating factors in the formation of logical-abstract reasoning in students of Computer and Information Technology courses. Three different strategies used, in a preliminary study, showed positive results for improving the teaching of algorithms. These strategies use the characterization of problem situations through associations between textual representations, and algorithmic graphics. It was observed that the difficulty presented by students in the algorithmic representation due to the difficulty in extracting the details that compose a problem situation.*

Resumo. *O presente artigo continua a discussão sobre estratégias de desenvolvimento da abstração como fator de motivação na formação do raciocínio lógico-abstrato em alunos de cursos de Computação e Informática. Três estratégias usadas, num estudo ainda preliminar, apresentaram resultados positivos para a melhoria do processo de ensino de algoritmos. Estas estratégias utilizam a caracterização de situações problema através de associações entre representações textuais, gráficas e algorítmicas. Foi possível observar que a dificuldade apresentada por estudantes na representação algorítmica deve-se à dificuldade dos mesmos em extrair os detalhes que compõem uma situação problema.*

Palavras-chave*: Raciocínio lógico-abstrato, Abstração, Ensino de Algoritmos.*

1. Introdução

A cada novo ano, a cada nova turma, o nível dos alunos que advém do ensino médio é mais e mais preocupante. A maioria chega ao nível universitário sem uma base adequada nas disciplinas de Português e Matemática. Não é incomum encontrarmos alunos com sérias dificuldades na interpretação de textos e na resolução de equações matemáticas.

Por ser a disciplina de algoritmos o primeiro contado do aluno ingressante com o mundo da computação, em termos acadêmicos, caso exista algum problema de

entendimento ou absorção de tal conteúdo, na maioria dos casos ocorre uma desmotivação por parte dos estudantes, fazendo-os pensar em desistir ou mudar de curso. O que se observa, de forma geral, é que o aluno ingressante em cursos superiores da área de computação e informática e que não possui ou não consegue desenvolver a competência de abstração de problemas, não consegue um bom desempenho em disciplinas iniciais do curso, tais como algoritmos, cálculo, álgebra entre outras. Desmotivado, a maioria dos alunos nessa situação escolhe dois possíveis caminhos: evadir do curso (e da instituição) ou mudar de curso.

As implicações disso são inúmeras. A mais grave, e cada vez mais crescente, é a falta de profissionais da área de computação no mercado. Uma outra, principalmente para os cursos de Instituições de Ensino Superior (IES) privadas, é o fechamento de cursos, pela inviabilidade financeira de se manter uma estrutura condizente com os padrões mínimos de qualidade.

Como mencionado por Piva Jr. e outros [PIVA JR. et al, 2009], é fundamental que se observe qual é o perfil dos alunos que estão ingressando ao ensino superior, para assim, traçar uma estratégia interdisciplinar e mais efetiva de desenvolvimento e retenção dos estudantes da área de computação e informática.

Uma análise nas entradas e saídas de estudantes na área de computação, tomando como base o ano de 2008, conforme análise disponibilizada no site do MEC/INEP, tem-se um total de 40.435 alunos ingressantes em cursos de Computação no Brasil. Nesse mesmo ano, um total de 14.194 alunos concluíram os estudos em Computação em IES no Brasil. Assim, em média, apenas 35,1% dos estudantes que ingressam em cursos superiores da área de computação terminam seus cursos.

Não é o objetivo deste artigo estabelecer estratégias para melhorar os índices de aprovação no curso, mas observa-se que, ao melhorar a qualidade do processo de ensino da disciplina de algoritmos, isso se reflete positivamente para todas as disciplinas e, consequentemente, para todo o curso. Neste contexto, vários são os trabalhos encontrados. Dentre eles pode-se destacar alguns que propõem estratégias de ensino/aprendizagem que para o ensino de algoritmos: BORGES 2002; FAYARD 2000; MENDES 2002; SILVEIRA 1998; TAROUCO, ROLAND, FABRE, KONRATH 2004; TAROUCO, 2005 entre muito outros.

Assim, este artigo continua a discussão sobre estratégias de desenvolvimento da abstração iniciada em [PIVA JR. et al, 2009], onde foi abordado a importância da leitura.

Neste artigo, além da problemática envolvida na falta de vocabulário em virtude da pouca leitura dos alunos dos cursos de Computação e Informática, ocasionando problemas no processo de abstração dos problemas no processo de resolução de algoritmos, é

centrado o foco no desenvolvimento de estratégias de ensino para o desenvolvimento da capacidade de abstração dos alunos, mesmo com deficiências de vocabulário.

2. Estratégias para desenvolver/mensurar a Abstração

Por vários semestres, foram aplicadas várias estratégias de ensino/aprendizagem nas turmas iniciais de cursos de computação e informática. Em virtude dos autores atuarem em Instituições públicas e privadas, as estratégias puderam ser aplicadas em vários contextos e públicos distintos, possibilitando assim, uma validação mais efetiva.

Estratégia 1: Do texto à sua representação

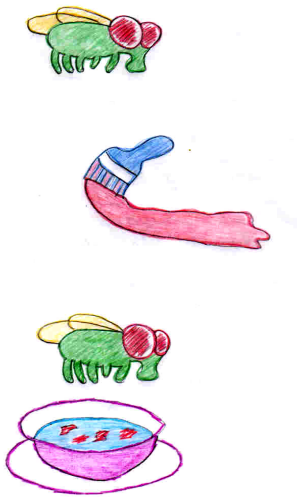
Em virtude da necessidade de mensurar o grau de leitura do estudantes e sensibilizá-los da necessidade de aprimoramento dessa competência, utilizamos a estratégia cognitiva em leitura para designar os princípios que regem o comportamento automático e inconsciente do leitor, enquanto que estratégias metacognitivas em leitura designarão os princípios que regulam a desautomatização consciente das estratégias cognitivas (KATO, 1980, p. 102).

Um dentre esses aspectos cognitivos é a capacidade de abstração, de associar imagens aos enunciados lidos. E é possível fazer um “treinamento” desse hábito, basta expor o leitor aprendiz a trechos de textos obtidos em jornais ou simplesmente a pequenas frases, e pedir que ele faça a interpretação desse trecho, por meio de um desenho. Obviamente que o que investiga não é a habilidade de desenhar do aluno, mas sua capacidade de abstrair uma imagem.

A aplicação dessa estratégia pode ser feita fornecendo o texto para o estudante, ou simplesmente solicitando que o próprio estudante leia e encontre um texto adequado. Para ilustrar seguem dois desenhos (a) e (b), resultados da interpretação dos textos (Ta) e (Tb), respectivamente.



(a)



(b)

Figura 1: Desenhos resultantes das interpretações dos estudantes

Os textos selecionados pelos estudantes são:

(Ta) *"Dona Fernanda ia começar mais um ditado, para cuidar de nossa ortografia.*
- Azedo, azeviche, azulzinho, caçapa, caçada, caçador.
E era uma avalanche de palavras com dois esses, cedilha e ze...
A reforma ortográfica se faz necessária aos olhos do governo para que os países de língua portuguesa possam se comunicar de forma única sem distorções de palavras ou entendimento delas. Porem para nós brasileiros será difícil acostumar-se com a nova linguagem.
Alguns alunos quando chegam a universidade, trazem muitas dúvidas, principalmente em escrita. De quem será a culpa? O mínimo de anos de estudo são 11, tempo suficiente para poder ter capacidade mental de escrever a palavra escrita. Duvidas em ss, s e ç, não deve haver".

Fonte: Sérgio Porto, Biografia de Rosamundo

(Tb) *"Mas eu sou a mosca*
Que pousou em sua sopa
Eu sou a mosca
Que pintou prá lhe abusar..."

Fonte: http://letras.terra.com.br/raul-seixas/48320/

Um exemplo dessa mesma estratégia, porém utilizando um texto único fornecido pelo professor para todos os alunos pode ser encontrado em [PIVA JR et al, 2009].

Estratégia 2: Do desenho a sua descrição

Uma segunda estratégia é promover o caminho inverso. Dado um determinado desenho, ilustração ou foto, os alunos devem construir um texto, explicando ou narrando o que é visualizado. Nessa atividade, o que conta não é a erudição do estudante, mas sim a capacidade de reconhecimento dos detalhes e as inter-relações existentes entre os elementos do desenho/ilustração/foto.

A estratégia 1 e a estratégia 2 podem ser utilizadas em conjunto, da seguinte maneira:

Passo1: Em Casa

1) Pede-se para os estudantes, em casa, selecionarem um ou dois parágrafos de uma notícia vinculada em algum veículo de comunicação; 2) uma vez selecionado, o estudante deve colocar esse texto em uma folha, identificando o título da notícia/reportagem, o parágrafo selecionado, a fonte, o nome e o RA do estudante; 3) em uma folha a parte, o estudante deve colocar apenas o RA no topo superior esquerdo da página e no restante, fazer um desenho que retrate com o máximo de detalhes a notícia/reportagem selecionada.

Passo 2: Em sala de aula – etapa 1

1) O professor deve receber todos os trabalhos e separar a folha com a notícia e o desenho sobre a notícia; 2) em seguida, deve embaralhar todos os desenhos e distribuí-los, um para cada estudante; 3) o estudante de posse do desenho, deverá escrever um parágrafo ou dois, descrevendo o desenho com o máximo de detalhes possível, em no máximo 15 minutos. O desenho deve ser identificado pelo número do RA, contante no canto superior

esquerdo da folha com o desenho; 4) o professor deve recolher os desenhos, embaralhá-los novamente e fazer a redistribuição, repetindo os passos 2 e 3 novamente; 5) ao final o professor deve fazer a junção dos dois textos produzidos pelos alunos para cada desenho, o desenho e o texto original.

Passo 3: Em sala de aula – etapa 2

1) O professor deve selecionar aleatoriamente dois desenhos, não informando a fonte; 2) deve ler a notícia/reportagem original e em seguida mostrar o desenho a todos da sala; 3) em seguida, deve ler os textos produzidos pelos estudantes; 4) por fim, o professor deve tecer alguns comentários sobre os processos de abstração envolvidos na modelagem.

Os passos 3 e 4 dessa etapa é parte central de toda a estratégia 2: sensibilização dos estudantes das dificuldades em concretizar uma abstração e/ou abstrair o real sentido de uma determinada situação.

O professor nessa etapa deve, de forma implícita utilizar os conceitos de modelagem de sistemas[1] e mostrar a importância e as dificuldades envolvidas no processo de abstração de uma determinada situação e a concretização dessa abstração em alguma forma concreta, no caso um desenho ou um texto.

Os estudantes devem ser sensibilizados da necessidade de se atentar aos detalhes, em descrevê-los com a máxima precisão, para que outras pessoas consigam entender e então abstrair corretamente, ou seja, formar as "imagens mentais"[2] apropriadas.

Estratégia 3: do problema a sua interpretação

A terceira estratégia tem como objetivo identificar qual o grau de entendimento, pelos estudantes, dos problemas (exercícios) a serem resolvidos.

A dinâmica é bem simples: uma vez passado um problema, os estudantes devem fazer o desenho da situação envolvida, com o máximo de detalhe possível, e então, após retratada a situação, partir para a solução do problema, utilizando uma linguagem algorítmica (português estruturado ou fluxograma).

Nessa estratégia, o professor terá condições de identificar possíveis problemas de interpretação: falta de vocabulário apropriado, linearidade de pensamento, habilidade de encadeamento de ações entre outros. Assim, uma vez identificada problemática, o professor deverá atuar pontualmente para que os alunos com determinadas deficiências possam identificá-las, trabalhá-las e então saná-las.

[1] Isso deve ser feito de forma implícita, pois trata-se de conteúdo da área de Engenharia de Software, a ser abordado em séries posteriores.

[2] "Imagens Mentais" podem ser associadas aos "interpretantes", conceito presente na Semiótica Peirceana em sua tríade interpretadora "signo-objeto-interpretante".

A atuação do professor poderá ocorrer de diversas formas. Uma delas é selecionando um problema e resolvendo-o em sala, observando a importância da observação dos detalhes no texto e seus reflexos na resolução, trabalhando o formalismo exigido e ilustrando o processo de interação humano-computador como o algoritmo criado gera a solução/interação com quem utilizará o sistema. Este último pode ser feito utilizando um projetor multimídia e um ambiente de programação. Muitos problemas acontecem em virtude da falta de consciência dos alunos do porquê estão fazendo determinadas coisas e o que isso implica em termos de interação humano-computador.

Uma outra forma de atuação pode ocorrer em conjunto com monitores da disciplina. O professor passa a esses monitores uma relação de alunos com as deficiências identificadas. Os monitores selecionam exercícios que exijam o conhecimento das deficiências abordadas. Numa primeira fase, eles desenvolvem em conjunto com os alunos tais exercícios. Numa segunda, os alunos desenvolvem e estes fazem a orientação/correção.

A Figura 2 ilustra um desenho, construído da interpretação da situação-problema, e o respectivo algoritmo. A situação-problema proposta foi a seguinte:

"Em uma Pizzaria, cada tulipa de chope custa R$ 2,80 e uma pizza grande custa 9 vezes o valor da tulipa. Existe um acréscimo de 15% ao valor da pizza para aquelas pedidas com borda recheada. Uma turma vai a pizzaria e pede uma determinada quantidade de chopes e pizzas (algumas delas com bordas recheadas). Escreva um algoritmo que calcule a conta, sabendo quantas pessoas estão na mesa, quanto cada uma deve pagar (já com 10% do garçom)".

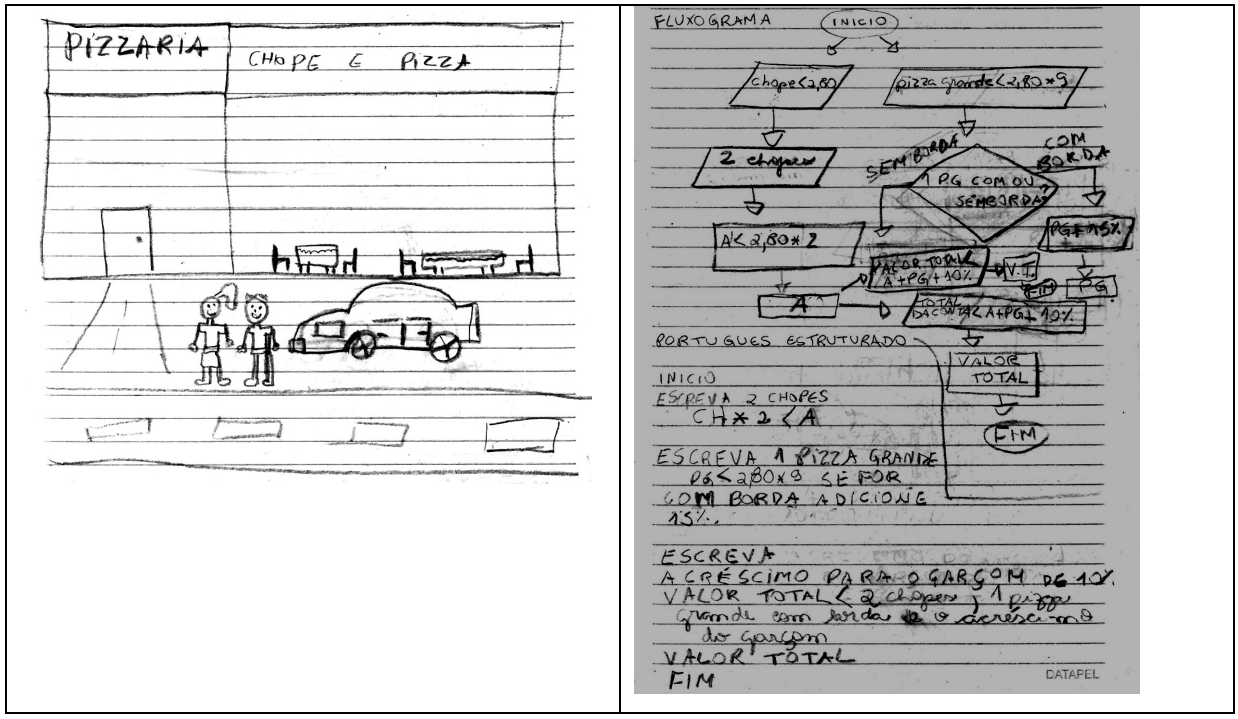


Figura 2: Interpretação da situação-problema pelos alunos

A ausência de detalhes no desenho, faz reflexo no processo de construção da solução algoritma. Problemas na linearidade geralmente são causados pela falta de percepção de todos os detalhes envolvidos no problema.

3. Implicações das estratégias nas habilidades dos estudantes

Os autores, ao formularem as estratégias mencionadas anteriormente, entendem que o ensino de algoritmos é um processo iterativo crescente. Para cada etapa, uma nova camada é construída. Se todos os aspectos envolvidos não forem bem absorvidos e fixados pelos estudantes, ocasionará falhas nessa camada, incorrendo em problemas futuros, principalmente quando envolve a resolução de problemas mais complexos e detalhados (teoria da “pedra na lagoa”).

A Figura 3, ilustra uma situação hipotética, onde foram identificados algumas deficiências nas partes fundamentais da disciplina (bolas vervelhas). Essas pequenas deficiências podem resultar em grandes falhas em etapas mais complexas do processo de ensino da disciplina (partes vermelhas nas demais camadas).

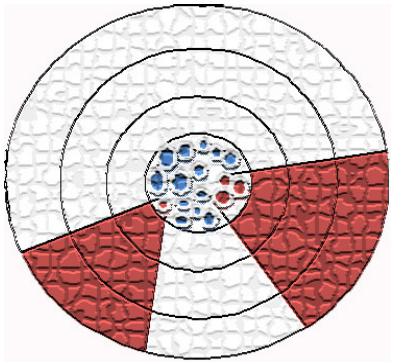


Figura 3: Pequenas deficiências x grandes implicações (problemas)

A título de ilustração, fazendo um resgate dos estudantes que produziram os textos e desenhos da Figura 1, em uma etapa posterior do processo de ensino de algoritmos, quando do início das construções algorítmicas, utilizando português estruturado e fluxograma, pode-se identificar algumas deficiências no processo de elaboração das abstrações e suas repercussões no processo de aprendizado de uma linguagem algorítmica.

Nessa etapa, os estudantes deveriam construir soluções algorítmicas em português estruturado e fluxograma da seguinte situação-problema:

Situação-Problema: *“Pedro comprou um saco de ração com peso em quilos. Pedro possui dois gatos para os quais fornece a quantidade de ração em gramas. Faça um algoritmo que receba o peso do saco de ração, e a quantidade fornecida para cada gato. Calcule e mostre quanto restará de ração após cinco dias.”*

A Figura 4 ilustra a solução encontrada pelo mesmo estudante que teve como produto da Estratégia 1, o desenho (a) da Figura 1, resultado do texto (Ta).

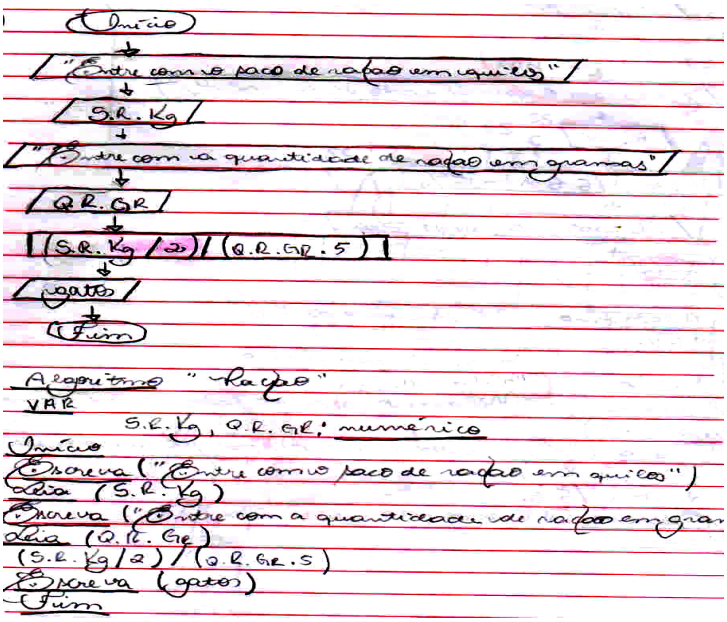


Figura 4: Algoritmo do estudante do desenho (a) da Figura 1 e texto (Ta)

Este estudante, quando da seleção do texto (Ta) não levou em consideração todas as possibilidades e detalhes ali envolvidos para uma possível representação. O que se observa é que o desenho não sugere, com todos os seus detalhes, a situação abordada no texto. Identifica-se nestes algoritmos uma possível deficiência do estudante em levar em consideração todos os detalhes envolvidos no processo de abstração. Como resultado, o que se vê na Figura 4, é o não entendimento completo da situação, e conseqüentemente, uma resolução incorreta do algoritmo. Note que o formalismo foi absorvido, entretanto, a seqüência de ações necessárias para resolver o problema não foi levado a cabo pelo estudante. Detalhes como a quantidade de ração necessária para os gatos por dia não foi identificada como necessária. Uma das implicações é a forma indevida encontrada pelo estudante para resolver matematicamente o problema, que além de misturar grandezas diferentes, não fez a guarda temporária do valor (em uma variável) para posterior exibição.

A Figura 5 ilustra a solução encontrada pelo mesmo estudante que teve como produto da Estratégia 1, o desenho (b) da Figura 1, resultado do texto (Tb). Assim como no desenho e no texto produzidos na Estratégia 1, o estudante no desenvolver os algoritmos mostra coesão, coerência e objetividade. Mesmo o cálculo estando errado, pois se esquece que existem dois gatos, tomando cuidado com os formalismos necessários e a seqüência correta de passos.

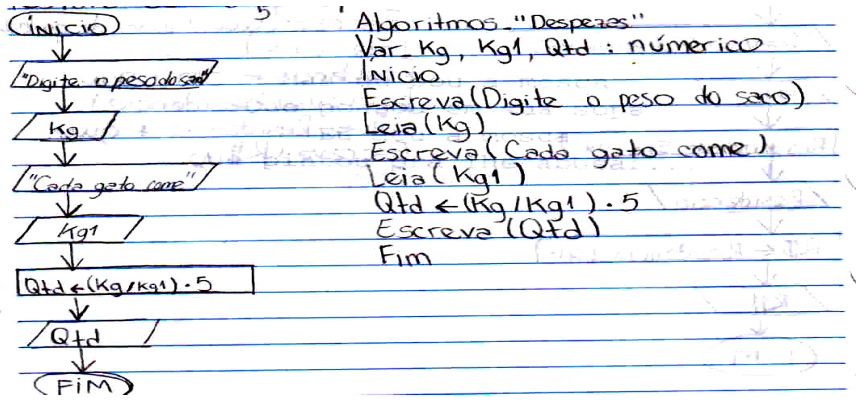


Figura 5: Algoritmo do estudante do desenho (b) da Figura 1 e texto (Tb)

5. Conclusões e próximos passos

É necessário considerar este trabalho como mais um ponto de partida para a exploração de novas propostas que visem estimular o aluno a buscar formas que lhe ofereçam maior capacidade para representação abstrata de um problema.

Um aspecto envolvido na leitura diz respeito ao jogo de imagens criadas durante o ato de leitura. Pêcheux (1990) aborda a questão do imaginário na leitura, sob um outro aspecto. O autor diz que a imagem que A tem de B, e que B tem de A, e que ambos têm de C (Sabendo-se que A é o locutor, B é o interlocutor e C o referente) interfere na recepção da mensagem. Que imagens os alunos produziram? De onde vieram essas imagens registradas nos desenhos deles?

Os exercícios aplicados aos alunos permitiram observar equívocos de natureza cognitiva, o que interferiu na coerência da mensagem traduzida em desenho. Além de um desconhecimento das implicações ou múltiplas possibilidades de interpretação de um texto com certa complexidade.

Ao longo da aplicação das Estratégias 1, 2 e 3, percebeu-se que ao despertar no aluno a preocupação com a correta compreensão do fato ou problema descrito, favoreceu fortemente na melhoria do resultado obtido na construção dos primeiros algoritmos por parte do aluno.

O grande ganho obtido pela aplicação dessas estratégias foi a de possibilitar pequenas deficiências nos processos de abstração que resultariam em posteriores deficiências na construção de algoritmos. Ao atacar as deficiências iniciais, consegue-se uma melhora significativa nos resultados finais.

Alguns resultados numéricos nos motivam a continuar os estudos e a elaborar formas de monitoramento e controle mais efetivos nas Estratégias propostas. Nas turmas de 1º e 2º semestre de 2009, 1º. e 2º. Semestres de 2010, a aplicação das Estratégias descritas neste trabalho e uma atuação mais efetiva dos professores nas deficiências

pontuais dos estudantes resultou em uma melhora de 23,6% na quantidade de aprovados, quando comparado com os anos anteriores (2007 a 2008). Passou de 54,4% de aprovados, em média, para 67,23%.

Como existe uma correlação entre as taxas de reprova na disciplina de algoritmo e a de evasão durante todo o curso, espera-se que esse aumento de 23% faça reflexo na taxa de conclusão. Se isso ocorrer, pensando na situação brasileira, passaríamos de 35,1% para 43,17%, um aumento de mais de 8 pontos percentuais. Em números, 8,07% estariam associados, estimativamente, a mais de 3.000 estudantes, que poderiam concluir o curso.

Estes resultados, embora preliminares, podem indicar um caminho a ser explorado nas aulas iniciais de disciplinas relacionadas ao aprendizado de algoritmos. Um próximo passo pretendido é a elaboração de mecanismos mais elaborados para pré-determinação dos problemas e formas abordagem para melhorar / corrigir o desempenho dos estudantes.

6. Bibliografia

BORGES, M. A. F. **Avaliação de uma Metodologia Alternativa para a Aprendizagem de Programação**. VIII Workshop de Educação em Computação – WEI 2000. Curitiba, PR, 2002.

FAYARD, P., **O jogo da interação: informação e comunicação em estratégia**. Caxias do Sul: EDUCS, 2000.

INEP. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. **Censo da Educação Superior**: **Sinopse 1995-2008**.
Site: http://www.inep.gov.br/superior/censosuperior/ [Acessado em 7/03/2010].

KATO, M. **O Aprendizado da Leitura**. São Paulo: Livraria Martins Fontes Editora Ltda., 1990.

MENDES, A. J. N. **Software educativo para apoio à aprendizagem de programação**, VIII Taller Internacional de Software Educativo- TISE**,** Santiago, Chile, 2002.

PÊCHEUX, M. **Análise Automática do Discurso. In: F. Gadet e T. Hak (orgs.)** ***Por uma análise automatica do discurso. Uma introdução à obra de Michel Pêcheux.*** **Trad. Bethânia S. Mariani et aL.** Campinas: Ed. Da Unicamp, 1990.

PIVA JR. D.; FREITAS, R.L.; PAULA, L.Q. **A Importância da Leitura e da Abstração do Problema no processo de formação do raciocínio lógico-abstrato em alunos de Computação**. XVII Workshop sobre Educação em Informática (WEI'2009). Bento Gonçalves, RS, Brasil, 2009.

SILVEIRA, S.R. **Estudo e Construção de uma ferramenta de autoria multimídia para elaboração de jogos educativos**. Dissertação Mestrado POA/PPGC/UFRGS 1998.

TAROUCO, L. **Jogos educativos via WWW**, Disponível em: http://penta3.ufrgs.br/animacoes/JogosEducacionais.

TAROUCO, L. M. R., ROLAND, L. C., FABRE , M. C. J. M., KONRATH, M. L. P. **Jogos educacionais,** RENOTE -Novas Tecnologias na Educação , v. 2, n. 1. 2004.

Desenvolvimento de um Interpretador de Comandos e Avaliador Gráfico para o Ensino de Estrutura de Dados (IGED)

Dorgival P. da Silva Netto[1], Thiago José S. Oliveira[1], Tiago Davi N. de Sousa[2], Gilberto F. de S. Filho[1], Andrei Formiga[1,2], Alisson V. Brito[1,2]

Departamento de Ciências Exatas – DCE [1]
Programa de Pós-Graduação em Informática – PPGI [2]
Universidade Federal da Paraiba - UFPB

tiagodvneves@yahoo.com.br, {dorgival.netto, thiago.oliveira, gilberto, andrei, alissonbrito}@dce.ufpb.br

Abstract. *The problem of evasion in Computer Science undergratuate courses is well known, and the courses related to programming are among the main culprits. This paper presents the architecture for a programming environment called IGED (Interpreter for Data Structures), which allows the students to develop algorithms to manipulate data structures and visualize the execution of such algorithms by graphical animations. This paper presents the proposed architecture for IGED, a sample high-level program, and a prototype GUI for the tool.*

Resumo. *O problema da evasão nos cursos superiores da área de Computação é conhecido, sendo as disciplinas de programação uma das maiores responsáveis pelo alto nível de reprovação. Para auxiliar no ensino e aprendizagem das Estruturas de Dados e seus algoritmos, esse trabalho apresenta a arquitetura para um Interpretador Gráfico de Comandos Específicos para Manipulação de Estrutura de Dados, o IGED, que permite a elaboração de algoritmos de Estrutura de Dados e a visualização da execução desses algoritmos através de animações.Este artigo apresenta a arquitetura geral do sistema, um exemplo de programa de alto nível, e um protótipo da interface gráfica com o usuário do IGED.*

1. Introdução

No processo de formação dos alunos de cursos da área de Computação e Informática, uma das habilidades esperadas dos egressos é a de construir programas para solucionar problemas variados. Para que isto seja possível no ensino de Estrutura de Dados, o professor deve auxiliar o aluno a sistematizar e organizar a estratégia de solução para os problemas na forma de um algoritmo, analisando as possíveis estruturas de representação do problema que tragam melhor eficiência na sua execução. A formulação de um algoritmo geralmente consiste em um texto contendo comandos que devem ser executados numa ordem prescrita. O texto é uma representação concreta e tem um caráter estático, enquanto que a sua execução com um conjunto de valores iniciais é um evento dinâmico [Guimarães 1994]. Fazer com que o aluno compreenda esses dois aspectos complementares geralmente é um processo complexo, pois é preciso "visualizar" as estruturas dinâmicas das execuções dos algoritmos a partir da estrutura estática do texto do algoritmo.

O estudo de Netto (2010), que teve como objetivo investigar e identificar as dificuldades encontradas por alunos de cursos de computação no aprendizado da disciplina

Estrutura de Dados, mostrou que 33% dos alunos tiveram dificuldade com a abstração e 17% dos alunos consideraram a complexidade do assunto como uma barreira ao aprendizado. Considerando as dificuldades com conceitos específicos, 24% dos entrevistados responderam que o assunto de maior dificuldade foram as estruturas de listas, 22% dos participantes responderam ter mais dificuldade com o conceito de recursividade e 15% afirmaram serem as estruturas de filas as mais difíceis. Também foi identificado que a maioria dos professores da disciplina não utiliza ferramentas educacionais para auxiliar as aulas, com a possível exceção de ferramentas gerais de programação.

Com o objetivo de facilitar o entendimento dos conceitos abstratos envolvidos, acredita-se que o uso de Objetos de Aprendizagem que permitam a exploração dos conteúdos de forma digital e interativa, com recursos midiáticos como simulações, gráficos, desenhos e animações,, possa fazer com que os alunos tenham uma melhor compreensão dos conteúdos e se sintam mais motivados ao estudo.

Segundo Monteiro (2006), Objetos de Aprendizagem podem se tornar um valioso recurso pedagógico, com possibilidades de facilitar e tornar mais eficaz o processo de ensino. Considerando isto, e conhecendo-se a dificuldade de assimilação de conteúdo dos alunos nas disciplinas de Estrutura de Dados [Netto 2010], faz-se necessário o uso de novos recursos para apoiar o ensino e a aprendizagem desses conteúdos considerados de fundamental importância para a área de Computação e Informática.

Neste sentido, o presente trabalho propõe a arquitetura para um Interpretador Gráfico de Comandos Específicos para Manipulação de Estrutura de Dados, o IGED, permitindo ao aluno uma efetiva ferramenta de programação e acompanhamento de seus programas através de animações das abstrações de cada estrutura de dados abordada. Após uma discussão sobre as motivações do trabalho e da ferramenta proposta (Seção 2) e sobre os trabalhos relacionados na literatura (Seção 3), é apresentada a arquitetura geral para o IGED (Seção 4), e mostra-se um protótipo da interface proposta para o sistema (Seção 5). Finalmente, a Seção 6 conclui o artigo com as considerações finais. .

2. Uso de Animações de Algoritmos no Ensino

Segundo Garcia (2007), os livros-texto e aulas convencionais que abordam as estruturas de dados utilizam-se de mecanismos que vão além da descrição em (pseudo) linguagem de programação, como ilustrações. Entretanto, o excesso dessas ilustrações, quando vistas estaticamente, contribui muito pouco para aumentar a compreensão dos algoritmos, dado que além das informações que estão sendo descritas, há uma grande quantidade de comunicação implícita na correlação entre as imagens. Esta correlação pode ser mostrada através da animação gráfica de tais imagens, melhorando a comunicação da informação ao aluno ao deixar as relações entre as imagens explícita.

Em [Brown 1987], M. Brown introduziu a ideia de ambientes dedicados à construção de animações gráficas computadorizadas como ferramenta para ensino em computação, como já são aplicadas com sucesso em outras áreas, como química, anatomia e simulação de circuitos.

Um primeiro uso de animação de algoritmos é a criação de filmes [Hopgood 1974] ou sequências pré-programadas da execução de eventos [Brown 1987]. Essa abordagem é denominada passiva, pois o usuário se comporta como mero espectador e é útil para a complementação de livros-texto e aulas.

Uma limitação clara, no entanto, está no fato de o usuário não poder testar novos conjuntos de dados e ter que ficar limitado aos que são fornecidos pelos criadores da

animação. Para remover essa restrição, alguns sistemas [Brown 1991], [Stasko 1990], [Amorim 1993] introduziram facilidades para que seus usuários pudessem fazer as suas próprias experiências, direcionando o andamento do algoritmo. Tal abordagem é denominada ativa.

Resultados positivos para o uso dessa abordagem foram relatados em uma experiência realizada com o ambiente POLKA [Lawrence 1994]. O exame desses resultados confirma que a abordagem ativa é mais poderosa, pois tem-se uma participação maior por parte do usuário.

Uma nova abordagem para o uso de animações no processo de ensino e aprendizagem foi introduzida pelo ambiente Astral [Garcia 1997]. Nesse ambiente, o usuário pode visualizar animações de algoritmos que ele mesmo implementa, além de visualizar animações preparadas previamente. Essa nova abordagem, denominada pelos autores de abordagem construtiva, foi implementada e exemplificada no ambiente Astral, que possui uma interface de programação de animações em uma plataforma adequada e amigável.

3. Trabalhos relacionados

O sistema Balsa (Brown Algorithm Simulator and Animator) [Brown 1987] foi desenvolvido na Brown University no início dos anos 80 e seu intuito era o de servir como um laboratório de experimentação com representações dinâmicas de algoritmos em tempo real. Seu sucessor, Balsa-II, introduziu uma interface mais amigável e uma maneira homogênea para a execução e interação com animações, executando sobre a plataforma Macintosh e utilizando a linguagem Pascal para entrada dos algoritmos.

No início dos anos 90, surgiu o projeto Zeus de M. Brown [Brown 1991] cujas contribuições incluíam uma proposta de separação entre algoritmo e visualizadores, colocando interfaces bem definidas entre eles; o suporte à construção de programas de grande porte utilizando o paradigma da orientação a objetos e a utilização da linguagem Modula 3. Em Zeus, múltiplos algoritmos podem ser executados concorrentemente, facilitando a análise de diferenças e similaridades entre vários algoritmos.

Xtango [Stasko 1990] é um sistema de animação de algoritmos de propósito genérico que permite a construção de animações em tempo real. O objetivo principal é prover facilidades para os usuários que irão construir as suas animações, fornecendo uma interface de alto nível. Utiliza a linguagem C e opera sobre plataformas Unix e X11 Window System. Polka[1] é o sucessor de Xtango e foi projetado para dar suporte ao desenvolvimento de animações concorrentes, de maneira a facilitar a exibição de algoritmos paralelos.

O projeto Astral [Garcia 1997] tem como sua principal inovação a produção sistemática de animações, na qual o estudante utiliza uma coleção pré-estabelecida de procedimentos de manipulação gráfica das estruturas de dados para criar algoritmos mais complexos. O aprendiz pode construir a animação completa de uma dada tarefa através da composição desses procedimentos básicos, podendo assim facilitar a ligação do conceito abstrato com a programação do algoritmo.

A abordagem construtiva proposta no projeto Astral está baseada em procedimentos básicos de manipulação gráfica pré-programados, sendo os detalhes de sua implementação totalmente abstraídos do usuário através de uma biblioteca de programação. Entretanto, embora o Astral possibilite ao usuário a construção de suas próprias animações, sendo assim um avanço sobre as outras abordagens, o uso de componentes básicos pré-estabelecidos para

[1] http://www.cs.sunysb.edu/~algorith/implement/xtango/implement.shtml

criar os algoritmos e suas animações associadas também limita o que pode ser representado no sistema. Desta forma, a ferramenta IGED aqui proposta objetiva tornar a abordagem construtiva mais efetiva ao aluno, disponibilizando uma linguagem de programação completa que permita a animação de qualquer algoritmo. A arquitetura do IGED é descrita a seguir.

4. Arquitetura do Interpretador Gráfico de Estrutura de Dados.

A arquitetura proposta para o Interpretador Gráfico de Estruturas de Dados (IGED) é ilustrada na Figura 1. Nela podem-se visualizar três camadas distintas: Camada Gráfica, responsável pela iteração com o usuário; o Interpretador de Comandos que avaliará e executará os comandos passados pelo usuário; Camada Avaliadora, responsável por comparar o resultado obtido pelos comandos passados pelo usuário com o resultado esperado em cada exercício. Além das três camadas citadas, o sistema inclui um componente chamado de Gerador de Tarefas, responsável pelo cadastro das tarefas criadas pelos instrutores e disponibilização dessas tarefas para os usuários.

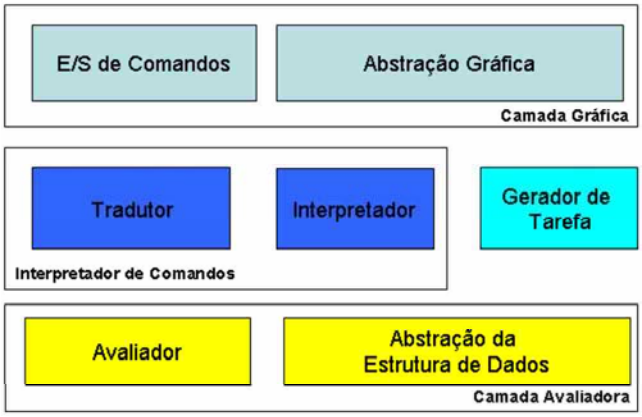


Figura 1. Arquitetura do interpretador Gráfico de Estruturas de Dados

4.1. Camada Gráfica

A Camada Gráfica, responsável pela interface com o usuário do sistema, é dividida em dois componentes: o primeiro é a E/S (Entrada/Saída) de Comandos, responsável por receber os comandos do usuário e entregá-los à camada Interpretador de Comandos (Entrada) e exibir os resultados dos comandos durante a execução de uma animação (Saída).

O segundo componente da Camada Gráfica, chamado Abstração Gráfica, é responsável por representar graficamente as estruturas de dados e suas operações, através do uso de animações. Esse componente deve se adaptar para as especificidades dos diferentes tipos de estruturas, podendo manipular estruturas estáticas como vetores ou dinâmicas como listas e árvores, e realizar tais manipulações através das ações básicas enviadas pelo componente Interpretador de Comandos.

4.2. Interpretador de Comandos

A camada responsável pela efetiva abordagem construtiva proposta neste trabalho é chamada Interpretador de Comandos, na qual se encontram dois componentes. O primeiro é o componente Tradutor, responsável pela tradução dos programas em linguagem de alto nível para uma linguagem intermediária; isso permite que o segundo componente, chamado Interpretador, execute o código intermediário para efetuar a manipulação das estruturas de dados. Os passos de computação realizados no nível da linguagem intermediária podem ser animados individualmente, e essa maior granularidade de passos básicos faz com que a animação se torne mais detalhada ao aprendiz, que tem assim uma experiência mais completa

dos seus passos e uma maior associação do código implementado ao seu efeito na abstração da estrutura de dados.

Para a implementação da camada Interpretador de Comandos, há a necessidade da definição de uma gramática para a linguagem de programação adotada pelo componente Tradutor. Também é necessário definir a linguagem intermediária que será usada na saída do Tradutor, pois essa linguagem define os passos básicos que serão executados pelo Interpretador e animados na Camada Gráfica. O fluxo das informações na ferramenta é ilustrado na Figura 2, iniciando com o programa de entrada do usuário, que é traduzido para os comandos da linguagem intermediária, que por sua vez são executados pelo Interpretador e animados pelo componente de Abstração Gráfica.

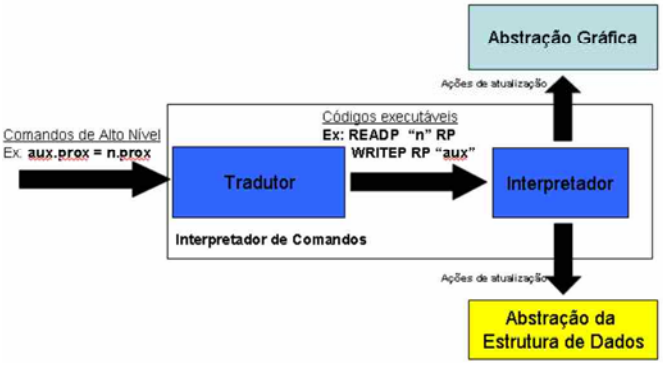


Figura 2. Fluxo de dados na camada Interpretador de Comandos.

Um dos benefícios diretos da divisão da camada Interpretador de Comandos em dois componentes é a flexibilidade que o sistema terá para trocar de linguagem de programação de alto nível, pela simples implementação de um novo componente Tradutor. Também é possível alterar a linguagem intermediária executada, embora isso tenha impacto em um maior número de componentes do sistema.

```
// Insere um número n em uma lista ordenada (crescente).
void InsereNo(int n, Lista<int> lista)
{
   var p = lista.frente;
   var q = null;

   while (p != null && p.valor < n) {
      q = p;
      p = p.proximo;
   }

   var novo = new Lista<int>.No;
   novo.valor = n;
   novo.proximo = p;
   if (q != null)
      q.proximo = novo;
   else
      lista.frente = novo;
}
```

Figura 3. Exemplo de programa na linguagem de alto nível do IGED.

Inicialmente, a sintaxe da linguagem de programação utilizada pelos usuários do sistema foi projetada para ser similar à linguagem Java e às linguagens da família C/C++, pois é um tipo de sintaxe com a qual a maioria dos alunos está familiarizado. Futuramente será possível implementar módulos Tradutores para outras linguagens de alto nível. Um exemplo de programa escrito na linguagem de alto nível do sistema IGED é mostrada na Figura 3. Este programa inclui apenas uma função chamada `InsereNo`, que insere um novo nó contendo um inteiro em uma lista encadeada que esteja ordenada em ordem crescente.

4.3. Camada Avaliadora

A Camada Avaliadora tem a responsabilidade de validar os comandos passados pelo aluno/usuário durante o processo de resolução de uma tarefa pré-cadastrada pelo instrutor no sistema. Logo, sua função é indicar ao aluno se a sua solução foi bem sucedida segundo a expectativa de resultado cadastrada anteriormente pelo instrutor.

Esta camada também está dividida em dois componentes: o primeiro componente é a Abstração de Estrutura de Dados, que contém a implementação de cada estrutura abordada e irá executar nestas estruturas os comandos recebidos e traduzidos pelo Interpretador de Comandos. Este componente irá trabalhar com duas instâncias idênticas da estrutura: a *original* e uma *réplica*. O objetivo da réplica é servir de controle da tarefa: o instrutor vai cadastrar um programa que serve de solução para a tarefa em questão; esse programa será executado sobre a réplica. Já a instância original receberá os efeitos da execução do programa inserido pelo usuário para resolver a tarefa.

O componente Avaliador terá acesso às duas instâncias (original e réplica) e irá comparar o seus estados finais. Se estiverem iguais, a resolução da tarefa será considerada com sucesso, caso contrário será considerada falha.

5. Protótipo de interface

Na Figura 4 é mostrado um protótipo da tela principal do IGED. Nessa tela, podemos identificar os campos: Exercício, responsável por mostrar a pergunta de um exercício previamente cadastrado, que deve ser resolvido pelo aluno; no centro da tela principal, encontramos o ambiente gráfico, responsável por apresentar a animação do código construído pelo aluno; na parte inferior da tela está a área Comandos, que é o campo onde o aluno escreverá o código para responder a questão mostrada no campo Exercício; no lado direito da tela, encontramos uma legenda, que mostrará o significado de todos os símbolos utilizados na animação para facilitar a compreensão da animação.

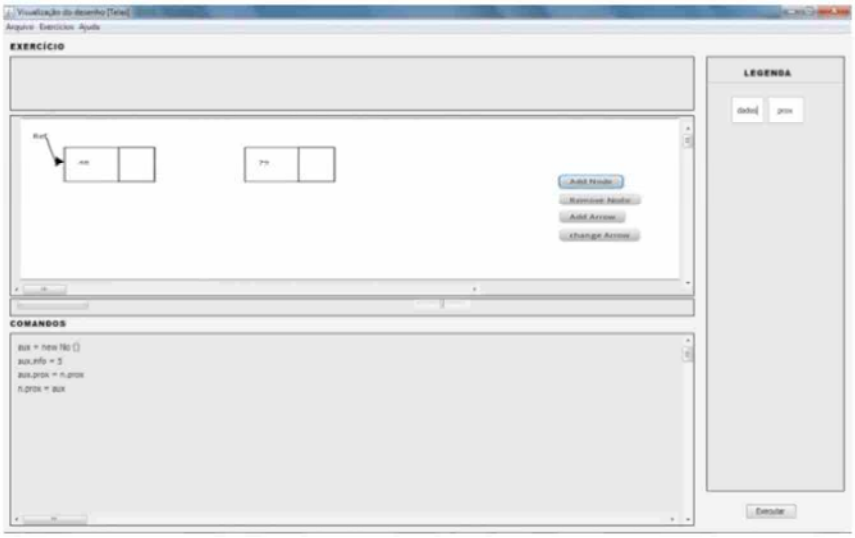


Figura 4. Protótipo da tela principal do Interpretador Gráfico de Estruturas de Dados

O JavaFX[2], foi escolhido para ser a plataforma utilizada para a construção da parte gráfica e as animações do IGED. Um fator importante para a escolha do JavaFX foi a possibilidade de desenvolver aplicações gráficas interativas de forma rápida e fácil.

Na Figura 5 é ilustrada uma animação feita em JavaFX que representa a inserção de nós em uma lista. O Passo 1 representa o estado inicial do ambiente, com dois nós, tendo o nó inicial (à esquerda) uma referência; no Passo 2, acontece a inserção de um novo nó, com um valor gerado aleatoriamente; o Passo 3 representa a ligação entre os nós através de setas; e por fim, o Passo 4 representa a ordenação gráfica dos nós, já que estes foram dispostos de forma desorganizada.

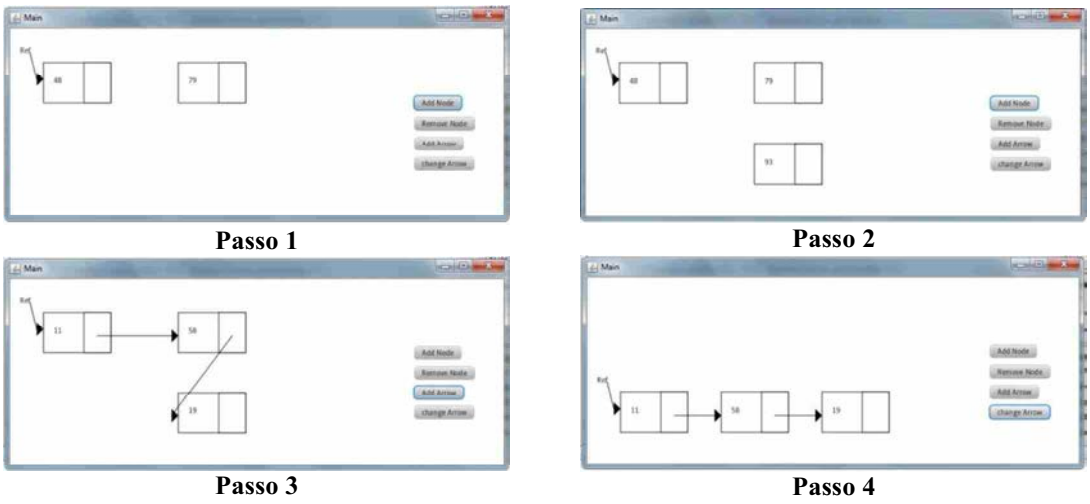


Figura 5. Representação de uma animação de Listas

6. Considerações finais e trabalhos futuros

O Interpretador Gráfico para Estruturas de Dados, IGED, é uma ferramenta proposta para uso no processo de ensino e aprendizagem de disciplinas dos cursos de computação que envolvem

[2] JavaFX: http://openjfx.java.sun.com/current-build/doc/reference/JavaFXReference.html

as Estruturas de Dados e seus algoritmos. Diferente de um livro-texto ou um conjunto de imagens estáticas, o IGED permite aos seus usuários uma abordagem ativa e construtiva, na qual os programas que manipulam as estruturas de dados podem ser construídos e seu resultado visualizado na tela imediatamente. Pretende-se que a visualização torne o assunto menos abstrato para os alunos, aumentando a retenção do conhecimento e diminuindo a evasão nas disciplinas relacionadas.

Os trabalhos futuros para o IGED compreendem a implementação de todas as camadas, inicialmente para as estruturas de dados mais simples como listas encadeadas. Tendo uma implementação terminada, será possível fazer um teste da ferramenta, através do uso da mesma em disciplinas relacionadas ao ensino das estruturas de dados. Em seguida, prevê-se a ampliação da implementação com novas estruturas e seguindo as sugestões que certamente surgirão do uso da ferramenta em sala de aula.

Referências

Amorim, R. V.; de Rezende, P. J. Compreensão de Algoritmos através de Ambientes Dedicados a Animação. XX Semish, 1993.

Brown M. H. Algorithm Animation. The MIT Press, 1987.

Brown, M. H. Zeus: A System for Algorithm Animation and Multi-View Editing. Proc. IEEE Workshop on Visual Languages, 1991.

Garcia, I. C.; De Rezende, P. J.; Calheiros, F. C. (1997). Astral: Um Ambiente para Ensino de Estruturas de Dados através de Animações de Algoritmos. In Revista Brasileira de Informática na Educação. Florianópolis: SC, Volume. 1, p. 71-80.

Guimarães, A. M.; Lages, N.A.C. (1994). Algoritmos e Estruturas de Dados. Rio de Janeiro , LTC.

Hopgood, F. Computer Animation Used as a Tool in Teaching Computer Science. Proc. 1974 IFIP Congress, pp. 889-892, 1974.

Lawrence, A. W.; Badre, A. N; Stasko, J. T. Empirically Evaluating the Use of Animations to Teach Algorithms. Technical Report GIT-GVU-94-07, Computer Science Department, 1994.

Monteiro, B. S., Cruz, H. P., Andrade, M., Gouveia, T., Tavares, R., Anjos, L. F. C.(2006) Metodologia de desenvolvimento de objetos de aprendizagem com foco na aprendizagem significativa, XVII Simpósio Brasileiro de Informática na Educação, Brasília.

Netto, Dorgival. Análise dos dados do questionário da disciplina Estrutura de Dados. Relatório Técnico. Departamento de Ciências Exatas, UFPB, 2010.

Stasko, J. T. Tango: A Framework and System for Algorithm Animation. Computer, 23(9): 14-36, sep 1990.

Takkou: Uma Ferramenta Proposta ao Ensino de Algoritmos

Leônidas S. Barbosa, Teresa C. B. Fernandes, André M. C. Campos

Departamento de Informática e Matemática Aplicada – Universidade Federal do Rio Grande do Norte (UFRN)
Caixa Postal 1679 – 59.072-970 – Natal – RN – Brasil

barbosaleonidas@aol.com, {tfernandes.cc, amccampos}@gmail.com

Abstract. *This paper presents a tool, named Takkou, to be used as support on algorithms classes. The tool aims to provide a playful learning experience based on puzzle games. Takkou is a game where the player must solve a puzzle through the actions performed by his/her game character. However, the player does not directly control his/her character, but instead provide a set of algorithmic instructions to it. Takkou is based on theory of Meaningful Learning, and its main aim is to motivate students to exercise and improve their algorithms skills.*

Resumo. *Este artigo apresenta uma ferramenta, chamada Takkou, a ser utilizada como ferramenta de suporte em aulas de algoritmos. O objetivo da ferramenta é criar uma experiência lúdica através de jogos de raciocínio. Takkou é um jogo onde o jogador deve resolver um problema através de ações realizadas por seu personagem no jogo. Entretanto, o jogador não controla diretamente seu personagem. Ele o controla através de um conjunto de instruções algorítmicas. Takkou é baseado na Teoria de Aprendizagem Significativa e seu principal objetivo é motivar os estudantes a se exercitarem e aperfeiçoarem suas habilidades em algoritmos.*

1. Introdução

A disciplina de algoritmos, em cursos de computação, representa a base na formação de estudantes em ciências da computação. Por meio de algoritmos são propostas soluções de problemas de ordem diversa, desde qual a melhor rota para um duto de petróleo até a definição de quantos passos um robô deve executar para trocar uma lâmpada. Por ser uma disciplina de base, aos alunos que ingressam no curso de computação, é exigido que se tenha boa habilidade algorítmica. Entretanto, durante os últimos anos o que se observa é que há um aproveitamento ineficiente dos alunos nesta disciplina. Em virtude do mal aproveitamento muitos alunos perdem o interesse pelo curso enxergando a disciplina como barreira para sua formação. Segundo Rocha (1993), os números de evasão e desistência na disciplina de algoritmos chegam a 60%. Ainda segundo o autor, é defendido que o mal aproveitamento na disciplina se deve a má metodologia abordada em sala de aula. Porém, em Jesus e Brito (2009), é afirmado que cabe aos alunos possuir certas habilidades requeridas no curso e a falta delas é a responsável pelo mal aproveitamento culminando em desistências e evasões. Dentre outras discussões o fator motivacional é apresentado como forte fator responsável pelo mal aproveitamento,

onde a falta de motivação faz com que os alunos sintam-se desencorajados em continuar no curso.

Como o objetivo de esclarecer quais os problemas que afetam o ensino de algoritmos, em Campos et al. (2011), foi realizado um levantamento sobre o aproveitamento na disciplina de algoritmos durante o período de setembro a novembro de 2010. Além desse abordagem, foi realizado um comparativo com os dados de 2003 a novembro de 2010 advindos de artigos que abordavam o tema, aproveitamento em algoritmos. Junto aos alunos foi realizado uma solicitação de dados através de um questionário quantitativo quanto ao aproveitamento e dificuldades vivenciadas durante a disciplina de algoritmos, e realizado um levantamento estatístico junto a algumas universidades brasileiras. Ao final, os autores realizaram uma revisão bibliográfica que frente as observações obtidas confirmavam os apontamentos.

Com os dados levantados, percebeu-se que há na verdade dois problemas principais responsáveis pelo mal aproveitamento na disciplina de algoritmos: o problema do ensino, que é referente a abordagem pedagógica aplicada em sala de aula, bem como a experiência dos alunos e a interação professor-aluno, dentre outros fatores relatados no trabalho citado; e o problema do raciocínio lógico, representando habilidade extremamente requerida nas disciplinas dos cursos de computação, onde, por vezes, o ingressante não a possuindo bem desenvolvida sente dificuldades em transcrever a resolução de problemas em algoritmos. Como forma de confrontar o problema do raciocínio lógico, no trabalho citado, foi proposta uma ferramenta com características de jogo que, por meio de *puzzles*, abordam conceitos de algoritmos e auxiliam os estudantes no exercício desta habilidade. Através de uma metodologia de uso onde o aluno não só soluciona os *puzzles*, mas, também, constrói desafios, e, assim exercita seu aprendizado, foi proposta a ferramenta *Takkou* a ser disponibilizada via web para os usuários.

Nesse trabalho será apresentada a ferramenta *Takkou*, ferramenta proposta como um protótipo do trabalho abordado em Campos et al. (2011), que intenta auxiliar o aluno no exercício do raciocínio lógico através de um jogo suscitando maior motivação com base em aspectos pedagógicos inspirados na teoria de David Ausubel.

2. Fundamentação Teórica

A teoria de David Ausubel, teoria do Aprendizado Significativo, aborda aspectos importantes para tornar a aprendizagem mais significativa, tais como a formulação do material potencialmente significativo, fazendo uso de signos que remetam a alguma situação real ou não, até a troca constante de informações entre professor-aluno e a disposição dos alunos em aprender conforme a teoria. Sendo uma teoria cognitivista, preconiza grande importância ao papel do aluno no modo como as informações são assimiladas por estes e como esta informação se adéqua e se reestrutura na sua estrutura cognitiva. Há, portanto, uma constante preocupação no aluno como fator do processo de aprendizagem, pois é desejável que o aluno aprenda de forma significativa, enriquecendo sua estrutura cognitiva, e não mecanicamente, onde os conceitos ficam dispersos nessa estrutura. Ao estruturar sua teoria, Ausubel afirma que quando um conceito menor se liga a um conceito maior, o qual identifica algum domínio de

conhecimento específico, esse conceito maior é chamado de **subsunçor**. O conceito subsunçor define uma das principais características da teoria de Ausubel, pois é através dele que o teórico justifica todo o processo empregado em sua teoria afirmando que, se os conceitos recém-adquiridos se ligam a um determinado subsunçor, há o aprendizado significativo, caso contrário o aprendizado é dito mecânico, pois não há nenhum conceito que esteja ligado ao conceito recém-adquirido e este ficará disperso na estrutura cognitiva do aluno sem nenhuma ligação, o que fará com que o esqueça com maior facilidade, Moreira e Mansini (2006).

Além do aspecto cognitivista da teoria, há o construtivista afirmando que os conhecimentos passados devem ser significativos, devem provocar no aluno alguma importância, Ausubel, em outras palavras, diz que cabe ao aluno em determinado momento tomar para si se um conteúdo é importante de acordo com a sua estrutura cognitiva e, a partir disso, construir o seu próprio conhecimento. Pelizzari (2002), afirma que há dois aspectos importantes que levam o aluno a aprender significativamente. O primeiro é o significado lógico, no qual para o aluno, a informação passada deve possuir algum sentido que se adeque ao conteúdo passado. O segundo é o significado psicológico, esse é particular ao aluno, é ele que, em dado momento, irá ligar na sua estrutura cognitiva algum conceito a um subsunçor próprio daquele conceito próprio ou temporário, onde subsunçores próprios são aqueles que ligam de forma direta um conceito menor a um conceito maior em um domínio especifico. Diz-se que um subsunçor **A** é próprio de **B** se os conceitos aos quais **B** referencia é subconjunto dos conceitos referenciados por **A**.

Para a criação da ferramenta *Takkou* preocupou-se no modo como esta iria transmitir ao aluno signos do dia a dia, mas que com o auxilio da ferramenta referenciassem conceitos de algoritmos. A criação da ferramenta com base na proposta de Ausubel se justifica por um outro ponto abordado nessa teoria, o de que o material passado deve ser potencialmente significativo. Para que fosse possível criar uma ferramenta como material potencialmente significativo pensou-se em como os objetos e o visual desta estariam dispostos de forma a relacionar conceitos de algoritmos. Na *Takkou*, os elementos que compõem o cenário do jogo são simples e representam algum objeto do cotidiano, uma pedra, um lago, um arbusto, uma chave, porta, etc. Entretanto, quando o aluno tenta alcançar algum dos objetivos do jogo, são apresentados, de forma tangencial, conceitos de algoritmos em forma de instruções, tais como: estruturas de condicionais, laços de repetição, variável, etc. Dessa forma para que o objetivo do jogo seja alcançado o aluno deve criar um algoritmo que solucione o *puzzle*. Em contra partida a resolução de problemas por meio dos alunos, há ainda a possibilidade do aluno criar o seus próprios desafios. Nessa abordagem, o aluno é colocado na condição de arquiteto e solucionador do problema, pois acredita-se que a medida que o aluno cria algum *puzzle* ele está, de certo modo, pensando duas vezes, a primeira em como criar e a segunda em como resolver a sua criação. Assim alcançasse, neste ponto, uma das características marcantes do construtivismo, fazendo com que o aluno construa o seu próprio conhecimento. Há ainda o fator de que, dessa forma, o aluno se esforça muito mais a aprender os conceitos de algoritmos, pois só entendendo como estes funcionam ele poderá criar *puzzles* suficientemente desafiadores aos seus colegas.

Com base no que foi apresentado, para a implementação da ferramenta em

questão, preocupou-se em atender aos seguintes aspectos voltados a teoria que a inspira.

•Apresentação de signos, símbolos do cotidiano: na ferramenta alguns símbolos e signos comuns do dia a dia dos alunos são transcritos visualmente deixando-os visualizar situações comum já viciadas por eles;

•Possibilidade de desafiar os outros usuários (criação de *puzzles*): está característica é de grande importância por tornar a abordagem pedagógica da ferramenta mais proveitosa, não o bastante, além desse aspecto pedagógico há o aspecto motivador, que surge ao permitir que os usuários possam desafiar-se;

•Temporizador de desafios: a questão tempo foi abordada na ferramenta de modo a tornar os desafios mais interessantes, não apenas isso, mas conforme se sabe, em cursos de algoritmos é comum que alunos resolvam problemas em um dado tempo. Com a ideia do temporizador em cada puzzle intenta-se atendar os fatores motivacionais e forçar o aluno a pensar em tempo hábil tornando o jogo mais interessante, o que também irá indiretamente implicar em uma melhoria nas suas habilidades deixando-o mais produtivo na resolução de problemas;

•Usabilidade: a ferramenta apresenta características interessantes no tocante a usabilidade. Esta preocupação deveu-se ao fato de “jogadores” sentirem-se mais atraídos por jogos com boa usabilidade facilitando o seu uso e deixando-os à vontade no ambiente da ferramenta. Outro ponto importante é que, como citado no item anterior, é dado ao aluno um tempo determinado para solucionar um puzzle, logo, o uso da ferramenta, a sua facilidade de uso, deve se dá a contento possibilitando que este consiga usá-la de forma produtiva no tempo que lhe foi dado;

No referente aos aspectos técnicos da ferramenta tem-se:

•Ferramenta usável via web: a criação da ferramenta *Takkou* está voltada ao ambiente web, uma vez que, em qualquer lugar onde o usuário estiver poderá fazer uso dela, exercitar sua habilidades, servindo até mesmo aos propósitos de educação a distância conforme interesse;

•Criação de um ambiente de registro de *puzzles*: conforme já fora mencionado, na *Takkou* há a possibilidade de serem criados *puzzles* pelos usuários, para que estes possam ser mantidos foi sugerido a criação de uma base de dados acessível por meio de serviço web que recupera e registra os *puzzles.* Essa base poderá futuramente ser usada com diversos propósitos, como por exemplo uma avaliação via ferramenta.

3. Trabalhos Correlatos

Há na literatura grande número de trabalhos voltados ao ensino de algoritmos. Alguns destes trabalhos abordam através de ferramentas formas novas e mais motivadores de se melhorar o ensino desta disciplina. Outras abordam o problema por meio de metologias com base em teorias pedagógicas. Dentre os trabalhos que abordam a criação de ferramentas voltadas ao ensino de algoritmos, ou que podem ser utilizadas com tal finalidade, destaca-se, aqui, alguns deles de acordo com a seguinte categorização.

1.Ferramentas de linguagem: ferramentas voltadas a alguma linguagem de programação especifica ou uma nova linguagem criada com o propósito de melhorar o

ensino de algoritmos. Enquadram-se nesta categoria as ferramentas: JavaTool e Tepequém. A primeira aborda o ensino algoritmos através de um subconjunto de instruções da linguagem Java por meio de uma ferramenta visual, Mota et al, (2008). A segunda aborda o ensino de algoritmos através de uma linguagem criada em português que possui características de acentuação e outros detalhes para o ensino de algoritmos e programação, Hinterholz Jr, (2009);

2.Ferramentas de simulação: são ferramentas que, de algum modo, simulam o comportamento de algoritmos, seja via depuração de código ou através do auxílio de fluxograma. As ferramentas CIFluxProg e VisuAlg se enquadram nesta categoria. A primeira aborda o ensino de algoritmos, simulação, através de fluxogramas e depuração, Santiago (2004) . Já a segunda faz uma abordagem mais voltada a depuração do conteúdo das variáveis em dado momento da execução do algoritmo, Souza (2009) ;

3.Ferramenta de interação com algoritmos: não necessariamente são ferramentas voltadas ao ensino de algoritmos, mas fazem constante referencia aos seus conceitos. Dentre as ferramentas que se enquadram nesta categoria cita-se o Scratch e o Robomind. A primeira é uma ferramenta do MIT de propósito geral onde o usuário pode interagir e criar animações e vídeos utilizando de pseudo código por aglutinação de comandos, possuindo como uma das suas características ter sido desenvolvida sob base pedagógica construtivista, Maloney et al, (2008) . O robomind por outro lado é uma ferramenta que simula um robô em um ambiente com obstáculos e objetos a serem capturados. Esse ambiente pode ser estendido a um robô real por meio do Lego Mindstorm. Para mover o robô virtualmente ou real é possível executar instruções algorítmicas por meio de código em uma linguagem inspirada na linguagem LOGO.

4.Jogos aplicados ao ensino de algoritmos: são jogos que possuem geralmente, de forma indireta, conceitos de algoritmos incorporados na interação do jogador. Como exemplo de jogos que envolvem o usuário, muitas vezes de forma tangencial, com os conceitos de algoritmos têm-se: Robozzle e o Light-Bot, estes são jogos similares que abordam através de *puzzles* ideias que se referem a algoritmos. Em ambos é apresentado um puzzle que deve ser solucionado por meio de comandos através de caixas laterais que fazem referencia a conceitos da disciplina de algoritmos.

A pesar dos inúmeros trabalhos voltados e não voltados ao ensino de algoritmos, mas que podem ser utilizados com tal propósito, a ferramenta *Takkou* foi proposta por apresentar diferenciais frente as já existentes. Dentre as ferramentas citadas as que mais se assemelham a discutida neste artigo são os jogos Robozzle, Light-Bot e Scracht. Para diferenciar a ferramenta *Takkou* das citadas especifica-se as diferenças entre elas e onde a ferramenta proposta é melhor as demais.

•*Takkou* x Robozzle: a ferramenta *Takkou* apresenta algumas similaridades com a Robozzle, entretanto, a primeira possui características pedagógicas fundamentadas em teoria, o que não é o caso da segunda, conforme o autor nos informou via e-mail. Além disso, ao contrário do robozzle, a ferramenta *Takkou* apresenta um ambiente potencialmente significativo no que concerne aos símbolos do dia a dia do usuário;

•*Takkou* x Light-Bot: por ser a Light-Bot similar a Robozzle, pode se reiterar o

que já foi afirmado no item anterior. Além das similaridades a ferramentas citada não possui um ambiente muito intuitivo, e se propõe a ser apenas um jogo, mesmo que abordando de forma direta conceitos de algoritmos;

•*Takkou* x Scratch: a grande diferença entre estas duas ferramentas se encontra nas teorias pedagógicas utilizadas em cada uma e no seu propósito. A ferramentra Scratch, como já fora citado, possui base pedagógica no construtivismo, entretanto, a ferramenta *takkou* vai além por abordar uma teoria diferencial, a do Aprendizado Significativo. Além disso, a ferramenta Scracth se enquadra na categoria de ferramentas de uso geral, por não está diretamente ligada a utilização no ensino de algoritmos e não se apresenta como um jogo, mas como uma ferramenta que possibilita a criação de animações, vídeos, etc. Já a ferramenta *Takkou*, se apresenta ligada a teoria de David Asubel e com características de um jogo de puzzle, ou antes, uma jogo onde se resolvem *puzzles* voltados diretamente para o ensino de algoritmos mesmo que de forma tangencial;

•*Takkou* x as demais ferramentas citadas: como pôde-se observar as ferramentas de linguagem e simulação apresentam-se como ferramentas técnicas. Já a ferramenta *Takkou* se enquadra no ramo das ferramentas com características de jogo, e, por sua vez, mais motivadora, partindo para uma abordagem diferencial, não técnica, mas lúdica.

Portanto, como fora explanado, percebeu-se que a ferramenta proposta neste artigo possui uma diferencial as já existentes, e, por isso, além de agregar um grande fator motivacional, apresenta-se como uma nova alternativa ao auxílio do ensino de algoritmos de forma lúdica.

4. Takkou

Takkou é uma ferramenta lúdica voltada ao ensino e prática de algoritmos e programação. Nela é preciso compor um bloco de instruções que, quando executado, seja possível cumprir uma determinada tarefa representada por um puzzle. O objetivo é fazer com o que o jogador cumpra com a tarefa em um menor tempo possível, refletindo uma situação em que ele terá de pensar rápido no desenvolvimento da lógica na resolução de um dado problema.

Cada puzzle contém um mapa representando um terreno e os obstáculos em que o jogador, representado por um boneco, deverá capturar itens que contribuem tanto para acumulo de pontos quanto para o cumprimento da missão. Esse último tipo de item, representado por uma chave, deve ser capturado antes de o jogador ir em direção a um objeto de fim de jogo, representado por uma porta. Além da chave e da porta, outro fator que afeta a condição de fim de jogo é a presença de um temporizador, relógio, em que é definido um tempo para que o jogador complete a missão. Assim que o jogador começa a trabalhar na resolução do puzzle, ou seja, quando a missão é carregada, o relógio inicia uma contagem regressiva.

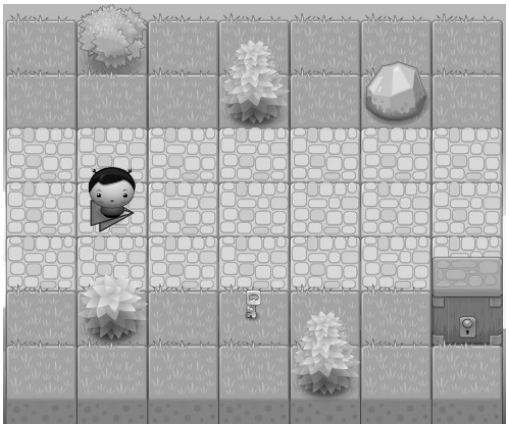


Figura 1 : Ilustra o mapa do jogo com o boneco representando o jogador e a chave e a porta como itens de fim de jogo.

A ferramenta foi projetada para possibilitar ao jogador uma interface simples e ao mesmo tempo eficiente. Para o que ela propõe, o jogador não deve ter dificuldades em compreender os elementos de interface assim como a sua manipulação, pois ele terá que focar apenas na missão do jogo.

Para fazer com que o *player*, boneco, se mova, deve-se montar uma estrutura formada por blocos de instruções. Uma vez inserido um bloco no painel, o mesmo pode ser movido de posição ou ser removido.



Figura 2: As opções de comandos (do lado esquerdo) podem ser utilizadas para inserir blocos de instrução no painel (do lado direito).

Os comandos são simples e fazem com que o *player* se mova em algumas direções como para frente, para o lado direito ou para o lado esquerdo. Além desses comandos há os comandos que permitem ao usuário checar uma dada condição de acordo com os elementos dispostos no mapa. É possível, por exemplo, verificar se há algum obstáculo no caminho em que se quer percorrer. E, por fim, é possível determinar quantas vezes determinado bloco de instrução deve ser executado através de um laço de repetição.

Durante a implementação dos comandos foram contrapostas duas abordagens em termos de eficiência. Levou-se em conta a seguinte questão: como fazer com que o jogador gaste pouco tempo manipulando os elementos de interface? Essa questão é bem colocada uma vez que a abordagem não deve afetar o tempo que se tem para cumprir a missão.

A primeira abordagem usa o artificio de *Drag and Drop*. Essa é a abordagem mais utilizada nos jogos de ensino de algoritmo. A sua maior vantagem é ser um mecanismo intuitivo. Em uma abordagem *Drag and drop* é dado ao usuário a

possibilidade de clicar em um botão, arrastá-lo e soltá-lo no painel de instruções. Muitas vezes, porém, o manuseio dos comandos torna-se um pouco demorado, o que acaba desmotivando o jogador e atrapalhando o objetivo da ferramenta, que no caso da *Takkou* é conseguir uma solução em um determinado tempo. Há, então, uma segunda abordagem que se baseia no uso do *scroll*, botão de rolagem do meio do mouse, para manipular os blocos de instrução. Apesar de ser menos intuitiva do que o uso do mecanismo de arrastar e soltar, ela se mostra mais adequada para a *Takkou*. Essa abordagem foi implementada disponibilizando um recurso de pré-visualização dos blocos no painel que é habilitado quando o mouse é passado por cima de algum comando. Para inserir um bloco no painel é necessário apenas um único movimento do *scroll*. A vantagem desta abordagem é que, quando não é movimentado o ponteiro do mouse, é possível deslocar o bloco recentemente inserido, mudando a sua ordem de execução. Assim, o jogador gastará menos tempo para deslocar um bloco. Na abordagem *Drag and drop* ele teria de mover o ponteiro do mouse até o bloco que se queira mudar de posição e arrastá-lo.

A Figura 3 mostra o funcionamento desta abordagem. Na situação **A** o mouse encontra-se apenas sobre o botão correspondente a instrução de “seguir em frente” e por isso o bloco é, temporariamente, inserido no painel até que o mouse saia da área do botão (situação **B**). Na situação **C** o *scroll* é movido uma vez e o bloco é inserido definitivamente no painel. Quando movido o *scroll* do mouse mais uma vez, a ordem do bloco em foco no painel é mudada de acordo com o sentido em que o scroll foi movido – para cima ou para baixo. Essa ação é representada na situação **D**.

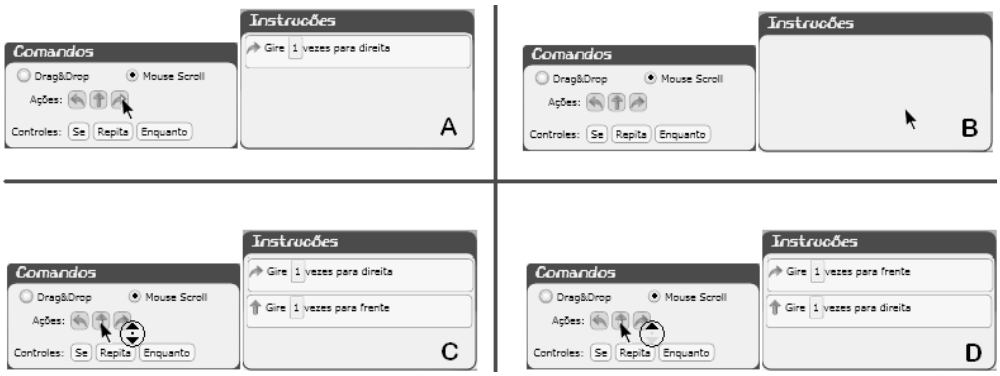


Figura 3: Ilustra as quatro situações possíveis com o uso do mouse *scroll*.

4.1. Criação de mapas

A criação de mapas ou *puzzles* é outro diferencial na *Takkou* possibilitando a criação de uma missão. Ao criar um mapa o usuário pode definir todos os elementos que farão parte da missão. Desta forma, quem o cria pode definir qual nível de “programação” deseja-se explorar do jogador, podendo construir mapas mais complexos ou simples.

Para criar um mapa, primeiro, é escolhida a sua dimensão, após esta definiçãoo usuário deve construir os mapas que se dividem em: terreno, obstáculos e itens. No mapa de terreno encontram-se disponíveis os tipos de terreno: água, terra, pedra ou grama; no mapa de obstáculo: pedra e arbustos; e, por fim, no de itens, que servem tanto para o jogador acumular pontos quanto para contribuir com as definições de fim de jogo, são definidos os itens: frutos, a chave e a porta.

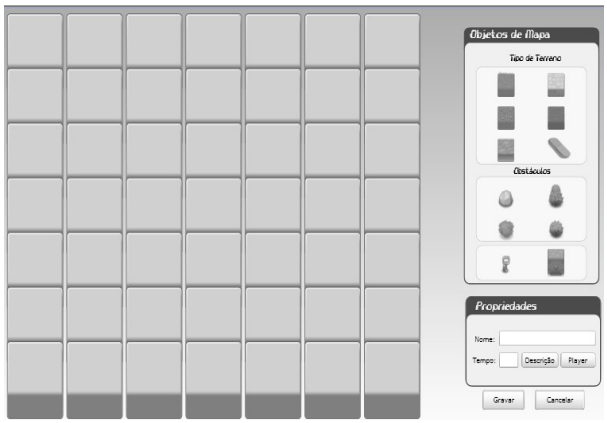


Figura 4: Página de construção de mapas de missão.

Na figura 4 ilustra-se a página de construção de mapas. A direita há as opções de itens que são usados para preenchê-lo. Os itens de fim de jogo são considerados obrigatórios, uma vez que sem eles o jogo não teria um objetivo a ser atingido. À esquerda está o mapa inicialmente sem nenhuma definição de preenchimento. Ao clicar em alguma opção de item (a direita) e em seguida clicar numa área do mapa, o item é incluso. Após a construção do mapa, pode-se inserir uma descrição da missão, que é ilustrada assim que o jogador a carrega; um nome e um tempo limite de conclusão da missão e a posição inicial do jogador.

5. Avaliação Preliminar

Em uma avaliação preliminar a ferramenta *Takkou* mostrou-se satisfatória frente ao fator motivacional. A isso atribui-se o modo como a ferramenta aborda os conceitos de algoritmos, de forma tangencial. Por ser uma ferramenta protótipo ainda em desenvolvimento, não foi possível realizar uma avaliação profunda do seu uso frente aos usuários.

Em trabalhos futuros, intenta-se aprimorar a ferramenta no que se refere aos elementos que a compõe tornando o uso direcionado através da especificação de níveis mais detalhado. Outra proposta para o futuro é adequar a ferramenta para ser disponibilizada para rodar em celulares. Dessa forma, possibilitando aos usuários fazer uso da ferramenta em qualquer lugar e a qualquer hora, retirando, deste modo, a obrigatoriedade do usuário está conectado em um computador para treinar suas habilidades.

6. Conclusões Finais

A ferramenta *Takkou* se apresenta como uma alternativa diferenciada das demais como forma de auxilar os alunos de computação no exercício de suas habilidades algoritmos concernentes ao raciocínio lógico. Como já fora citado em Campos et al. (2011), a ferramenta surgiu como uma abordagem frente ao problema enfrentado por inúmeros alunos de sentirem dificuldades em raciocinar logicamente na resolução de problemas. A forma como a ferramenta se apresenta foi proposta com o intuito de melhor motivar os seus usuários. Houve, também, a preocupação em torno da usabilidade por parte do usuário criando-se mecanismos de facilitação no manuseio da ferramenta como *drag and drop* e uso do botão central do mouse, *scroll*, para se adicionar comandos a serem

executados.

Além das características já citadas, a *Takkou* se distância das outras ferramentas por se apresentar com uma abordagem diferenciada de ensino e auxilio ao aprimoramento das habilidades na disciplina algoritmos. Por se apresentar como uma ferramenta com uma seria de jogos, *puzzles*, esta encoraja o usuário a aprender algoritmos sem tomar conhecimento que assim está fazendo, ao menos não de forma direta.

Por fim, acredita-se que no que se propõe esta ferramenta ela atende em sua totalidade os objetivos almejados: motivar os alunos a exercitar os conhecimentos de algoritmos; buscar conhecer mais sobre resolução de problemas; tornar a prática do exercício em algoritmos mais proveitosa e prazerosa.

Referencias

Campos, A. M., Barbosa, L. S. and Fernandes, T. C. B. (2011). A new approach for teaching algorithms . Submitted to The 2011 International Conference on Frontiers in Education: Computer Science and Computer Engineering.

Hinterholz, JR, O. (2009). Tepequém: uma nova Ferramenta para o Ensino de Algoritmos nos Cursos Superiores em Computação. XVII WEI – Anais do Workshop sobre Educação em Informática. Bento Gonçalves – RS.

Jesus, A. e Brito, G.S. (2009). Concepção de Ensino-aprendizagem de Algoritmos e Programação de Computadores: A Prática Docente. I ENINED - Encontro Nacional de Informática e Educação. Universidade Estadual do Oeste do Paraná (UNIOESTE). Cascavel, Paraná.

Maloney. J., Peppeler, K. and Kafai, Y. Resnick, M. Rusk, N. (2009). Programming by Choice: Urban Youth Learning Programming with Scratch. In ACM SIGCSE Bulletin archive, Vol 40, 367-371, ISSN:0097-8418, 2008.

Moreira, M.A. e Masini, E.F.S. (2006). Aprendizagem Significativa - A Teoria de David Ausubel. São Paulo: Editora Centauro.

Mota, M.P., Pereira, L.W.K. e Favero, E.L. (2008) . JavaTool: Uma Ferramenta para o Ensino de Programação. In: Congresso da Sociedade Brasileira de Computação. Belém. XXVIII Congresso da Sociedade Brasileira de Computação. p. 127-136.

Pelizzari, A. et al. (2002) Teoria da aprendizagem significativa segundo Ausubel. Rev.PEC, Curitiba, v.2, n.1, p.37-42.

Rocha, H. (1993). “Representações Computacionais Auxiliares ao Entendimento de Conceitos de Programação”, In: “Computadores e Conhecimento: Repensando a Educação”. Livro organizado por Valente, J. A. Editora Unicamp.

Santiago, R. (2004). Ferramenta de apoio ao ensino de algoritmos. XII Seminário de Computação. Blumenau/SC.

Souza, C.M. (2009). VisuAlg – Ferramente de Apoio ao Ensino de Programação. Revista TECCEN – Vol. 2, no.2; ISSN 1984-0993.

Robótica: aprendizado em informática de forma lúdica

Renato Ferreira Soares, Marcos Augusto Francisco Borges

Faculdade de Tecnologia – Universidade Estadual de Campinas (Unicamp)

Caixa Postal: 13484-332 – Limeira – SP – Brasil

Resumo. *Este resumo de artigo apresenta a robótica como uma ferramenta para o auxilio no aprendizado em informática. O artigo apresenta experiências de uso da robótica e os resultados obtidos.*

Abstract. *This Short-Paper presentsrobotics as a tool to aid learning in informatics. This paper demonstrates the experiences of the useof robotics and the results obtained.*

1.Introdução

Há uma grande demanda por profissionais que atuem na área de computação. Este é um problema difícil de resolver devido à dificuldade envolvida no processo de aprendizagem. O aprendizado em programação pode ser bastante complexo, chegando a ser desmotivador (Dijkstra, 1989).Pesquisadores estudam metodologias para ensinar informática de uma forma motivadora. Seymour Papert foi um dos precursores nesta área. Ele criou o Logo, um *software* livre(Solomon & Papert, 1976), no qual pessoas poderiam a partir das linhas de código que escreviam movimentar um objeto. A idéia principal do Logo é oferecer a oportunidade a crianças de aprender programação de uma forma lúdica e motivadora(Solomon & Papert, 1976).Outra pesquisa feita, utilizando-se de robótica para ensinar conceitos de informática, concluiu que a utilização de robótica é satisfatória e motivadora para o ensino de computação (Williams, 2003). Por outro lado, há trabalhos que indicam que o ensino de informática não pode ser feito apenas com robótica, utilizando-a como única fonte de aprendizado, pois ensinar informática somente utilizando a robótica pode não ser positivo (Fagin & Merkle, 2003).

Baseados nestes estudos, este artigo apresenta experiências com a robótica como uma ferramenta complementar no aprendizado de programação.A robótica, neste contexto, foi utilizada como uma ferramenta complementar no processo de aprendizado. A hipótese é que, com robótica, mantém-se a motivação, diminuindo a dificuldade com o aprendizado. Serão apresentadas dinâmicas com o intuitode aumentar o interesse dos alunos.

Este trabalho está estruturado como se segue: a seção 2 apresenta as plataformas utilizadas ao longo da pesquisa, a seção 3 apresenta exemplos de dinâmicas conduzidas e a seção 4 conclui.

2. Plataformas robóticas

Uma das plataformasanalisadas neste trabalho é o LEGO *MINDSTORMS*(LEGO, 1998). LEGO *MINDSTORMS* é um "brinquedo" programável, formado por um controlador lógico, sensores e diversas peças que se encaixam em um padrão LEGO. Esta plataforma oferece a oportunidade de montar diversos tipos de robôs de forma bastante intuitiva, com peças de encaixar. A plataforma LEGO oferece um ambiente para programação formado por uma linguagem visual e em blocos, adequada para crianças e

jovens sem conhecimento na área de programação. Com o LEGO *MINDSTORMS*, pode-se usar todas as estruturas básicas de programação. Um problema para a utilização é seu alto custo no Brasil. O valor de um único kit é de R$ 1999,00 (BrinquedosMania, 2009).

Uma alternativa estudada por este trabalho é a utilização de plataformas robóticas de baixo custo, utilizando-se placas *open-hardware*(Rubow, 2008) e sucata. Placas *open-hardware* possuem toda a documentação para sua construção disponível, tornando-as assim facilmente reproduzíveis. Existe uma placa *open-hardware*, chamada Arduino(Brock, Bruce, & Reiser, 2009). A Arduino é bastante utilizada, e um dos possíveis motivos do seu sucesso é o fato de sua programação ser inteiramente feita em C e C++, linguagens bastante conhecidas. A Arduino possui um ambiente próprio e gratuito de desenvolvimento e o seu custo é baixo (por volta de R$ 100,00), sendo mais acessível a diversos usos (Cuartielles, 2005).

Um exemplo de plataforma feita por este trabalho para avaliar o potencial de uso de materiais de baixo custo e sucata é um elevador (Figura 1.a, Figura 1.b), controlado por uma placa Arduino. Este elevador demonstra para os aprendizes uma plataforma inserida no cotidiano. Podem ser programados diversos algoritmos associados à plataforma de *hardware*, desde a utilização mais simples, onde o usuário iria pressionar um botão em um andar e o elevador se desloca até lá. Ou programas mais complexos, utilizando-se de estruturas de dados, para que o usuário possa apertar diversos botões em vários andares, e o algoritmo selecionaria o caminho ótimo para percorrer todos os andares.

3. Dinâmicas

As dinâmicas realizadas com alunos do ensino superior em informática seguiram a metodologia *ProblemBased Learning* (PBL) (Bellström & Kilbrink, 2010). Esta metodologia sugere que o responsável pela criação do aprendizado são os próprios alunos. Na PBL, o professor age como um facilitador e também selecionao problema aser resolvido. Este problema precisa proporcionar a necessidade de manipular diversos conceitos e trabalhar diretamente a motivação dos alunos. Durante as dinâmicas conduzidas foram propostos dois problemas aos alunos.

No primeiro problema, foram formados três grupos. Eles deveriam construir um robô móvel utilizando a plataforma LEGO *MINDSTORMS*. O objetivo destes robôs era simular uma batalha (Figura 1.c). Para que esta batalha acontecesse, foi delimitada uma área com uma faixa preta no chão, onde os robôs seriam colocados. Quem venceria a batalha seria o robô melhor programado. O robô deveria conseguir se manter dentro da faixa e localizar os outros robôs. Este problema foi proposto aos grupos para que eles se familiarizassem com a robótica e alguns conceitos que poderiam precisar para o segundo problema

No segundo problema foram propostos três desafios, um para cada grupo: o primeiro grupo deveria desenvolver um robô móvel que localizava “bolas de papel” em um ambiente fechado e as coletava. O segundo deveria desenvolver uma estrutura mecânica para um jogo chamado “Jogo da Fábrica” que é integrado a uma placa Arduino (Bortholotto & Borges, 2009). O terceiro desafio proposto foi a criação de um robô móvel que foi integrado via uma placa Arduino a um jogo chamado “Mundo do Robô” (Bortholotto & Borges, 2009). Os alunos estavam livres para resolver os

problemas da forma que lhes parecesse mais adequada. Foi dada uma base introdutória aos conceitos de programação. Todos os alunos utilizaram materiais de baixo custo ou sucata na resolução dos desafios. São exemplos de materiais utilizados: cantoneiras, monitores de computador não funcionais, motores de brinquedos antigos, etc. Este problema explora diversos conceitos.Um exemplo é a necessidade de avaliar a integração *hardware/software* durante todo o processo de criação da solução, pois só assim é possível queo resultado contemple as restrições de cada plataforma. Este conceito é difícil de ser repassado com o ensino tradicional, pois conectar duas plataformas gera um tipo de dificuldade mais simples de ser observado em uma atividade prática do que explicado em uma aula teórica.



Figura 1. (a , b) Elevador construido com sucata, (c) Batalha de robôs durante as dinâmicas

Estes problemas tiveram como objetivo facilitar o aprendizado de programação e integração *hardware/software*. Um dos resultados almejados com esta dinâmica foi mostrar para os alunos que eles têm potencial para construir seu próprio conhecimento, estando aptos para aprenderem e se adaptarem a qualquer tipo de linguagem que forem utilizar.As dinâmicas ocorreram em um período de um semestre. Os resultados mostraram ser satisfatóriaa utilização de robótica como ferramenta educacional. As análises foram desenvolvidas a partir de uma avaliação sobre os resultados das dinâmicas, na visão dos alunos, realizada no término do semestre. Também foram consideradas observações realizadas pelo professor durante as dinâmicas.Participaram das dinâmicas,treze alunos de ensino superior em informática. Os resultados das avaliaçõesque eles preencheram mostram que quatro destes alunosindicaram se sentirem motivados, doze desafiados e dois curiosos (podendo marcar múltiplas opções). Dois destes alunos acreditavam que dinâmicas como estas são essenciais, dez que são importantes e um que a dinâmica foi interessante, mas não seria mais utilizada.

4. Conclusão

Durante as dinâmicas foi possível mostrar para os alunos o potencial que possuem para gerar seu próprio aprendizado. Em todo o processo de criação de uma plataforma robótica o aluno é desafiado a entender novos conceitos e descobrir novas formas de solucionar os problemas propostos.

Este trabalho utilizou-se de diferentes plataformas. Assim, os alunos foram incentivados a procurar este conhecimento de uma forma mais ampla. Em outras dinâmicas citadas, os trabalhos se restringiam a uma plataforma robótica específica, com componentes de simples manipulação.

Também foi possível mostrar aos alunos conceitos que dificilmente são adquiridos em sala de aula. Um exemplo é o conceito de integração de plataformas, normalmenteaprendido apenas com a experiência profissional. Trabalhando conceitos

como esses de forma lúdica e motivadora, as dinâmicas com robótica mostram que podem ser úteis, como ferramenta de apoio ao aprendizado.

Os resultados das dinâmicas realizadas mostram que a robótica pode ser uma forma de se complementar o processo de ensino de informática. A robótica pode apoiar os professores, facilitando o aprendizado dos alunos e também o entendimento dos conceitos ensinados em sala de aula.

5.Bibliografia

Bellström, P., & Kilbrink, N. (2010). Problem-Based Learning in a Programming Context– Planning and Executing a Pilot Survey on Database Access in a Programming Language. In: G. A. Papadopoulos, W. Wojtkowski, G. Wojtkowski, S. Wrycza, & J. Zupancic, *Information Systems Development* (pp. 867-875). Springer US.

Bortholotto, J. C., & Borges, M. A. (2009). *Integração de dispositivos robóticos a sistemas de apoio ao aprendizado utilizando a plataforma Arduino.* Relatório final de projeto de iniciação científica: FT - Universidade Estatual de Campinas.

BrinquedosMania. (2009). *Brinquedos Mania.* Acesso em 03 de 04 de 2011, http://brinquedosmania.com.br

Brock, J. D., Bruce, R. F., & Reiser, S. L. (2009). Using Arduino for introductory programming courses. *Journal of Computing Sciences in Colleges* , 129-130.

Cuartielles, D. (21 de Abril de 2005). *Arduino.* Acesso em 01 de Abril de 2011, disponível em Arduino: http://arduino.cc/en/

Dijkstra, E. W. (1989). On the Cruelty of Really Teaching Computing Science. *Communication of ACM* , 1398-1404.

Fagin, B., & Merkle, L. (2003). Measuring the effectiveness of robots in teaching computer science. *SIGCSE '03 Proceedings of the 34th SIGCSE technical symposium on Computer science education* (pp. 307-311). Reno: ACM.

LEGO. (1998). *Lego Mindstorms*. Acesso em 20 de 03 de 2011, disponível em LEGO: http://mindstorms.lego.com/en-us/Default.aspx

Rubow, E. (20 de 11 de 2008). *Open Source Hardware*. Acesso em 03 de 04 de 2011, disponível em Cseweb.ucsd.edu: http://cseweb.ucsd.edu/classes/fa08/cse237a/topicresearch/erubow_tr_report.pdf

Solomon, C. J., & Papert, S. (1976). A case study of a young child doing turtle graphics in LOGO. *AFIPS '76 Proceedings of the June 7-10, 1976, national computer conference and exposition* (pp. 1049-1056). New York: ACM.

Williams, A. B. (2003). The qualitative impact of using lego mindstorms robots to teach computer engineering. *IEEE Transaction on Education* , 206.

Gestão de Direitos Autorais em Educação a Distância na Área de Computação

Dilermando Piva Jr.[1,2], Joni de Almeida Amorim[3]

[1]Faculdades de Tecnologia do Estado de São Paulo e Universidade Virtual do Estado de São Paulo (UNIVESP) – São Paulo, SP, Brasil.

[2]Pontifícia Universidade Católica de Campinas (PUC-Campinas) – Campinas, SP, Brasil.

[3]Universidade Estadual de Campinas (UNICAMP) – Campinas, SP, Brasil.

PivaJr@gmail.com, Joni.Amorim@reitoria.unicamp.br

Abstract. *Increasingly, the use of multimedia both in face-to-face education and in distance education turns to be a reality. Both students and teachers benefit from the falling costs of hardware, from the increase in the bandwidth and from the easiness to share and reuse digital media. In this sense, it becomes essential to develop policies and strategies for the management of copyright. While focusing on the computing area, this study presents the essential elements to be considered. This research summarizes the practice of the authors while participants on different projects from public institutions. The projects demanded in-depth research on copyright and suggested relevant strategies.*

Resumo. *Cada vez mais, o uso de multimídia na educação presencial e na educação a distância passa a ser uma realidade. Alunos e professores se beneficiam da queda nos custos de hardware, do aumento da largura de banda e da facilidade de compartilhamento e reutilização de mídia digital. Nesse sentido, torna-se essencial desenvolver políticas e estratégias para a gestão dos direitos de autorais. Com foco na área de computação, este estudo apresenta os elementos essenciais a serem considerados. Esta pesquisa resume a prática dos autores enquanto participantes em diferentes projetos de instituições públicas. Os projetos exigiram investigação em profundidade sobre direitos autorais e sugeriram estratégias pertinentes.*

1. Introdução

A melhoria do processo de ensino e de aprendizagem em Computação passa tanto pela inovação curricular como pela incorporação de novas estratégias, meios e mídias ao contexto da Educação Superior. No momento atual, ao mesmo tempo em que se discutem as Diretrizes Curriculares (http://formularios.mec.gov.br/consulta-diretrizes-curriculares) dos cursos de Bacharelado em Ciência da Computação, Engenharia de Computação, Engenharia de Software e Sistemas de Informação, assim como dos cursos

de Licenciatura em Computação, também se discutem as inovações relativas à incorporação de multimídia à educação, seja a educação presencial face-a-face, seja a educação a distância amparada por tecnologias como a Internet.

Tal contexto tende a favorecer a produção em maior escala de multimídia, seja para o apoio à educação presencial, pelo seu enriquecimento, seja para a viabilização da educação a distância, pela potencialização do uso de recursos baseados em áudio e em vídeo, dentre outros. Este favorecimento é reflexo de uma característica da produção de multimídia: custos fixos altos e custos variáveis baixos. Em outras palavras, quanto maior a similaridade entre os cursos da área de Computação, maior o potencial de reutilização de produtos multimídia como áudios, vídeos, softwares com animações e simulações, jogos educacionais, imagens animadas e estáticas, textos e hipertextos, etc. Este potencial de reutilização se torna relevante no momento em que muitas instituições passam a compartilhar seus produtos multimídia entre si e/ou com todos aqueles ligados à rede mundial. Tal compartilhamento pode ocorrer, por exemplo, através de repositórios específicos onde os produtos podem ser encontrados e carregados ("download") com relativa facilidade.

Neste cenário de uso crescente de multimídia em uma perspectiva de compartilhamento, torna-se fundamental a gestão de direitos autorais. Como é sabido, os produtos multimídia podem ter todos os direitos reservados ou, de maneira a potencializar o intercâmbio e a distribuição, podem ter apenas alguns direitos reservados. Assim sendo, uma organização pode tanto optar por ter todos os direitos reservados sobre seus produtos, o que lhe permitiria comercializá-los, como pode também incentivar os seus colaboradores a disponibilizar estes produtos com apenas alguns direitos reservados. Quando apenas alguns direitos estão reservados, pode ser possível, por exemplo, reutilizar e até mesmo modificar um determinado produto desde que certos direitos, como o de autoria original do produto, sejam preservados. Com isso, dentre outras possibilidades, um autor de um podcast de áudio ou vídeo poderia, por exemplo, disponibilizar todo o seu material na Internet exigindo somente que seu nome fosse mencionado como autor original deste material.

Tomando por base estas novas possibilidades que hoje surgem, inclusive com a produção crescente de material multimídia para a área de computação, este artigo pretende discutir a gestão de direitos autorais em educação a distância. Tal discussão é, certamente, de interesse não apenas aos contextos de produção em larga escala, como aqueles onde políticas públicas fomentam a produção com o financiamento de projetos de produção, mas também aos contextos de menor escala, onde mesmo um docente da Educação Superior pode optar por gerar material multimídia que seria publicado na Internet. Mesmo nos casos em que um docente não visa lucro com sua produção, há o interesse pela preservação de seu direito de autor, com citações ao seu trabalho tal qual ocorre com outras produções, como artigos para eventos ou capítulos para livros.

Em um certo sentido, tal discussão é análoga a uma outra discussão já bastante conhecida na área de computação, neste caso a que se refere ao desenvolvimento de software livre, muitas vezes gratuito, em contraposição ao software proprietário, quase sempre pago e com uso autorizado mediante licenças bastante restritivas. Ainda que a discussão apresentada neste texto não substitua uma consulta à legislação vigente, a qual é bastante detalhada, contemplando casos bastante específicos, pretende-se que este trabalho contribua enquanto referência preliminar a ser utilizada pelos autores de

materiais instrucionais. Os resultados aqui apresentados refletem as investigações realizadas pelos autores em sua prática na Educação Superior enquanto integrantes de projetos de pequeno e de grande porte em organizações como as Faculdades de Tecnologia do Estado de São Paulo, a Universidade Virtual do Estado de São Paulo (UNIVESP), a Pontifícia Universidade Católica de Campinas (PUC) e a Universidade Estadual de Campinas (UNICAMP).

2. Creative Commons e a Visão Contemporânea da Propriedade Intelectual

A literatura (BRASIL, 2008) define diferentes conceitos hoje importantes, como o de Propriedade Intelectual, que seria um ramo do Direito que trata da propriedade dos bens imateriais ou incorpóreos resultantes da manifestação intelectual do ser humano, englobando: (1) Propriedade Industrial, (2) Direito Autoral e outros Direitos sobre bens imateriais de vários gêneros, tais como (3) Direitos Conexos e (4) Proteções Sui Generis. Por Propriedade Industrial se entende o conjunto de direitos que compreende as patentes de invenção e de modelo de utilidade, os registros de desenho industrial, as marcas e as indicações geográficas, bem como a repressão da concorrência desleal.

O Direito Autoral (BRASIL, 2008), por sua vez, considera as obras intelectuais, redutíveis à noção de artístico ou literário, assim como aquelas de caráter puramente científico, qualquer que seja seu modo de expressão. Alguns exemplos seriam (a) obras literárias, como livros, textos e poesias, (b) obras artísticas, como composições, fotografias e artes cinematográficas, (c) obras científicas, como projetos, cartas geográficas e programas de computador, (d) conferências, sermões e outras obras de mesma natureza e (e) adaptações, traduções e outras transformações de obras originais. Neste caso, a proteção está disciplinada na Lei nº 9.610, de 19 de fevereiro de 1998, que altera, atualiza e consolida a legislação sobre direitos autorais e dá outras providências. Ainda assim, no caso de programas de computador (software), a proteção está disciplinada na Lei nº 9.609, de 19 de fevereiro de 1998.

No caso de Direitos Conexos, busca-se proteger a pessoa jurídica ou física que contribui para tornar as obras autorais acessíveis ao público. No que se refere às Proteções Sui Generis, temos "híbridos jurídicos", ou figuras jurídicas intermediárias entre a Propriedade Industrial e o Direito Autoral, resultado do surgimento de novas criações intelectuais, como topografia de circuitos integrados, cultivares vegetais e conhecimentos tradicionais associados aos recursos genéticos. A patente, enquanto título de propriedade temporário outorgado pelo Estado, é válida apenas nos países onde foi requerida.

A Lei nº 9.279, de 14 de maio de 1996, que regula direitos e obrigações relativos à propriedade industrial, indica que pode ser patenteada a invenção que atenda aos requisitos de novidade, atividade inventiva e aplicação industrial. Deste modo, ficam excluídos, dentre outros, descobertas, teorias científicas e métodos matemáticos, assim como esquemas, planos, princípios ou métodos comerciais, contábeis, financeiros, educativos, publicitários, de sorteio e de fiscalização. Também se excluem as obras literárias, arquitetônicas, artísticas e científicas ou qualquer criação estética assim como programas de computador em si.

No caso específico de programas de computador, o código-fonte é protegido por direito autoral, não sendo patenteável; somente uma criação industrial que faça uso de programas de computador, como uma máquina, um processo ou um sistema, poderia vir

a ser patenteada. Internacionalmente, para o caso específico de software livre, merece destaque a iniciativa "Free Software Foundation" (FSF - http://www.fsf.org/), que pretende promover o desenvolvimento e o uso de software livre assim como pretende promover o desenvolvimento da documentação concernente.

A discussão acima deixa evidente a complexidade relativa à proteção da propriedade intelectual. No ambiente acadêmico, a temática da propriedade intelectual tem ganhado destaque por diferentes motivos, mas em especial pelo fato de hoje existirem iniciativas importantes focadas no livre acesso a diferentes obras, incluindo-se aí obras educacionais. Um exemplo é o da iniciativa internacional denominada "Creative Commons" (CC - http://creativecommons.org/), com licenças que permitem a autores de vários tipos de obras, como textos, imagens, áudios e vídeos, publicar suas obras na Internet por conta própria nos locais de sua preferência indicando quais direitos estão reservados. Os seis principais tipos de licença seriam: "Attribution", que exige menção à obra original mas permite desde distribuição até a realização de alterações; "Attribution Share Alike", que permite desde distribuição até a realização de alterações, desde que a nova criação seja licenciada nos mesmos termos; "Attribution No Derivatives", que não permite alterações na obra mas permite redistribuição com fins comerciais ou não; "Attribution Non-Commercial", que permite desde distribuição até alterações, mas com a restrição de que a nova obra seja não comercial; "Attribution Non-Commercial Share Alike", que exige os mesmos termos nas licenças das novas obras, sendo o uso não comercial; e "Attribution Non-Commercial No Derivatives", que seria o tipo de licença mais restritivo, pois não permite alterações ou uso comercial, permitindo somente a distribuição citando o autor. Um dos principais exemplos de organização que utiliza licenças CC seria o "Massachusetts Institute of Technology" (MIT - http://web.mit.edu/), com a iniciativa "MIT OpenCourseWare" (http://ocw.mit.edu/), com uso de licenças "Attribution", "Noncommercial" e "Share Alike". No Brasil, merecem destaque os portais governamentais, como o Banco Internacional de Objetos Educacionais do Ministério da Educação (http://objetoseducacionais.mec.gov.br/).

Também tem merecido grande destaque no meio acadêmico a discussão sobre inovação e sobre o registro da propriedade intelectual decorrente da inovação. A importância de tal discussão se reflete na legislação, como exemplifica a chamada Lei Paulista de Inovação (http://www.al.sp.gov.br/legislacao/norma.do?id=98136), ou Lei Complementar Nº 1049, de 19 de junho de 2008, a qual dispõe sobre medidas de incentivo à inovação tecnológica, à pesquisa científica e tecnológica, ao desenvolvimento tecnológico, à engenharia não-rotineira e à extensão tecnológica em ambiente produtivo, no Estado de São Paulo, e dá outras providências correlatas. Tal Lei passa a permitir, por exemplo, a atuação de pesquisadores do setor público nos setores da produção, inclusive com a prestação de consultoria técnico-científica.

Seja como for, da mesma forma que hoje se percebe um interesse crescente das Instituições de Ensino Superior no que se refere à proteção de sua Propriedade Intelectual, inclusive através de patenteamento e de proteção de programas de computador no contexto de colaborações Universidade X Empresa, deve existir reciprocidade, com as Faculdades e Institutos respeitando a Propriedade Intelectual relativa a livros, a filmes, etc. Deste modo, o uso não autorizado de cópias de produtos multimídia, ou mesmo de simples trechos de textos de livros, deve ser desincentivada. Assim sendo, é de se esperar que o controle da Propriedade Intelectual passe a ter

importância crescente no meio acadêmico conforme se desenvolve a chamada Sociedade da Informação.

3. DRM e Multimídia

Segundo Paranaguá e Branco (2009), apenas no século XVI começaram a ser atribuídas licenças aos livreiros para que publicassem determinados livros. Em 1710, foi publicado o notório Estatuto da Rainha Ana, que concedia aos editores o direito a cópia de determinada obra pelo período de 21 anos. Segundo Paranaguá e Branco (2009), "... somente em 1886 é que surgiram as primeiras diretrizes para a regulação ampla dos direitos autorais." No Brasil, segundo Manso (1987), a primeira disposição legal data de 1827. A Lei 9.610, de 1998, "regula os direitos autorais, entendendo-se sob esta denominação os direitos de autor e os que lhes são conexos", sendo uma das principais referências atualmente, ainda que também seja relevante considerar em certos casos as especificidades das legislações sobre marcas, sobre software etc.

No que se refere ao direito autoral, cabe notar que a gestão de direitos é uma área vasta que hoje inclui aspectos afins aos meios digitais. Moskowitz (2006), ao tratar especificamente de gestão de direitos digitais, salienta a diferença entre as propriedades mais tradicionais, como imóveis e bens físicos, e as propriedades intelectuais, as quais se referem a itens que de modo geral podem ser compartilhadas com facilidade, gerando problemas hoje já comuns, como a pirataria de produtos digitais, destacando-se neste caso músicas e filmes.

Por diferentes razões, discussões sobre "Digital Rights Management" (DRM), ou Gestão de Direitos Digitais, demandam considerações sobre o contexto no qual uma obra é produzida ou distribuída. De modo geral, os criadores de uma obra a produzem com o intuito de comercialização, fato que leva à cobrança de direitos de cópia. Nos casos em que não se visa o lucro com a obra, de modo geral se pretende que a distribuição da mesma cite seu autor original. Independentemente da existência de intuito comercial, a legislação prevê direitos para o autor de um trabalho original, inclusive pelo fato de ser necessário algum controle sobre a criação de trabalhos derivados com base em um determinado trabalho original.

Com o advento da Internet, surgem a cada dia diferentes serviços de compartilhamento de arquivos que tornam cada vez mais complexo o controle do direito autoral, fato que tem levado ao desenvolvimento de tecnologias específicas que permitam rastrear a autoria de obras através de soluções como marcas d'água digitais. Ainda assim, o conceito internacionalmente disseminado de "fair use", ou uso aceitável, gera desafios adicionais, dado que muitas vezes não é evidente o que poderia ser visto como uso aceitável. Para tanto, Moskowitz (2006) indica que são quatro os fatores principais a serem considerados para se determinar se uma determinada utilização de uma obra seria ou não aceitável: (1) o propósito da utilização, o que inclui diferenciar o uso comercial do uso educacional em contextos específicos; (2) a natureza do trabalho original; (3) a proporção, ou porcentagem, do trabalho original que se utilizará; e (4) o efeito da utilização de um trabalho original relativamente ao mercado potencial para tal trabalho, o qual seria copiado totalmente ou parcialmente.

Moskowitz (2006) salienta que a determinação do que seria uso aceitável é algo tão dependente do contexto que simplesmente citar os autores originais de uma obra não

seria suficiente, sendo indicado que em cada caso específico fossem pedidas autorizações aos detentores dos direitos. Tal perspectiva trás diferentes restrições a projetos envolvendo a produção e/ou a utilização de multimídia dado que a produção de novos produtos que não sejam 100% originais demandaria autorizações para o uso de cada obra citada ou adaptada enquanto a utilização de produtos multimídia existentes demandaria autorizações para reprodução. Também no que se refere a multimídia, Moskowitz (2006) ressalta que considerações sobre a segurança devem ter em mente certos aspectos: (1) multimídia é compressível e facilmente transferível; (2) a digitalização de material em formato analógico é cada vez mais rápida e barata; e (3) a manufatura de material digital tem se tornado mais barata, aumentando as margens de lucro das organizações, mas ao mesmo tempo também permite que não detentores dos direitos copiem de maneira descontrolada, o que pode diminuir as margens de lucro. Tal autor sugere que a maioria dos setores econômicos deve sofrer algum efeito das mudanças relativamente à convergência da base tecnológica para a digital, visto que cada vez mais não apenas músicas e filmes, mas também livros, artigos e todo tipo de materiais passam a ser disponibilizados em formato eletrônico através de redes como a Internet.

4. Uma Breve Orientação para Autores de Conteúdos EaD

As obras intelectuais são aquelas criadas pelo ser humano e expressas ou publicadas por um determinado meio de comunicação a fim de que sejam conhecidas. Ainda assim, pretende-se algum controle relativamente ao acesso a tais obras, em especial no caso da existência de lucro potencial. A Lei de Direitos Autorais, ou Lei n. 9.610 de 19 de fevereiro de 1998, relaciona os tipos de obras intelectuais protegidas pelo direito autoral, sendo importante ressaltar que as mesmas são tão somente exemplificativas, não se limitando a elas. Assim, segundo Gandelman (2004), são elas texto, conferências, obras dramáticas, obras coreográficas, obras audiovisuais, obras fotográficas, desenhos, pinturas, gravuras, ilustrações, mapas, obras plásticas, traduções com criação intelectual nova, programas de computador, coletâneas, dicionários, dentre outras obras possíveis. Uma vez que sabemos quais são as obras protegidas, cabe ressaltarmos quais são os cuidados necessários que os autores de conteúdos digitais e/ou instrucionais devem tomar no seu trabalho. Cabe registrar que os programas de computadores estão protegidos na lei n. 9.609 de 19/02/1998 (artigo 2º).

Com o objetivo de orientar os autores de conteúdos digitais e/ou autorais, principalmente para a EaD na área de Computação, a seguir são sugeridas algumas ações e cuidados que os autores devem tomar no decorrer do desenvolvimento de seus trabalhos. São sugeridas situações ideais, aceitáveis e aquelas em que uma autorização se faz imprescindível.

De início, temos a situação “ideal”, relativa a domínio público e afins. No trabalho de desenvolvimento de conteúdos EaD, o ideal é utilizar obras de terceiros que sejam distribuídas sobre licenças CC (Creative Commons – http://www.creativecommons.org.br/) ou que já estejam em Domínio Público. Para o caso de fotos, por exemplo, existem portais como o Flickr (http://www.flickr.com/creativecommons/), onde existem mais de 20 milhões de fotos disponíveis para reprodução ou mesmo para a criação de obras derivadas. Exemplos adicionais de portais que utilizam licenças CC podem ser encontrados acessando

diretórios de livre acesso na Internet como o que é mantido pela organização responsável pela elaboração dos padrões das licenças Creative Commons (http://wiki.creativecommons.org/Content_Directories).

No Brasil, o principal portal com material de livre utilização é o Portal Domínio Público (http://www.dominiopublico.gov.br/). Também são de domínio público obras cuja última versão é muito antiga e/ou, cujos autores já faleceram faz mais de 70 anos. A seguir, são apresentados alguns exemplos para um melhor entendimento desta problemática:

- Exemplo 1: uma partitura de Mozart já se tornou de domínio público, mas uma interpretação recente por uma orquestra não é de domínio público.
- Exemplo 2: uma poesia de Fernando Pessoa já se tornou de domínio público, mas uma recitação recente por um ator em um filme não é de domínio público.
- Exemplo 3: uma escultura grega em um museu pode ser fotografada livremente, mas se um fotógrafo ainda vivo tiver tirado uma foto de tal escultura, ninguém pode reutilizar ou alterar tal foto sem a autorização do referido fotógrafo.

Assim, para os casos de material sob domínio público ou sob licença CC, quase sempre basta citar a fonte de onde a obra foi obtida, porém respeitando-se restrições eventualmente indicadas. Para os demais casos, uma análise mais cuidadosa deve ser feita antes que a obra seja utilizada parcialmente ou totalmente, como se discute a seguir.

Em segundo lugar, temos a situação "aceitável", relativa às partes de obras. A legislação brasileira ainda não está atualizada a ponto de considerar todos os casos possíveis, em especial no que se refere a direitos morais, autorais, patrimoniais e de distribuição via Internet. Assim, os advogados costumam trabalhar com analogias quando os casos não estão previstos em Lei. No caso dos materiais didáticos produzidos em múltiplas mídias, como software, hipertextos ou vídeos, quase sempre podem ser vistos como materiais educacionais destinados ao trabalho acadêmico e sem fins lucrativos. Nestes casos, a lei permite o uso de partes de obras de terceiros, desde que tal uso não gere prejuízos aos detentores dos direitos morais, autorais, patrimoniais e de distribuição.

Seja qual for o caso, deve ser citada a fonte de onde a obra foi retirada, citando o autor da obra, o ano de publicação da obra, etc. No Brasil, sugere-se o uso das normas ABNT (http://www.abnt.org.br/) mesmo nos casos em que o material seja obtido na Internet, neste caso citando-se em especial o endereço de onde se tirou o material e a data do acesso a esse material. Para fazer uso de partes de obras de terceiros sem pedir autorização, sugere-se usar o mínimo possível, tal como 10 segundos de um filme ou 10 segundos de uma música. Exemplos:

- Exemplo 4: um produto multimídia pode citar um pequeno trecho de um filme, de até 10 segundos, desde que se cite a fonte e de que não seja possível identificar quais são os atores; se o ator puder ser identificado visualmente, e estiver vivo, é preciso pedir autorização ao menos para o ator.
- Exemplo 5: um produto pode citar um pequeno trecho de uma letra de música, de até 10 segundos, desde que se cite a fonte e o autor da música.

No caso de textos, tem-se maior liberdade desde que nunca se use uma obra completa sem pedir autorização, evidenciando-se claramente na citação qual é o autor original da obra e explicando-se qual é o trecho exato que foi copiado. Exemplos:

- Exemplo 6: se uma poesia aparece publicada na Internet, é aceitável utilizar um pequeno trecho desde que se cite a fonte e a data do acesso a tal fonte.
- Exemplo 7: se um livro apresenta textos de interesse, a citação de uma parte dentro do produto multimídia é aceitável desde que seja citada a fonte.

Para os casos em que se queira fazer uso de uma grande parte de uma obra, ou mesmo da obra completa, é necessário pedir autorização, como se descreve a seguir.

Em terceiro lugar, temos a situação que "demanda" autorização, relativa ao uso de obras completas. O grande problema se refere ao uso de obras completas, pois neste caso é necessário pedir autorização aos detentores dos direitos. São exemplos de obras completas: (a) texto completo de uma poesia; (b) áudio ou vídeo com a recitação do texto completo de uma poesia; (c) texto completo de uma música; (d) áudio ou vídeo com o som relativo ao texto completo de uma música; (e) textos de um modo geral, mesmo que tenham sido publicados na internet; (f) capas de revistas, de livros ou de jornais; (g) tirinhas de histórias em quadrinhos, mesmo que com apenas um quadro; (h) fotografia, desenho, representação esquemática ou diagrama; (i) logotipos indicando visualmente as marcas de produtos e afins; e (j) peça publicitária que comparece como propaganda em uma revista, livro ou jornal.

Em todos os casos de obras completas de terceiros que não estejam em domínio público, é necessário pedir autorização tanto para uso no produto multimídia específico, quanto para sua posterior distribuição em meios como Internet, TV, CDs, DVDs, etc.

Aconselha-se que seja evitada a citação direta de nome de pessoas envolvidas em casos polêmicos ou situações desfavoráveis à sua imagem. Para tanto, pode-se citar a história, mas sem mencionar diretamente o nome de pessoas ou organizações envolvidas, evitando-se também a apresentação de imagens de pessoas vivas sem que essas autorizem. No caso de tirinhas de histórias em quadrinhos, a eventual solução é a de que a mesma seja refeita, utilizando softwares ou portais específicos como, por exemplo, o Stripgenerator (http://stripgenerator.com/strip/create/), para que a história original sirva tão somente de base para a criação de uma nova tirinha.

Na sua quase totalidade, projetos no âmbito público não contam com recursos para pagamento por autorizações para uso de obras de terceiros e/ou para a reutilização com vistas à criação de obras derivadas. Nestes casos, se os detentores do direito demandarem qualquer forma de pagamento, deve ser feita a opção pela não utilização da obra no produto multimídia em concepção, em especial caso tal produto vá ser distribuído digitalmente.

Uma observação importante se refere aos atalhos ("hyperlinks") para portais, páginas, "frames" específicos e arquivos. De maneira geral, deve-se buscar citar os portais ou, no máximo, páginas de portais onde estejam "frames" ou arquivos de interesse. Com isso, fica evidente ao usuário quais são os autores e/ou qual é a organização que realiza a distribuição do material de interesse. Ou seja, o ideal é colocar uma URL que leve o usuário de maneira explícita ao sítio onde oficialmente ocorre a distribuição da obra, como a página ou o portal. Com isso, um recurso sugerido é o da

abertura de janelas separadas com materiais de terceiros, mostrando a página ou o portal, mas não um "frame" específico ou um arquivo separado onde não se percebe qual o sítio de distribuição oficial.

5. Considerações Finais e Trabalhos Futuros

Os projetos de produção de produtos digitais/educacionais para EaD, na área de Computação, por suas especificidades, demandam a definição de uma política específica de propriedade intelectual que esteja em conformidade com as demais políticas da instituição educacional em questão.

No caso dos projetos de produção de multimídia, os principais cuidados a serem observados tendem a se referir à coleta de autorizações para uso de obras de terceiros antes da produção de produtos que façam uso de materiais com direitos reservados assim como à coleta das cessões de direitos autorais, patrimoniais e de distribuição dos integrantes das equipes de produção. Ou seja: é preciso ter autorizações para usar obras de terceiros e é preciso que aqueles envolvidos na produção de multimídia doem os direitos dos produtos que criaram à organização onde trabalham. Já no caso dos projetos envolvendo o uso de produtos multimídia, como por exemplo cursos a distância e cursos presenciais enriquecidos com uso de multimídia, o cuidado fundamental tende a se referir a conseguir as autorizações para tal uso, quando os direitos são reservados, ou a armazenar apropriadamente as informações que certificam que uma obra pode ser utilizada livremente, quando os direitos não estão necessariamente reservados.

Assim sendo, as considerações sobre propriedade intelectual devem comparecer desde as fases iniciais de cada projeto, com reflexos diretos na definição dos requisitos dos produtos multimídia a serem produzidos e no gerenciamento dos custos afins à produção e/ou à utilização. Pesquisas recentes (AMORIM e MISKULIN, 2010) indicam que seis ações fundamentais são necessárias na maioria dos contextos: (1) planejar a gestão da propriedade intelectual, (2) definir a política de propriedade intelectual, (3) identificar as autorizações necessárias, (4) adquirir as autorizações e/ou licenças de uso, (5) coletar os termos de cessão dos recursos humanos envolvidos na produção e (6) arquivar a documentação.

Com base na argumentação apresentada, este texto pretendeu contribuir para a discussão em torno do tema propriedade intelectual na perspectiva da produção e da utilização de produtos multimídia em educação a distância na área de computação, podendo ser extensível, com algumas adaptações, a outras áreas do saber. Percebe-se pela argumentação apresentada que se trata de um tema amplo e complexo que por certo demanda constante consulta à legislação vigente, a qual tende a ser atualizada com alguma freqüência com base na evolução tecnológica da sociedade.

Estabelecidas as estratégias e as políticas de direitos autorais de uma instituição educacional, surge uma questão adicional: Como selecionar produtos multimídia apropriados ao contexto da educação a distância na área de computação?

Por certo, uma tendência é a de que se prefiram produtos já disponíveis na rede mundial, preferencialmente em repositórios onde se permita tanto o uso como a distribuição. Opcionalmente, pode-se verificar que seria de interesse produzir novos produtos reutilizando partes de produtos existentes. Dessa maneira, a questão acima pode ser respondida com base em vários critérios, sendo um deles a existência ou não

de autorizações para reutilização de material de terceiros. Um outro critério se refere ao design instrucional de um determinado curso, já que certos tipos de produto, como áudio, podem ser mais adequados ao ensino de um tema do que outros, como vídeos ou simulações. Um critério adicional se refere ao grau de abstração envolvido, dado que vídeos e simulações, por exemplo, são menos abstratos que áudios e hipertextos.

De maneira geral, pesquisas recentes (AMORIM et al., 2011) indicam que materiais menos abstratos devem ser preferidos em um momento inicial no qual os alunos ainda não adquiriram uma familiaridade mínima com o conteúdo de interesse.

Na perspectiva de que a seleção de produtos mais apropriados a um certo contexto representa um problema relevante, pesquisas futuras poderão envolver o estabelecimento de critérios para a tomada de decisão relativamente à seleção de um produto em detrimento de outro, em especial caso se assuma que em breve inúmeros repositórios na Web oferecerão material de qualidade a um custo baixo ou nulo.

Dada a importância de que a temática da propriedade intelectual seja tratada em disciplinas de cursos de graduação e de pós-graduação da área de Computação, investigações adicionais poderão envolver a apresentação de uma proposta de como incorporar esta discussão ao contexto dos cursos. Em paralelo, seria importante buscar identificar como as instituições de ensino têm tratado o tema atualmente. Tal abordagem pode vir a permitir a apresentação de conselhos e/ou de orientações ainda mais gerais que as discutidas neste texto, com a consideração de aspectos relevantes tanto para os professores como para os alunos, seja no ensino presencial seja no ensino a distância.

Referências Bibliográficas

AMORIM, J. A.; MISKULIN, M. S. (2010). **Multimedia Engineering: Methodological Contributions to Digital Content Production and Use in Education**. In: Anais do XXI SBIE: CTD-IE. João Pessoa - PB.

AMORIM, J. A.; REGO, I. M. S.; DE SIQUEIRA, J. M.; GIMENO, A. (2011). **Defining the design parameters of a teacher training course on the incorporation of ICT into teaching practices**. WCES´2011. Elsevier. Istanbul, Turkey.

BRASIL. (2008). **Guia de Depósito de Patentes.** Ministério do Desenvolvimento, Indústria e Comércio Exterior. Instituto Nacional da Propriedade Industrial - INPI. DIRPA. Disponível em < http://www.inpi.gov.br/ >. Acesso em: 10 set. 2009.

GANDELMAN, H. (2004). **O que você precisa saber sobre direitos autorais**. Rio de Janeiro:Ed. Senac Nacional.

LIMA, J. A. A. (2006). **Curso de Propriedade Intelectual para Designers**. João Pessoa: Ed. Novas Idéias.

MANSO, E. J. V. (1987). **O que é direito autoral**. São Paulo: Brasiliense.

MOSKOWITZ, S. (2006). Introduction - Digital Rights Management. In: Zeng, W. & Yu, H. & Lin, C. (Editors). **Multimedia Security Technologies for Digital Rights Management**. Academic Press.

PARANAGUÁ, P.; BRANCO, S. (2009). **Direitos Autorais**. Rio de Janeiro: Ed. FGV.

O *Design* Instrucional nos cursos de Informática na modalidade a distância: a Experiência do Instituto Federal do Espírito Santo

José Mário Costa Junior, Danielli V. C. Sondermann, Saymon Castro de Souza

Centro de Educação a Distância - Instituto Federal do Espírito Santo (Ifes)
Serra – ES – Brasil

{jcjunior,danielli,saymon}@ifes.edu.br

Abstract. *This work discusses the application of Instructional Design in the planning of computing courses offered at distance by the Instituto Federal do Espírito Santo (Ifes). It is presented the activities map, an instrument used to plan the content, activities and resources for course subjects. In addition, it discusses the Ifes experience in the production of books, animations, videos and web conferencing. It was conclude that the methodologies proposed by Instructional Design can help in distance computing courses planning, which rely on different strategies to achieve the desired quality.*

Resumo. *Este artigo discute a aplicação do design instrucional no planejamento dos cursos da área de Informática oferecidos a distância pelo Instituto Federal do Espírito Santo (Ifes). É apresentado o mapa de atividades, instrumento utilizado para planejamento dos conteúdos, atividades e recursos para as disciplinas. Além disso, discute-se a experiência do Ifes na produção de materiais impressos, animações, vídeos e webconferência. Conclui-se que as metodologias propostas pelo Design Instrucional podem contribuir no planejamento dos cursos de Informática a distância, os quais dependem de estratégias diferenciadas para alcançar a qualidade desejada.*

1. Introdução

O ato de educar é um desafio constante, seja no nível fundamental, médio ou superior. Da alfabetização às disciplinas mais complexas das graduações, o professor precisa desenvolver estratégias para que, na interação com o aluno, os mais diversos tipos de conteúdos sejam aproveitados da melhor maneira possível. Esse processo de ensino-aprendizagem tem se transformado continuamente, em especial com a popularização da Internet e sua infinidade de recursos e informações. O professor, antes visto como detentor do conhecimento, agora precisa se adaptar a uma realidade na qual o conhecimento está propagado e disponível aos estudantes a qualquer momento e em diferentes formas. Sendo assim, as barreiras físicas da sala de aula já não são mais vistas como limite para a Educação.

Neste contexto, a Educação a Distância (EaD) se destaca como modalidade de educação no Brasil. Segundo o Instituto de Pesquisa Econômica Aplicada (2011), quase sete milhões de brasileiros estudam ou já estudaram a distância pela Internet, o que mostra que aprender online já é comum no Brasil. É importante ressaltar que, para um país com as dimensões do Brasil, essa se constitui uma das soluções para diminuir o déficit educacional e difundir o conhecimento em cidades afastadas dos grandes centros

e que, de outra forma, não teriam possibilidade de ofertar cursos técnicos ou superiores à população.

O Instituto Federal do Espírito Santo (Ifes), diante da demanda de interiorização da Educação Superior no estado, iniciou em 2006 o processo de implantação do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas (TADS) a distância, por meio do projeto Universidade Aberta do Brasil (UAB). Neste mesmo ano, foi criado o Centro de Educação a Distância (CEAD) da instituição. Em 2009, iniciou-se a oferta do segundo curso de graduação a distância do Ifes, a Licenciatura em Informática. Ainda em 2009, foi aberto o curso Técnico em Informática a distância, por meio do projeto Escola Técnica do Brasil (E-tec). Como visto, os primeiros cursos ofertados pelo Ifes são da área de Informática. Buscando não apenas tornar a educação nesta área acessível, mas também garantir a qualidade dos cursos ofertados, o CEAD/Ifes desenvolveu diversas estratégias de planejamento, utilizando-se principalmente das técnicas do *design* instrucional.

Este artigo tem por objetivo apresentar a experiência do CEAD/Ifes no planejamento dos cursos de Informática a distância, no intuito de oferecer Educação de qualidade aos alunos.

2. O *Design* Instrucional

O desenvolvimento da tecnologia trouxe diversas possibilidades ao campo da Educação. Hoje é possível utilizar vários recursos em uma aula, seja presencial ou a distância. Há quadros interativos, realidade aumentada, computadores com acesso à Internet e esta, por sua vez, possui fóruns, *chats*, redes sociais e inúmeras outras ferramentas. No entanto, esses recursos serão eficazes se houver um planejamento adequado e contextualizado com o que se deseja ensinar e aprender.

Segundo Moran (2009), uma das atividades fundamentais dos educadores é ajudar os aprendentes a encontrar uma lógica dentro do caos de informações atualmente disponíveis. Com tantos recursos e conteúdos, o professor deixa de ser o detentor do conhecimento, mas seu papel não é diminuído. Pelo contrário, o seu trabalho pode ser ainda maior ao assumir, agora, a responsabilidade de orientar o aluno para o que é importante e qual caminho deve ser seguido, o que exige, certamente, um planejamento coerente com os objetivos do processo de ensino-aprendizagem.

Na EaD, o planejamento didático ganha características diferenciadas, e dentre tantos fatores está o desenvolvimento dos materiais a serem disponibilizados aos alunos. Se na modalidade presencial o aluno passa grande parte do tempo fisicamente ao lado do professor, na modalidade a distância há a interação também com os materiais disponibilizados, que devem ser interessantes e despertar a autonomia do aluno, embora o papel do docente continue sendo imprescindível. Para que os cursos a distância não sejam apenas uma aplicação das antigas técnicas em ambientes tecnológicos, uma solução é utilizar as bases do *design* instrucional.

O *design* instrucional consiste na sistemática de planejar, desenvolver e aplicar métodos, técnicas, atividades, produtos, entre outros, em situações didáticas com o objetivo de promover a aprendizagem humana. É o conjunto de atividades aplicado para encontrar uma necessidade educacional e desenvolver uma solução para o problema [Filatro, 2008].

Muitas instituições que oferecem cursos a distância contam com um profissional específico para auxiliar no planejamento dos cursos: o *designer* instrucional. Segundo Moore e Kearsley (2008), a EaD não precisa apenas de especialistas em conteúdo, mas também de profissionais que cuidem dos aspectos da instrução, utilizando-se da teoria e da prática do gerenciamento de informações. Além disso, esse profissional deve ser capaz de identificar a aplicação das tecnologias disponíveis, de forma que facilitem a aprendizagem, e não apenas que sejam utilizadas sem contexto algum, tornando-se dispersivas. Ou seja, o trabalho do *designer* instrucional é potencializar o conteúdo desenvolvido pelo professor.

3. Estratégia de *Design* Instrucional no CEAD/Ifes

Um dos grandes desafios do CEAD/Ifes na implantação dos cursos a distância da área de Informática foi o planejamento das disciplinas. A maioria delas envolve processos de cálculo ou programação de computadores, o que geralmente, no ensino presencial, o professor explica passo a passo no quadro negro. Não significa que essa seja a forma mais efetiva de se ensinar, porém, é a mais tradicional.

O ensino de disciplinas dos cursos de computação a distância envolve, então, um planejamento detalhado e cuidadoso, para que o aluno consiga aprender o que precisa e a instituição evite a grande evasão, característica dos cursos. Por outro lado, os profissionais da área de Informática precisam desenvolver a autonomia, pois há mudanças constantes e bruscas no mercado.

Para que a divisão dos conteúdos e o planejamento das atividades fossem pensados com um maior nível de detalhes, o CEAD/Ifes desenvolveu um artefato denominado Mapa de Atividades, com base em outros mapas já existentes. A Figura 1 apresenta um exemplo de mapa:

MAPA DE ATIVIDADES

S	Tema principal	Subtemas	Objetivos específicos	Atividades	T	P	Recurso do Moodle	Grau de Dificuldade*	Nota	%	Observações
	Apresentação da Disciplina.	- Conteúdo Programático. - Metodologia de ensino. - Critérios de Avaliação. - Datas de provas. - Bibliografia utilizada.	Evidenciar para o aluno os objetivos e propostas da disciplina.	Visualizar o vídeo do professor, realizar a leitura dos Boas-vindas e dinâmica da disciplina. **Fórum 1** : Expectativas da disciplina.	x	X	Link a arquivo Página Web Link a arquivo Fórum Geral	Baixa	-	-	Reforçar na escrita sobre a importância da leitura da agenda e dedicação aos estudos (De 1 a 2 horas diárias).
01	1 Arquitetura de Software	1.1 Características 1.2 Arquitetura Distribuída 1.2.1 Sistemas Distribuídos 1.2.2 Arquiteturas de Sistemas Distribuídos	Compreender as características básicas de uma arquitetura de Software. Exemplificar alguns tipos de Arquitetura: Computação Distribuída, P2P,	**Atividade 1** : Questionário sobre características de arquiteturas.(incluir questão do material impresso). **Fórum 2** : Fórum sobre Arquiteturas. Cada	X	X X	Questionário Fórum Geral grade de correção - Fórum 2 Matriz individual - Fórum 2.	Baixa Média	1.0 1.0	2	

Figura 1. Mapa de Atividades

Cada coluna atende a um requisito do planejamento, a saber: a coluna "S" (Semanas) indica a forma de divisão do curso, e no caso do CEAD/Ifes é utilizada a divisão semanal; a coluna "Tema Principal" informa o título do conteúdo mais importante da semana; já a coluna "Subtemas" é a subdivisão do tema principal em unidades menores; em "Objetivos Específicos" devem ser inseridas as metas

pedagógicas para o módulo; em "Atividades", são listadas as tarefas que os alunos deverão realizar no módulo; já as colunas "T" e "P" indicam se cada tarefa proposta é prática ou teórica, o que é essencial em cursos de Informática; a seção "Recursos do Moodle" indica quais recursos do ambiente serão utilizados para implementar as tarefas; em "Grau de Dificuldade" é possível indicar a complexidade das tarefas; em "Nota" devem ser inseridos os valores para cada atividade; a coluna "%" indica a quantidade de horas gastas com o módulo em relação ao total de horas do curso. Por fim, a coluna "Observações" permite que sejam especificadas informações adicionais sobre as atividades propostas. [Carneiro et al., 2010].

O mapa de atividades traz algumas vantagens importantes para o *designer* instrucional. É possível identificar se há um grande número de atividades com alto grau de dificuldade em um módulo específico. Ele permite, ainda, visualizar se há diversidade de atividades e recursos. Além disso, é possível comparar os mapas de disciplinas concomitantes e identificar se o aluno terá muito trabalho em um determinado módulo e pouco em outros.

A coluna "Recursos" deve receber, além de ferramentas do Ambiente Virtual de Aprendizagem, solicitações de mídias (vídeos, animações, tutoriais, realização de webconferências, etc.). É dever do *designer* instrucional planejar, junto ao professor, o desenvolvimento dos mais diversos recursos multimídias. O CEAD/Ifes conta com uma equipe de produção preparada para desenvolvê-los.

4. Diversificando recursos em busca da aprendizagem

Uma das grandes preocupações na elaboração de disciplinas a distância - principalmente na área de informática - deve ser a utilização de múltiplos recursos em busca da melhoria do processo de ensino-aprendizagem. Muitas disciplinas do campo da computação apresentam - tanto na modalidade presencial, como a distância - certa complexidade para os alunos, uma vez que tratam de abstração e exigem habilidades cognitivas mais avançadas.

Quando se utilizam tecnologias na educação, deve-se ter o cuidado de não tornar a simples aplicação de recursos como estratégia de planejamento. A Internet, por exemplo, tornou a comunicação rápida e dinâmica. No entanto, na maioria das vezes, essa comunicação se dá por meio de textos, o que implica certo distanciamento, já que há uma associação da palavra escrita com a formalidade [Carlini e Tarcia, 2010). Assim, levar os alunos a somente ler na Internet não apresenta grandes diferenças de dar-lhes um livro. O que muda é apenas a tecnologia e não a modelagem do processo de ensino-aprendizagem.

Há inúmeras boas metodologias de ensino, independente da modalidade a que são aplicadas. O designer instrucional, junto ao professor, deve ser capaz de identificar as melhores práticas de cada modalidade e aplicá-las, sempre tendo em mente facilitar o desenvolvimento dos alunos. O uso de tecnologias e mídias não implica o abandono das boas práticas do ensino tradicional da sala de aula. A tendência é agregar boas práticas da Educação presencial, semipresencial e a distância.

O CEAD/Ifes vem buscando se utilizar da diversidade de recursos e tecnologias nos cursos de Informática a distância oferecidos pela instituição. Nas seções a seguir serão apresentadas algumas experiências desenvolvidas para disciplinas dos cursos TADS e Licenciatura em Informática.

4.1. O uso do Material Impresso

A EaD no Espírito Santo trouxe possibilidades a pessoas do interior do estado, as quais, de outra forma, não teriam acesso a um curso superior. No entanto, há, ainda, grandes desafios para esses alunos, tais como a necessidade de trabalhar e estudar e o acesso, por vezes precário, à Internet.

Muitos estudantes viram na EaD a chance de conciliar o trabalho e os estudos. Assim, o conceito de flexibilidade se faz valer, pois qualquer tempo disponível é tempo para estudar. Nesse sentido, a disponibilização do material impresso se faz indispensável porque permite que o aluno possa ter acesso ao conteúdo a qualquer hora e lugar, amenizando a necessidade do computador para estudar.

O uso do computador e o acesso à Internet são fundamentais para o bom desempenho nos cursos. Porém, em algumas cidades ainda há dificuldade de conexão, especialmente em áreas mais interioranas, seja por questões técnicas ou devido à situação financeira dos alunos. Há estudantes que só conseguem acessar o Ambiente Virtual de Aprendizagem nos polos de apoio presencial. Para eles, o uso do material impresso siginifica uma grande oportunidade.

O CEAD/Ifes procura facilitar a leitura nos materiais impressos adotando alguns padrões, tais como: linguagem dialógica, iconografia para destacar os conteúdos, margens diferenciadas que permitem anotações ao lado dos textos, produção de ilustrações específicas para o material e atividades a cada três páginas de conteúdo com o objetivo de evitar o cansaço mental da leitura. Além disso, foi desenvolvido um padrão de cores para cada nível (Técnico, Graduação e Pós-Graduação), de forma que o aluno se habitue e tenha o estudo facilitado.

4.2. A utilização das animações

Um tipo de objeto de aprendizagem interessante e com muitas possibilidades é a animação. O CEAD/Ifes possui uma equipe de *design* gráfico preparada para produzi-las, o que traz a vantagem da personalização de roteiros integrados ao conteúdo da disciplina.

Muitas vezes, os alunos não conseguem entender um determinado assunto apenas por meio de um texto. Há, ainda, alguns tópicos difíceis de transpor em forma de linguagem escrita, isso porque ocupariam muito tempo de leitura e exigiriam muitas explicações. Geralmente, esses conteúdos podem ser mais bem representados em forma de animações ou vídeos.

No caso das animações, é possível criar personagens, inserir efeitos especiais, enfatizar graficamente um determinado ponto de interesse, etc. As possibilidades são muitas, dependendo da criatividade e da capacitação técnica da equipe.

Outra utilização relevante é permitir a interação do aluno com o objeto de aprendizagem. Nos cursos da área de informática há sempre muitas disciplinas práticas, por isso os simuladores são importantes ferramentas, visto que permitem ao estudante visualizar, na prática, o que só poderiam executar presencialmente no polo de apoio presencial.

A Figura 2 mostra uma animação utilizada na disciplina Programação II, do curso TADS. A ementa define assuntos como variáveis dinâmicas (ponteiros), conceito

de tipos abstratos de dados, lista estática x lista dinâmica, lista simplesmente encadeada, entre outros. Uma estratégia muito utilizada na modalidade presencial para ensino desses conteúdos são os testes de mesa. A animação procura transpor essa técnica para o ambiente virtual utilizando-se da personagem Sofia, criada para atuar como professora virtual.

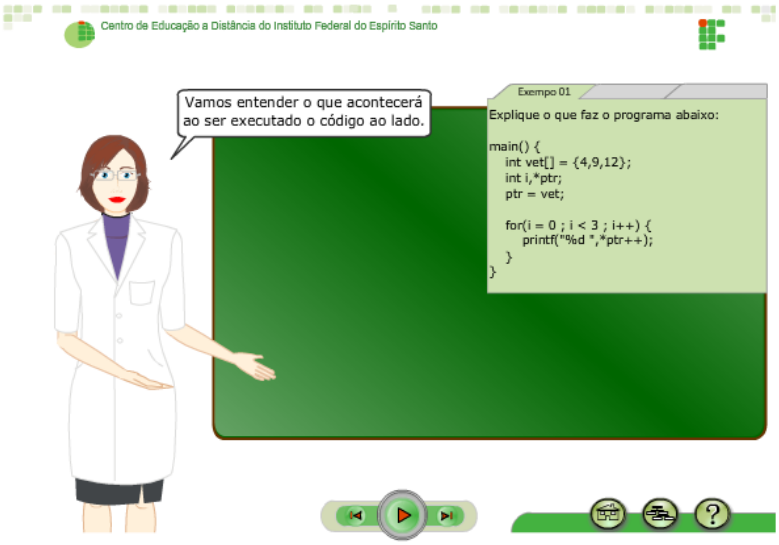


Figura 2. Animação da disciplina de Programação II

Um desafio maior foi a disciplina de Laboratório de Montagem e Manutenção de Computadores, do 3º período do curso Licenciatura em Informática. Como o próprio nome da disciplina indica, o conteúdo é muito prático. Os alunos tinham acesso aos equipamentos no polo de apoio presencial, mas foi preciso pensar em alguma estratégia para que pudessem treinar virtualmente.

Foi desenvolvido, então, um simulador de montagem de um computador. Nele, o aluno pode interagir com a animação e arrastar as peças para os devidos lugares. Se todas as peças forem encaixadas corretamente, o aluno conseguirá ligar o computador e ver a especificação dos componentes. Se não, recebe a mensagem de erro adequada e deve tentar novamente. A Figura 3 mostra o simulador em ação.

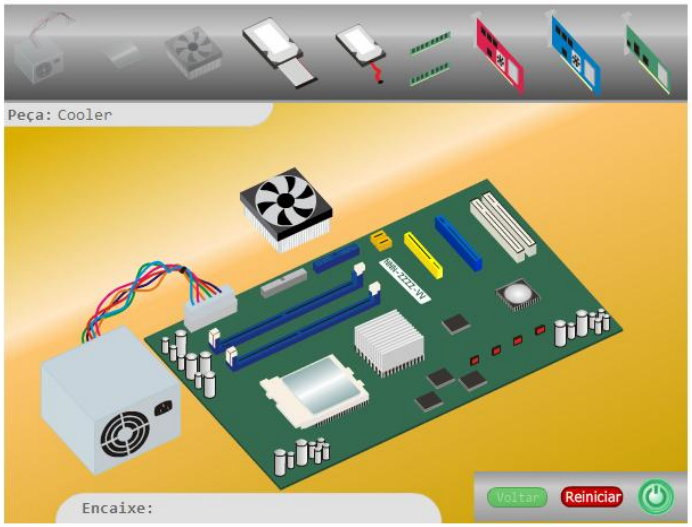


Figura 3. Simulador de montagem em manutenção

É importante salientar que as animações são construídas a partir de roteiros escritos pelos professores com a ajuda dos *designers* instrucionais. Os roteiros são fornecidos juntamente com o mapa de atividades.

4.3. A Utilização do Vídeo

Os vídeos são capazes de expressar aspectos culturais como valores, espaços e liguagem de forma atraente, aproximando o conteúdo e o professor do universo dos alunos. Se utilizados de forma adequada, os vídeos podem acelerar o processo de ensino-aprendizagem [Carvalho e Ivanoff, 2010].

Em disciplinas práticas ou que exigem processos muito complexos para resolução de problemas, o vídeo se torna adequado como recurso. Isso porque, por meio dele, o aluno pode assistir ao professor como em uma explanação presencial, embora sem a interação imediata. Além disso, os vídeos são um tipo de mídia muito popularizado pela televisão e pelo cinema. Ou seja, o aluno certamente terá familiaridade com o uso desse recurso, o que facilitará a aprendizagem.

O CEAD/Ifes, pela complexidade de gravação, exigência de equipamentos e necessidade de pessoal, só há pouco tempo começou a desenvolver videoaulas. Um exemplo de aplicação foi o vídeo desenvolvido para a disciplina Laboratório de Montagem e Manutenção de Computadores, da Licenciatura em Informática. Nele, o professor explica pontos importantes da manipulação das peças. Para garantir a presença do professor, o vídeo foi dividido em uma tela maior, mostrando os componentes com mais detalhes e uma pequena tela, à esquerda inferior, com a presença do professor. A Figura 4 exemplifica o modelo de vídeo.



Figura 4. Vídeo da disciplina Laboratório de Montagem e Manutenção de Computadores

No contexto da busca por inovação no que se refere aos vídeos, o CEAD/Ifes participou do Serviço Experimental de Educação a Distância (SE-EDAD), promovido pela Rede Nacional de Pesquisa (RNP). O projeto consiste em disponibilizar e operar uma infraestrutura de disseminação de material multimídia no formato de videoaulas,

destinado a qualquer instituição educacional ou de pesquisa que deseje criar e disponibilizar as aulas. O grande diferencial está no modelo dessas aulas: o ambiente proposto permite que o aluno assista aos vídeos e interaja com as animações simultaneamente. O CEAD/Ifes pretende, futuramente, disseminar a produção de videoaulas juntamente com os demais recursos já utilizados. A Figura 5 mostra uma captura da videoaula experimental do Ifes, gravada pra o projeto.

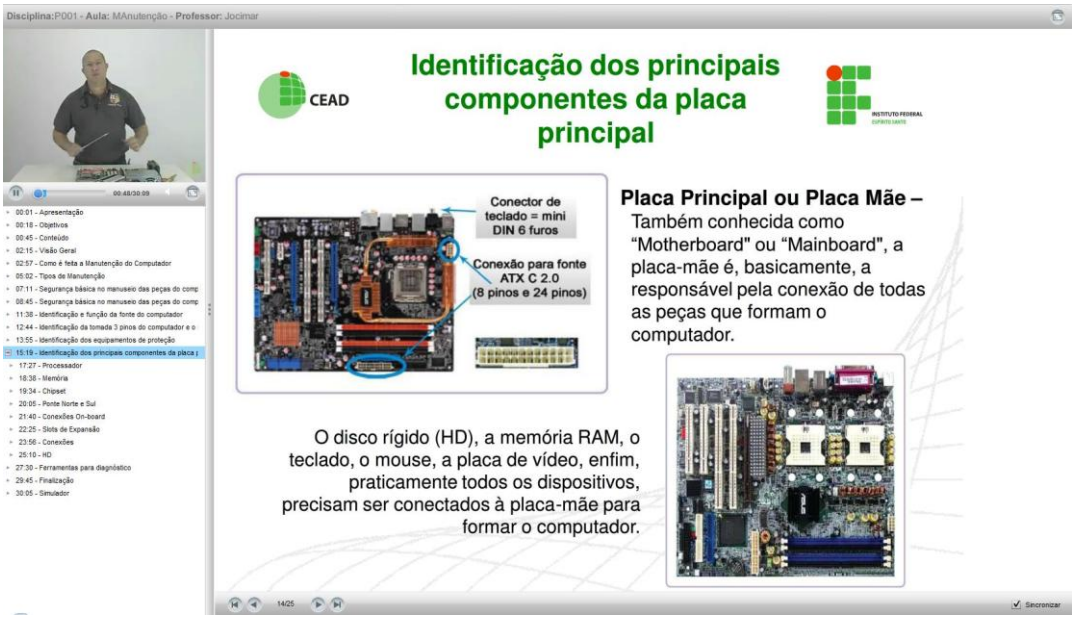


Figura 5. Videoaula experimental gravada pelo Ifes para o projeto SE-EDAD

4.4. Webconferência: interação em tempo real

Ao permitir que participantes situados em lugares geograficamente diferentes possam realizar uma reunião síncrona com transmissão de vídeo e outros recursos, a videoconferência e a webconferência são, dentre as mídias aplicadas na EaD, as mais próximas do presencial [Cruz, 2009].

Essa aproximação, em tempo real, entre quem ensina e quem aprende, traz muitos dos benefícios da modalidade presencial à modalidade a distância, dentre os quais o fato de poder tirar dúvidas instantaneamente, a possibilidade de uma comunicação espontânea e a grande interação entre os participantes.

Os momentos de webconferência podem ser utilizados pela equipe dos cursos para esclarecimento de dúvidas após um determinado módulo de conteúdo. Podem, ainda, ser utilizados para aulas ao vivo, nas quais o professor expõe os assuntos e os alunos participam, como em uma aula presencial.

Uma aplicação importante é na apresentação de trabalhos, prática muito comum na modalidade presencial, mas que ainda encontra a barreira da distância nos cursos não presenciais. A webconferência consiste em uma solução eficiente, já que os professores e os tutores têm a possibilidade de marcar encontros virtuais e avaliar a apresentação dos estudantes, enfatizando, assim, a capacidade de comunicação e expressão dos alunos.

O CEAD/Ifes planejou webconferências para os momentos de tutoria da disciplina Programação II, do curso TADS. Os tutores vasculhavam os fóruns de dúvidas da disciplina, no ambiente virtual, e reuniam os principais questionamentos surgidos durante a semana. Após isso, uma aula de esclarecimentos era preparada e os alunos eram convidados a participar. Nela, era possível a resolução ao vivo de exercícios de programação, por meio do compartilhamento de telas. A Figura 6 mostra um momento de tutoria via webconferência.

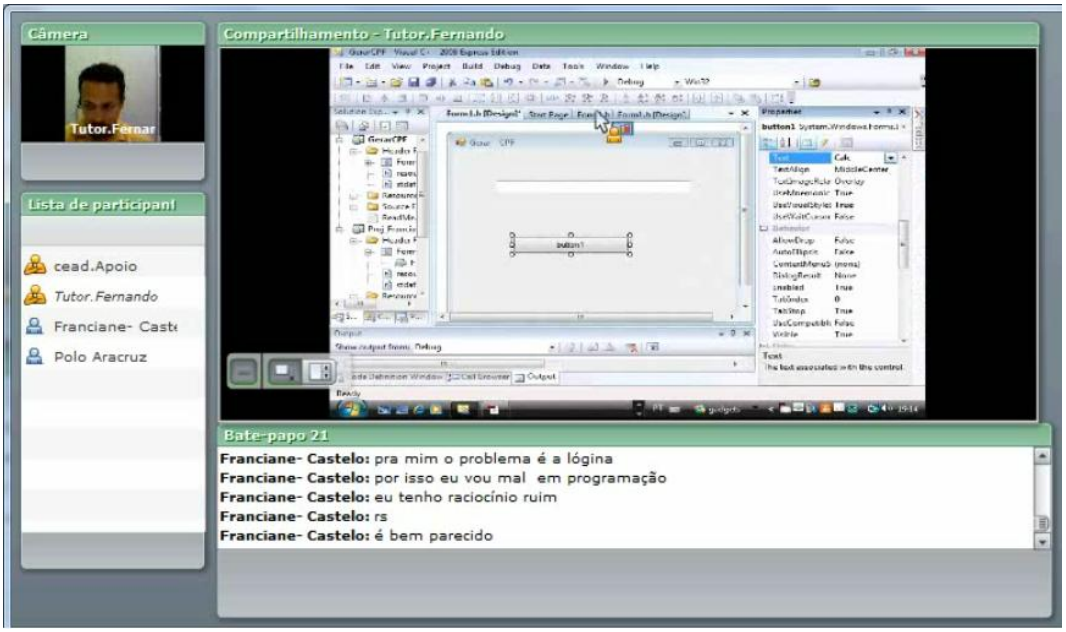


Figura 6. Tutoria via webconferência

5. Considerações finais

Planejar adequadamente um curso, seja presencial ou a distância, é um passo importante para garantir sua qualidade. A busca pela melhor forma de ensinar e aprender sempre será um desafio, pois a Educação é muito dinâmica. Metodologias e recursos utilizados em um determinado grupo podem não funcionar com outro, e técnicas aplicadas a uma disciplina podem ser muito inadequadas a outras.

Por isso, o trabalho do *designer* instrucional junto aos professores é essencial, visto que esse profissional deve estar atento às mudanças e particularidades de cada conteúdo trabalhado. As escolhas que esses profissionais fazem podem determinar o sucesso ou o fracasso de uma disciplina. E escolher estatégias corretas depende da análise de um conjunto formado pelo conteúdo, recursos disponíveis, perfil do grupo, entre outros.

O CEAD/Ifes busca, por intermédio das medidas apresentadas neste trabalho, facilitar a aprendizagem dos alunos e, ao mesmo tempo, torná-la um processo prazeroso. É preciso tentar diminuir a distância entre o aprendiz, o conteúdo, seus tutores e colegas. E esse é um dos objetivos centrais do planejamento: mostrar ao aluno que a EaD pode ser interessante, interativa, colaborativa e flexível, e que há outras forma de aprender diferentes da leitura e da exposição. Nesse sentido, o *Design* Instrucional tem papel fundamental.

Como trabalho futuro, pretende-se realizar uma avaliação mais ampla do impacto dos recursos desenvolvidos na aprendizagem dos alunos. Não basta apenas

fazer uma comparação entre turmas ricas em recursos e turmas com mídias escassas. As realidades dos alunos, o atendimento dos tutores e professores, as condições dos pólos de apoio presencial, entre outros, são fatores que também precisam ser avaliados. Por exemplo: uma turma pode ter um bom desempenho em uma sala virtual pobre de recursos, mas com um atendimento eficaz e competente dos tutores. Outra turma pode ter inúmeros recursos midiáticos interessantes, mas ter um desempenho baixo por conta do atendimento pouco eficiente dos tutores.

Utilizar recursos diferenciados para ensinar Informática a distância é um desafio constante para os *designers* instrucionais e os professores. São muitos os conteúdos que precisam de intervenção para que sejam aprendidos. Por isso, é importante buscar sempre a diversidade de tecnologias, sua aplicação e a avaliação de sua efetividade. Cabe aos docentes e *designers* instrucionais avaliarem suas práticas, trabalhando de forma cooperativa no planejamento de uma EaD de maior qualidade.

Referências

Carlini, A. L., Tarcia, R. M. L. (2010) "20% a distância e agora? Orientações práticas para o uso de tecnologia de educação a distância", São Paulo: Pearson Edcation do Brasil.

Carneiro, D. V., Costa Junior, J. M., Nunes, V. B., Nobre, I. A. M, Baldo, Y. P. (2010) "Uma proposta de planejamento para criação de salas no ambiente virtual de aprendizagem Moodle: mapa de atividade adaptado", In: XVI Congresso Internacional ABED de Educação a Distância, Foz do Iguaçu – PR.

Carvalho, F. C. A., Ivanoff, G. B. (2010) "Tecnologias que educam: ensinar e aprender com tecnologias da informação e comunicação", São Paulo: Pearson Prentice Hall.

Cruz, D. M (2009) "Aprendizagem por videoconferência", In: Litto, F.M., Formiga, M. (Orgs.) "Educação a distância: o estado da arte", São Paulo: Pearson Education do Brasil.

Filatro, A. (2008) "Design instrucional na prática", São Paulo: Pearson Education do Brasil.

Instituto de Pesquisa Econômica Aplicada. (2011) Secretaria de Assuntos Estratégicos da Presidência da República. Diretoria de Estudos e Políticas Setoriais, de Inovação, Regulação e Infraestrutura. Radar: Tecnologia, Produção e Comércio Exterior. n. 13. In: Kubota, L. C., Demoliner, V. L., Figueiredo, C. O. "Um perfil do uso da Educação on-line no Brasil". Brasília: 10 mai. 2011.

Moore, M. G., Kearsley, G. (2008) "Educação a distância: uma visão integrada", São Paulo: Cengage Learning.

Moran, J. M. (2009) "A educação que desejamos: novos desafios e como chegar lá", 4ª ed., São Paulo: Papirus.

Serviço Experimental de Educação a Distância. (2010) "Sobre o EDAD: o que é", http://edad.rnp.br, Abril.

Avaliação do Comportamento do Aluno de Educação à Distância (EAD) na Área de Computação

Janaína R. Penedo[1], **Morganna C. Diniz**[1], **Flávio de Brito Pinheiro**[2]

[1]Universidade Federal do Estado do Rio de Janeiro - UNIRIO
Av. Pasteur, 296 – Urca – CEP 22290-240 – Rio de Janeiro – RJ – Brazil

[2]Fundação CECIERJ - Consórcio CEDERJ
Rua da ajuda, 5 – Centro – CEP 20040-000 – Rio de Janeiro – RJ – Brazil

{janaina.penedo,morganna}@uniriotec.br,brito@cederj.rj.gov.br

Abstract. *Understanding the characteristics of the user behavior of educational sites is an important question in the search for the improvement of e-learning courses. This type of application demands a considerable amount of resources, therefore checking how these resources are used is crucial to ahead foresee the system workload increase. In this work, a case study of an e-learning site is presented. This application holds about two thousand users of a graduation course in the area of computation spread by 17 different localities in the State of Rio de Janeiro.*

Resumo. *Caracterizar o comportamento dos usuários de sites educacionais é uma questão importante na busca pela melhoria dos cursos EAD que tem na Internet o seu meio de comunicação. Esse tipo de aplicação demanda uma considerável quantidade de recursos, por isso verificar como esses recursos são utilizados é essencial para prever o comportamento do sistema diante do aumento de carga de trabalho. Neste trabalho é apresentado um estudo de caso de uma aplicação EAD que comporta cerca de dois mil alunos de um curso de graduação na área de computação espalhados por 17 diferentes localidades no Estado do Rio de Janeiro.*

1 Introdução

Uma das consequências do advento da Internet foi o crescimento extraordinário dos cursos universitários oferecidos na modalidade de Ensino a Distância (EAD). Como a Internet é considerada um meio relativamente rápido e barato para a transmissão de dados multimídia, o seu uso para a distribuição de conteúdo de aulas parece ser a resposta natural (e já disponível) para a implementação de cursos não presenciais ou parcialmente não presenciais pelas instituições de ensino. De acordo com o MEC, em 2009, 145 instituições estavam credenciadas para a oferta de cursos superiores na modalidade a distância com mais de 760.000 mil alunos matriculados [MEC 2011].

O ambiente de ensino à distância usando a Internet parece, a primeira vista, bastante simples e fácil de implementar. As aulas de um determinado curso são armazenadas em um computador conectado à Internet. Este computador é na realidade um servidor de arquivos multimídia (documentos, vídeos, animações, etc) que tem como função entregar estes arquivos (as aulas) à medida em que forem sendo solicitados. Quando um

aluno deseja ver uma aula do curso, ele acessa a Internet e, dentro da página do curso, seleciona uma aula. Para tirar dúvidas sobre o conteúdo da aula, o aluno pode marcar um horário com um professor e utilizar uma ferramenta de transmissão de voz ou um chat que deve estar disponível na página do curso. Ferramentas colaborativas podem ser disponibilizadas permitindo a interação entre grupos de alunos, que podem discutir temas e realizar tarefas propostas pelos professores em equipes. Além disso, o ambiente pode disponibilizar instrumentos de avaliação da aprendizagem, que ajudam os professores no acompanhamento da evolução dos alunos. Resumindo, a instituição que oferece um curso à distância precisa ter um servidor de arquivos ligado à Internet. Para participar do curso, o aluno precisa apenas se conectar à Internet.

É importante enfatizar que EAD é educação e tem que ser de qualidade, tanto quanto a educação presencial [Nascimento and Carnielli 2009]. Um dos problemas encontrados pelas instituições que provêem ensino na modalidade a distância está em decidir como compartilhar, dimensionar e alocar os recursos computacionais a seus sistemas de forma a melhor atender a esta premissa. Uma hipótese para solução deste problema está em entender como a interação dos usuários com o sistema afeta seu desempenho. A determinação dos processos que representam este comportamento e suas características estatísticas nos permitirá entender o que motiva um aluno a utilizar o sistema, assim como descobrir quais são os pontos fortes e fracos que podem responder perguntas referentes a aumentos de escalabilidade, mantendo o desempenho satisfatório e consequentemente a qualidade de serviço.

Um exemplo de sucesso na modalidade de EAD é o Consórcio CEDERJ da Fundação CECIERJ [CEDERJ 2011]. Este consórcio reúne seis universidades públicas sediadas no Estado do Rio de Janeiro:

1. Universidade do Estado do Rio de Janeiro - UERJ;
2. Universidade Estadual do Norte Fluminense Darcy Ribeiro - UENF;
3. Universidade Federal do Estado do Rio de Janeiro - UNIRIO;
4. Universidade Federal do Rio de Janeiro - UFRJ;
5. Universidade Federal Fluminense - UFF;
6. Universidade Federal Rural do Rio de Janeiro - UFRRJ.

O aluno do Consórcio CEDERJ está matriculado em uma das universidades do consórcio, dependendo do curso e do pólo regional a que esteja vinculado. Os pólos são os locais onde os alunos matriculados podem realizar atividades presenciais obrigatórias como aulas de laboratório, avaliações, entre outros. Atualmente, o Consórcio possui mais de 30 mil alunos matriculados e oferece graduação em: Matemática, Ciências Biológicas, Pedagogia, Física, Computação, Administração, Química, História e Turismo.

O objetivo deste trabalho é apresentar o estudo do comportamento dos alunos de graduação do curso de Tecnologia de Sistemas de Computação do CEDERJ feito nos meses de agosto, setembro e outubro de 2010. Embora a pesquisa ainda esteja em sua fase inicial, é possível verificar que o comportamento dos alunos de cursos de EAD tem características próprias que precisam ser melhor analisadas.

Este trabalho está estruturado da seguinte forma. Na seção 2 são discutidos os trabalhos relacionados com o tema. A seção 3 comenta a metodologia usada pelo curso de Tecnlogia de Sistemas de Computação do CEDERJ. A seção 4 apresenta a análise do

comportamento dos alunos do curso feita nos meses de agosto, setembro e outubro de 2010. As conclusões e os trabalhos futuros estão na seção 5.

2 Trabalhos Relacionados

Na literatura são encontrados trabalhos sobre o comportamento dos usuários de websites com diversos objetivos. A seguir, são discutidos alguns destes trabalhos.

Em [Mor et al. 2007], os autores propõem uma metodologia de três níveis para estudar um ambiente de *e-learning* real: nível de sessão, nível de atividade e nível de desejo. No primeiro nível, são analisados os caminhos de navegação dentro do site e identificados os padrões de navegação dos usuários. No segundo nível, é avaliado o progresso de aprendizagem dos estudantes dentro do ambiente. E no terceiro nível, o trabalho faz a comparação entre o que foi pretendido e o que foi obtido pelos alunos nos cursos matriculados. O foco deste trabalho é a avaliação e melhoria da usabilidade em ambientes de *e-learning*.

Em [Weirich et al. 2007], os autores apresentam uma ferramenta de análise de log desenvolvida para um ambiente de EAD. Neste trabalho, é feita a caracterização dos usuários em relação ao modo de navegação no site classificando os tópicos das aulas de acordo com a quantidade de acessos recebidos. O objetivo do trabalho é auxiliar o professor no aperfeiçoamento do conteúdo das aulas do curso EAD.

Em [Alves et al. 2007], os autores caracterizam o comportamento dos alunos de um curso de EAD que acessam às video-aulas em um servidor. Neste ambiente, os alunos tem total controle sobre a aula que estão assistindo, podendo se movimentar livremente sobre todos os tópicos de cada aula, podendo parar e retomar a exibição a qualquer instante e de qualquer ponto da aula. A principal contribuição deste trabalho é a caracterização da interatividade dos alunos em um ambiente real de operação.

Em [Lutteroth and Weber 2008], os autores apresentam um método para calcular a carga de trabalho de um sistema web a partir da caracterização do comportamento dos usuários.

Em [Marques-Neto et al. 2009], os autores propõem uma metodologia para caracterizar usuários de banda larga da Internet a partir do estudo de sete itens: processo de chegada das sessões dos usuários; processo de saída das sessões dos usuários; duração das sessões; quantidade de bytes recebidos durante as sessões dos usuários; quantidade de bytes enviados nas sessões; os principais serviços; atividades de comércio eletrônico executadas durante o acesso.

Em [Menascé et al. 1999], é proposta uma metodologia de caracterização de carga de trabalho dos sites de comércio eletrônico baseada no comportamento dos usuários. São critérios usados neste trabalho para a definição do modelo: requisições que geram ações no servidor, taxa média de requisições por sessão, e intervalo médio entre requisições.

Nenhum dos trabalhos discutidos acima se preocupa em caracterizar o comportamento dos usuários de sites de EAD ou em verificar se os métodos já propostos também são adequados na avaliação desses ambientes. Embora seja fácil verificar que um aluno de curso de EAD não tem o mesmo comportamento de um cliente de loja virtual. Afinal, o aluno precisa executar um conjunto de atividades próprias da área de ensino como ler/assistir aulas, discutir conteúdo didático, tirar dúvidas e fazer exercícios. Portanto,

entender o comportamento desses usuários é fundamental para o planejamento, desenvolvimento e manutenção de sistemas de *e-learning*.

3 Curso de Tecnologia de Sistemas de Computação

O Curso Superior de Tecnologia de Sistemas de Computação do CEDERJ, sob a responsabilidade conjunta da UFF e da UFRJ, utiliza basicamente os seguintes instrumentos na formação do aluno:

- aulas gravadas pelos professores da UFF e da UFRJ;
- tutoria presencial ou à distância;
- plataforma EAD com e-mails, foruns, listas de discussão, material didático complementar, etc;
- avaliação individual presencial (sem consulta) e à distância (com consulta).

O formato das aulas gravadas é o mesmo encontrado nos cursos presenciais: o aluno visualiza o professor e as transparências do tópico em discussão. Entretanto, o aluno pode interagir como estivesse vendo um filme em DVD. Ele pode parar, avançar, voltar ou ir direto a uma parte da apresentação. Além disso, o aluno pode ser convidado durante a aula a realizar alguma tarefa no próprio computador com o objetivo de consolidar a matéria em estudo.

Os alunos podem assistir as vídeo-aulas tanto no pólo como em casa. Em cada um dos pólos, onde o curso é disponibilizado, existe um laboratório exclusivo para uso dos alunos da Computação. Se o aluno possuir um computador, ele pode assistir as aulas em casa usando os DVDs recebidos na matrícula ou pode assistir as aulas acessando a Internet.

O material didático para o aluno consolidar os conceitos apresentados nas video-aulas é o livro comercial, ou seja, os alunos do curso EAD usam os mesmos livros-texto dos alunos presenciais da UFF e da UFRJ. Na plataforma EAD é disponilizado apenas material didático complementar ao livro-texto. Por isso, cada pólo possui um determinado número de livros das disciplinas do curso para empréstimo aos alunos.

Os alunos podem tirar as dúvidas sobre a matéria indo ao pólo e conversando pessoalmente com os tutores presenciais ou podem usar o 0800 e a plataforma EAD para conversar com os tutores à distância.

O curso tem uma carga de 2.120 horas distribuída em seis semestres (duração de 3 anos) e teve início no primeiro semestre de 2005 com três turmas de 35 alunos. Atualmente, são 1.995 alunos matriculados em 17 cidades no Estado do Rio de Janeiro (Angra dos Reis, Barra do Piraí, Belford Roxo, Duque de Caxias, Itaguaí, Itaocara, Itaperuna, Niterói, Nova Iguaçu, Piraí, Rio Bonito, Rio das Flores, São Fidélis, São Gonçalo, Saquarema, Três Rios e Volta Redonda).

Na avaliação do último ENADE, o curso de Tecnologia de Sistemas de Computação do CEDERJ recebeu nota 5.

4 Estudo de Caso

Para fazer este estudo, foram disponibilizados pelo CEDERJ o log da plataforma EAD com o acesso dos alunos do curso de computação nos meses de agosto, setembro e outubro de 2010. A Tabela 1 mostra as quatro principais métricas calculadas para o período

analisado. A primeira métrica corresponde ao número de diferentes usuários que acessaram o sistema. É interessante observar que a quantidade de usuários diminui à medida que o semestre vai progredindo. Este fato é explicado pelo número de trancamentos e abandonos do curso durante o semestre. A segunda métrica mostra a quantidade de sessões verificadas no período, onde uma sessão corresponde ao intervalo de tempo que o usuário fica conectado ao sistema. A terceira métrica especifica o número de requisições processadas pelo sistema em cada mês, onde cada requisição corresponde a um pedido de serviço ao sistema como, por exemplo, leitura de e-mails, acesso às salas de tutoria, etc. A quarta métrica mostra o tempo médio das sessões durante os três meses.

Métricas	**Agosto**	**Setembro**	**Outubro**
Número de usuários ativos	1.956	1.545	1.484
Quantidade de sessões	20.935	21.434	14.681
Quantidade de requisições recebidas	375.959	194.801	140.187
Média de tempo conectado	00:58:50	00:53:00	01:05:55

Tabela 1. Métricas usadas na pesquisa.

A plataforma CEDERJ disponibiliza 81 diferentes módulos de serviço para os seus usuários. A partir da avaliação dos logs, é possível observar que os pedidos de serviço se concentram em onze módulos. A descrição dos módulos mais utilizados pelos alunos pode ser vista na Tabela 2.

Módulo	**Ações disponíveis**
Login	Identificação do usuário e do ambiente a ser usado
Forum	Discussão de um determinado assunto pelos alunos, tutores, professores e coordenadores do curso
Sala de tutoria	Discussão das disciplinas com os tutores
Gerenciador de arquivos	Upload/Download de arquivos
Aviso	Visualização de mensagens do CEDERJ
Email	Acesso e utilização dos emails pela plataforma
Grade de disciplina	Acesso às disciplinas do curso
Mural do curso	Visualização de mensagens da coordenação do curso
Mural dos pólos	Visualização de mensagens do pólo
Sala de disciplina	Acesso a todo o ferramental alocado para as disciplinas
Material didático	Acesso ao material complementar disponível para as disciplinas

Tabela 2. Módulos mais utilizados.

A Figura 1 mostra o número de requisições para os módulos mais utilizados pelos alunos do curso de Tecnologia de Sistemas de Computação. O módulo login não é representado por ser obrigatório para todos os acessos.

Seja sessão a sequência de ações consecutivas de um mesmo usuário no período de uma visita ao sistema [Duarte et al. 2008]. No log utilizado, esse tempo corresponde

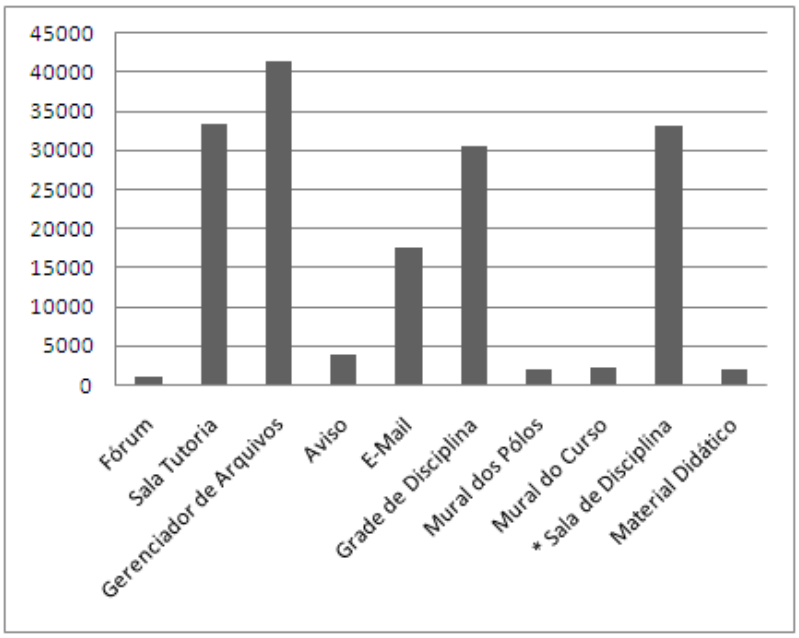


Figura 1. Quantidade de requisições por módulo.

ao intervalo entre uma ação de Login e uma ação de Logout (ou uma ação qualquer que resulte na saída do sistema) de um determinado usuário. A Figura 2 mostra a probabilidade de duração de uma sessão no período estudado. Em mais de 50% dos registros, os alunos ficaram em torno de 90 minutos logados na plataforma a cada acesso. Entretanto, é interessante comparar esta informação com a Figura 3 que apresenta o tempo médio das sessões nos meses analisados. No estudo do log, foram detectadas dezenas de sessões com mais de 6 horas de duração. A ocorrência de tais sessões coincidem com a semana em que as ADs (avaliação à distância) foram disponibilizadas pelo curso e com a semana de entrega das ADs pelos alunos nos pólos. Portanto, a plataforma EAD tem uma carga maior de serviço nos períodos relacionados a alguma atividade de avaliação feita nas disciplinas. Nestes momentos, os alunos aumentam a participação nos foruns e nas salas de tutoria para pesquisar e discutir as questões das provas.

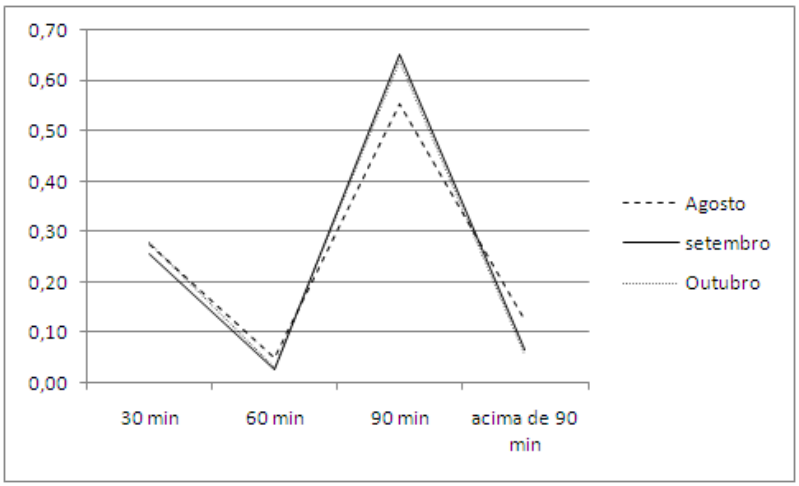


Figura 2. Duração das sessões.

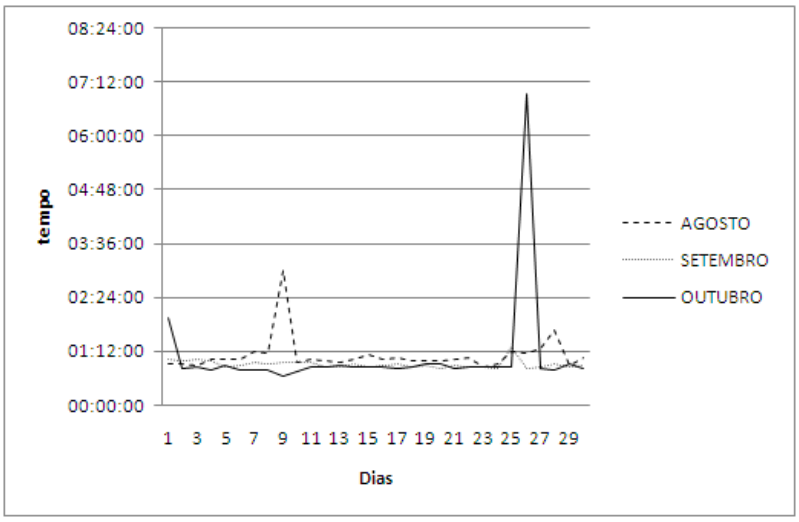


Figura 3. Tempo médio de duração das sessões.

A Figura 4 mostra o número médio de requisições por dia no sistema. Note que, normalmente, existe pouco acesso à plataforma pelos alunos da Computação durante o fim de semana. Enquanto, um acesso maior ao site é detectado entre as terças e quintas-feiras. Isto pode ser explicado pelo fato das tutorias presenciais estarem mais concentradas nos dias de sábado. Assim, os alunos procuram os tutores nos pólos nesses dias para tirar suas dúvidas deixando para acessar a plataforma nos dias úteis.

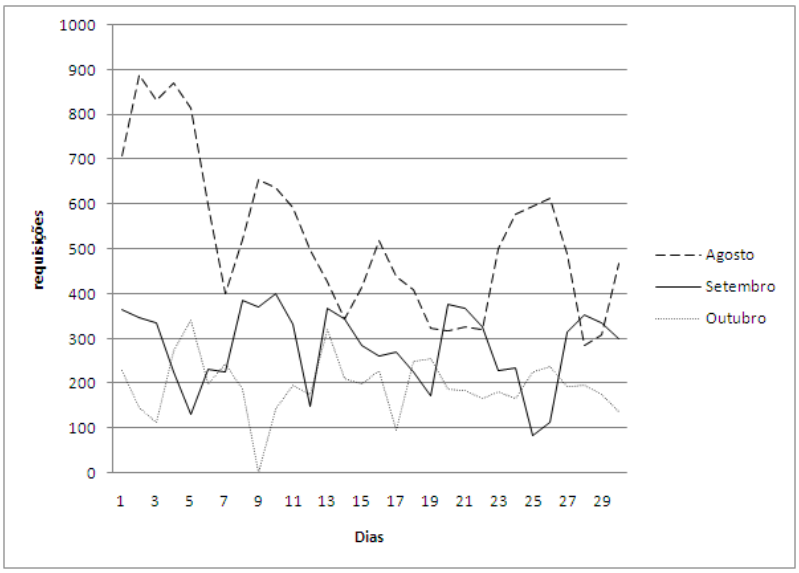


Figura 4. Número médio de requisições por dia.

A Figura 5 mostra o número médio de requisições por hora. Apesar da diferença na quantidade de requisições observadas nos três meses, é possível notar um comportamento padrão no acesso à plataforma. Há um acesso quase uniforme das 11 horas da manhã às 8 horas da noite. A partir das 21 horas, o número de requisições vai diminuindo, quase chegando a zerar no horário da madrugada. Isto pode ser explicado pelo

fato do aluno de Computação normalmente ter acesso a um computador durante as suas atividades profissionais. O que permite o acesso à plataforma no horário comercial.

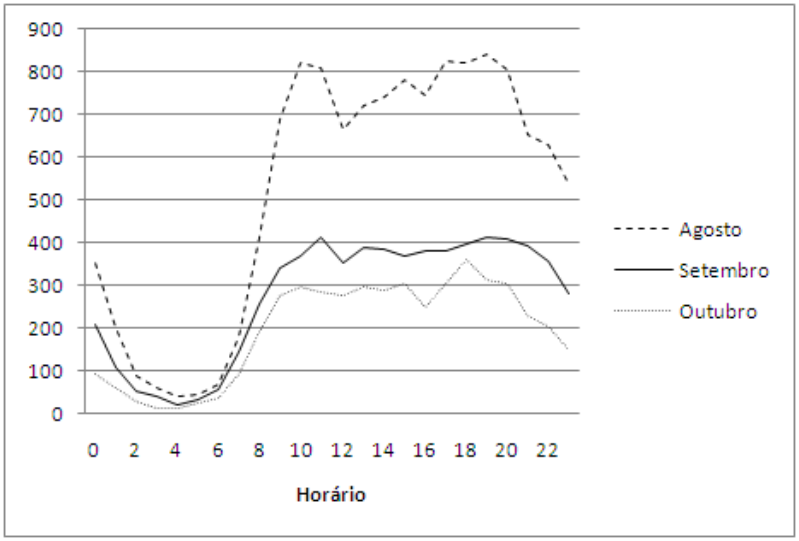


Figura 5. Número médio de requisições por hora.

5 Conclusões e Trabalhos Futuros

Este trabalho contribui com uma caracterização atual de um sistema de educação a distância com especial ênfase ao comportamento dos usuários do sistema. Os resultados da caracterização foram obtidos utilizando as principias métricas obtidas através da revisão sistemática dos trabalhos encontrados na literatura com a finalidade de entender como os usuários interagem com o sistema.

A análise dos logs revelou uma alta interatividade por parte dos usuários do sistema. Através da caracterização foi possível identificar algumas propriedades interessantes.

- Os alunos costumam acessar, em média, a plataforma por 90 minutos, sendo que este tempo pode triplicar na época de atividades que estão relacionadas à aplicação de provas. Isto pode ser explicado pelo fato das video-aulas não estarem disponíveis na plataforma. Logo, o sistema é usado pelos alunos principalmente como canal de contato com os tutores, professores e coordenação.
- O horário de maior requisição ao sistema ocorre entre 11 e 12 horas e entre 19 e 20 horas. O sistema se torna mais ativo a partir das 11 horas, caindo a frequência após às 23 horas. Isto ocorrer porque o aluno do curso normalmente trabalha, assim a disponibilidade para acessar o sistema é no horário de almoço ou depois do expediente.
- As funções mais utilizadas no sistema são as ações referentes a: sala de disciplina, sala de tutoria, gerenciador de arquivos, grade de disciplina e email. Portanto, os alunos utilizam o sistema principalmente para esclarecer dúvidas em relação ao conteúdo das matérias. As outras funções fornecidas pelo sistema também foram mapeadas, no entanto elas possuem baixa utilização.

Como trabalho futuro, pretende-se:

- coletar dados de um ano inteiro ampliando a análise para outras características dos usuários;
- criar um modelo matemático para representar o comportamento do aluno de EAD;
- incluir planejamento de capacidade e demanda de serviço no modelo.

Agradecimentos
Este trabalho não teria sido possível sem a ajuda da equipe de suporte da plataforma do CEDERJ que forneceu os logs do servidor EAD e que tirou dúvidas sobre a infraestrutura e o sistema utilizados.

Referências

Alves, B., Leão, R., and de Souza e Silva, E. (2007). Caracterizando variáveis de interatividade dos alunos do curso de computação do CEDERJ baseado no servidor multimídia RIO. In *In: V Workshop de Desempenho em Sistemas Computacionais e de Comunicação (WPerformance) / XXVII SBC 2007*.

Benevenuto, F., Rodrigues, M. C., and Almeida, V. (2009). Characterizing user behavior in online social networks. In *IMC '09 Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*.

CEDERJ (2011). Acessado em Fevereiro (2011), `http://portal.cederj.edu.br/fundacao/`.

Duarte, O., Mattos, B., Almeida, J., Almeida, V., Curiel, M., and Bestavros, A. (2008). Hierarchical characterization and generation of blogosphere workloads. In *Elsevier*.

Jonack, M. and Murta, C. (2002). Caracterização de carga de cache da www. In *Revista Eletrônica de Iniciação Científica*.

Lutteroth, C. and Weber, G. (2008). Modeling a realistic workload for performance testing. *Enterprise Distributed Object Computing Conference, IEEE International*, 0:149–158.

Marques-Neto, H., do Valle, E., Castilho, L., Almeida, J., and Almeida, V. (2009). Caracterização hierárquica do comportamento dos usuários de sistemas par-a-par na internet de banda larga. In *27º Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.

MEC (2011). Acessado em Fevereiro (2011), `http://portal.mec.gov.br/`.

Menascé, D., Almeida, V., Fonseca, R., and Mendes, M. (1999). A methodology for workload characterization of e-commerce sites. In *Proceedings of the ACM Conference on Electronic Commerce*, New York. ACM.

Milani, A., Jassó, J., and Suriani, S. (2008). Modeling online user behavior. In *Proceedings of the 2008 IEEE International Conference on e-Business Engineering*, pages 736–741, Washington, DC, USA. IEEE Computer Society.

Mor, E., Garreta-Domingo, M., Minguillón, J., and Lewis, S. (2007). A three-level approach for analyzing user behavior in ongoing relationships. In *Proceedings of the 12th international conference on Human-computer interaction: applications and services*, HCI'07, pages 971–980, Berlin, Heidelberg. Springer-Verlag.

Nascimento, F. and Carnielli, B. (2009). Ensino superior: expansão com qualidade? In *São Paulo em Perspectiva, 2000 - SciELO Brasil*.

Pereira, A., Franco, G., Silva, L., and Jr., W. M. (2004). A hierarchical characterization of user behavior. pages 2–9.

Weirich, R., Gasparini, I., and Kemenczinski, R. (2007). Análise de log para avaliação do comportamento do aluno em um ambiente web. In *XVIII Simpósio Brasileiro de Informática na Educação*. SBC.

BrasilOS: Um Ambiente Didático para Auxílio ao Ensino e Aprendizado de Sistemas Operacionais

Marcos Aurélio Medeiros Silva[1], Rogério Guerra Diógenes Filho[1], Emanuel Feliciano Queiroz[1], Hedwio Carvalho e Silva[1], Verônica Pimentel de Sousa[1,2]

[1] Instituto Federal de Educação Ciência e Tecnologia (IFCE) – 60.050-531 – Fortaleza – CE – Brasil

[2] Universidade Estadual do Ceará (UECE) – Campus Itaperi – Fortaleza – CE – Brasil

{marcosmedeiros31, rogerio.diogenes.f, emanuelfqueiroz}@gmail.com,
{hedwio, veronica}@ifce.edu.br

Abstract. *This paper presents a tool to assist the teaching of Operational Systems about the comprehension of Process Scheduling Algorithms. Called BrasilOS, the tool allows the user to run process and see results with the help of graphics, and with small text alerts, it's possible show the behavior of the algorithms that are often considered difficult by students. Thus the student can view, compare and understand the operation of different algorithms in a clear and objective manner.*

Resumo. *Este trabalho apresenta uma ferramenta para auxiliar o ensino de Sistemas Operacionais, quanto à compreensão dos Algoritmos de Escalonamento de Processos. Chamada de BrasilOS, ela permite que o usuário execute processos e veja resultados com ajuda de recursos gráficos, e com pequenos textos de avisos, é possível mostrar o comportamento dos algoritmos que muitas vezes são considerados difíceis pelos alunos. Desta forma o estudante pode visualizar, comparar e compreender o funcionamento de diferentes algoritmos de uma maneira clara e objetiva.*

1. Introdução

As disciplinas iniciais de ensino de algoritmos e programação de computadores destacam-se como sendo de grande importância para a formação básica dos alunos dos cursos de graduação da área de computação e informática. Isso é confirmado segundo o Currículo de Referência da SBC para Cursos de Graduação em Computação e Informática [SBC, 1999]. Nele, as matérias Linguagens de Programação, Técnicas de Programação e Estruturas de Dados são dadas como parte do núcleo de Fundamentos da Computação.

Esse primeiro contato com o mundo da computação, onde é exigido o desenvolvimento de um bom raciocínio lógico gera um cenário de dificuldade para o aluno [Rocha, 1991], que desestimulado, tende a ser reprovado em disciplinas, desistir do curso, ou seguir adiante sem motivação para os estudos.

Esta dificuldade também é observada nas disciplinas de sistemas operacionais. Para [Prior, 2003] a habilidade de programação de computadores não pode ser adquirida sem um significativo esforço em atividades práticas de laboratório. Segundo [Mercado, 2002], a computação, quando aplicada no ensino, traz flexibilidade na aprendizagem, une a teoria e a prática, onde os alunos aprendem e sabem como, por que, onde e

quando eles aprendem. Nesse cenário, foi desenvolvida uma ferramenta didática com o objetivo de facilitar o aprendizado de sistemas operacionais.

2. Trabalhos Relacionados

O desenvolvimento de softwares com fins educativos em computação, como o MINIX [Tanenbaum, 1996] e o SOsim [Maia, 2001], que visem os temas de maior dificuldade para os estudantes, vem sendo bem trabalhados nos últimos anos. O primeiro tem como principal objetivo permitir ao usuário entender o funcionamento com uma visão mais profunda e complicada, uma vez que requer conhecimentos de linguagens de programação. O segundo, focado em simular o ambiente de sistemas operacionais de forma amigável para o usuário, não entra em detalhes de implementação.

Nesse contexto, outras implementações foram publicadas. Em [Carvalho *et al*, 2006], percebe-se a intenção de apresentar um simulador nos moldes do SOsim, mas que se detém à política de escalonamento de processos e não usa de recursos gráficos e textuais, assim, limitando o fornecimento de informações que tornem mais fácil o aprendizado do aluno.

3. A Ferramenta BrasilOS

Com o intuito de avançar e contribuir no aprendizado de sistemas operacionais, o BrasilOS é uma ferramenta de simulação de sistemas operacionais. O sistema faz uso de recursos gráficos para que a dinâmica de cada algoritmo seja bem visualizada durante sua simulação. Para tornar o software mais completo, são inseridos pequenos textos durante a execução do algoritmo, o que possibilita uma formação conceitual mais abrangente para o efetivo aprendizado. Os recursos gráficos utilizados pela ferramenta apresentam-se como um dos diferenciais do BrasilOS, pois de acordo com [Borba e Villarreal, 2005], o principal *feedback* dado pelos *softwares* se refere ao aspecto visual.

A construção da ferramenta em módulos e fases distintas ocorre devido a preocupação em construir uma aplicação capaz de atender as necessidades dos usuários do ponto de vista didático. Nesse sentido, é parte integrante deste projeto a utilização da ferramenta e a aplicação de questionários de avaliação nas turmas de sistemas operacionais do IFCE a fim de coletar a experiência do usuário, as qualidades identificadas e as eventuais dificuldades encontradas.

Implementada em C#, a fase atual contempla o módulo de algoritmos de Escalonamento de Processos, com simulação dos algoritmos FCFS (*First Come, First Served*), SJF (*Shortest Job First*), alternância circular e prioridade. Além disso, contém textos objetivos para bom embasamento teórico e um guia de utilização da ferramenta, tornando possível que todos os seus recursos sejam bem utilizados.

3.1. Simulação de algoritmos de escalonador de processos

Em sistemas multiprogramáveis, torna-se necessário a execução de vários programas que são executados por vários processos distintos [Deitel *et al*, 2005]. Em grande parte das aplicações tornar-se-ia inviável esperar que cada processo terminasse de executar para outro começar. Pensando nisso, todos os processos são executados de maneira a simular um paralelismo. A tarefa do escalonador de processos é otimizar ordem e tempo de execução de cada processo. Para isso, existem vários algoritmos.

A ferramenta BrasilOS se apresenta essencial para compreender gerenciamento de processos, pois considera várias características do processo, como: tamanho, tipo, prioridade, tempo de CPU, tempo em que permanece bloqueado e tempo de alternância entre processos. Esses parâmetros podem ser configurados pelos usuários bem como a quantidade de processos e qual algoritmo simular.

A Figura 1 mostra como o usuário pode configurar estas opções. Para isso, é necessário que as informações de cada processo sejam adicionadas antes de inseri-lo no escalonador. Após adicionar os processos necessários para a simulação é possível selecionar o algoritmo de escalonamento na lista de seleção do lado direito da imagem. A partir daí, o usuário pode executar a simulação e comparar os resultados.

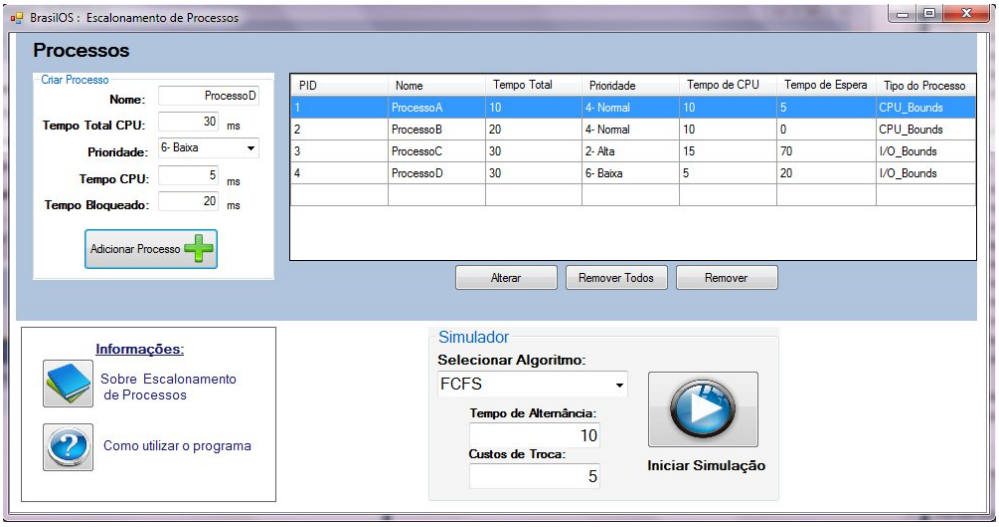


Figura 1 – Interface BrasilOS para simulação de escalonamento de processos

A Figura 2 mostra os resultados dos algoritmos simulados. São apresentadas informações sobre os processos como: quantidade de processos e identificadores de processos executados. Além disso, informações relacionadas ao tempo do processo: processo/unidade de tempo, tempo médio de execução por processo e tempo total do algoritmo. Esse conjunto de informações possibilita maior entendimento do aluno ao analisar e comparar os resultados de diferentes tipos de simulação.

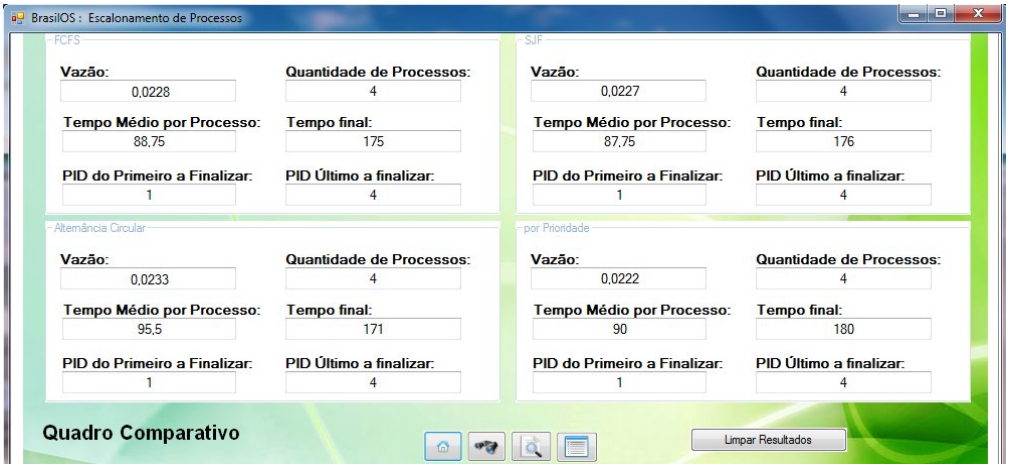


Figura 2 – Resultado da simulação de escalonamento de processos

3.2. Resultados alcançados

Na aplicação dos questionários, para um grupo de estudantes da disciplina de Sistemas Operacionais, foram considerados os quesitos: manuseio, recursos gráficos, aprendizado e expectativas. Em um esquema de avaliação baseado nas opções ruim, regular, bom e ótimo, o simulador foi considerado ótimo por 40% do grupo pesquisado, bom por 58% e regular por 2%. Nenhum dos estudantes considerou a ferramenta ruim. As expectativas foram consideradas atingidas por 85% desse mesmo grupo.

4. Conclusões e Trabalhos Futuros

O BrasilOS mostra-se capaz de contribuir com um melhor aproveitamento dos alunos durante as aulas, além de funcionar como uma ferramenta de apoio fora da sala de aula. A abordagem teórica, os recursos gráficos, as animações e os *pop-ups* utilizados na ferramenta permitem um estudo mais agradável.

Com intuito de aumentar a contribuição com o ensino de Sistemas Operacionais, tem-se como objetivo de trabalhos posteriores a ampliação do escopo da ferramenta para contemplar outros algoritmos de escalonamento de processos, além do gerenciamento de memória. A produção de tutoriais escritos e aplicá-los em mini-cursos dentro da universidade e em eventos como encontros de estudantes e de iniciação científica também compõem os trabalhos futuros.

Referências

Borba, M. C.; Villareal, M. E. “Humans-With-Media and the Reorganization of Mathematical Thinking: information and communication technologies, modeling, experimentation and visualization”. v. 39, New York: Springer, 2005.

Carvalho, D. S., Balthazar, G. R., Dias, C. R., Araújo, M. A. P., Monteiro, P. H. R. (2006) “S²O: Uma Ferramenta de Apoio ao Aprendizado de Sistemas Operacionais”. http://www.natalnet.br/sbc2006/pdf/arq0107.pdf

Deitel, H. M., Deitel, P. J., Choffnes, D. R. (2005) “Sistemas operacionais”. 3. ed. São Paulo, Pearson Prentice Hall.

Maia, L. P. (2001) “SOsim: Simulador para o Ensino de Sistemas Operacionais”. Rio de Janeiro.

Mercado, L. P. L., “Novas Tecnologias na Educação: Reflexões Sobre a Prática”, Maceió: Edufal, 2002.

Prior, J. C. (2003) “Online Assessment of SQL Query Formulation Skills”. In Proceedings of the Fifth Australian Conference on Computing Education. Adelaide, Australia.

Rocha, H. V. (1991) Representações Computacionais Auxiliares ao Entendimento de Conceitos de Programação, UNICAMP.

SBC – Sociedade Brasileira de Computação. (1999) “Currículo de Referência da SBC para Cursos de Graduação em Computação e Informática”.

Tanenbaum, A. S. (1996) “Minix Information Sheet”. http://www.minix3.org/.

Um instrumento de reflexão sobre a ação do homem no meio ambiente desenvolvido no contexto do ensino de Computação

Rosangela L. Lima[1], Heron de S. Marques[1], Alisson P. Quintanilha[1], João Paulo P. Tonelli[2]

[1]Instituto de Computação – Universidade Federal Fluminense (UFF)
24.210-240 – Niterói – RJ – Brasil

[2]TIC/INFRA-TIC – Petróleo Brasileiro S/A (Petrobras) – Macaé – RJ – Brasil

lima@dcc.ic.uff.br, hmarques@ic.uff.br, alissonpq@id.uff.br, joaotonelli@petrobras.com.br

Abstract. *This article is about a game developed in the context of a didactic experiment in which research-action in classrooms was adopted as methodology. That experiment targets the students of the Computer Science Course of the Fluminense Federal University, enrolled in an elective discipline in which the theme Computing and Environment was developed. As a result, it has been produced a game, presented in this article, which addresses environmental problems focusing on mankind's responsibility towards the sustainability of life on Earth.*

Resumo. *Este artigo trata de um jogo desenvolvido no contexto de uma experiência didática em que se adotou como metodologia a pesquisa-ação em sala de aula. Essa experiência teve como público-alvo os alunos do curso de Ciência da Computação da Universidade Federal Fluminense inscritos em uma disciplina optativa na qual se desenvolveu o tema Computação e Meio Ambiente. Como resultado, foi produzido um jogo, apresentado neste artigo, que aborda a problemática ambiental focalizando a responsabilidade do homem com a sustentabilidade da vida no Planeta.*

1. Contextualização e justificativa

O tema – Computação e Meio Ambiente – surge a partir da necessidade de abordar a sustentabilidade ecológica para a formação de profissionais da área de computação. Constitui-se objeto de uma disciplina optativa do currículo de um curso de graduação em Ciência da Computação oferecida a cada semestre letivo desde 2010. Essa proposição se justifica principalmente pelo fato de a indústria de informática, caracterizada pelo seu crescimento exponencial em todo o mundo, ter-se tornado uma grande fonte de problemas relacionados ao resíduo eletrônico depositado no meio ambiente.

Dentre os problemas focalizados, destacam-se o descarte e a reciclagem de lixo eletrônico. Ordinariamente, este lixo é separado inconforme às normas relativas a saúde e segurança, sem receber o tratamento adequado, e é depositado em locais impróprios. Esta prática nociva atinge a espécie humana diretamente enquanto autora e também indiretamente pela contaminação dos recursos essenciais à manutenção da vida na Terra. A resolução de problemas como este depende de altos investimentos em tecnologia, no

que diz respeito a processos de separação e desmonte dos equipamentos, aliados a ações integradas entre governo e comunidades.

> A reciclagem de computadores não é algo fácil. É uma tarefa que deve ser realizada de maneira apropriada. Se um componente ao ser retirado é danificado ele perde o seu valor econômico. Tem-se como agravante, também, que é difícil separar os materiais presentes no computador para reciclagem. Basta pensar no circuito impresso. [Beiriz 2005]

Atualmente, na maioria dos países que não dominam as tecnologias mais avançadas de separação, a reciclagem é realizada por meio de coleta e desmanche manual destes resíduos, constituindo-se em atividade informal realizada por pessoas despreparadas em condições inadequadas. O resultado deste trabalho insalubre é uma separação ineficaz do material coletado. Essa situação advém da omissão do poder público, pois falta legislação que regulamente essa atividade, bem como ação fiscalizadora que garanta seu cumprimento. [UNEP-STEP 2009]

No Brasil, foi sancionada recentemente a lei que cria a Política Nacional de Resíduos Sólidos[1] e já existem alguns exemplos que podem ser seguidos, tal como a iniciativa pioneira da Universidade de São paulo (USP) que vem obtendo sucesso com o Projeto de Cadeia de Transformação de Lixo Eletrônico. Atuando em práticas de reuso, descarte e reciclagem dos bens de informática e telecomunicações que ficam obsoletos, teve sua implementação iniciada dentro da própria universidade. Segundo dados da Fundação Getúlio Vargas (FGV), apresentados no referido projeto, o Brasil, em uso de computadores, já ultrapassou 50 milhões em 2008. [CCE/USP 2008]

No contexto de uma economia baseada em um modelo em que se programa a obsolescência dos equipamentos eletroeletrônicos, estima-se que são geradas no mundo, a cada ano, 20 a 50 milhões de toneladas de lixo eletrônico [Greenpeace 2010]. As consequências da manutenção desse modelo, que favorece e investe no consumo desenfreado, vem causando graves prejuízos para a sociedade como um todo. É necessário que se adquira uma nova percepção de mundo, dotada de uma visão processual e sistêmica, a fim de impulsionar uma mudança estrutural de valores capaz de transformar o estilo de vida das pessoas . [Lima 2009]

Entende-se que a transformação desse estilo de vida possibilitará ao indivíduo da sociedade contemporânea interferir na crescente ciranda financeira mantenedora do consumo desenfreado. Somente uma mudança de valores desencadeará transformações efetivas na estrutura da sociedade. Este é um cenário que indica a complexidade de um sistema econômico, regido por uma demanda artificialmente criada, que se vale da mídia como principal veículo, para gerar um consumo exacerbado. Pela mídia as pessoas são levadas a comprar o que de fato não precisam. Essa demanda baseada em valores supérfluos sustenta o fluxo financeiro base do regime capitalista. [Leonard 2007]

Com o intuito de auxiliar na criação de uma cultura baseada no agir em prol do bem comum, focalizando a sustentabilidade da vida no planeta, buscou-se na referida disciplina desenvolver a aprendizagem do conteúdo proposto através de estudos que se transformassem em ações concretas, identificadas como promotoras de mudanças essenciais na relação entre computação e meio ambiente.

1 Notícia publicada em 02/08/2010 fonte:http://www.globalgarbage.org/blog

Para iniciar a discussão sobre o tema da disciplina buscou-se inicialmente desenvolver uma etapa de sensibilização com o propósito de fazer com que os alunos pudessem refletir sobre o impacto da sua ação como profissional de informática, não somente em termos individuais e locais, mas também de um modo global, através da percepção do mundo como uma rede de relações, cuja configuração indica que a ação de um indivíduo sempre interfere no outro e vice-versa.

Após a etapa de sensibilização, foram formados grupos de trabalho em que os alunos se organizaram em torno de projetos concretos decorrentes das referidas ações. O jogo, objeto deste artigo, integrou um painel sobre sustentabilidade da vida no Planeta apresentado na referida instituição de ensino superior durante a Semana Nacional de Ciência e Tecnologia em 2010, ocasião em que foi apreciado e avaliado por alunos de diferentes níveis de formação, oriundos de escolas da região e da própria instituição – vide figura 1. Das trinta e duas fichas de avaliação respondidas pôde-se inferir que o jogo agradou o público, tendo somente 5% de rejeição. Outro destaque é que 90% dos estudantes que responderam ao questionário consideraram que o jogo lhes ensinou algo que desconheciam. Nas respostas à questão aberta foram registradas sugestões de melhorias, que já estão sendo implementadas.

A principal motivação do grupo que desenvolveu o jogo apresentado neste artigo ocorreu a partir de diversificadas atividades pedagógicas, que incluíram a leitura do livro "Capitalismo Natural" [Hawken, Lovins and Lovins 1999], escolhido para colocar os estudantes diante da necessidade de refletir sobre a responsabilidade de cada um perante a eco-sustentabilidade.



Figura 1. Apresentação do jogo no campus da UFF

2. A complexidade da realidade e a metodologia adotada

[Barnett 2005] relaciona três grandes responsabilidades da universidade sobre o que ele define como *supercomplexidade*[2]. São elas: (I) ajudar a gerar a supercomplexidade,

2 Ordem maior de complexidade em que se tem que descobrir formas de viver e de prosperarem um mundo onde as estruturas são testadas e desafiadas.

interrogar e informar a respeito; (II) abrir caminhos e criar histórias justificáveis sobre o mundo e sobre o ser humano; e (III) desenvolver as estruturas de ego de modo que os indivíduos possam viver em paz uns com os outros e consigo mesmo, ainda que, em meio a essa *supercomplexidade*, possam fazer intervenções significativas e reflexivas do mundo. Ele diz ainda que:

> Uma universidade digna do nome, portanto, certamente irá acrescentar complexidades a este admirável mundo novo (não seria uma "universidade" se ela simplesmente ecoasse as descrições existentes no mundo); mas por meio do ensino superior que ela oferece, ela também fornece os meios, as qualidades humanas de coragem e resistência que permitirão que os indivíduos enfrentem e vivam neste mundo.

A compreensão da sala de aula pela ótica da complexidade, como um ambiente constituído de relações entre indivíduos autônomos e implicados na construção do conhecimento, evidencia a necessidade de expor o estudante a um conjunto ampliado de informações, métodos de aprendizagem e ferramental tecnológico, de modo que, sob a orientação docente, ele possa se defrontar com desafios do mundo real, pois "o currículo é determinado pela vida, pelos desafios que se encontram no momento, dados pelo ambiente". [Alves 2002]

> Autores como Ilya Prigogine, Isabelle Stengers, Edgar Morin têm demonstrado a necessidade de considerar a complexidade do real não como moda passageira, mas como necessidade de se superar o reducionismo das ciências puras e "duras" que pretendem dominar seu objeto e generalizar suas conclusões. [Morin 2004]

No intuito de trazer para a sala de aula a complexidade da realidade, buscou-se adotar, no desenvolvimento da disciplina, uma metodologia que focalizasse o processo de construção do conhecimento na rede de relações entre alunos e professor e não no processo reducionista de transmissão de conteúdos.

Como seres plurais em sua relação com os outros e com o mundo, os estudantes foram situados no contexto como agentes de mudança, autônomos, embora dependentes do outro. Utilizar a proposta da pesquisa-ação visou uma mudança de foco, do objeto para as relações entre atores, o que permitiu além de romper com a concepção determinista-mecanicista, verificar que de fato não se pode predeterminar os resultados finais do processo educativo.

Pela compreensão da complexidade da interação, os estudantes, em sua relação com os outros e com o contexto, em grupos de no máximo quatro integrantes, foram capazes de escolher livremente os temas de seu interesse e as ações que culminaram no aprendizado por projeto dentro do tema Computação e Meio Ambiente.

As orientações fornecidas aos estudantes para a construção dos projetos consistiram dos seguintes pontos: Identificar claramente as necessidades do grupo: definir o objetivo geral da pesquisa; compreender bem a finalidade do projeto como uma mudança para conduzir a transformações completas ou parciais, ou melhorias em situações e comportamentos; definir tema, título, planejamento e forma de implantação; indicar forças humanas e dinâmicas para a realização do projeto; determinar a duração do projeto em um semestre letivo; definir os papéis dos participantes, que como autores participam ativamente com as suas reflexões, conhecimentos, falas e escutas.

3. A ação pedagógica: O jogo no contexto da disciplina

É importante destacar os diversos aspectos relevantes desta experiência em sala de aula. O primeiro se relaciona ao aprendizado do grupo considerando o desenvolvimento do jogo como atividade de avaliação da disciplina. Ao escolher o jogo como ferramenta para desenvolver a atividade, os alunos se tornaram capazes de:

a) construir um projeto detalhado, pela sistematização das etapas de desenvolvimento, aprofundando-se no conhecimento do tema abordado, uma vez que realizaram pesquisas aprofundadas para construir o conjunto de conhecimentos necessários ao desenvolvimento da história do jogo;

b) compartilhar a escolha da tecnologia necessária à construção do jogo, conjugando habilidades e conhecimentos preexistentes;

c) desenvolver uma proposta que atinge adequadamente o público jovem, pois como nativos digitais[3] [Mattar 2010], utilizaram elementos que fazem parte da rotina do público alvo, também nativos digitais;

d) produzir uma atividade lúdica envolvente e atrativa tendo por parâmetro as exigências e expectativas que eles mesmos têm enquanto usuários de jogos.

Um segundo aspecto trata da possibilidade da aplicabilidade do conteúdo da disciplina como ação transformadora com foco na educação para a sustentabilidade, permitindo aos estudantes a participação em uma atividade de pesquisa em que não somente se buscou a pesquisa pela pesquisa, como geralmente acontece nas atividades de aprendizagem, mas sim a contribuição direta através da pesquisa aplicada, cujo foco está mais ligado à efetividade da ações.

Um outro importante aspecto no desenvolvimento do jogo como ferramenta educativa é a possibilidade de colocar os responsáveis pelo seu desenvolvimento frente a diferentes ações, tais como: a elaboração da proposta pedagógica, a construção da ferramenta e a pesquisa para aprofundar o conhecimento do tema abordado. O jogo apresentado neste trabalho pode ser chamado de 'ilha de expertise' que, segundo [Mattar 2010], é aquele que possibilita ao jogador o acesso a livros, sites, notícias e outras fontes, direcionando o jogador a um aprendizado em um nível mais complexo.

O último aspecto identificado como foco principal do desenvolvimento do jogo foi a possibilidade de utilizá-lo para colocar o jogador perante a necessidade de conhecer o que [Morin 2000] chamou de identidade terrena, ou seja, o reconhecimento do indivíduo como parte integrante do Planeta. A partir desse reconhecimento, o jogador busca interagir com o aspecto planetário, partindo do local para o global e vice-versa, adquirindo consequentemente a noção da crise que é hoje partilhada por toda a humanidade.

Durante o desenvolvimento da disciplina as principais etapas solicitadas ao grupos de estudantes foram duas atividades consideradas fundamentais: o planejamento da proposta de projeto e a construção do mapa conceitual. A seguir serão apresentados o projeto e o mapa conceitual do jogo “Terra: Preserve ou Morra”, um dos produtos da ação pedagógica anteriormente referida.

3 Segundo Mattar : “são pessoas que incorporaram mídias digitais em seu cotidiano de maneira significativa”

O projeto desenvolveu-se dentro do tema Preservação do Meio Ambiente e Sustentabilidade com o seguinte título: “Terra: Preserve ou Morra”.

i. O objetivo geral do projeto foi ampliar a consciência ambiental, alertando para os perigos iminentes a que o planeta está exposto. Utilizou-se uma abordagem informativa e provocativa, levantando-se as questões mais relevantes da atual pauta ambiental, expondo os principais problemas do presente, os graves prognósticos e as mudanças que se fazem progressivamente mais urgentes e imperativas.

ii. O objetivo específico consistiu na construção de um jogo educativo que pudesse contribuir para a expansão da consciência ambiental dos jogadores de maneira lúdica, agradável e eficaz. Tendo como público alvo estudantes do ensino médio e superior, “Terra: Preserve ou Morra” objetiva incutir nos futuros profissionais a devida atenção às questões ambientais, ampliando a visão sobre a necessidade de uma melhor interação entre homem e meio ambiente, de modo a alertar sobre os perigos do padrão de comportamento vigente na sociedade contemporânea.

iii. O desenvolvimento do jogo justificou-se pelo fato de que jogos eletrônicos constituem uma importante e eficaz ferramenta para a disseminação do conhecimento. Por meio de uma atividade prazerosa, consegue-se dispor afetivamente os jogadores para que possam absorver o conteúdo transmitido com mais disposição e satisfação.

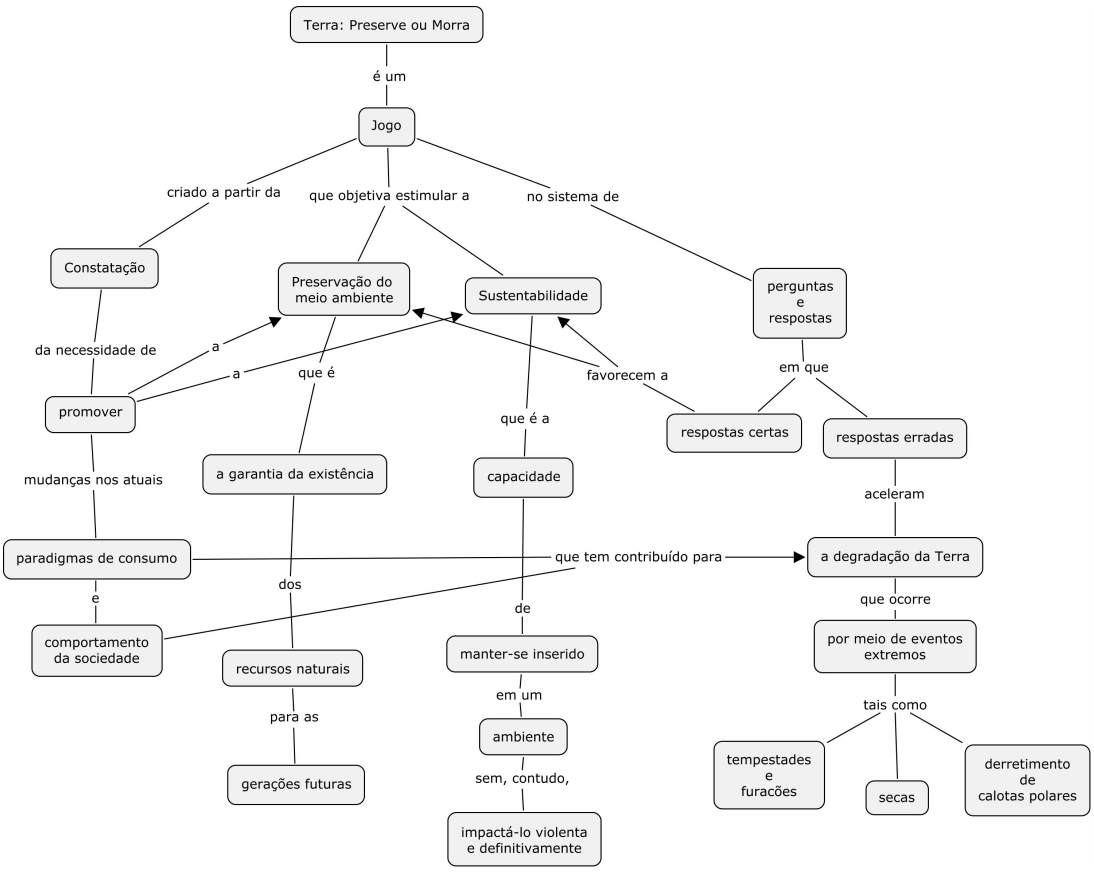


Figura 2. Mapa Conceitual do projeto

O mapa conceitual, baseado na teoria do conhecimento significativo[4], foi

4 Teoria desenvolvida por David Ausubel

utilizado como um método de representação gráfica para organizar e representar o conhecimento. A figura 2 apresenta o mapa conceitual que consistiu da primeira estratégia para a organização do corpo de conhecimentos utilizado pelo grupo. Através da construção do mapa, os alunos puderam relacionar os principais conceitos envolvidos com o tema escolhido. Construído à medida que o conhecimento se ampliava através das pesquisas desenvolvidas, serviu de base para a implementação do jogo.

4. Desenvolvimento do jogo

Este jogo visa representar a interação de cada indivíduo com o ambiente por meio de respostas a questões relacionadas à preservação ambiental e ao desenvolvimento sustentável do Planeta. O objetivo principal do jogo é garantir a perenidade do planeta Terra. Este é o prêmio conquistado pelos jogadores: a garantia de um futuro para o Planeta. A meta é fazer com que o jogador entenda que o futuro será tão bom quanto forem as ações praticadas no presente.

O jogo é desenvolvido com base no sistema de perguntas e respostas. O objetivo é aferir não só o conhecimento dos jogadores, mas a maneira de pensar e agir desses jogadores em relação ao tema. Para isso, são apresentadas perguntas objetivas e subjetivas. Podem jogar até quatro pessoas em turnos. Cada um dos jogadores responde a uma pergunta alternadamente, até o término do jogo. As perguntas são selecionadas aleatoriamente de um banco de questões e cada uma oferece 4 opções de resposta. Sempre haverá uma alternativa totalmente adequada e uma totalmente inadequada. As demais alternativas oferecem respostas intermediárias.

O mecanismo utilizado para calcular a pontuação dos jogadores funciona da seguinte forma: cada alternativa possui um modificador associado a ela. O modificador confere um percentual de pontuação e reflete o nível de adequação da resposta, tendo em vista a proposta do jogo, que é preservação e sustentabilidade. Esse modificador está restrito ao intervalo entre -1 (inclusive) e 1 (inclusive). A pontuação negativa é usada para acionar os eventos, enquanto que a pontuação positiva é somada ao placar dos jogadores.

Essa estratégia procura sensibilizar os jogadores ao mostrar, tomando por medida o modo de pensar e agir deles, o quanto estão, ou não, contribuindo para a degradação da Terra. O jogo evidencia a interação que cada indivíduo tem com o meio e, dependendo de como se caracteriza essa interação, acontecem eventos, que evidenciam os impactos causados ao Planeta.

O jogo possui placar individual para mostrar a pontuação de cada jogador de maneira independente, e placar coletivo, o qual mostrará a soma das pontuações dos jogadores. Desta maneira, procurou-se ressaltar o aspecto coletivo das ações. Se a resposta é adequada, o pensamento do jogador é adequado e contribui para a preservação da Terra. Logo, todos ganham. O termômetro, que indica o nível de destruição do Planeta, reforça essa ideia. Existe somente um termômetro para todos os jogadores. Logo, qualquer jogador que escolha uma resposta inadequada contribui para o crescimento da pontuação negativa.

Os eventos são acionados quando se atinge determinada faixa no termômetro. São ao todo nove eventos. O último evento coincide com o máximo de pontuação negativa e, portanto, com a derrota dos jogadores. Os eventos mostram catástrofes climáticas e desastres ecológicos e sociais decorrentes das ações humanas. Assim,

evidencia-se que o modo de pensar do jogador, refletido na resposta dada, influencia negativamente o destino da Terra.

Em contrapartida, caso o jogador responda adequadamente às perguntas, serão fornecidas mensagens de incentivo. Essas mensagens apresentam ações, eventos ou políticas favoráveis ao meio ambiente que já vêm sendo implantadas em algum lugar do Planeta. O texto de cada mensagem é uma adaptação de manchetes ou notícias já veiculadas na mídia. No caso de o jogador escolher uma alternativa próxima da mais adequada, ele será parabenizado, mas será também incentivado a aprimorar seu modo de pensar, recebendo um *link* para uma fonte de pesquisa. Para a seleção destas indicações foram consultadas cerca de 100 fontes de pesquisa, incluindo livros, revistas e, majoritariamente, sites da Internet.

Os eventos foram programados de modo a apresentar uma cronologia de mudanças climáticas. Inicia-se com as mudanças já ocorridas até o presente e prossegue-se com as perspectivas futuras para a Terra. Procurou-se evidenciar que se nada for feito para evitar essas alterações, as consequências se tornarão catastróficas. Desta forma escolheu-se trabalhar, no caso de eventos ainda hipotéticos, com os piores prognósticos, aqueles que prevêem as piores consequências para o futuro e que acontecerão somente se nada for feito para evitar. Como o título “Terra: Preserve ou Morra” enfatiza, o homem está morrendo por suas próprias mãos, devido às suas próprias atitudes, destruindo seu próprio *habitát* e dos demais seres vivos.

5. O passo a passo do jogo

A tela inicial do jogo, mostrada no canto superior esquerdo da figura 3, apresenta três opções: Instruções, Jogar e Créditos. Em créditos, o jogador poderá conhecer a equipe responsável pela criação do jogo, a orientadora e a universidade onde nasceu este projeto. Em jogar, inicia-se a partida. Em instruções, o jogador será conduzido por telas explicativas nas quais poderá se familiarizar com a interface do jogo.

Ao se iniciar o jogo escolhe-se o número de jogadores. Em seguida aparece como apresentada no canto superior direito da figura 3: a tela do jogo, que contém o mapa-*mundi* ao fundo e, sobre ele, o quadro de perguntas. No topo da tela, ao centro, é mostrada a pontuação coletiva. Ao lado desta, à direita e à esquerda, são mostradas as pontuações individuais.

O termômetro, localizado no lado esquerdo da tela, mostra o nível de destruição atingido. Os eventos disparados no jogo são marcados por pequenas figuras sobre o mapa-*mundi* acompanhadas de um box descritivo. O canto inferior esquerdo da figura 3 mostra a tela de jogo após erro, com box de evento.

O jogador pode ler novamente o texto do evento, bastando clicar sobre a figura que representa o evento no mapa. No canto inferior direito da figura 3, apresenta-se a tela de jogo após erro, com os eventos no mapa.

Na figura 4, à esquerda e à direita, são apresentadas respectivamente as telas que indicam a derrota e a vitória dos jogadores.



Figura 3. Telas do jogo. Projeto gráfico: Pedro Thiago Mourão.

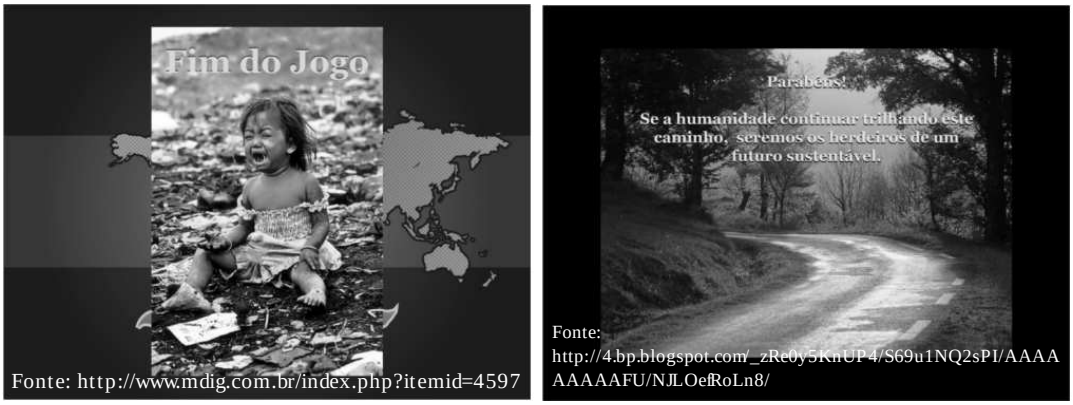


Figura 4. Finais do jogo

6. Conclusão

Em termos pedagógicos, o planejamento e desenvolvimento do jogo em sala de aula propiciou o estreitamento da relação professor-alunos, possibilitando uma interação onde a empatia teve um importante papel no decorrer da proposta pedagógica. Nessa relação, os ganhos obtidos foram efetivos em termos de aquisição de conhecimentos, uma vez que se buscou aprofundar as pesquisas sobre a problemática relacionada à intervenção sem limites do homem rumo à destruição do ambiente em que vive.

Os estudos desenvolvidos também fortaleceram a tomada de consciência sobre a

imprevisibilidade do futuro da humanidade. Em termos da questão ambiental, o trabalho está totalmente concernente com o momento dramático a que toda humanidade está exposta, haja vista o que vem acontecendo recentemente no mundo devido à omissão das autoridades competentes.

Finalmente, como se pode comprovar na prática, verificou-se que é possível transformar o modelo tradicional de aprendizagem, em que o aluno de modo passivo recebe os conteúdos transmitidos pelo professor. Modificando essa prática, no caso específico, adotou-se o método de pesquisa-ação. Ganham os alunos e o professor. O professor efetivamente se transforma em um orientador da busca do aprendizado pelos estudantes, em um processo de construção do conhecimento significativo.

Referências

Beiriz, F. A. S. (2005). “Gestão Ecológica de Resíduos Eletrônicos: Proposta de Modelo Conceitual de Gestão”, Ed. Latec, Brasil.

Morin, E. (2000). “Os sete saberes necessários à educação do futuro”, Ed. Cortez, Brasil.

CCE/USP (2008). “Projeto de Criação de Cadeia de Transformação de Lixo Eletrônico da Universidade de São Paulo”, Prêmio Mário Covas, Ed. USP, Brasil.

Hawken, P.; Lovins, A. and Lovins, L. H. (1999). “Capitalismo Natural”, Ed. Cultrix - Amana Key, Brasil.

Leonard, A. (2010). “Story of Stuff Project’s”, http://www.storyofstuff.com/, Maio.

Lima, R. L. (2009). “O Lixo Computacional na Sociedade Contemporânea”, Ed. ENINED, Brasil.

Mattar, J. (2010). “Games em educação: como os nativos digitais aprendem”, Ed. Pearson Prentice Hall, Brasil.

UNEP/STEP (2009). “RECYCLING – From e-waste to resources - Sustainable Innovation and Technology Transfer, Industrial Sector Studies”, Ed. UNEP/STEP, Alemanha.

Alves, R. (2002). “Por uma educação romântica”, Ed. Papirus, Brasil.

Barnett, R. (2005). “A Universidade: em uma era de complexidade”, Ed. Anhembi Morumbi, Brasil.

Greenpeace Brasil (2010). “Adeus ao lixo tóxico: saiba o que grandes marcas podem fazer pela sua saúde”, http://www.greenpeace.org/brasil/pt/Noticias/Adeus-ao-lixo-toxico-saiba-o-que-grandes-marcas-podem-fazer-pela-sua-saude/, Maio.

Educação para além do mercado de trabalho

David Moises Barreto dos Santos

Departamento de Ciências Exatas – Universidade Estadual de Feira de Santana (UEFS)
Avenida Transnordestina, s/n – Novo Horizonte – 44.036-900 – Feira de Santana – BA

davidmbs@uefs.br

Abstract. *This article questions the importance of a broader education, without to neglect aspects inherent in the being human. For this, we point out some limitations of education pressured by the market and discuss the importance of an education that goes beyond the work and its technical issues, covering other dimensions of being human. Finally, we contextualize the humanistic formation in publications and guidelines of Computing.*

Resumo. *Este artigo questiona a importância de uma educação mais ampla, que não negligencie aspectos inerentes ao ser humano. Para isso, apontamos algumas limitações da educação pressionada pela economia de mercado, além de discutir a importância de uma educação que vá além do trabalho e de suas questões técnicas, que contemple também outras dimensões do ser humano. Finalmente, contextualizamos a temática na formação humanística em publicações e diretrizes de referência da Computação.*

1. Introdução

Atualmente, dado o cenário hostil que vivemos, virou rotina fazermos e ouvirmos queixas a respeito da falta de ética, respeito e amor para com o outro. Bauman (2008) aponta em direção a uma verdadeira degradação da condição humana, onde os vínculos humanos se revelam frágeis, rompidos facilmente, bastando que o outro não mais satisfaça aos desejos aspirados pelo indivíduo. Ele é mais incisivo, quando diz que "na sociedade de consumidores, ninguém pode se tornar sujeito sem primeiro virar mercadoria. A característica mais proeminente da sociedade de consumidores – ainda que cuidadosamente disfarçada e encoberta – é a transformação dos consumidores em mercadorias" [Bauman, 2008, p. 19]. De fato, é comum ser tratado assim, em que o valor está no *ter*, sobretudo quando se tem dinheiro e prazer a oferecer, e não no *ser*. E o pior, isso tem sido cada vez mais recorrente.

Reflexos como estes também têm sido notados no âmbito educativo. Alunos que tratam professores como mercadoria e vice-versa, cada qual exigindo do outro que "funcione" conforme o manual de instrução sob pena de devolução, de descarte, traduzido em termos de indiferença, agressão e até homicídio.

Podemos então questionar qual nosso papel, enquanto educadores, diante de problema tão proeminente. Não podemos fechar os olhos perante tal realidade, abrindo-os apenas para as questões relativas estritamente à técnica, aos conteúdos pertinentes

exclusivamente à Computação – tema também abordado recentemente na lista da SBC[1] por ocasião da reforma de diretrizes curriculares. Não queremos afirmar, em hipótese alguma, que o que diz respeito à Computação é irrelevante, pelo contrário, é muitíssimo importante, porém, é preciso ampliar o horizonte, buscar compreender melhor o ser humano para além de mera mão-de-obra.

Em outras palavras, não podemos dizer que o problema não é nosso; pode até ser que não seja mesmo, mas ele bate à nossa porta, nos envolve, nos exige uma resposta. Frankl (1991) afirma que não somos nós que questionamos a vida, mas é a vida que nos questiona e nos intima a dar uma resposta. Assim, nosso objetivo neste trabalho, ainda que embrionário e sem pretensões de oferecer uma solução, é discutir a relevância de uma formação mais ampla – no sentido humanístico – no contexto da Computação.

Para tanto, na Seção 2, descrevemos um breve histórico da educação ocidental, além da relação da educação e trabalho e suas limitações, especialmente, na Computação. Na Seção 3, discutimos a importância de uma formação para além do mercado de trabalho para, na Seção 4, apontarmos um breve diagnóstico da formação humanística na Computação. Finalmente, traçamos as conclusões finais na Seção 5.

2. Educação para o mercado de trabalho

Antes de prosseguirmos com a discussão, é importante afirmar que o ser humano pode ser caracterizado através de quatro dimensões [Costa, 2004]: (1) *logos*, dimensão do pensamento, relativa à razão; (2) *pathos*, dimensão do sentimento, relativa à afetividade, geradora da empatia e antipatia da relação consigo mesmo e com os outros; (3) *eros*, dimensão do desejo, relativa às pulsões, à corporeidade; e (4) *mytho*, dimensão da relação do homem com o mistério da vida e da morte, do bem e do mal.

A educação ocidental, em diferentes momentos da história, deu destaques distintos a cada uma dessas dimensões. Na verdade, na Grécia era buscado um equilíbrio entre elas. Na Idade Medieval, quando surge a universidade, todas as quatro também eram valorizadas, porém o *mytho* se sobressaía. Já na Modernidade, principalmente, após a Revolução Industrial, o *logos* foi predominante e tem sido até os dias atuais. Mais especificamente, a partir desta época, surge a relação da educação para o trabalho, moldada de acordo com a economia de mercado [Santos, 1989].

Essa realidade não é exclusiva da universidade, mas também da escola, que não está em nosso escopo; todavia, apenas a título de exemplo, podemos citar escolas de ensino médio cuja meta primordial é fazer com que os estudantes sejam aprovados em vestibulares e bem colocados no Enem (Exame Nacional do Ensino Médio). Em outras palavras, o direcionam quase que exclusivamente para o profissional e não para a vida, não importando tanto a vocação do educando, suas angústias, mas o renome da universidade na qual será aprovado para estampar em outdoors, buscando elevar o "gabarito" comercial da escola.

Já no âmbito universitário, e aqui já mergulhando na Computação, notamos, por um lado, a ênfase que é dada à questão da técnica articulada com uma certa preocupação com o mercado de trabalho. O que não poderia ser diferente, afinal, como aponta a

[1] https://grupos.ufrgs.br/mailman/private/sbc-l/2011-March/015368.html. Acesso: 30 mar 2011.

própria chamada de trabalhos deste evento, a Computação tem sido o "terceiro pilar" da pesquisa científica, ajudando no avanço de outras áreas do conhecimento, além de também ser fundamental para o desenvolvimento econômico, tecnológico e social de um país. Sendo assim, nossos educandos investigam os principais algoritmos para solução de problemas, aprendem a construir colaborativamente – seguindo processos adequados – os mais variados e sofisticados softwares, se debruçam para entender a lógica de diferentes arquiteturas de hardwares, entre tantas outras tarefas.

Entretanto, por outro lado, não raro observa-se uma certa negligência – por razões até históricas, como as que descrevemos – quanto a importância de uma formação mais ampla, que abarque várias dimensões do ser humano ou, mais especificamente, do educando, que não apenas a profissional. Boaventura Santos (1989) afirma que formação e desempenho profissional tendem a fundir-se num só processo produtivo, demandando educação permanente, reciclagens, etc. Mas será que, mesmo sem ser propositalmente, aquele educando que está ali só pode ser vislumbrado apenas como mão-de-obra bruta a ser lapidada, como um novo talento para a Computação? A própria ciência muitas vezes se encarregou de imprimir, através de sua racionalidade analítico-instrumental, uma visão exageradamente positivista e parcial do homem, como apontaram – e combateram tal percepção – intelectuais como Husserl e Heidegger.

Embora os progressos científico-tecnológicos sejam de grande valor, há uma preocupação cada vez mais crescente com o desenvolvimento do ser humano, que, por diferentes acontecimentos nas variadas esferas (social, econômica, política, cultural, etc), parece estar em desequilíbrio [Costa, 2004]: na relação consigo mesmo, marcada pela ansiedade e medo; na relação com os outros, cheia de individualismo, força bruta e uso instrumental de outrem; na relação com a natureza, desestabilizando ecossistemas, comprometendo o que era garantia de vida, para se tornar ameaça; na relação com o transcendente da vida, com a perda de sentido e sacralidade da vida.

Diante de tal sociedade, que está perdendo os valores, que a falta de respeito e dignidade é evidente, qual a nossa resposta? Basta formar educandos com grande capacidade de resolver problemas estritamente técnicos e/ou compor currículos com o que há de mais novo na Computação? Claro que aptidões mais gerais como espírito crítico, criatividade, atitude positiva perante o trabalho árduo e em equipe, ambição e motivação pessoal, flexibilidade, capacidade de negociação, entre outras, também são valorizadas, mas, insistimos que, mesmo assim, ainda estão direcionadas às questões técnicas. O espírito crítico apenas o é para questões profissionais (ex. "a solução A é mais adequada que a B para a situação S por isso e aquilo"), a flexibilidade também – será que sinais dessa flexibilidade são demonstrados no trânsito tão estressado de hoje em dia ou nos diversos graus de relacionamentos? –, e assim por diante. Por exemplo, em um trabalho anterior [Santos & Burnham, 2010], ao defrontarmos um dos métodos de ensino-aprendizagem de nosso curso com a teoria de Paulo Freire, constatamos o quanto nos falta valorizar potenciais que não apenas aqueles técnicos.

Reiteramos que todas essas premissas técnico-profissionalizantes são verdadeiramente relevantes e justas, importantes para construção da sociedade. O que colocamos em cheque é se apenas elas são suficientes para agir em prol de uma sociedade mais humana, com ideais de paz, justiça social e solidariedade. Além do bem fazer também precisamos do bem agir. Enquanto que o primeiro está mais para a

eficácia da técnica o segundo está mais para a atitude pautada na ética [Santos, 2010b]. Desta forma renovamos aqui os questionamentos de Costa (2008): que tipo de ser humano queremos formar? Que tipo de sociedade, para cuja construção, queremos contribuir?

3. Educação para além do mercado de trabalho

É inspirado nestes questionamentos que refletiremos sobre uma educação que vá além do mercado de trabalho, e quando falamos de mercado não estamos nos referindo apenas ao meio empresaria, mas também o acadêmico, da pesquisa científica. Desde já, elucidamos que, neste artigo, não pretendemos dar respostas, apontar o que podemos e devemos mudar, mas *tornar explícito que a educação pode ser mais ampla e a sua importância*.

A vida é muito maior que a ciência. Portanto, por que não incluir outras dimensões do ser humano? Por que não discutir também sobre uma ética mais ampla e não apenas ética para a Computação? Por que não debater a afetividade dos relacionamentos, desde os familiares até os desconhecidos? Por que não abordar as necessidades do ser humano como aquelas da Pirâmide de Maslow? Por que não discutir sobre o sentido da vida (morte) e suas incertezas? Questões como estas abrem o leque para uma formação mais ampla, que contemple, por exemplo, as dimensões supracitadas *logos*, *pathos*, *eros* e *mytho*. Essa, todavia, não é a única visão interdimensional do homem, outras perspectivas também são defendidas por organizações como a Organização Mundial de Saúde ou autores como Morin (2000) e Frankl (1991) – por limitações de espaço, não entraremos em detalhes. De antemão deixamos claro que nossa intenção, neste momento, não é discutir a melhor teoria ou a mais adequada, mas reforçar que o ser humano não pode ser compreendido parcialmente, mas de uma maneira mais abrangente.

Sendo assim, para Morin (2000, p. 11) "uma educação só pode ser viável se for uma educação integral do ser humano. Uma educação que se dirige à totalidade aberta do ser humano e não apenas a um de seus componentes". De outra forma, uma educação voltada apenas para o domínio da técnica, para o mercado de trabalho não é suficiente. Paulo Freire entra nessa discussão e critica tal postura:

> transformar a experiência educativa em puro treinamento técnico é amesquinhar o que há de fundamentalmente humano no exercício educativo: o seu caráter formador. Se se respeita a natureza do ser humano, o ensino dos conteúdos não pode dar-se alheio à formação moral do educando. Educar é substantivamente formar. Divinizar ou diabolizar a tecnologia ou a ciência é uma forma altamente negativa e perigosa de pensar errado. De testemunhar os alunos, as vezes com ares de quem possui a verdade, um rotundo desacerto. Pensar certo, pelo contrário, demanda profundidade e não superficialidade na compreensão e na interpretação dos fatos. [Freire, 1996, p. 33]

Foi percebendo tal realidade que a UNESCO lançou o relatório "Educação: um tesouro a descobrir" deixando claro que considera as políticas educativas como "um processo de enriquecimento dos conhecimentos [...] mas também e talvez em primeiro lugar, como uma via privilegiada de construção da própria pessoa, das relações entre

indivíduos, grupos e nações" [Delors, 1996, p. 12]. Por isso, em tal documento são propostos quatro pilares para a educação contemporânea: aprender a conhecer, aprender a fazer, aprender a conviver e aprender a ser. O primeiro, *aprender a conhecer*, mais do que a aquisição de uma gama de saberes codificados, valoriza, antes de tudo, aprender a aprender, destacando a autonomia do educando, além de exercitar a atenção, memória e pensamento. Ter uma abertura para o conhecimento, compreendê-lo e (re)construí-lo.

O segundo pilar é indissociável do primeiro. O fazer, ou melhor, o *aprender a fazer* objetiva por em prática os conhecimentos oriundos da qualificação profissional, além de desenvolver competências que permitam o educando trabalhar em equipe e enfrentar situações diversificadas e inusitadas uma vez que as demandas profissionais têm mudado constantemente. Até aqui não temos muita novidade tendo em vista que estas duas categorias de aprendizagens constituem as bases dos currículos de Computação.

Então, vamos ao terceiro pilar: *aprender a viver juntos, a viver com os outros*. Este é um dos maiores desafios, pensar em uma educação voltada para evitar e/ou contornar conflitos de maneira pacífica, respeitando a diferença seja qual for ela, política, religiosa, cultural, etc. Esta aprendizagem é extremamente desafiadora dado que "seres humanos têm tendência a supervalorizar as suas qualidades e as do grupo a que pertencem, e a alimentar preconceitos desfavoráveis em relação aos outros" [Delors, 1996, p. 97]. Soma-se a isso a lógica perversa da economia de mercado, que imprime uma competitividade estimulante do espírito de concorrência e do sucesso individual.

Neste sentido, o aprender a conviver pode ser dividida em dois níveis [Delors, 1996]. O primeiro diz respeito à compreensão do outro. A (evolução da) comunicação não garante compreensão, como nos alerta Morin (2000, p. 93), que também aponta esta habilidade como um dos (sete) saberes necessários à educação do futuro:

> Lembremo-nos de que nenhuma técnica de comunicação, do telefone à Internet, traz por si mesma a compreensão. A compreensão não pode ser quantificada. Educar para compreender a matemática ou uma disciplina determinada é uma coisa; educar para a compreensão humana é outra. Nela encontra-se a missão propriamente espiritual da educação: ensinar a compreensão entre as pessoas como condição e garantia da solidariedade intelectual e moral da humanidade.

É importante tanto ter conhecimento da diversidade inerente dos seres humanos quanto ter consciência das semelhanças e interdependências entre eles. O caminho dessa descoberta do outro ainda passa pela descoberta de si mesmo. É compreendendo a si próprio que podemos compreender o outro e suas ações, aumentando a capacidade de abertura à alteridade. Portanto, este aprendizado deve passar por uma viagem interior contínua a fim de amadurecer crescentemente a personalidade.

No segundo nível está a participação em projetos comuns que revela-se uma estratégia significativa na resolução de conflitos; "se existirem objetivos e projetos comuns [num contexto igualitário], os preconceitos e a hostilidade latente podem desaparecer e dar lugar a uma cooperação mais serena e até à amizade" [Delors, 1996, p. 97]. Em outras palavras, a convivência motivada por projetos em comum tendem a fazer com que os conflitos – hostilidade, preconceitos, etc – sejam dirimidos. Tratar questões

como estas no processo de formação, incluindo o universitário, pode se tornar uma boa referência para os educandos, independentemente do contexto social no qual está inserido – das amizades, profissional, conjugal, familiar, etc.

O quarto e último pilar é o *aprender a ser*, reforçando a idéia de uma educação para o desenvolvimento total da pessoa, contemplando as diferentes dimensões do homem, envolvendo, desta forma, espírito e corpo, inteligência, sensibilidade, sentido estético, responsabilidade pessoal, espiritualidade, etc. "O desenvolvimento tem por objeto a realização completa do homem, em toda a sua riqueza e na complexidade das suas expressões e dos seus compromissos: indivíduo, membro de uma família e de uma coletividade, cidadão e produtor, inventor de técnicas e criador de sonhos" [Faure, 1972, p. XVI apud Delors, 1996, p. 101]. Assim preparado, o educando pode agir cada vez mais com maior autonomia, discernimento e responsabilidade de modo a responder, por si mesmo, aos diferentes questionamentos que a vida lhe impõe. É importante ressaltar que isso diz respeito a competências pessoais; autonomia não significa auto-suficiência ou individualidade exacerbada.

Finalmente, todas as quatro dimensões não devem ser tratadas isoladamente nem em uma única fase da vida, mas de maneira harmoniosa e durante toda a vida. Ademais, uma pessoa mais bem preparada nessas aprendizagens também pode ser um melhor profissional. Será uma conseqüência. Para ilustrar, a sua hipotética habilidade de bem conviver com as diferenças naturalmente será exercitada no trabalho e trará benefícios.

4. Breve diagnóstico da formação humanística na computação

Nesta seção, buscamos de forma breve encontrar elementos na formação do educando de Computação referentes às questões que levantamos na seção anterior, de uma educação para além do mercado de trabalho. Para isso, investigamos diretrizes de referência e pesquisas realizadas sobre o assunto.

Quanto às diretrizes, consideramos três documentos: (1) o currículo de referência para cursos de graduação em Computação e Informática da SBC [SBC, 1999], (2) o currículo de referência para cursos de graduação em Engenharia de Computação da ACM (*Association for Computing Machinery*) [ACM, 2004][2], e (3) Diretrizes Curriculares dos cursos de Bacharelado em Ciência da Computação, Engenharia de Computação, Engenharia de Software e Sistemas de Informação e dos cursos de Licenciatura em Computação do Ministério da Educação (MEC) do Brasil [MEC, 2003].

Inicialmente, como era esperado, constatamos que as questões que se aproximam do que discutimos na seção anterior são tratadas como uma área (eixo) específica do currículo comumente denominada de humanística. A Tabela 1 resume um contraste entre os currículos analisados, focando a formação humanística.

Primeiramente, extraímos trechos que destacam a importância deste tipo de formação para os educandos – linha "Perfil desejado dos educandos" da Tabela 1. De maneira unânime, todos os documentos concordam na importância de uma atuação

[2] O currículo de referência geral da ACM não dispõe de dados relevantes sobre o assunto; basicamente, expõe as diferenças dos perfis das graduações existentes na Computação. Assim, escolhemos um currículo mais específico, optando aleatoriamente pelo do curso de Engenharia de Computação.

consciente, crítica e ética do profissional na sociedade e que questões pertinentes a esse perfil desejado, que ajude ao educando desenvolvê-las, devem fazer parte do currículo.

Tabela 1. Comparação entre diretrizes relativas à formação humanística para currículos de Computação.

	[SBC, 1999]	[ACM, 2004][3]	[MEC, 2003]
Perfil desejado dos educandos	"Os egressos de cursos que têm a computação como atividade-fim devem ser profissionais com as seguintes características: [...] formação humanística, permitindo a compreensão do mundo e da sociedade, e o desenvolvimento de habilidades de trabalho em grupo e de comunicação e expressão [...] respeitar os princípios éticos da área de computação [...] Ter uma visão humanística crítica e consistente sobre o impacto de sua atuação profissional na sociedade". (p. 2-3)	"garantir que os estudantes possuam um conjunto de habilidades transferíveis ou pessoais como as habilidades de comunicação, trabalho de grupo e apresentação". (p. 23) "Os estudantes devem desenvolver uma compreensão do contexto social e profissional no qual eles aplicam seus estudos de engenheiro da computação. Considerações éticas devem ser cobertas no contexto de tópicos técnicos". (p. A.10)	"[...] cabendo aos profissionais da Área [de Computação] a responsabilidade pelo desenvolvimento de soluções, ferramentas e processos coerentes com a moral, bons costumes, valores éticos e interesse social, e que também busquem o bem-estar do homem e o avanço tecnológico". (p. 6-7)
Componentes curriculares	Filosofia	Estruturas filosóficas	Filosofia
	Sociologia	Questões profissionais/sociais	Sociologia
	Direito e Legislação	Propriedade Intelectual, Crimes virtuais	
	Economia, Contabilidade e Custos	Questões econômicas na Computação, Compromissos financeiros e riscos	
		Responsabilidades éticas e profissionais	Questões éticas
		História e visão geral	História da Computação
	Empreendedorismo, Administração, Métodos Quantitativos Aplicados à Administração		Empreendedorismo
	Computadores e Sociedade		Impactos da Automação na Sociedade
	Comunicação e Expressão, Inglês, Estágio, Informática na Educação	Políticas públicas, Métodos e ferramentas de análise	
Carga horária (% em relação ao total)	14%-19%	3%-4%	–

Sendo assim, mais uma vez de maneira unânime, todos indicam alguns componentes curriculares (disciplinas) que poderiam fazer parte do currículo auxiliando nessa tarefa de formação humanística. Porém, como pode ser visto na Tabela 1, nem todas as indicações coincidem. Mesmo assim, tentamos fazer um agrupamento apenas para auxiliar em termos de visualização. Assim, percebemos facilmente que os componentes Filosofia e Sociologia estão presentes nos três documentos. Todavia, pode acontecer de haver equívocos, pois fizemos o agrupamento baseado no título do componente e não em sua ementa. Por exemplo, um componente específico para ética não foi indicado pela SBC, mas tal tema pode ser facilmente absorvido pelo componente Computadores e Sociedade, como realmente acontece. Além disso, no último item da

[3] Tradução livre do autor.

linha "Componentes curriculares" colocamos os componentes "isolados", isto é, que não encontramos correspondência alguma com outro documento.

É válido a formação de um olhar profissional sobre questões da sociedade que envolvem a Computação como crimes virtuais, softwares e/ou hardwares relacionados à vida humana (Informática na Medicina), ética no tratamento de dados pessoais, entre tantas outras. São de fato questões relevantes e ao mesmo tempo delicadas, que precisam de muita dedicação. Porém, pouco – ou praticamente nada – encontramos a respeito daquilo que discutimos na seção anterior, que contemple uma educação interdimensional. O foco ainda está muito nos aspectos técnicos, no *logos*. Para agravar, temos dois fatores. O primeiro é que a carga horária sugerida pela ACM para componentes de cunho humanístico é baixa, entre 3% e 4% do total de créditos do currículo. Já a da SBC é mais elevado, entre 14% e 19%, enquanto que o do MEC nada indica. O segundo é que os componentes são muito abrangentes e pouco profundos. Isso acontece também porque a maioria dos componentes ofertados tem pouquíssima relação entre si, por exemplo, Empreendedorismo e Filosofia. O educando tem contato com fragmentos de saberes, geralmente, sem a oportunidade de desenvolvê-los melhor.

Neste ínterim, é importante também ressaltar um importante fato: a resistência dos educandos quanto à aprendizagem e reflexão dos temas humanísticos (inclusive, comprometendo a freqüência nos componentes cursados) [Pinto & Burnham, 2010] – o tema é tão relevante que virou tese de doutorado da primeira autora. Após analisar algumas turmas de um determinado curso, as autoras perceberam que os educandos até gostam dos temas humanísticos, porém acabam por dar prioridade aos temas técnico-profissionalizantes. Na verdade, essa resistência parece estar presente não apenas nos educandos que "não estão mais interessadas na tarefa de ilustração e de elevação espiritual do povo [mas também nos educadores e intelectuais, que] pararam em grande parte de se definir pela responsabilidade que têm para com 'o povo', a nação e a humanidade" [Bauman, 2004, p. 311, grifo nosso].

Prosseguindo, além das diretrizes de referência, também buscamos encontrar discussões sobre educação interdimensional em trabalhos científicos na área de Educação em Computação. Neste intuito, pesquisamos artigos publicados nos últimos três anos (2008-2010) em duas publicações relevantes na área de Educação em Computação: nos anais deste evento, do Workshop sobre Educação em Computação (WEI), e no periódico *IEEE Transactions on Education* (IEEE-TE). Um extrato do resultado está na Tabela 2. As colunas "#t" e "#h" apontam, respectivamente, a quantidade total de publicações no ano indicado e a quantidade de publicações com temáticas humanísticas naquele mesmo ano. Para chegar a este último número, primeiramente, avaliamos os títulos dos artigos. Se não encontrássemos correlação, não contávamos; se houvesse correlação ou nos casos de dúvida (tal trabalho é correlato ou não?), então líamos o resumo e, se necessário, todo o texto.

Tabela 2. Extrato de publicações na área de computação sobre humanística.

	2008		2009		2010		TOTAL	
	#t	#h	#t	#h	#t	#h	#t	#h
WEI	24	1	27	1	25	1	76	3
IEEE-TE	66	1	74	1	122	0	262	2

Os resultados surpreenderam pela pouca quantidade de trabalhos correlatos. Do total de 76 artigos do WEI, apenas 3 tinham uma veia mais humanística. Os dois primeiros, dos anos de 2008 e 2009, são do mesmo grupo de pesquisa que desenvolve projetos sócio-filantrópicos em atividades práticas de computação. Em um deles os autores chegam a seguinte conclusão: "os alunos dos cursos de Computação e Informática começam a perceber que a sua formação deve ser mais ampla que somente a preparação técnica, centrada no mercado. Além de possuir habilidades específicas, o profissional não pode se apartar do seu papel como cidadão" [Oliveira Neto *et al*, 2008, p. 235]. O outro é de nossa autoria [Santos & Burnham, 2010], que já citamos na Seção 2, mas destacamos novamente que uma das conclusões é que o conhecer-fazer não deve estar desvinculado do ser-conviver.

Quanto ao IEEE-TE, é mais discrepante, dos 262 artigos, apenas 2 tratam do assunto. Na verdade, ambos se preocupam com questões psicológicas dos educandos. Um descreve uma estratégia para motivação do ensino-aprendizado e de forma tangencial aborda que tal estratégia pode atenuar sentimentos de isolamento entre os educandos. O outro investiga a ansiedade relacionada com o aprendizado de programação e reconhece que necessidades psicológicas como estas devem ser identificadas e apoiadas no âmbito universitário.

Quanto ao restante dos trabalhos (os não correlatos), de maneira sucinta, a grande maioria se concentra em ampliar a capacidade cognitiva e reduzir dificuldades cognitivas em contextos específicos, por exemplo, de componentes curriculares técnicos. É importante ressaltar que, alguns deles, focou habilidades preciosas como comunicação e trabalho em grupo, mas voltado para aspectos técnicos, por isso não foram contados.

5. Conclusão

Neste trabalho elucidamos brevemente a tradição da educação ocidental de aproximar ferozmente a relação entre educação e trabalho, incluindo contextos relacionados à Computação. Também buscamos fazer uma conscientização de uma *educação interdimensional, para além do mercado de trabalho, dos aspectos técnicos*, valorizando, por exemplo, não apenas aprender a conhecer e fazer mas também aprender a ser e conviver. Aprendizagens como estas podem contribuir não apenas para o enriquecimento da pessoa em si mas também da sociedade, incluindo aí o mercado de trabalho que faz parte dela.

Esta é uma perspectiva que tem ganhado força nos últimos anos, que aperfeiçoa a educação que geralmente nos deparamos atualmente, a qual frequentemente tem negligenciado questões inerentes do ser humano. Em especial, esta constatação ficou mais evidente ao analisarmos pesquisas e diretrizes para currículos de Computação, as quais dão pouco – ou quase nenhum – destaque para uma educação interdimensional.

Obviamente, o que foi abordado aqui ainda é muito pouco frente ao potencial que a temática exige, demandando, desta forma, reflexões mais profundas de como tais questões poderiam ser contempladas nos currículos de Computação com o menor impacto possível, sem comprometer, por exemplo, a carga horária. São novos componentes curriculares que precisamos? É mudança de atitude? Com certeza, é uma tarefa árdua, constituindo, a partir de nosso olhar, um dos (grandes) desafios futuros.

Referências Bibliográficas

ACM (2004). Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering, Disponível em: http://www.acm.org/education/education/curric_vols/CE-Final-Report.pdf. Acesso: 21 mar 2011.

Bauman, Z. (2004). Entrevista com Zygmunt Bauman. In *Revista Tempo Social*, v. 16, n. 1, p. 301-325.

Bauman, Z. (2008), Vida para consumo: a transformação das pessoas em mercadorias, Jorge Zahar Ed.

Costa, A. (2008), Educação, Ed. Canção Nova.

Costa, A. C. (2004). Por uma educação interdimensional. In *Revista de Educação CEAP*, v. 12, n. 45, p. 27-38.

Delors, J. (org.) (1996). Educação: um tesouro a descobrir. *Relatório da Comissão Internacional sobre Educação para o século XXI,* UNESCO.

Faure, E. (org.) (1972). Apprendre à être. *Relatório da Comissão Internacional sobre o Desenvolvimento da Educação*, UNESCO.

Frankl, V. (1991), Em busca de sentido, Ed. Sinodal, Ed. Vozes.

Freire, Paulo. (1996). Pedagogia da Autonomia, Ed. Paz e Terra, São Paulo, 35ª ed.

MEC (2003). Diretrizes curriculares para cursos da área de computação e informática. Disponível em: http://ramec.mec.gov.br/index.php?option=com_docman&task=doc_download&gid=1723&Itemid=1. Acesso: 21 mar 2011.

Moraes, R. J. (2006), As chaves do inconsciente, Ed. Vozes.

Morin, E. (2000), Os sete saberes necessários à educação do futuro, Ed. Cortez, UNESCO.

Oliveira Neto, J. *et al* (2008). Uso de projetos sócio/filantrópicos em atividades práticas: mecanismo de construção do profissional e da cidadania. In *Workshop sobre Educação em Computação – Anais do XXVIII Congresso da SBC*, Belém.

Pinto, G. & Burnham, T. (2010). Disciplinas de formação humanística em Engenharia: resistência do estudante, recursos educacionais motivacionais e alguns resultados. In *Anais do COBENGE 2010*, Fortaleza.

Santos, B. S. (1989). Da Idéia de Universidade a Universidade de Idéias. In *Revista Crítica de Ciências Sociais*, n. 27/28, p. 11-62.

Santos, D. M. & Burnham, T (2010). O pensamento de Paulo Freire e PBL: primeiras aproximações e afastamentos. In: *XVIII Workshop sobre Educação em Computação – Anais do XXX Congresso da Sociedade Brasileira de Computação*, p. 991-1000.

Santos, L. C. (2010b). Ética no cotidiano: notas de aula. Salvador, Outubro.

SBC (1999). Currículo de Referência da SBC para Cursos de Graduação em Computação e Informática. Disponível em: http://www.sbc.org.br/index.php?option=com_jdownloads&Itemid=195&task=finish&cid=52&catid=36. Acesso: 21 mar 2011.

Ensino de IHC – Atualizando as Discussões sobre a Experiência Brasileira

Sílvia Amélia Bim[1], Raquel O. Prates[2], Milene Selbach Silveira[3], Marco Winckler[4]

[1]Dept. de Ciência da Computação – UNICENTRO, Guarapuava – PR – Brasil

[2]Dept. de Ciência da Computação – UFMG, Belo Horizonte – MG – Brasil

[3]Faculdade de Informática – PUCRS, Porto Alegre – RS – Brasil

[4]Dept. de Informática – Université Paul Sabatier, Toulouse, França.

sabim@unicentro.br, rprates@dcc.ufmg.br, milene.silveira@pucrs.br, winckler@irit.fr

Abstract. *This paper summarizes the outcomes of the workshop on education in Human-Computer Interaction (WEIHC) that was held in conjunction with IHC 2010. Thirty participants including researchers, professors, students and professionals attended the workshop. The results point out some challenges that should be overcome to promote HCI education in Brazil, as well as a proposal of the next steps in that direction.*

Resumo. *Este artigo resume os resultados do workshop sobre o ensino de Interação Humano-Computador (WEIHC), realizado com o IHC 2010. Trinta participantes, incluindo pesquisadores, professores, estudantes e profissionais acompanharam o workshop. Os resultados indicam alguns desafios que devem ser superados para impulsionar o ensino de IHC no Brasil e uma proposta inicial de linhas de ações para lidar com eles.*

1. Introdução

O ensino de Ciência da Computação é um assunto amplamente discutido pela comunidade acadêmica brasileira e internacional. No contexto brasileiro o Workshop sobre Educação em Computação (WEI) tem sido um dos principais fóruns para esta discussão. Outra iniciativa mais recente é o Curso de Qualidade (CQ), evento realizado pela Comissão de Educação da SBC, que em 2011 será realizado em conjunto com o WEI em uma iniciativa de proporcionar maior integração do público interessado e atuante na educação em Computação.

Além das iniciativas de discussão sobre o ensino de Ciência da Computação em geral, as características particulares de cada disciplina que compõe o currículo dos cursos demandam discussões mais específicas. No Brasil algumas áreas já promovem eventos anuais organizados pelas suas Comissões Especiais da SBC e que acontecem anualmente juntamente com o evento Brasileiro da área. Este é o caso do Fórum de

Educação em Engenharia de Software[1] (FEES) e do Workshop sobre Educação de Arquitetura de Computadores[2] (WEAC).

A área de Interação Humano-Computador (IHC) também possui seus próprios desafios de ensino para os professores desta disciplina. Entre estes desafios pode-se citar: a natureza interdisciplinar da área que envolve disciplinas distintas (tais como Psicologia Cognitiva, Design Gráfico, Ergonomia, Ciência da Computação, etc); a adaptação destes conhecimentos aos programas de graduação e pós-graduação nos quais disciplinas de IHC são oferecidas; e o acompanhamento de como novas tecnologias (tais como a telefonia móvel e a TV interativa) introduzem novos contextos de uso e novos requisitos para a construção de interfaces. Conscientes destas dificuldades, associações científicas tais como o ACM SIGCHI[3] [ACM 1992] e o comitê técnico da IFIP TC13[4] [IFIP Working Group 13.1 2007] organizam regularmente discussões e grupos de estudo para acompanhar a evolução do ensino de IHC. No entanto, estes esforços não levam em consideração o contexto brasileiro.

No Brasil, o primeiro grupo de trabalho para discutir o ensino de IHC se reuniu durante o IHC 2002[5], mas não foi gerado nenhum relatório formal do encontro. No IHC 2006 um novo encontro foi organizado com o objetivo específico de se discutir uma proposta de ementa para disciplinas de IHC. Os resultados deste encontro deram origem a uma publicação no XV Workshop sobre Educação em Computação [Silveira e Prates 2007] que atualmente constitui o único relato oficial sobre aspectos de ensino de IHC no Brasil. Na edição seguinte do IHC, em 2008, não houve grupo de trabalho sobre o ensino de IHC, contudo o tópico emergiu naturalmente durante o painel "IHC no Brasil: lições aprendidas e novas perspectivas" [de Souza et al. 2008].

Este artigo descreve os resultados da iniciativa mais recente para dar continuidade à discussão sobre o ensino de IHC no Brasil. O Workshop sobre o Ensino de IHC (WEIHC) foi realizado em 2010 em conjunto com o IHC 2010. Na seção 2 descrevemos a organização do WEIHC. Os relatos de experiências são apresentados na seção 3. A seção 4 apresenta os temas discutidos durante o workshop, seguida, na seção 5, da lista de ações geradas. As considerações finais são feitas na seção 6.

2. A Organização do Workshop sobre o Ensino de IHC (WEIHC)

O Workshop de Ensino de IHC (WEIHC) teve como seu principal objetivo discutir o ensino de IHC tanto sob a perspectiva curricular (quais são os conteúdos abordados nas disciplinas de IHC), quanto sob a perspectiva das práticas pedagógicas (como estes conteúdos são trabalhados, como o conhecimento dos alunos é avaliado, quais as dificuldades encontradas por alunos e professores no processo de ensino-aprendizagem). Para tanto, visou-se como público-alvo professores de disciplinas relacionadas à área de Interação Humano-Computador, em cursos de graduação e pós-graduação, e alunos de mestrados e doutorados da área de IHC que pretendem atuar como professores desta disciplina.

[1] http://www.each.usp.br/cbsoft2011/portugues/sbes/fees_pt.html
[2] http://sbac-pad-2011.lsc.ic.unicamp.br/
[3] *ACM Special Interest Group in Computer-Human Interaction* (http://www.sigchi.org)
[4] *International Federation for Information Processing – Technical Committee 13 on Human Computer Interaction (http://csmobile.upe.ac.za/ifip).*
[5] Simpósio de Fatores Humanos em Sistemas Computacionais

Através de listas de difusão (ihc-l e sbc-l[6]) foram solicitadas contribuições na forma de artigos curtos. Todas as submissões foram então avaliadas pelos coordenadores do workshop e receberam comentários. Os artigos selecionados foram publicados no website do workshop e nos anais estendidos do IHC2010 [Silveira et al. 2010].

2.1 Agenda

O WEIHC que teve duração de oito horas. O programa contou com a apresentação de trabalhos selecionados, de grupos de discussão e de intervenções convidadas. Os trabalhos selecionados foram agrupados em três sessões temáticas: estratégias de ensino, IHC e outras disciplinas, e IHC e mercado de trabalho.

O workshop contou com a presença de 30 participantes, incluindo os autores dos artigos selecionados, os ouvintes e os organizadores. Além da apresentação oral, os participantes responderam a uma breve consulta sobre suas experiências em IHC (ver Tabela 1). Com relação aos estados de origem dos participantes, quatro das cinco regiões brasileiras foram representadas, com a seguinte distribuição: Sudeste (62%), Sul (17%), Nordeste (14%) e Norte (7%).

Tabela 1. Experiência dos participantes em IHC

Atividades	Leciona IHC		Faz pesquisa em IHC	Trabalha na indústria com IHC	Estudante
	Graduação	Pós-graduação			
Participantes	20	09	25	04	07

Após a apresentação dos participantes, os organizadores fizeram uma breve introdução sobre o cenário brasileiro e internacional de ensino de IHC. Em seguida foram apresentados os artigos selecionados. Dando continuidade às atividades do workshop houve uma discussão guiada em torno de questões selecionadas. O evento foi encerrado com a apresentação de um levantamento sobre a bibliografia de IHC em português. As seções a seguir apresentarão os temas discutidos.

3. Relatos de Experiências

O WEIHC iniciou com a apresentação de palestra convidada sobre um levantamento de disciplinas sendo ensinadas na área de IHC no país [Prates e Filgueiras 2011]. A pesquisa foi feita através de questionário distribuído através das listas ihc-l e sbc-l no final de 2009 e contou com a participação de 91 professores, afiliados a 63 universidades distintas com representantes de todas as regiões do país. Foram descritas 141 disciplinas, sendo que destas, 57% são destinadas à graduação, 23% à pós-graduação, e 18% a ambas (graduação e pós). As disciplinas foram ainda classificadas em introdutórias (apresentam conceitos básicos e visão geral da área) – 57% das disciplinas descritas; avançadas (tratam de tópico específico de IHC de forma mais aprofundada) – 28%; e módulo de IHC (IHC é apresentado como módulo de uma disciplina de outra área) – 15%.

Após esta visão geral do ensino de IHC no Brasil, passou-se para a apresentação dos artigos selecionados. Os oito artigos selecionados foram agrupados em três temas principais: (i) estratégias de ensino, (ii) IHC e outras disciplinas e (iii) IHC e o mercado de trabalho. Entretanto, muitas vezes os artigos abordavam mais de um tema. A seguir,

[6] ihc-l: ihc-l@sbc.org.br e sbc-l: sbc-l@sbc.org.br

apresentamos resumidamente cada um dos artigos apresentados e as principais questões levantadas pelos demais participantes do WEIHC.

3.1. Estratégias de ensino

Dentre as estratégias apresentadas, Cerutti [2010] descreve a realização de um exercício de desconstrução de uma interface utilizando a técnica proposta por Nielsen [Nielsen e Tahir 2002]. Neste exercício, os alunos, individualmente, deveriam apenas descrever a interface segundo os critérios propostos. Um dos objetivos do exercício era despertar nos alunos a percepção sobre o processo de interação de uma interface pronta, para auxiliá-los em seus próprios projetos. A principal dificuldade encontrada pelos alunos foi realizar uma descrição pura da interface, sem avaliá-la.

A disciplina de Interface Homem-Máquina, para a qual o exercício foi proposto, é ofertada em paralelo com a disciplina de Projeto de Sistemas de Informação. Com relação a este contexto, a autora observa que a proposta de programa sugerida em [Silveira e Prates 2007] não pôde ser aplicada, no que diz respeito à ordem do conteúdo. O programa [Silveira e Prates 2007] recomenda que o conteúdo sobre projeto de IHC seja trabalhado após o conteúdo de avaliação de IHC. Cerrutti [2010] ressalta, entretanto, que para acompanhar as atividades de desenvolvimento de software da disciplina de Projeto de Sistemas de Informação, foi necessário abordar primeiramente o tema de projeto de interfaces na disciplina de Interface Homem-Máquina.

O relato feito por Leite [2010] apresenta a proposta de programa que está sendo adotada na disciplina de IHC no curso de Ciência da Computação da UFRN. O autor faz considerações sobre o excesso de conteúdo que a disciplina deve abordar e sobre a falta de formação dos professores sobre todos eles. Diante destas considerações a proposta é trabalhar com apenas um recorte do conteúdo, que no caso apresentado, trata da prática de design como processo comunicativo. O objetivo é oferecer uma abordagem mais prática de IHC que não exija o conhecimento de teorias de diferentes disciplinas. Dentre os conteúdos trabalhados estão: fundamentos da teoria da Engenharia Semiótica; práticas das áreas de design e de comunicação - *brainstorming*, esboços de telas e prototipagem (design) e elaboração de roteiros (*scripts*), *storyboarding* e princípios de comunicação visual (comunicação); técnicas de análise e modelagem da computação; técnicas de avaliação com ênfase no processo de interpretação dos significados.

O terceiro relato compartilhou a experiência do uso de um quadro interativo (*Smart Board*) nas disciplinas de Design de IHC e Computação Gráfica [Militão 2010]. A vantagem do uso deste recurso tecnológico é estimular a participação dos alunos através da interação direta com o quadro durante as aulas. As atividades colaborativas através do *Smart Board* assumem um tom lúdico, o qual incentiva a participação dos alunos nas discussões sobre as propostas de design. Além disto, toda interação no *Smart Board* é gravada e enviada aos alunos logo após a aula.

Não foi feita ainda uma análise do impacto da tecnologia no aprendizado dos alunos. No entanto, Militão afirma que o recurso tecnológico ajuda os alunos na compreensão do que é um trabalho em grupo e que é possível perceber a diminuição do número de desistências da disciplina. Embora os resultados desta experiência sejam promissores, duas ressalvas foram feitas: (1) o custo do hardware que impede que este seja amplamente utilizado; (2) nem todas as aulas podem ser convertidas para uso do

Smart Board. O desafio é adaptar o conteúdo para o novo veículo de comunicação, o que muitas vezes exige um considerável tempo de pesquisa.

3.2. IHC e outras disciplinas

Um dos relatos sobre a relação entre IHC e outras disciplinas do currículo de Ciência da Computação compartilha a experiência de integração entre IHC, Engenharia de Software e Banco de Dados. No contexto de um curso com disciplinas anuais Bim [2010] descreve o projeto desenvolvido em conjunto pelas três disciplinas, as quais são cursadas, pela maioria dos alunos, ao mesmo tempo, o que facilita o desenvolvimento dos projetos. Outro fator que contribui para o sucesso da proposta de integração são turmas pequenas, em média 15 alunos. A disciplina de IHC é responsável pela parte de avaliação de IHC dos projetos, aplicando os métodos de avaliação propostos pela Engenharia Semiótica. A disciplina de Engenharia de Software por sua vez trabalha na modelagem em UML dos projetos. E na disciplina de Banco de Dados os diagramas conceitual, lógico e físico dos projetos são avaliados.

Furtado e Kampf [2010] fazem reflexões sobre a integração entre IHC e uma disciplina fora do currículo de Ciência da Computação, a Psicologia Ambiental. As autoras refletem sobre um trabalho desenvolvido na disciplina de Psicologia Ambiental (do curso de Psicologia). Elas propõem que os conceitos da Psicologia Ambiental possam ser relevantes para a elicitação de requisitos de sistemas complexos e para o estudo da apropriação pelos pretensos usuários das soluções de interação com esses sistemas.

O terceiro relato aborda a questão de uma disciplina avançada de IHC. Melo [2010] compartilha a experiência da oferta da Disciplina de Contexto Social "Acessibilidade e Inclusão Digital", em caráter optativo, no contexto de um curso de Ciência da Computação em uma universidade pública. A disciplina foi cursada por estudantes de diferentes períodos e estratégias metodológicas diversificadas foram usadas para explorar seu conteúdo. A autora relata que é possível perceber, na disciplina subseqüente, uma postura mais crítica dos alunos que cursaram a disciplina optativa, com relação à importância de IHC.

3.3. IHC e mercado de trabalho

Dentre os artigos selecionados para o workshop, dois trataram da relação com o mercado de trabalho. O primeiro relato [Aquino Junior 2010] descreve a inserção da disciplina de IHC em um programa de computação em uma universidade privada brasileira e o mercado de trabalho para alunos que seguem estas disciplinas. Já o segundo relato [Winckler & Palanque 2010] resume a experiência de dez anos de um programa de mestrado profissional em Toulouse, França, dedicado a IHC.

A primeira apresentação discorreu inicialmente sobre as dificuldades de fazer evoluir o programa de graduação para incluir novas disciplinas tais como IHC. A disciplina de IHC foi introduzida em nível de graduação, inicialmente como parte da disciplina de multimídia. Após esforços para aumentar a carga horária (atualmente 4 horas de aulas teóricas e 2 horas de aulas práticas), ainda é necessário melhorar a relação da disciplina de IHC com as demais disciplinas do programa. O autor comentou como atualmente os professores estão trabalhando para despertar o interesse dos alunos pela área de IHC e para evangelizar empresas para investir na área. Os primeiros frutos

começam a aparecer e, recentemente, através do Centro de Pesquisas da Universidade em questão, empresas têm investido na formação e demonstrado interesse nos trabalhos tratados na disciplina de IHC. As empresas começam a oferecer estágios e o número de trabalhos de conclusão de curso na área de IHC tem crescido.

O relato sobre o mestrado em IHC da França apresenta um trabalho para formar profissionais de IHC. Tal curso se caracteriza pelo grande número de horas de ensino em IHC (~450 horas) e uma estreita cooperação com a indústria através de estágios e estudos de casos reais que são tratados em projetos em grupo (os chamados *chef d'oeuvre*). É interessante notar que a sobrevivência de cursos de mestrado na França está diretamente relacionada à procura de alunos e ao número de propostas de emprego que os alunos recebem. Cursos que não conseguem empregar alunos na área de formação podem perder a autorização para funcionar. Para tanto o programa pedagógico do curso de mestrado tem evoluído constantemente para acompanhar as oportunidades de emprego, em especial na indústria aeroespacial local (e.g. Airbus, CNES, Thalles Avinonics). O balanço tem sido positivo e o curso é bem avaliado (i.e. A+) pelo ministério da educação francês. Contudo, os autores relatam que a reorganização do programa de ensino (que ocorre a cada 2-4 anos) é exigente e que a participação de indústrias no programa foi importante para o reconhecimento, não apenas do curso em si, mas da área de IHC como um todo.

4. Discussões e Reflexões

As apresentações dos trabalhos descritas acima geraram diversas discussões sobre os desafios vivenciados no ensino de IHC. Nesta seção organizamos as discussões em 4 tópicos e apresentamos os principais pontos discutidos em cada tópico.

4.1. Uso de Tecnologia no Ensino de IHC

O uso de novas tecnologias no ensino de IHC pode trazer benefícios para o aprendizado do aluno. No entanto, a introdução destas novas tecnologias requer adaptação das metodologias e materiais utilizados no ensino. Se por um lado, a introdução já traz um grande interesse devido à novidade, faz-se necessário também a pesquisa associada ao uso possível destas, assim como sobre seu impacto no aprendizado do aluno.

A tecnologia é uma questão não apenas em relação ao seu uso em sala de aula, mas também pelos alunos. Atualmente, cada vez mais tecnologias estão sendo utilizadas no cotidiano das pessoas, mas os trabalhos de curso costumam considerar como objeto de estudo as interfaces mais conhecidas e de fácil acesso, como sistemas web. A consideração da interação em diferentes dispositivos é um desafio, tanto pela dificuldade de se oferecer ao aluno acesso a algumas destas tecnologias, quanto pela abordagem das disciplinas, que na sua maioria são introdutórias, não tendo tempo de tratar assuntos mais avançados.

4.2. Integração com Outras Disciplinas

Um ponto amplamente discutido foi o desafio de integrar a disciplina ou conteúdo de IHC com os de outras disciplinas, como, por exemplo, Engenharia de Software. Algumas soluções e experiências de sucesso foram relatadas. Por exemplo, Cerrutti alterou a ordem sugerida pela ementa gerada pela comunidade de IHC [Silveira e Prates

2007] para que o conhecimento obtido na disciplina de IHC fosse utilizado em outra disciplina. Já Bim relatou a integração dos trabalhos de três diferentes disciplinas.

Esta integração pode trazer outros benefícios para o aluno, como uma visão de como as diferentes áreas e métodos podem ser combinados na prática. Além disso, até mesmo os professores envolvidos se beneficiam de um maior entendimento de outras áreas e de como integrar os diversos métodos. No entanto, esta integração teve sucesso em um contexto onde o grupo de alunos e corpo de professores é pequeno facilitando o processo de integração. Em outros contextos, esta integração pode ser dificultada por se ter várias turmas da mesma disciplina, diferentes professores ministrando as disciplinas o uma baixa quantidade de alunos matriculados em um mesmo conjunto de disciplinas.

Este desafio pode ser ainda maior quando se considera não apenas as diferentes disciplinas de um curso, mas um contexto interdisciplinar, no qual os alunos e professores envolvidos também são de áreas de conhecimento distintos. Furtado e Kampf fazem reflexões e apresentam uma proposta de integração entre IHC e a Psicologia Ambiental para o projeto de sistemas sustentáveis.

4.3. Disciplinas Avançadas

Na apresentação sobre o levantamento de ensino de IHC no Brasil, viu-se que a grande maioria das disciplinas são oferecidas para graduação e são introdutórias. No workshop, a discussão também focou mais em disciplinas introdutórias de IHC. Ainda assim, Melo levantou na sua apresentação a dificuldade de se oferecer disciplinas avançadas que possibilitam o aprofundamento em um tópico específico da área. Os desafios muitas vezes se iniciam com questões administrativas relacionadas à possibilidade de se propor uma disciplina nova ou mais avançada. Ainda dependendo da disciplina, quando se tem um grupo de alunos heterogêneo (de diferentes cursos ou períodos) este também pode ser um desafio. Vale ressaltar que, muitas vezes, professores vivenciam estes desafios ainda para criarem uma primeira disciplina de IHC no curso.

4.4. Integração com Mercado de Trabalho

As discussões em torno dos dois relatos sobre mercado de trabalho indicam que o cenário brasileiro é carente de profissionais de IHC. Embora, existam pessoas atuando na área, existe uma distância grande entre a universidade e a indústria. Embora tenhamos visto que já existem disciplinas de graduação em IHC em todo o país, em um levantamento recente sobre os profissionais da área de IHC atuando na indústria, a maior parte deles relata ter obtido seu conhecimento da área na prática (já no mercado de trabalho) [Prates e Filgueiras 2011]. Além disso, esta distância pode ser confirmada pelo baixo número de relatos de estágios ou propostas de emprego na área no Brasil. Experiências como a francesa relatada por Winckler são muito interessantes e devem ser bem analisadas para que possam contribuir para se definir estratégias para que se consiga uma integração melhor entre universidade e indústria no contexto brasileiro.

Os desafios discutidos durante o WEIHC levantam algumas das principais questões vivenciadas no ensino de IHC no Brasil. A discussão de como lidar com eles e trocas de experiências nesta direção devem ser continuadas para que o esforço não seja isolado de grupos ou professores, mas que todos possam contribuir para a comunidade.

5. Lista de Ações para a Comunidade de IHC

A partir da discussão realizada, foi elaborada pelos participantes, durante o workshop, uma lista de sugestões e ações com o objetivo de se endereçar os desafios identificados. As propostas foram organizadas em 4 linhas de ação: IHC em sala de aula; apoio ao ensino; divulgação da área e evolução da área no país. Vale ressaltar que as duas últimas linhas vão além de aspectos da educação, mas a comunidade entende que são esforços que têm impacto também no ensino de IHC. A Tabela 2 apresenta as proposta para cada um das linhas sugeridas.

Tabela 2. Propostas de ação resultantes do WEIHC

Linha	Propostas de ações e sugestões
IHC em sala de aula	• Tentar pensar a disciplina de IHC de forma interdisciplinar e/ou interdepartamental, buscando uma "língua comum" entre os professores das diversas disciplinas/práticas relacionadas; • Compartilhar e/ou dividir a disciplina, seja no mesmo departamento ou entre departamentos (tais como design e ciência da computação, por exemplo); • Diversificar as estratégias pedagógicas utilizadas em sala de aula; • Estabelecer uma rotina de seminários, apresentando as tendências da área; • Discutir – entre professores e alunos de IHC - a Competição de Avaliação do IHC quanto a retorno para os alunos e benefícios ao processo de aprendizagem; • Discutir as etapas do ciclo de vida do desenvolvimento de software que devem ser tratadas na disciplina de IHC; • Discutir como tratar a questão da inovação tecnológica (e.g. iPhone) e emergência de novas disciplinas (e.g. mobile HCI, iTV).
Apoio ao ensino	• Criar um espaço de memória, que permita aos professores, que ministram disciplinas na área, compartilhar materiais e experiências (tanto para apoio a estes professores quanto a aqueles estreantes na área); • Realizar a revisão do currículo de IHC recomendado pela SBC; • Discutir como construir um currículo sem o "engessamento" de IHC; • Criar uma comissão permanente para acompanhamento deste tema (por exemplo, dentro da Comissão Especial de IHC da SBC (CEIHC)); • Organizar da bibliografia existente em português na área.
Divulgação da área	• Aumentar o espaço de discussão para os trabalhos acadêmicos de alunos no IHC; • Pensar na criação de: Cursos de Extensão (para conscientização da comunidade para a área); uma Escola Regional de IHC e/ou uma Escola Itinerante de IHC; um Curso de Especialização - Interinstitucional - em IHC.
Evolução da área no país	• Atentar para a preservação de nossa diversidade cultural e identidade enquanto comunidade brasileira de IHC; • Criar uma comissão permanente de pesquisa e outra de educação (dentro da CEIHC).

Dentre as sugestões acima listadas algumas já estão em processo de discussão como as que dizem respeito à definição de novos papéis dentro da CEIHC – especificamente quanto a representantes de educação e pesquisa, com o intuito de acompanharem mais diretamente estas questões. Outro ponto recorrentemente discutido e atualmente em estudo pela CEIHC é ter-se o Workshop de Ensino de IHC como uma das categorias já pré-estabelecidas do evento de IHC nacional, e não passível de seleção como acontece atualmente.

Além disso, em relação ao levantamento da bibliografia existente em português na área, durante o evento Bruno S. da Silva se propôs a realizar esta pesquisa. Ele

coletou informações através de pesquisa na internet e solicitação de indicação de referências na lista ihc-l. A partir das informações, no final de 2010 uma lista preliminar foi organizada e disponibilizada por ele para a comunidade de IHC através da ihc-l. Na lista divulgada foi possível verificar que, desde 1999, todo ano há novas publicações em português sobre IHC, exceto 2001. E em 2010 houve a maior publicação na área de material por autores nacionais (3 livros); este crescimento possivelmente se dá devido a um aumento da demanda no mercado e no ensino da área de IHC.

6. Considerações finais

O WEIHC permitiu a discussão de trocas de experiências e desafios vivenciados por professores de IHC. Além dos relatos de como lidam com os desafios vivenciados, a discussão entre os participantes permitiu a identificação de questões a serem tratadas pela comunidade, assim como a proposta de linhas de ações para endereçá-las. Enquanto algumas podem ser completadas em um curto prazo (como vimos, algumas já estão sendo executadas), outras propostas envolvem discussões mais amplas e a mais longo prazo, como é o caso da questão de tratamento da interdisciplinaridade que pode envolver inclusive aspectos da administração e estruturação das instituições de ensino superior.

Além disto, uma análise detalhada do mercado de trabalho, com informações sobre o perfil dos profissionais que estão atuando na área e sobre as necessidades do mercado de trabalho para a formação dos alunos é imprescindível tanto para direcionarmos as ações do ensino de IHC, quanto para investirmos em um diálogo melhor entre universidade e mercado. Com relação à questão de acessibilidade, por exemplo, já é possível identificar a carência de profissionais para desenvolverem sistemas que respeitem este critério de qualidade.

A discussão sobre o ensino de IHC no âmbito do WEI é relevante e traz contribuições para a comunidade de computação como um todo. Primeiramente, como ainda há disciplinas de IHC sendo criadas e ministradas por professores sem formação na área, os resultados apresentados aqui são uma importante referência para que entendam os desafios com que vão lidar e identifiquem experiências que podem se aplicar ao seu contexto de ensino. Embora a discussão seja específica sobre a área de IHC, diversos dos problemas e soluções podem ser interessantes para ensino de disciplinas de outras áreas (e.g. Engenharia de Software e Bancos de Dados). Além disso, a discussão sobre a integração entre diversas disciplinas que lidam com projeto e avaliação de software requer a participação destas outras áreas. Assim, este trabalho contribui também para a divulgação desta questão entre outras comunidades para que se inicie um diálogo mais amplo de como tratá-la.

Diante da extensa agenda de questões a serem discutidas e investigadas nos parece urgente que haja uma regularidade nos eventos dedicados ao tema do ensino de IHC. Ao compreendermos os desafios do ensino desta área, poderemos traçar estratégias para mitigá-los e consequentemente contribuiremos com profissionais melhor capacitados para atuarem no desenvolvimento de sistemas interativos com maior qualidade de interação. Além disso, a ampla divulgação destas discussões é relevante para que contribua não apenas para o ensino de IHC, mas também para sua integração com outras disciplinas contribuindo para a melhoria dos cursos da área de computação de forma geral.

Agradecimentos

Os autores agradecem a todos os participantes do WEIHC 2010. Raquel Prates agradece o apoio da Fapemig à sua pesquisa.

Referências

ACM SIGCHI. Curricula for Human-Computer Interaction. ACM SIGCHI Report, ACM, NY. 1992. Revisado em 1996. Disponível online em http://sigchi.org/cdg/

Aquino Junior, P. T. Papéis do Docente de IHC: do conhecimento ao Mercado. Anais Estendidos do IHC 2010, SBC, 2010. v. II. p. 79-82.

Bim, S. A. Uma experiência de integração entre as disciplinas de IHC, Engenharia de Software e Banco de Dados. Anais Estendidos do IHC 2010, SBC, 2010. v. II. p. 83-86.

Cerutti, D. M. L. Ensino de IHC: Desconstruindo interfaces em sala de aula. Anais Estendidos do IHC 2010, SBC,, 2010. v. II. p. 75-78.

de Souza, C. S., Baranauskas, M. C., Prates, R. O., and Pimenta, M. S. 2008. HCI in Brazil: lessons learned and new perspectives. In Proc. of IHC 2008, ACM International Conference Proceeding Series, vol. 378, 358-359.

Furtado, E., Kampf, T. A Psicologia Ambiental no Projeto Sustentável de Sistemas Interativos. Anais Estendidos do IHC 2010, SBC,, 2010. v. II. p. 59-62.

IFIP Working Group 13.1. 2007. Education in HCI and HCI Curriculum. Disponível em http://www.hcieducation.com/

Leite, J. C. Aplicando técnicas práticas de comunicação para o design e a avaliação de IHC na formação de profissionais de computação. Anais Estendidos do IHC 2010, SBC, 2010. v. II. p. 67-70.

Melo, A. M. Acessibilidade e Inclusão Digital: Disciplina de Contexto Social para Estudantes de Ciência da Computação. Anais Estendidos do IHC 2010, SBC, 2010. v. II. p. 63-66.

Militão, J. G. O. Use of Smart Boards for Undergraduate Teaching in Computer Graphics and Interface Design. Anais Estendidos do IHC 2010, SBC, 2010. v. II. p. 87-90.

Nielsen, J., Tahir, M. Homepage: 50 websites desconstruídos. Campus, 2002. 315p.

Prates, R. O. e Filgueiras, L. V. L. Usability in Brazil. Global Usability – Human-Computer Interaction Series. Douglas, I., Liu, Z (Eds), 2011, Springer.

Silveira, M. S., Prates, R. O. Uma Proposta da Comunidade para o Ensino de IHC no Brasil. Anais do XV WEI, XXVII Congresso da SBC, SBC, páginas 76-84.

Silveira, M. S., Prates, R. O., Pádua, C. I. P. S. (eds.). Anais Estendidos do IX Simpósio de Fatores Humanos em Sistemas Computacionais (Vol. II). Sociedade Brasileira de Computação. Belo Horizonte, Brasil, 5/10/2010. ISSN 2178-7700.

Winckler, M., Palanque, P. Desafios para Programas de Mestrado em Interação Humano-Computador (IHC): a experiência do M2IHM. Anais Estendidos do IHC 2010, SBC, 2010. v. II. p. 71-74.

A Sustentabilidade como Componente Curricular do Curso de Bacharelado em Ciência da Computação

Luciana A. M. Zaina[1], Katti Faceli[1]

[1]Universidade Federal do São Carlos (UFSCar) – campus Sorocaba
Rdv. João Leme dos Santos (SP-264), Km 110 - Sorocaba - São Paulo - Brasil

{lzaina,katti}@ufscar.br

Abstract. *The transversal approach to the sustainability theme has been used in some Computer Science courses. This work presents the Sustainability as a curricular component, adding this thematic in the educational backgrounds of the graduate of Computer Science. The experience was realized by the Federal University of São Carlos, Sorocaba campus.*

Resumo. *A abordagem transversal do tema sustentabilidade tem sido uma realidade em alguns cursos de Ciência da Computação. Este trabalho apresenta a Sustentabilidade como um componente curricular, adicionando esta temática na formação do Bacharel em Ciência da Computação. A experiência de implantação do componente foi realizada na Universidade Federal de São Carlos, campus de Sorocaba.*

1. Introdução

A sustentabilidade tem sido um tema discutido amplamente no mundo todo. Países desenvolvidos e em desenvolvimento tem procurado planejar ações que tragam contribuições importantes para o planeta. Já há alguns anos o assunto deixou de ser algo discutido apenas entre os estudiosos ou ambientalistas para ser debatido em diversos setores da sociedade.

Nesse contexto, o tema sustentabilidade também tem sido uma preocupação dentro da formação superior nas áreas de Computação. Iniciativas de diversos grupos brasileiros têm demonstrado que a Computação tem muito a contribuir dentro do contexto de sustentabilidade. Como exemplos no ano de 2010 podem ser destacados a criação do "Centro de Descarte e Reuso de Resíduos de Informática" (Cedir) do Centro de Computação Eletrônica (CCE) da USP, São Paulo [Cedir 2010] e "XXX Congresso da Sociedade Brasileira de Computação" [CSBC 2010] cujo tema foi "Computação Verde – Desafios Científicos e Tecnológicos". No que tange a elaboração de Projetos Pedagógicos dos cursos de graduação, o tema tem sido evidenciado como elemento transversal dos currículos de formação na área de Computação.

Embora existam diversas definições para sustentabilidade todas convergem para a capacidade de utilizar os recursos do nosso ambiente sem comprometê-los para as gerações futuras. Os princípios da sustentabilidade começaram a serem discutidos na Conferência das Nações Unidas sobre o Meio Ambiente Humano em Estocolmo em 1972 [United Nations Environment Programme 1972] focando principalmente na vertente de poluição e degradação ambiental. Porém, com o passar dos anos os

princípios foram se expandindo principalmente para que fosse possível haver uma sinergia entre os interesses relacionados ao meio ambiente e os interesses econômicos de um país. Foi a partir disto que surgiu o conceito de desenvolvimento sustentável, que é definido como a integração entre o crescimento econômico, a equidade social e a preservação do meio ambiente, tornando-os elementos interdependentes de suporte ao desenvolvimento de longo prazo [Hedstrom, Poltorzycki e Stroh 1998].

Considerando as discussões sobre Sustentabilidade, o Projeto Político Pedagógico (PPP) do curso de Bacharelado em Ciência da Computação, da UFSCar - campus Sorocaba foi elaborado considerando que o tema não deveria ser implementado somente transversalmente, mas também como parte integrante da formação do estudante. O projeto foi construído respeitando as diretrizes curriculares determinadas pela Sociedade Brasileira de Computação [Diretrizes Curriculares SBC 2005] e em consonância com o Projeto Político Institucional do campus que determina que o eixo principal que orquestra as atividades do campus é a temática da Sustentabilidade. Além disto, o PPP se pauta no documento do "Perfil do profissional a ser formado na UFSCar" [Perfil UFSCar 2008] que determina que a formação do estudante deve ser pautada não somente de elementos técnicos, permitindo que este trabalhe não somente como desenvolvedor ou difusor de tecnologia, mas também como agente transformador da sociedade.

O objetivo deste artigo é apresentar a experiência que o curso de Bacharelado em Ciência da Computação, da UFSCar - campus Sorocaba teve ao introduzir uma disciplina que trate dos aspectos de sustentabilidade através de uma disciplina técnica da Computação. O artigo apresentará não somente a descrição do componente curricular como também a experiência e os resultados obtidos com uma turma do curso. O restante do artigo está organizado da seguinte forma: seção 2 apresenta o componente curricular de Tecnologia da Informação e Sustentabilidade, a implantação do componente no curso é apresentada na seção 3, a seção 4 apresenta os resultados obtidos durante a implantação do componente curricular e a seção 5 realiza as conclusões do artigo.

2. Componente Curricular – Tecnologia da Informação e Sustentabilidade

O componente curricular "Aplicações em Tecnologia da Informação para a Sustentabilidade" , de quatro créditos semanais, foi elaborado com o intuito de expandir a visão dos estudantes sobre o tema e suas aplicações. O objetivo geral deste componente é capacitar o aluno para o desenvolvimento de soluções na área de tecnologia da informação que tenham impacto em sustentabilidade, além de proporcionar aos alunos meios de analisar e avaliar soluções de tecnologia da informação através de indicadores de sustentabilidade.

Tópicos foram planejados com o intuito de apresentar o tema sustentabilidade através de um contexto amplo e de mostrar como ele atinge diversos ramos do conhecimento, com destaque para a área da Computação. Dentro deste viés foram planejados os conteúdos para o componente curricular: fundamentos sobre sustentabilidade, alicerces da sustentabilidade: social, ambiental e econômico, computação e sustentabilidade, TI verde, indicadores de sustentabilidade em TI, arquiteturas para sistemas computacionais e sustentabilidade e gestão do conhecimento e sustentabilidade. Estudos de caso e projetos de diversas áreas da Computação devem

ser utilizados como mecanismos de acompanhamento, permitindo que o aluno desenvolva a visão da responsabilidade das tecnologias da informação em segmentos transversais.

A inserção desse componente curricular no projeto pedagógico do curso tem um impacto importante na formação do estudante. Podem-se destacar como competências e habilidades que se deseja desenvolver no aluno: a de compreender a atuação profissional como uma forma de intervenção do indivíduo na sociedade, devendo esta intervenção refletir uma atitude crítica, de respeito aos indivíduos, à legislação, à ética, ao meio ambiente, tendo em vista contribuir para a construção da sociedade presente e futura.

2.1 Planejamento da Disciplina

A disciplina foi oferecida aos alunos do sexto semestre do curso de Bacharelado em Ciência da Computação no ano de 2010, sendo que dezoito alunos cursaram a disciplina. Nesta edição, a turma oferecida teve duas docentes responsáveis (autoras deste artigo), que trabalharam conjuntamente na elaboração e aplicação das atividades teóricas e práticas da disciplina.

Durante o planejamento as docentes procuraram elaborar atividades em que os alunos tivessem uma participação efetiva. O objetivo era que os alunos não fossem apenas agentes passivos dentro do processo de ensino, mas que também contribuíssem de maneira ativa na construção de seu conhecimento. As docentes buscaram auxílio em alguns trabalhos que discutem o tema dentro da formação do profissional da computação em cursos de graduação. Dentre estes trabalhos dois deles podem ser destacados e são descritos a seguir [Mann, Smith e Muller 2008][Mankoff, Kravets e Blevis 2008].

O resultado de uma pesquisa que foi realizada no "*The 13th Annual Conference on Innovation and Technology in Computer Science Education*" sobre políticas para Educação da Computação para Sustentabilidade é apresentada no primeiro trabalho [Mann, Smith e Muller 2008]. O evento é promovido pelo ACM *Special Interest Group on Computer Science Education* (SIGCSE). O objetivo principal desse trabalho foi comparar as diferentes concepções que os educadores da área de Computação tinham sobre o tema Sustentabilidade. A partir dessa pesquisa, foi possível pontuar ações pedagógicas que podem ser adotadas pelos docentes da área para incorporar a Sustentabilidade com um foco transversal na formação do profissional de computação. Dentre estas ações foram sugeridos diversos temas que podem ser trabalhos nas disciplinas, incentivando os alunos a construírem soluções que possam contribuir de alguma forma para a sustentabilidade do planeta. O segundo trabalho apresenta algumas áreas da computação que podem contribuir para a sustentabilidade através de projetos [Mankoff, Kravets e Blevis 2008]. Dentre outras, podem ser destacadas as áreas: redes de computadores, *datacenters*, hardware, computação em nuvens.

Baseando-se nesses trabalhos as docentes responsáveis pela disciplina procuram planejar um direcionamento que demonstrasse que diferentes temas na Computação poderiam ser abordados considerando a visão corporativa da aplicação das Tecnologias da Informação através do desenvolvimento sustentável. Para que esta abordagem fosse efetivamente utilizada decidiu-se que no tópico "alicerces da sustentabilidade: social,

ambiental e econômico" seriam apresentados alguns modelos e abordagens que buscam dar suporte às empresas que desejam atuar de forma sustentável. Dentre esses modelos foram selecionados o *Natural Step*, o Capitalismo Natural e o *Triple Bottom Line*. Ficou decidido que a disciplina iria se concentrar na abordagem *Triple Bottom Line*, um dos modelos mais adotados pelas empresas. O modelo aborda o retorno de investimentos da empresa a longo prazo a partir de determinadas ações que visam o crescimento econômico, preservação ambiental e equidade social [Almeida, 2002].

Finalmente, com o objetivo de garantir a participação efetiva dos alunos no processo de aprendizado, bem como de avaliar esse aprendizado, três tipos diferenciados de atividades foram planejadas para serem propostas aos alunos: estudos de caso, seminários e projeto. O objetivo fundamental dessas atividades era estimular a participação dos alunos na solução de problemas de maneira que eles visualizassem como estas soluções poderiam contribuir para a sustentabilidade. Tais atividades serão descritas na seção a seguir.

3. Condução da Disciplina - Tecnologia da Informação e Sustentabilidade

No início da disciplina foram apresentados aos alunos os objetivos da disciplina, os tópicos que seriam abordados e as atividades que seriam desenvolvidas durante o semestre. Um ponto interessante foi que, embora os alunos do curso tivessem contato com o tema em disciplinas anteriores, a expectativa em torno da disciplina era grande.

A cada encontro eram propostos estudos de caso, onde os alunos tinham que utilizar os conceitos vistos e as discussões realizadas durante a aula para propor uma solução. Era apresentado um cenário aos alunos e estes tinham que discutir em grupo e apresentar uma solução ao final do encontro.

No primeiro encontro foi realizada uma atividade onde os alunos deveriam responder a duas questões: "o que é sustentabilidade para você?" e "como a computação pode contribuir para a sustentabilidade?". As respostas foram desenvolvidas individualmente e entregues às docentes em formato eletrônico. As docentes leram as respostas e observaram que os alunos tinham a visão de que a sustentabilidade estava presente apenas em ações que preservavam o meio ambiente, sem possuir a visão plural de que é necessário um equilíbrio econômico e social também. No sentido de motivar os alunos e ilustrar a importância do tema tanto para a sua formação pessoal, quanto para sua formação enquanto profissional da Computação, foram realizadas ainda as atividades: realização de um teste que avalia a quantidade de recursos naturais gastos por uma pessoa com base no seu estilo de vida [Pegada Ecológica 2008]; exibição do vídeo "A história das coisas" [Ciclo Vivo 2009]; apresentação de um conjunto de notícias recentes relacionando os temas sustentabilidade e computação.

No segundo encontro as docentes buscaram trabalhar juntamente com os alunos em discussões que demonstrassem a necessidade do equilíbrio do tripé econômico, social e ambiental. Inicialmente, foi discutido sobre o eixo central da sustentabilidade corporativa: a responsabilidade social empresarial. Segundo o *World Business Council for Sustainable Development* [WBCSD, 2000], a responsabilidade social empresarial se caracteriza pelo "comprometimento permanente dos empresários de adotar um comportamento ético e por contribuir para o desenvolvimento econômico, melhorando, simultaneamente, a qualidade de vida de seus empregados e de suas famílias, da

comunidade local e da sociedade como um todo". A partir destas definições foram apresentados os modelos que procuram apoiar o desenvolvimento sustentável nas empresas, focando especialmente no modelo *Triple Bottom Line*. Como atividade, ao final do encontro os alunos tiveram que apontar quais seriam as contribuições para o tripé da sustentabilidade a partir de um dado cenário que empregava uma solução computacional. A partir da definição do modelo foram discutidas diversas áreas da Computação que poderiam contribuir para a sustentabilidade: *Cloud Computing*, Redes Sociais, *e-Learning*, Computação Ubíqua, Cidades Inteligentes, Mineração de Dados, Reciclagem, Virtualização, Sensores, entre outros. Durante as discussões os próprios alunos interagiram apresentando exemplos a partir dos diversos temas.

Ao final das discussões os alunos formaram grupos com três integrantes para que os temas para os seminários fossem distribuídos. Os alunos teriam então dois meses para estudar o assunto e preparar um artigo de seis páginas que descrevesse esses resultados, para em seguida apresentarem oralmente os resultados do estudo. Os alunos deveriam estudar e apresentar os conceitos computacionais relacionados ao tema, identificar e discutir em quais índices de sustentabilidade havia impacto, finalizando com a apresentação de uma solução computacional que impactasse na sustentabilidade (estudo de caso ou produto) e que já tenha sido empregada em algum cenário. Os grupos também teriam uma semana para definir um tema para um projeto que seria apresentado ao final da disciplina. Em duas semanas haveria uma apresentação oral onde os grupos proporiam o tema, apresentando quais eram os impactos da solução considerando o modelo *Triple Bottom Line*. Nesta apresentação as docentes responsáveis pela disciplina e o grupo de alunos discutiriam as propostas realizando direcionamentos no projeto. Além disto, as docentes se dividiram na orientação dos grupos de maneira que houvesse um acompanhamento mais próximo do desenvolvimento do trabalho. Os resultados obtidos com os seminários e com os projetos serão apresentados na seção posterior.

No terceiro encontro foi abordado o tema de TI Verde, que está diretamente ligado ao aspecto ambiental da sustentabilidade, vez que foca redução do impacto ambiental de um departamento ou instituição que empregue tecnologia da informação. Esse tema permite facilmente que os alunos visualizem a conexão da sustentabilidade com a tecnologia da informação. Essencialmente foram discutidos temas como consumo de energia, uso de materiais perigosos, descarte de lixo eletrônico, dentro do ciclo projeto, fabricação, utilização e descarte verde dos recursos de tecnologia, sempre focando na importância de se avaliar os impactos das soluções. Como estudo de casos desse tema os alunos deveriam fazer um levantamento das práticas de TI Verde aplicadas em alguma grande empresa da área de TI.

Após as diversas discussões anteriores foi apresentado aos alunos como utilizar indicadores de sustentabilidade de maneira corporativa. Para isto foi apresentada a "matriz de sustentabilidade corporativa" [SustainAbility 2010], baseada no *Triple Bottom Line*, que permite realizar análises estratégicas que relacionam aspectos fundamentais da sustentabilidade social e ambiental a uma série de reconhecidos fatores de sucesso nos negócios, como: reduzir custos pela diminuição dos impactos ambientais e pelo aumento da eco-eficiência; aumentar receitas pela melhoria dos processos do ponto de vista ambiental e pelo favorecimento da economia local; melhorar a imagem da empresa pela oferta de produtos e serviços ambientalmente sustentáveis; desenvolver

o capital humano pela gestão efetiva de recursos humanos, abrangendo toda a força de trabalho; entre outros. O viés corporativo sobre a sustentabilidade permitiu que os alunos visualizassem que as contribuições podem ser trabalhadas de maneira atrativa para as empresas. Embora a implantação do tema responsabilidade social empresarial possa acarretar opiniões divergentes, quando existem ações aderentes as estratégias corporativas estas podem otimizar oportunidades para as empresas, trazendo benefícios que vão além dos financeiros, como atrair públicos-alvos, melhorar as relações com as comunidades onde a empresa atua, melhorar a imagem e reputação da marca da empresa. Outro ponto importante que foi apresentado aos alunos diz respeito à adoção de indicadores de sustentabilidade pelo mercado financeiro. Desde que o desenvolvimento sustentável passou a ser encarado como algo possível pelas empresas e com retornos em diversos aspectos, o mercado financeiro incorporou índices que avaliam empresas socialmente responsáveis, sustentáveis e rentáveis. Estes índices procuram demonstrar que empresas sustentáveis geram valor para os acionistas no longo prazo, pois estão mais preparadas para enfrentar riscos econômicos, sociais e ambientais. Em 1999, foi lançado o Índice *Dow Jones* de Sustentabilidade - *Dow Jones Sustainable Index* [DJSI 1999], que acompanha o desenvolvimento financeiro de empresas mundiais que são orientadas a sustentabilidade. Em 2005, a BOVESPA juntamente com outras instituições criou o Índice de Sustentabilidade Empresarial [ISE 2005].

Os últimos tópicos abordados foram as "Arquiteturas para Sistemas Computacionais" e a "Gestão do Conhecimento" estes voltados para TI e Sustentabilidade. No primeiro tópico foram explorados alguns modelos sobre arquitetura como o TOGAF (*The Open Group Architecture Framework - Framework arquitetural do Open Group*) [The Open Group 2006], convergindo para as arquiteturas SOA (*Service Oriented Architecture – Arquitetura Orientada a Serviços*) [Erl 2009] utilizada em conjunto com *BPM (Business Process Managemen*t) [OMG, 2006]. Foi discutido como a adoção destas arquiteturas pode contribuir para um maior controle do processo de negócio e da infraestrutura envolvida, e como estes impactam e auxiliam as empresas no maior controle de TI. No tópico de Gestão de Conhecimento foram discutidas as relações entre os temas TI, conhecimento e sustentabilidade, focando nas contribuições que o profissional de TI tem a dar na produção e gerenciamento do conhecimento, e nos benefícios que a sociedade pode obter a partir disso. A partir das discussões, em duplas, os alunos deveriam selecionar, estudar e apresentar ao grupo todo um artigo relacionado ao tema, proveniente do Workshop de Computação Aplicada em Governo Eletrônico (WCGE) ou do II Workshop de Computação Aplicada a Gestão do Meio Ambiente e Recursos Naturais (WCAMA), ambos ocorrido no Congresso da Sociedade Brasileira de Computação [CBSC 2010], sempre identificando a relação entre TI, conhecimento e sustentabilidade.

Ao final da disciplina os alunos apresentaram os seminários e projeto final. Os resultados bem como a discussão acerca destes é realizada a seguir.

4. Resultados das Atividades Desenvolvidas

Ao término da disciplina as docentes solicitaram aos alunos que estes respondessem novamente as duas perguntas que foram realizadas no início da disciplina: "o que é sustentabilidade para você?" e "como a computação pode contribuir para a

sustentabilidade?". Observou-se que as respostas foram bem diferentes em relação às iniciais demonstrando que os alunos passaram a ter uma visão mais madura sobre o tema, pontuando diversos elementos e não mais somente o meio ambiente como foco da sustentabilidade.

Observou-se também que as atividades propostas durante a condução da disciplina (estudos de caso, seminários e projeto final) tiveram contribuições significativas e diferenciadas para os alunos. Os estudos de caso permitiram que os alunos discutissem e exercitassem seu entendimento sobre os assuntos abordados de maneira imediata.

Já os seminários permitiram que os alunos pudessem estudar temas de Computação sobre o viés da sustentabilidade. A elaboração do artigo permitiu que os alunos desenvolvessem a capacidade argumentativa, mas com embasamento teórico, possibilitando que eles transitassem entre assuntos técnicos e não técnicos dentro de um mesmo contexto. A Tabela 1 apresenta os seminários apresentados pelos alunos.

Tabela 1. Seminários apresentados

Grupo	Tema	Foco do Seminário	Aplicação/Produto
1	Reciclagem de Produtos de TI	Lixo eletrônico	Iniciativas implementadas por empresas como Nokia, Sony Ericsson, Itautec, Motorola, Toshiba e Empresas americanas focadas na reciclagem de componentes eletrônicos.
2	Mineração e Análise de Dados	Análise de dados em Hospitais	Estudo de uma ferramenta para análise de dados em hospitais, com objetivo de identificação de problemas e possíveis soluções e aumento de eficiência em áreas como assistência (hospitalizações, urgências, etc), gestão econômica, recursos humanos, identificação de fatores de risco de patologias, análise de perfis de doadores de sangue , entre outros.
3	Datacenters	Consumo de Energia	Implementação realizada pelo Itaú Unibanco e *Datacenters* flutuantes
4	Computação Ubíqua	Melhoria da qualidade de vida e gestão dos recursos naturais	Casas inteligentes: iluminação, climatização, gestão do sistema hidráulico, etc
5	Redes Sociais	Diminuir as distâncias e disseminação da informação	*Facebook*: Consumo de energia eficiente na computação – softwares que melhoram a performance dos sistemas e otimizam o armazenamento de dados.
6	Sensores	Inovações que aplicam sensores na sustentabilidade	Sensor para controlar o desperdício de água nas lavouras; óculos-mouse que garante a inclusão digital de deficientes físicos e sensores em cidades inteligentes (Detector de Enchentes, Detector de Deslizamentos, Detector de poluição, Iluminação Inteligente de Vias Públicas, Semáforos Inteligentes, Carros Inteligentes, SinalizaSom).

Como os temas dos seminários eram abrangentes os grupos tiveram a oportunidade de focar em elementos específicos dentro do contexto explorado. Através de uma análise dos impactos dentro do tripé da sustentabilidade os alunos conseguiram descrever as contribuições que as aplicações/produtos trouxeram para o tema.

O desenvolvimento do projeto foi importante para que eles próprios propusessem soluções. A apresentação inicial do projeto, durante o período inicial da disciplina, permitiu que os alunos visualizassem melhor as contribuições que as propostas poderiam trazer. A Tabela 2 apresenta os projetos desenvolvidos.

Tabela 2. Projetos Desenvolvidos

Grupo	Tema	Descrição do Projeto
1	Green Building	Sistema que permite avaliar indicadores que podem influenciar no aspecto sustentável na construção de uma casa: área permeável, número mínimo de uma janela por cômodo, número mínimo de uma porta por cômodo, etc.
2	Web Collaborative Learning	Ambiente que permite que um grupo de pessoas possa compartilhar conhecimentos. Os grupos são formados de acordo com o perfil das pessoas que participam. As informações encontradas pelos participantes são compartilhadas entre todos que possuem um mesmo tema de interesse e perfil.
3	Virtual Home	Ambiente de comercio eletrônico que apresenta móvel e eletrodomésticos que possuem certificações de sustentabilidade. Os alunos utilizaram selos para avaliar a sustentabilidade, onde o primeiro era baseado em uma instituição especializada em certificação determina para o produto e o segundo são os selos criados pelas próprias empresas.
4	Análise da água	Desenvolvimento uma base de conhecimento sobre características da água para utilização em sensores que realizarão medições afim de avaliar sua potabilidade.
5	Redes Sociais de Animais Silvestres	Rede social onde é possível trocar informações, por meio de notícias, eventos publicados pelos próprios usuários e por ONGs focadas na preservação animal.
6	Jogo Educativo	Jogo para conscientização de crianças com idade entre cinco e sete anos sobre questões ambientais, abrangendo os temas de economia de energia e água, reciclagem de resíduos e preservação de plantas e animais.

Durante a apresentação do projeto os alunos deveriam não só apresentar as soluções bem como utilizar a matriz de sustentabilidade corporativa para defender os pontos fortes e fracos da solução apresentada. Dentre os trabalhos apresentados pode-se destacar algumas soluções consideradas interessantes. No *Virtual Home* os alunos propuseram uma maneira de validação do selo de sustentabilidade. No caso do primeiro tipo de selo, era realizada uma validação a partir do código de controle fornecido pela empresa no momento de cadastrar o seu produto na plataforma. Esta validação seria realizada através de *Web Services* disponibilizados pelas próprias instituições certificadoras. Na análise da água uma base de conhecimento foi analisada com o auxílio de uma ferramenta *open source* de mineração de dados desenvolvida pela Universidade de Waikato, chamada WEKA [WEKA 2010]. Inicialmente foi simulada uma coleta de dados através de sensores, com base em um conjunto de informações disponíveis a respeito de diversos atributos mensuráveis que indicam a qualidade e potabilidade da água. Uma vez gerados os dados, estes foram analisados através de vários algoritmos de aprendizado com o objetivo de encontrar o melhor algoritmo de classificação para ser utilizado nos sensores para o reconhecimento da qualidade da amostra. Os parâmetros analisados estão em conformidade com o determinado pelo Ministério da Saúde, na portaria n. 518 de 25 de Março de 2004. A Rede Social de animais silvestres foi motivada por uma situação real: o trabalho desenvolvido pela ONG Mata Ciliar [Mata Cilar, 2010] que trabalha com preservação de matas ciliares e educação ambiental e reabilitação de animais silvestres. O objetivo principal do projeto proposto é fomentar as discussões e a disseminação das informações sobre os trabalhos realizados pelas entidades.

5. Considerações Finais

Este artigo apresentou a experiência do curso de Ciência da Computação da UFSCar, campus Sorocaba em implantar o tema Sustentabilidade como um componente curricular de formação do aluno. Em consonância tanto com as diretrizes da universidade bem com a da Sociedade Brasileira de Computação, essa proposta buscou fomentar a formação de profissionais que sejam socialmente responsáveis. A primeira edição da disciplina ocorreu no segundo semestre de 2010. Uma nova turma será oferecida no segundo semestre de 2011.

A implantação de um componente curricular que reúne elementos técnicos e sociais é um desafio. Buscando sempre realizar o relacionamento dos elementos técnicos com os sociais, as docentes adotaram estudos de casos reais como forma de gerar discussões no grupo.

A retroalimentação dos alunos sobre as atividades da disciplina foram importantes e pontuaram que eles consideram fundamental a abordagem de tal tema no curso. Segundo os alunos, a disciplina permitiu que eles ampliassem a visão sobre o tema, especialmente dentro do contexto da Computação. Foi realizada uma avaliação sobre a disciplina. Nessa avaliação foi constatado que 50% dos alunos consideraram que os temas abordados na disciplina foram pertinentes para a sua formação e que 52% dos alunos disseram que a disciplina contribuiu para sua formação. Isto se deve ao fato de que os alunos de um modo geral reconhecem a importância do tema não só para sua formação pessoal, como também profissional. Para eles o perfil profissional que está sendo formado faz com que eles possam atuar como agentes ativos, contribuindo para a sustentabilidade através de soluções computacionais. Outro ponto interessante foi que muitos alunos que cursaram a disciplina acabaram por escolher temas relacionados à sustentabilidade para o desenvolvimento de trabalhos de outras disciplinas.

Referências Bibliográficas

Almeida, M. F. L. (2006) “Sustentabilidade Corporativa, Inovação Tecnológica e Planejamento Adaptativo - Dos princípios à ação”, Tese de Doutorado, PUC-Rio.

Cedir (2010). “Centro de Reciclagem de Lixo Eletrônico”, Disponível em: http://www.usp.br/imprensa/?p=1591, Acessado em: 20/01/2011.

Ciclo Vivo (2010) “História da coisas”, Disponível em: www.ciclovivo.com.br/video.php/823/a_historia_das_coisas/, Acessado em: 13/08/2010.

CSBC (2010) “Anais do XXX Congresso da Sociedade Brasileira de Computação”, Disponível em: http://www.inf.pucminas.br/sbc2010/anais/index.htm, Acessado em: 21/09/2010.

Diretrizes Curriculares SBC (2005) “Currículo de Referência para Cursos de Graduação em Bacharelado em Ciência da Computação e Engenharia de Computação”, Disponível em: http://portal.sbc.org.br/educacao/lib/exe/fetch.php?media=documentos:cr2005.pdf, Acessado em: 20/10/2009.

DJSI (1999) “Dow Jones Sustainable Index”, Disponível em: http://www.sustainability-index.com/, Acessado em: 20/08/2010.

Erl, T. (2009) “SOA: Princípios de Design de Serviços”, Prentice Hall.

Hedstrom, G., Poltorzycki, S., Stroh, P. (1998) “Sustainable development: the next generation - how real, how soon and who’s doing what?”, Prism - The Arthur D. Little Journal, vol 4, pp.5-19.

ISE (2005) “Índice de Sustentabilidade Empresarial”, Disponível em: http://www.bmfbovespa.com.br/indices/ResumoIndice.aspx?Indice=ISE&idioma=pt-br, Acessado em: 20/08/2010.

Mankoff, J.; Kravets, R.; Blevis, E. (2008) “Some Computer Science Issues in Creating a Sustainable World”, Journal Computer, vol. 41, n. 8, pp. 102-105.

Mann, S.; Smith, L.; Muller, L. (2008) “Computing Education for Sustainability”, Inroads — SIGCSE Bulletin, vol 40, n. 4, pp. 183 – 193.

Mata Cilar (2010) “Associação Mata Cilar”, Disponível em: http://www.mataciliar.org.br/mata/, Acessado em: 20/09/2010.

OMG (2006) “Business Process Management”, Business Process Management Initiative, Object Management Group, Disponível em: http://www.bpmi.org/ , Acessado em: 20/06/2010.

Pegada Ecológica (2008) “Que marca você quer deixar no planeta?”, Disponível em: http://www.pegadaecologica.org.br/index.php, Acessado em: 13/08/2010.

Perfil UFSCar (2008) “Perfil do profissional a ser formado na UFSCar”, Disponível em: http://www.prograd.ufscar.br/arquivos/perfil_profissional_ufscar.pdf, Acessado em: 30/09/2009.

SustainAbility (2010) Disponível em: http://www.sustainability.com/, Acessado em: 10/06/2010.

The Open Group (2006) “The Open Group Architecture Framework”, Disponível em: http://www.opengroup.org/architecture/togaf8-doc/arch/, Acessado em: 08/05/2010.

United Nations Environment Programme (1972) “Report of The United Nations Conference on the Human Environment”, Stockholm, Disponível em: http://www.unep.org/Documents.Multilingual/Default.asp?DocumentID=97, Acessado em: 20/08/2010.

WBCSD (2000) “World Business Council for Sustainable Development. Social responsibility: making good business sense”, Disponível em: http:// www. wbcsd.ch/DocRoot/IunSPdIKvmYH5HjbN4XC/csr2000.pdf>, Acessado em: 10/03/2010.

WEKA (2010) “Data Mining Software in Java”, Disponível em: http://www.cs.waikato.ac.nz/ml/weka/, Acessado em: 10/08/2010.

Boas Práticas de Ensino de Gerência de Projetos: Bolsa de Projetos

Rogério Güths[1], Cláudia Santos Fernandes[2], Bárbara Barros[1]

[1] Departamento de Ciências Exatas – Universidade Federal do Mato Grosso do Sul (UFMS) – Caixa Postal 252 – 79.304-902 – Corumbá – MS – Brasil

[2] Telecentro de Bodoquena – Secretaria de Assistência e Promoção Social de Bodoquena - Caixa Postal 63 – 79.390 – Bodoquena – MS – Brasil

rguths@gmail.com, fazvaleverde@hotmail.com, brgbarros@yahoo.com

***Abstract**: **Abstract**: This article describes the application of Stock Market's concept to help the Project Management teaching. It was reported Stock Market and Project Management fundamental concepts. It was related experiences in the application of this concept in the Project Management discipline that is a part of the Information System course at the Lutheran University of Brazil through an activity followed by the execution of a stock exchange, where students could invest on projects and ran the risk of making bad investments.*

***Resumo**: O presente artigo descreve a aplicação de conceitos de Mercado de Ações no auxílio ao ensino de Gerência de Projetos. São abordados conceitos fundamentais de Mercado de Ações e Gerência de Projetos. São relatadas experiências de aplicação destes conceitos na disciplina de Gerência de Projetos no curso de Bacharelado em Sistemas de Informação da Universidade Luterana do Brasil através de uma atividade de elaboração de projeto, seguida pela execução da bolsa de projetos, no qual os alunos investiam nos projetos estando sujeitos ao risco de realizarem maus investimentos.*

1. Introdução

Um projeto é um empreendimento não repetitivo, caracterizado por uma sequência clara e lógica de eventos, com início, meio e fim, que se destina a atingir um objetivo claro e definido, sendo conduzido por pessoas dentro de parâmetros pré-definidos de tempo, custo, recursos envolvidos e qualidade. De forma mais sucinta, o *Project Management Institute – PMI* (2002) define projeto como "*um empreendimento temporário com o objetivo de criar um produto ou serviço único*".

Estuda-se Gerência de Projetos – GP para se ter maior possibilidade de sucesso nos projetos. O ensino é caracterizado pelo uso intensivo de exemplos práticos, que permitem ao acadêmico enriquecer sua percepção da área e adquirir habilidades para detectar e resolver problemas, mesmo que não tenha experiência profissional em GP. Conforme PMI (2002), quanto mais se usa a metodologia, aumentam a maturidade e a chance de sucesso nos próximos projetos.

Neste artigo, primeiro são feitas considerações sobre GP; na seção seguinte são relatados conceitos básicos do Mercado de Ações; posteriormente, é mostrada a

aplicação dos conceitos abordados nas seções anteriores em experiências para três turmas da disciplina de GP, do curso de Sistemas de Informação da Universidade Luterana do Brasil – ULBRA; por último são apresentados os resultados alcançados.

2. Gerência de Projetos

Desde os anos 60, vem sendo estudada GP. O PMI é o pioneiro na regulamentação e distribuição deste conhecimento. Sendo uma entidade internacional sem fins lucrativos, ela congrega os profissionais de áreas relacionadas à GP. Fundado nos EUA, atualmente está presente praticamente no mundo inteiro.

O PMI classifica os processos necessários a um projeto em nove áreas do conhecimento: escopo, custo, prazo, qualidade, recursos humanos, comunicação, riscos, aquisições e integração. Cada uma destas deve ser conhecida, entendida e praticada pelo gerente de projetos para que seu projeto tenha êxito. Por outro lado, os processos de GP podem ser organizados em cinco grupos obedecendo a uma ordem cronológica:

- Iniciação – processos de organização inicial e autorização do projeto.
- Planejamento – processos no qual ocorrem a definição e o refinamento dos objetivos e seleção da melhor das alternativas de ação.
- Execução – processos em que pessoas e outros recursos são coordenados para realizar o planejamento proposto.
- Controle – processos que asseguram que os objetivos do projeto estão sendo atingidos, através da monitoração regular, identificando variações do planejamento e demandando ações corretivas, quando necessárias.
- Encerramento – processos que formalizam a aceitação do projeto e encerramento de uma forma organizada.

Conforme Martins (2005), as áreas estão sujeitas às interações até o fim do projeto. Por exemplo, se a data prevista para o término for antecipada, a qualidade, o custo, ou o escopo podem necessitar de alteração. Além disto, planejamento não é uma ciência exata – equipes distintas podem gerar planos diferentes para o mesmo projeto.

3. Mercado de Ações - Capitais

Para Bovespa (2010), o mercado de capitais é um sistema de distribuição de valores mobiliários, com o propósito de proporcionar liquidez aos títulos de emissão de empresas e viabilizar seu processo de capitalização. Os principais títulos negociados são representativos do capital de empresas — as ações — e permitem a circulação de capital para custear o desenvolvimento econômico destas. Ações são títulos de renda variável, emitidos por empresas, que representam a menor fração de seu capital. O investidor em ações é um co-proprietário da empresa, participando dos seus resultados. As ações são conversíveis em dinheiro, a qualquer tempo, pela negociação em bolsas de valores.

Os investidores buscam a melhor relação entre três aspectos: retorno, prazo e proteção. Ao escolher uma ação para investir, deve-se analisar a rentabilidade, liquidez e grau de risco. Normalmente, rentabilidade é diretamente proporcional ao risco. O investidor ao tornar-se acionista da empresa está apostando no sucesso desta. Ele recebe dividendo caso a empresa obtenha lucro e suas ações se desvalorizam caso contrário.

4. Bolsa de Projetos e sua Execução

O uso de exemplos práticos busca enriquecer, no acadêmico, a percepção da área e habilidade de resolver problemas, mesmo sem experiência profissional no tema.

Visando a formação completa do aluno em GP devemos levar em conta que:

- Segundo Shenhar, Raz e Dvir (2003), o planejamento é fundamental para o sucesso de um projeto, portanto deve ser uma das áreas mais exigidas.
- No âmbito de uma disciplina fica inviável propiciar a cada aluno, a experiência completa de ser gerente de um projeto na área de desenvolvimento, devido a restrições de recursos humanos e financeiros.
- Utilizando-se da análise de diversos planejamentos o acadêmico aguça a capacidade crítica e de distinção entre um projeto bom e um mal.
- De acordo com Dinsmore e Cavalieri (2003) riscos são muito importantes para o sucesso do projeto GP, e estão associados à ganhos ou perdas.

Desta forma, planejou-se a seguinte estratégia de ensino:

- Os acadêmicos foram os responsáveis pela escolha dos temas, coleta de informações, elaboração e apresentação do projeto.
- Cada projeto continha: Viabilidade, Descrição do Produto a ser desenvolvido, Recursos Necessários (financeiros, prazos e humanos) e Planejamento.
- Na data de apresentação dos projetos, cada acadêmico foi convidado a investir recursos de R$ 5,00, que eram convertidos em papéis, porém os alunos que não desejassem participar por quaisquer motivos tinham liberdade para tal.
- Ao final das apresentações, cada acadêmico escrevia no verso de seus papéis seu nome e o projeto no qual aquele papel estava sendo investido. Na entrega dos seus papéis, cada aluno justificava seu investimento.
- O acadêmico não podia investir em seu próprio grupo.
- Os três projetos que mais recebessem investimentos retornariam o valor em reais investido através dos papéis.
- Os papéis investidos nos demais projetos perdiam o valor, que era revertido em um lanche na próxima aula a ser repartido com todos.
- O professor apresentava: seu parecer sobre os projetos indicando os melhores na sua avaliação, a correlação entre esta e os projetos vencedores, uma analise das justificativas dos investimentos e sugestões de melhoria a cada projeto.
- Realizava-se debate sobre a atividade, as avaliações e sugestões de melhoria.

A avaliação do professor analisou as seguintes questões: o escopo está bem definido? Foi comprovada viabilidade financeira? Definiu corretamente cronograma, atribuição de responsabilidades e marcos de projeto? Comprovou a exeqüibilidade do cronograma? Previu adequadamente os recursos humanos necessários? Está prevista comunicação interna e externa do projeto? Nos marcos de projeto estão previstos critérios de qualidade para aprovação? Os riscos previstos e trabalhados são coerentes com o projeto?

Nem todos os projetos vencedores em investimentos foram os melhores na avaliação do professor, o que propiciou amplo debate sobre os pontos analisados bem como sobre as justificativas para o investimento.

Uma alternativa à devolução dos investimentos seria conferir lucros baseados nos investimentos perdidos, de forma a não sobrarem valores. Para não frustrar os perdedores, esta abordagem não foi utilizada.

Tabela 1- Resumo das execuções da estratégia nas turmas

Turmas	Qtd de alunos	Qtd de grupos	Comparação entre as avaliações dos alunos e do professor
2008/1	14	6	Dos 3 projetos vencedores na avaliação dos alunos, somente um o professor não considerou vencedor
2008/2	19	9	Os 3 projetos vencedores na avaliação dos alunos também foram os melhor avaliados pelo professor
2009/1	13	6	Dos 3 projetos vencedores na avaliação dos alunos, somente um o professor não considerou vencedor

5. Conclusões

A metodologia utilizada, em que os alunos vivenciavam um ambiente real de GP mostrou-se muito eficiente, tanto na motivação e comprometimento demonstrado pelos alunos quanto em seu desempenho na disciplina, corroborado através das avaliações finais. Porém, destaca-se que a melhora no desempenho de análise dos alunos depende muito do perfil da turma e da variação na qualidade dos projetos apresentados. Uma turma na qual os projetos são todos muito bons ou todos muito ruins terá menor amadurecimento neste aspecto. Particularmente na turma de 2008/2, ocorreu a maior discrepância na qualidade dos projetos apresentados e, consequentemente o debate que finalizou a atividade foi o mais extenso de todos.

Para que a atividade seja bem sucedida, sugerem-se algumas limitações: número máximo de oito grupos; apresentações entre quinze a vinte minutos cada, com até cinco minutos para questionamentos; idealmente a atividade deve ter início e término na mesma aula, alocando três a quatro horas, porém se não for possível, os acadêmicos devem compor *checklist* de avaliação para os projetos, anotando as principais características observadas em cada projeto.

Este tipo de atividade é altamente recomendado para turmas de GP, pois propicia importante prática no planejamento e análise de projeto. Acredita-se que esta atividade possa ser estendida a qualquer outra disciplina que envolva a idéia da venda de produto ou convencimento do público ou clientes por parte dos apresentadores.

Referências

BOVESPA. Mercado de Capitais. Disponível em: http://www.bmfbovespa.com.br/Pdf/merccap.pdf Acesso em 20 de Janeiro de 2011.

PMI (2002) Tradução livre do PMBOK 2000 Disponível em: http://www.mp.ap.gov.br/tecnologia/wp-content/uploads/2010/03/608-_PMBOK2000Portugues.pdf . Acesso em 15 de setembro de 2010.

DINSMORE, P., C., CAVALIERI, A.. Como se Tornar um Profissional em Gerenciamento de Projetos, Editora Qualitymark, SP, 2003.

SHENHAR, A., RAZ, T. e DVIR, D.. An Empirical Analysis of the Relationship Between Project Planning and Project Success. *International Journal of Project management*, 21, 2 : 89-95, (2003)

MARTINS, J.C. Gerenciando projetos dedesenvolvimento: de software com PMI, RUP eUML. 4ª ed. Brasport, Rio de Janeiro, 2007.323p

Uma nova metodologia para o desenvolvimento de sistemas de correção automática de listas de exercícios de programação baseada em testes de unidade

Thiago F. Noronha[1], **Natã Goulart**[2],
Matheus A. Pacheco[3], **Gabriel G. A. Barbosa**[3]

[1]Departamento de Ciência da Computação, UFMG
Av. Antônio Carlos, 6627 - 31270-010 - Belo Horizonte - MG - Brazil

[2]Mestrado em Modelagem Matemática e Computacional - CEFETMG
Av. Amazonas 7675 - 30.510-000 - Belo Horizonte - MG - Brasil

[3]Centro de Pesquisas e Desenvolvimento em Engenharia Elétrica, UFMG
Av. Antônio Carlos, 6627 - 31270-010 - Belo Horizonte - MG - Brazil

tfn@dcc.ufmg.br, natan@dppg.cefetmg.br,

gabriel.guedesaz@gmail.com, matheuspacheco100@gmail.com

Abstract. *This paper proposes a new methodology for developing systems that automatically correct computer programming exercises based on unit tests. Using this methodology, students have each unit of their code evaluated just after they program it, without the need of professor intervention. In the case of some piece of code (procedure or function) is incorrect, an error message is displayed to the student, accompanied by an explanation that may help him to implement that piece of code correctly. A case study in classes of Algorithms and Data Structures of the Federal University of Minas Gerais showed that the unit tests helped the students to rapidly identify and correct the errors in their codes without the interference of the professor.*

Resumo. *Este trabalho propõe uma nova metodologia para o desenvolvimento de sistemas de correção automática de listas de exercícios de programação baseada em testes de unidade. Utilizando esta metodologia, os alunos tem cada unidade de seu código avaliada imediatamente após a sua codificação sem a necessidade de intervenção do professor. No caso de alguma unidade (procedimento ou função) estar incorreta, uma mensagem de erro é exibida para o aluno, acompanhada por uma explicação que pode ajudá-lo a implementar aquela unidade corretamente. Um estudo de caso em turmas de Algoritmos e Estruturas de Dados da UFMG mostrou que o uso dos testes de unidade permitiu que os alunos rapidamente identificassem e corrigissem os erros de seu código sem a ajuda do professor da disciplina.*

1. Introdução

Com a adoção de várias universidades federais ao programa de Reestruturação e Expansão das Universidades Federais - REUNI, espera-se um grande aumento no número de alunos por turma nas disciplinas de ensino de programação de computadores. A estimativa

é que o número médio de alunos por turma em disciplinas como Algoritmos e Estrutura de Dados I e II da Universidade Federal de Minas Gerais - UFMG chegue a aproximadamente 100 alunos. Estas disciplinas são oferecidas em diversos cursos de graduação em instituições de todo o país. Na UFMG, vários cursos oferecem estas disciplinas em sua grade curricular, dentre os quais podemos citar: Ciência da Computação, Engenharia de Controle e Automação, Engenharia da Produção, Engenharia Elétrica, Estatística, Matemática Computacional e Sistemas de Informação.

O ensino de programação de computadores exige dos alunos a resolução de um grande número de exercícios, a fim de que eles desenvolvam o raciocínio lógico sequencial necessário para solucionar problemas computacionais. A correção destes exercícios demanda muito tempo do professor da disciplina. Com o crescimento do número de alunos nas disciplinas de ensino de programação, a correção manual de todos os exercícios se tornará impraticável.

A abordagem mais empregada atualmente para correção automática de exercícios de programação é baseada nos chamados *Juízes Online* (do inglês *Online Judge*) [Cheang et al. 2003]. Nesta abordagem, o aluno escreve um programa de computador que soluciona um problema computacional e submete seu programa via Internet para um servidor de testes. Este servidor trata o programa do aluno como uma ”caixa preta”, aplicando-o a vários casos de teste. Se o resultado do programa do aluno for igual ao resultado esperado, o juiz *online* atesta que o programa do aluno está correto. Uma vez que o programa do aluno não produz o resultado esperado, o juiz *online* não tem como indicar qual é o ponto do programa que está errado, nem sugerir uma possível solução. Além disso, nenhuma pré-condição pode ser avaliada dentro do programa do aluno. Mesmo que o enunciado do exercício exija que o aluno utilize um determinado algoritmo ou estrutura de dados específica, isto não pode ser avaliado pelo juiz *online*. Uma outra desvantagem desta abordagem, está na necessidade da compra e manutenção de um servidor de testes que demanda recursos financeiros e de pessoal técnico da universidade, o que impede sua utilização em centros de ensino com poucos recursos financeiros. Além disso, se por alguma razão o servidor estiver congestionado ou inoperante, os alunos não podem avaliar os seus exercícios. Surge então a necessidade de se estudar outros mecanismos alternativos ou complementares para correção automática de exercícios de programação.

Este trabalho propõe uma nova metodologia para o desenvolvimento de sistemas de correção automática de listas de exercícios de programação baseada em testes de unidade (do inglês *unit tests*). A Seção 2 apresenta uma revisão bibliográfica sobre os sistemas utilizados para avaliar algoritmos codificados em determinada linguagem de programação e suas aplicações em sistemas de correção automática de exercícios de programação. A Seção 3 descreve a metodologia para o desenvolvimento de sistemas de correção automática de listas de exercícios de programação proposta neste trabalho. Na Seção 4 é discutido o estudo de caso dessa metodologia aplicada em cursos de Algoritmos e Estruturas de Dados da UFMG. Por fim, a Seção 5 apresenta as considerações finais sobre este trabalho.

2. Trabalhos Relacionados

Esta seção apresenta uma revisão dos principais sistemas e tecnologias que podem ser aplicadas no ensino de linguagens de programação e em outros cursos *online*.

2.1. Juízes Online

Os sistemas baseados em juízes *online* são capazes de receber o código fonte de um programa codificado em uma linguagem de programação qualquer e verificar se este programa apresenta, para determinadas entradas pré-definidas, as saídas esperadas. Estes sistemas foram inicialmente concebidos para uso em concursos e competições de programação [ICPC 2010]. Nos últimos anos porém, instituições de ensino têm utilizado e adaptado estes sistemas para o ensino de linguagens de programação em cursos técnicos e superiores [Kurnia et al. 2001, Kurnia et al. 2008]. Estes sistemas estão sendo integrados a sistemas de ensino e acompanhamento *online* chamados *Learning Management System - LMS* [Woo et al. 2008]. Um dos sistemas LMS mais utilizado para o acompanhamento acadêmico e também em cursos a distância é o *Moodle* [MOODLE 2010], que faz uso das amplas possibilidades da Internet para o ensino presencial ou a distância.

Atualmente, existem propostas de adicionar aos juízes *online* ferramentas com o objetivo de identificar a cópia dos trabalhos que são submetidos. A detecção desta cópia, conhecida na literatura como plágio, é uma função comum nos sistemas de avaliação utilizados no meio acadêmico. Dentre as opções de detecção de plágios disponíveis, podemos citar um *plugin* desenvolvido para o *Moodle* baseado no sistema chamado *Measure Of Software Similarity - MOSS* [Schleimer et al. 2003] que está disponível em [Schleimer et al. 2010].

Dentre as aplicações práticas de sistemas baseados em juízes *online* encontradas na literatura podemos citar o uso do sistema de correção automática que é utilizado na escola de computação da Universidade de Singapura, descrito em [Cheang et al. 2003]. Nesta universidade, até o final da década de 90, os trabalhos acadêmicos na área de ensino de linguagens de programação eram corrigidos manualmente. A partir do ano 2000, esta e outras instituições [Kurnia et al. 2001, Kurnia et al. 2008] começaram a utilizar ferramentas de correção automática de listas de exercício de programação.

Um sistema de correção automática baseado em juízes *online* exige que as saídas geradas pelos programas sigam um padrão e formato rígido. Caso a saída não esteja formatada exatamente igual a do gabarito esperado, o sistema pode considerar que o programa não resolve o problema corretamente. Se por exemplo, o programa espera como saída dados formatados em duas colunas, o resultado pode ser considerado errado caso o aluno use espaços ao invés de tabulações.

2.2. Testes de unidade

Testes de unidade (do inglês *unit tests*) são amplamente utilizados atualmente no desenvolvimento e teste de sistemas comerciais. Eles tem este nome porque cada teste de unidade testa uma unidade lógica do programa independentemente das outras. Em programação estruturada esta unidade lógica pode ser definida como um procedimento ou função [Dustin et al. 1999]. Ao longo do desenvolvimento e manutenção do software, se alguma mudança resultar num erro em uma determinada função os testes relacionados a esta função falham e o erro pode ser imediatamente identificado. Podem existir vários

testes para uma mesma unidade de código, cada um deles avaliando uma propriedade que deve ser atendida por aquela unidade.

Os testes de unidade fazem parte de um conjunto de testes chamados de testes de *caixa branca* [Misra 2003]. Diferentemente dos testes de *caixa preta*, onde o sistema recebe uma entrada e após o processamento gera uma saída, os testes de caixa branca permitem a avaliação do funcionamento interno do sistema. Os testes de unidade são utilizados para verificar se o *software* faz corretamente e adequadamente todos os detalhes especificados no projeto ou, no caso do uso acadêmico, no enunciado do exercício. Na prática, como são testes que irão avaliar os detalhes da implementação, são codificados normalmente pelo próprio desenvolvedor que tem o conhecimento da lógica utilizada. Em sistemas grandes e complexos formados pelas camadas de lógica, dados e interface com o usuário, os testes de unidade devem verificar todos os possíveis caminhos do código, avaliar telas, menus, mensagens de usuário devidamente formatadas, validar campos e verificar se as exceções são devidamente codificadas.

Segundo [Dustin et al. 1999], testes de caixa branca como os testes de unidade geralmente utilizam cerca de 60% do esforço e custo dos testes em um sistema, por outro lado, pode garantir um aumento de cerca de 40% na qualidade do *software* desenvolvido. Devido ao alto custo gerado na elaboração dos testes de unidade, é importante a utilização de algumas ferramentas e metodologias para o seu desenvolvimento e aplicação. Com o objetivo de melhorar a produtividade dos desenvolvedores de testes de unidade foram criados *frameworks* que auxiliam no desenvolvimento e no controle dos testes.

3. Metodologia

A metodologia utilizada para implementar corretores automáticos de listas de exercícios proposta neste trabalho consiste em quatro etapas. Primeiramente, seleciona-se um conjunto qualquer de exercícios relacionados a um tópico dentro do programa da disciplina. Em seguida, cabe ao professor da disciplina especificar qual a interface de cada função que deve ser implementa pelos alunos, especificando os seus parâmetros de entrada e saída. O próximo passo consiste em programar um ou mais testes de unidade para testar cada propriedade que deve ser atendida por cada função. Por fim, é solicitado ao aluno que ele implemente as funções das listas de exercício da forma como especificado no enunciado da questão. Como cada teste de unidade verifica uma única propriedade da função, quando esta falha, é possível exibir para o aluno uma mensagem de erro precisa sobre o que está errado em sua implementação.

Para facilitar a implementação dos testes de unidade foi utilizado o *framework* para desenvolvimento de testes de unidade conhecido como *Google Test*. *Frameworks* são bibliotecas de *software* que auxiliam na implementação de sistemas que tem naturezas semelhantes. Dentre os principais *frameworks* utilizados para a codificação de testes de unidade estão o *JUnit* [JUNIT 2010, Hamill 2004] para programação em *Java* e o *Google Test* [Google Test 2010] para programação em *C++*. O *Google Test* é um *framework* que pode ser compilado e executado em várias plataformas como *Windows, Linux, Mac OS X*, entre outras. Ele é desenvolvido como um projeto de software aberto e distribuído gratuitamente sob os termos da licença *New BSD License* [BSD 2010]. Esta biblioteca dispões de várias, funções, classes e macros em *C++* que permitem a implementação rápida e fácil de teste de unidade. Dentre estas ferramentas, a mais utilizadas são co-

nhecidas como *assertions*, que verificam uma determinada condição da função testada. Quando esta condição não é satisfeita, é gerada automaticamente uma mensagem de erro que pode ser personalizada para cada tipo de erro gerado, informando, por exemplo, os dados de entrada e a saída esperada.

A Figura 1 apresenta o gabarito do exercício que consiste em implementar uma função que recebe um vetor `v` com `n` elementos e retorna a média dos elementos de `v`. Na Linha 2, a variável auxiliar `soma` é inicializada. No laço das linhas 3-5, o somatório dos elementos de `v` é acumulado na variável `soma`. Por fim, a média dos elementos de `v` é retornada na Linha 6. Um erro muito comum neste exercício ocorre quando os alunos escrevem `i <= n`, na Linha 3, em vez da forma correta `i < n`. Uma vez que os vetores em `C++` são indexados de 0 a $n - 1$, isto resulta no acesso ilegal a posição de memória `v[n]`.

Este erro é verificado pelo teste de unidade apresentado na Figura 2. Nas linhas 2 e 3, é criado um vetor `v` com $n = 3$ elementos que será utilizado no teste. Entretanto, para verificar se o aluno não está acessando a posição `v[n]`, um elemento a mais é alocado para este vetor. O valor retornado pela função implementada pelo aluno é armazenado na variável `atual` na Linha 4, enquanto o valor que a função retornaria se estivesse errada é calculado e armazenado na variável `inesperado` na Linha 5. Caso os valores das duas variáveis sejam iguais, uma mensagem de erro é mostrada ao aluno indicando que sua função pode estar acessando a posição inválida `v[n]`. A Figura 3 mostra a mensagem de erro que é exibida ao aluno quando o teste de unidade falha. Note que este teste de unidade testa exclusivamente este tipo de erro. Outros testes de unidade devem ser implementados para testar outras propriedades que devem ser satisfeitas pela função que calcula a média de um vetor. Note também que os detalhes sobre a teoria e implementação de testes de unidade não são apresentados aos alunos, uma vez que esta teoria está além do escopo do curso de ensino de programação de computadores e exige conhecimentos ainda não ensinados para os alunos.

Cada lista de exercícios é disponibilizada na forma de um pacote de arquivos que é composto de (i) um arquivo com o enunciado das questões, (ii) um arquivo com as interfaces das funções que devem ser implementadas pelos alunos (e.g. vetores.h), (iii) um arquivo que contém um ou mais testes para cada função da lista (e.g. vetores_test.h), além da (iv) biblioteca do *Google Test*, necessária para compilar os teste de unidade. Cabe ao aluno escrever o arquivo que contém a implementação de todas as funções referentes a lista de exercícios (e.g. vetores.cc). Aos alunos são dadas duas opções de compilação. No modo `Main`, o programa do aluno é executado normalmente. Já no modo `EasyTesting` o programa do aluno não é executado. Em vez disso, todos os testes de unidade são executados e um relatório é exibido para o aluno com a lista das funções que passaram ou não passaram nos testes (Figura 3). Para cada teste que falha, uma mensagem de erro é exibida acompanhada por uma explicação que pode ajudar o aluno a solucionar erro.

4. Resultados e discussões

Um estudo de caso foi realizado em turmas de Algoritmos e Estruturas de Dados II da UFMG em 2010 e 2011. Estas turmas continham até 70 alunos matriculados. Foram desenvolvidas listas de exercício com os respectivos testes de unidade para os seguintes

```
1  float Media(float v[], int n) {
2    float soma = 0;
3    for (int i = 0 ; i < n ; i++) {
4      soma += v[i];
5    }
6    return soma / n;
7  }
```

Figura 1. Uma possível solução para a função que calcula a média de um vetor `v` com `n` elementos.

```
1  TEST_F(Teste, Testa_acesso_a_uma_posicao_invalida_do_vetor_na_funcao_Media) {
2    int n = 3;
3    float v[3 + 1] = {4.0, 8.0, 14.0, 1000.0};
4    float atual = Media(v, n);
5    float inesperado = (4.0 + 8.0 + 14.0 + 1000.0) / n;
6    ASSERT_NE(atual, inesperado)
7      << "-------------------------------------------------------------------\n"
8      << "Erro na funcao \"float Media(float v[], int n)\".                  \n"
9      << "-------------------------------------------------------------------\n"
10     << "  Sua função pode estar acessando a posição inválida v[n].         \n"
11     << "-------------------------------------------------------------------\n";
12 }
```

Figura 2. Teste de unidade que verifica se a função que calcula a média de um vetor não acessa um posição inválida do vetor.

```
vetores
[==========] Executando 5 testes 1 test case.
[----------] Iniciando testes.
[EXECUTANDO] Testa_acesso_a_uma_posicao_invalida_do_vetor_na_funcao_Media

-------------------------------------------------------------------
Erro na funcao "float Media(float v[], int n)".
-------------------------------------------------------------------
  Sua função pode estar acessando a posição inválida v[n].
-------------------------------------------------------------------

[  FALHOU  ]
[EXECUTANDO] Testa_funcao_Media_de_um_vetor_unitario
[    OK    ]
[EXECUTANDO] Testa_funcao_Media_com_mais_de_um_valor
[    OK    ]
[EXECUTANDO] Testa_funcao_Media_com_numeros_nulos
[    OK    ]
[EXECUTANDO] Testa_funcao_Media_numeros_positivos_e_negativos
[    OK    ]
[----------] Finalizando testes.
[==========] Foram executados 5 testes 1 test case.
[ PASSARAM ] 4 testes.
[ FALHARAM ] 1 teste. Veja a baixo a lista de testes que falharam:
             Testa_acesso_a_uma_posicao_invalida_do_vetor_na_funcao_Media
```

Figura 3. Mensagem de erro exibida quando o teste de unidade apresentado na Figura 2 falha.

tópicos referentes a disciplina: (i) procedimentos e funções, (ii) instruções condicionais e operadores lógicos, (iii) comandos de repetição, (iv) vetores, (v) matrizes, (vi) registros, (vii) recursão, (viii) listas encadeadas, (ix) pilhas, (x) filas, (xi) árvores binárias de busca, (xii) tabelas de dispersão, entre outros. Estas listas de exercício e os respectivos testes de unidade estão disponíveis para a comunidade acadêmica através do projeto de código aberto EasyTesting (`http://code.google.com/p/easytesting`).

As listas de exercício foram disponibilizadas semanalmente após cada tópico da matéria ser abordado em sala de aula. Nenhuma informação sobre como os testes de unidade foram implementados foi apresentada aos alunos. Foi informado apenas como executar os testes e consultar as mensagens de erro. Após a aplicação das listas de exercícios observou-se que o uso de testes de unidade foi um agente motivador para o aprendizado dos alunos, que tiveram menos dificuldade na resolução das listas de exercícios, pois tiveram um retorno imediato sobre a corretude dos seus programas ou possíveis pontos de falha em suas implementações. Isto permitiu que os alunos pudessem solucionar os erros em suas questões e avançar para os próximos tópicos sem a necessidade de aguardar instruções do professor.

Em aulas de laboratório, a correção automática dos exercícios possibilitou que o professor da disciplina pudesse dar uma atenção adequada para uma turma com grande número de alunos, uma vez que os erros mais fáceis e mais comuns puderam ser solucionados pelos alunos apenas com as mensagens de erro exibidas pelos testes de unidade. Isto permitiu ao professor concentrar seu tempo em auxiliar os alunos apenas na solução dos problemas mais complexos. Já fora da sala de aula, a correção automática dos exercícios permitiu que o professor gastasse menos tempo na correção de listas de exercícios e investisse este tempo na preparação das aulas e em outras atividades acadêmicas. Além disso, não foi necessária a compra nem a manutenção de um servidor de testes, uma vez que os testes de unidade são compilados e executados junto com o código do aluno.

Dentre os pontos negativos observados neste estudo de caso pode-se citar: (i) a metodologia não permite o teste da função principal de cada programa (*main()* em C++). Para a verificação desta função pode-se fazer uso complementar de sistemas como os juízes *online*. Além disso, (ii) durante a implementação dos testes de unidade pode ocorrer que determinados casos de teste não sejam codificados, dando a impressão ao aluno que sua codificação incorreta esteja correta. Este é um problema inerente a todos os sistemas de teste de *software* que neste caso pode ser contornado confrontando o aluno com o gabarito da questão após o encerramento da atividade. Ao surgir uma dúvida por parte do aluno, a divergência pode ser esclarecida e um novo teste de unidade que trata este caso pode ser adicionado. Por fim, uma outra desvantagem desta abordagem, é que as interfaces das classes e funções a serem implementadas são previamente definidas no enunciado do exercício e não podem ser alteradas pelos alunos. Isto não é um problema em cursos básicos de programação e podem ser contornados em cursos avançados com a adoção de trabalhos práticos complementares.

5. Considerações Finais

Neste trabalho foi proposta uma nova metodologia para implementação de sistemas de correção automática de listas de exercícios de programação baseada em testes de unidade. O *framework Google Test* foi utilizado na construção de testes de unidade para listas

de exercícios sobre diversos temas relacionados ao ensino de programação em cursos de graduação em computação e em engenharia. Através da aplicação deste método em turmas da disciplina Algoritmos e Estruturas de Dados da UFMG, observou-se que o método auxilia no ensino de linguagens de programação, principalmente para turmas com um grande número de alunos. Isto reduziu consideravelmente o esforço do professor da disciplina na correção de exercícios e o tempo gasto por ele tirando dúvidas dos alunos nas aulas de laboratório. O uso dos testes de unidade permitiu que os alunos rapidamente identificassem e corrigissem os erros de seu código sem a ajuda do professor da disciplina.

Trabalhos futuros podem estudar a utilização em conjunto da abordagem proposta neste trabalho com outras baseadas em sistemas de juízes *Online* e com ferramentas de verificação de cópias de trabalho para uma avaliação completa de listas de exercícios.

Referências

BSD (2010). New bsd license. http://www.opensource.org/licenses/bsd-license.php (referência on-line).

Cheang, B., Kurnia, A., and Oon, A. L. W. (2003). On automated grading of programming assignments in an academic institution. *Computers and Education*, 41:121–131.

Dustin, E., Rashka, J., and Paul, J. (1999). *Automated Software Testing - Introduction, Management and Performance*. Addison-Wesley, Reading.

Google Test (2010). http://code.google.com/p/googletest/w/list (referência on-line).

Hamill, P. (2004). *Unit Test Frameworks - Tools for High-Quality Software Development*. O'Reilly Media, Reading.

ICPC (2010). Acm-icpc. http://cm2prod.baylor.edu/welcome.icpc (referência on-line).

JUNIT (2010). http://www.junit.org/ (referência on-line).

Kurnia, A., Lim, A., and Oon, W.-C. (2001). Online Judge. *Computers and Education*, 36:299–315.

Kurnia, A., Malafiejsk, M., and Noinsk, T. (2008). Application of an Online Judge and Contester System in Academic Tuition. *Advances in Web Based Learning in ICWL 2007*, 4823-2008:343–354.

Misra, S. (2003). Evaluating four white-box test coverage methodologies. In *Evaluating four white-box test coverage methodologies*, volume 3, pages 1739 – 1742 vol.3.

MOODLE (2010). http://moodle.org/ (referência on-line).

Schleimer, S., Wilkerson, D. S., and Aiken, A. (2003). Winnowing: local algorithms for document fingerprinting. In *SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 76–85, New York, NY, USA. ACM.

Schleimer, S., Wilkerson, D. S., and Aiken, A. (2010). Moss plugin. http://code.google.com/p/sunner-projects/w/list (referência on-line).

Woo, Y.-H., Hong, S.-W., and Kim, S.-B. (2008). A study on the self-directed learning management system. In *NCM '08: Proceedings of the 2008 Fourth International Conference on Networked Computing and Advanced Information Management*, pages 119–122, Washington, DC, USA. IEEE Computer Society.

Uma Proposta para Auxiliar Alunos e Professores no Ensino de Programação: O Ambiente AIIP

Cledson Calaça Cavalcante Gomes[1], David Henrique de Souza Lima[2], Raphael Pereira Ribeiro[2], Eliana Silva de Almeida[2], Patrick Henrique da Silva Brito[2]

[1]Mestrado em Modelagem Computacional do Conhecimento
Universidade Federal de Alagoas (UFAL)
CEP: 57072-970 – Maceió – Alagoas – Brasil

[2]Instituto de Computação – Universidade Federal de Alagoas (UFAL)
CEP: 57072-970 – Maceió – Alagoas – Brasil

{cledsoncalaca, dhs.lima, raphaelpr01}@gmail.com

{eliana.almeida, patrickhenrique}@gmail.com

Abstract. *This paper presents AIIP, an intelligent environment for beginners in programming. AIIP is an environment that aims to help students in their programming learning process. AIIP unites in a single environment for teaching programming desired features by the community, such as a practical teaching module, a theoretical teaching module and an assistent able to act according to user behavior, providing appropriated hints and feedbacks and performs calculations in accordance with pre-established metrics to evaluate the student's solutions. Thus, it is expected that with the combination of these features in one environment, the students could develop skills and reduce their difficults in learning programming.*

Resumo. *Este artigo apresenta o AIIP (Ambiente Inteligente para Iniciantes em Programação). O AIIP é um ambiente que tem o objetivo de auxiliar os alunos em seu processo de aprendizagem de programação. O AIIP une em um só ambiente funcionalidades para o ensino de programação desejáveis pela comunidade da área, tais como: um módulo prático de ensino, um módulo teórico e um assistente capaz de agir de acordo com o comportamento do usuário, fornecendo dicas e feedbacks apropriados, além de realizar cálculos, de acordo com métricas pré-estabelecidas, para avaliar as soluções dos alunos. Desta forma, espera-se que com a junção destas funcionalidades em um único ambiente, o aluno possa desenvolver habilidades e diminuir suas dificuldades de aprendizado de programação.*

1. Introdução

Parece ser consenso entre especialistas da computação que a programação de computadores é uma das disciplinas mais importantes nessa área. Porém, para a maioria dos alunos iniciantes, seu aprendizado e o desenvolvimento de habilidades é um processo complexo, exigente e cercado de dificuldades que podem ser diagnosticadas não somente pelo alto grau de repetência nas disciplinas introdutórias, mas também por deficiências

demonstradas pelos mesmos nas disciplinas mais avançadas e que exigem o pré-requisito de programação [Nobre and Menezes 2002].

Segundo [Gómez-Albarrán 2005] são fatores que afetam negativamente o processo de aprendizagem de programação: a complexidade dos conceitos e estruturas de programação, as dificuldades atribuídas à sintaxe e à semântica das linguagens de programação, bem como a complexidade dos ambientes de programação.

Cientes das dificuldades neste processo, o que realmente podemos observar nos últimos anos é a discussão acerca da utilização de novas metodologias de ensino com objetivo de melhorar o processo de ensino/aprendizagem de programação. Esta discussão tem sido foco de pesquisas no Brasil e no mundo há alguns anos, onde diversas ferramentas e metodologias são propostas tendo como objetivo amenizar tais dificuldades e tornar o processo de ensino e aprendizagem de programação de computadores menos complexo.

Pode-se comprovar o interesse da comunidade no tema no trabalho de [Júnior and Rapkiewicz 2006], no qual verificamos que pelo menos um artigo por ano sobre o ensino de programação é publicado tanto no SBIE como no WEI, eventos ligados à Sociedade Brasileira de Computação. Os autores analisaram vários artigos e concluíram que existem três vertentes na busca por soluções para as dificuldades vivenciadas por professores e alunos neste processo: metodologias, ferramentas computacionais e a integração de ambas. Segundo os autores, apesar de terem encontrado evidências de que a união de ferramentas computacionais e metodologias adequadas culminam em melhores resultados, o que se observa na maioria dos trabalhos é uma tendência em tratá-las separadamente.

Com o intuito de ampliar as discussões sobre o tema, este artigo apresenta um ambiente para o ensino de programação apoiado por um assistente inteligente como proposta para auxiliar tanto alunos como professores com relação ao ensino de programação. Vale ressaltar aqui, que o ambiente não se encontra totalmente implementado, ou seja, alguns componentes citados no decorrer do artigo ainda estão em fase de implementação. Sendo assim, não descrevemos detalhadamente o funcionamento desses componentes por não possuírmos, ainda, conclusões relevantes para discussão, porém, as idéias são expostas para dar ao leitor uma noção de como pretendemos que tais componentes se comportem dentro do ambiente.

O artigo está estruturado da seguinte maneira: a Seção 2 apresenta alguns conceitos que servem como base para o entendimento posterior do artigo e do ambiente apresentado. A Seção 3 apresenta alguns trabalhos relacionados ao ensino e aprendizado de programação. A Seção 4 apresenta o AIIP, incluindo uma visão geral da sua arquitetura, o assistente inteligente e a forma de avaliação dos alunos. Finalmente, a Seção 5 apresenta algumas considerações finais e trabalhos futuros.

2. Embasamento Teórico

Os conceitos apresentados nessa seção dizem respeito à alguns aspectos teóricos relacionados com o tema do artigo, bem como com o desenvolvimento do ambiente a ser apresentado. Através da breve apresentação de tais conceitos será possível um melhor entendimento por parte dos leitores, no que se refere às formas e metodologias de projeto e implementação do ambiente, levando em consideração fatores educacionais relacionados com o ensino de programação.

2.1. Metodologias de Ensino de Programação

Quando falamos sobre o ensino de programação, alguns cuidados básicos por parte dos professores são necessários em diversos fatores, que vão desde o entendimento do problema abordado, do paradigma selecionado, passando pela linguagem escolhida e chegando às metodologias de ensino utilizadas.

Alguns estudos apresentam diferentes soluções metodológicas na tentativa de melhorar o processo de ensino de programação, dentre as quais podemos citar:

Atividades de programação em laboratório: Atividades em laboratório fazem parte de um curso típico de introdução à programação [Robins et al. 2003]. Esta metodologia possui características pedagógicas úteis, uma vez que os alunos podem trabalhar e aprender por conta própria e no seu próprio ritmo, mas não supre todas as necessidades dos alunos, visto que aspectos teóricos da solução do problema muitas vezes não são contemplados, focando apenas a codificação dos algoritmos.

Programação em par: Na programação em par [Han et al. 2010], dois programadores trabalham juntos em um computador no mesmo projeto, algoritmo, código ou teste. É uma metodologia de aprendizagem colaborativa, uma vez que as duas pessoas envolvidas estão trabalhando em conjunto para um objetivo comum [Han et al. 2010]. Esta metodologia é bastante interessante, pois os alunos podem inverter os papeis no processo de aprendizagem, alternando entre codificar e observar a codificação do seu companheiro. Ao mesmo tempo um fator negativo de tal metodologia é que o desempenho do par pode ser afetado de acordo com a discrepância de conhecimento entre os integrantes do par.

Utilização de Tutores de Programação Inteligentes: Tal metodologia também é um meio eficaz e viável para auxiliar programadores iniciantes a superar as dificuldades de aprendizagem. No entanto, sua utilização generalizada tem sido inibida pelos elevados custos e pela complexidade de desenvolvimento associados com a construção de tais sistemas [Pillay 2003].

É importante deixar claro a existência de outras metodologias de ensino de programação, e que, além disso, tais metodologias podem ser combinadas para um uso em conjunto.

2.2. Sistemas Tutores Inteligentes

Técnicas de inteligência artificial aplicadas à ambientes educacionais são geralmente representadas por Sistemas Tutores Inteligentes (`STI`) [Wenger 1987]. Tais sistemas fazem parte de uma classe de programas que possuem certo grau de autonomia de seus módulos e componentes, onde suas funcionalidades variam desde auxiliar o aluno na resolução de um problema até a personalização da assistência provida a um aluno em um determinado momento de sua interação com o ambiente.

No entanto, nem todos os ambientes educacionais são implementados utilizando alguma técnica de inteligência artificial, e mesmo aqueles que se utilizam de alguma técnica de inteligência artificial não necessariamente se enquadram como `STI's` completos, uma vez que a implementação destes sistemas, independente do tamanho do projeto, demandam um grande esforço e um bom tempo, muitas vezes não atingindo os resultados esperados pela comunidade acadêmica da área. Desta forma, a implementação de um

assistente inteligente, menos complexo do que um `STI`, mas não menos eficaz, pode simplificar o projeto e implementação do ambiente atingindo resultados semelhantes àqueles esperados.

Estamos desenvolvendo e implementando aos poucos para o AIIP, um assistente inteligente capaz de tomar ações de acordo com o comportamento e o estado do usuário no momento de interação com o ambiente. Explicaremos melhor sobre este componente do AIIP na Seção 4.2.

2.3. Arquitetura de Software

A arquitetura de *software*, através de um alto nível de abstração, define o sistema em termos de componentes, a interação entre eles e os atributos e funcionalidades de cada um [Sommerville 2001]. Além disso, a arquitetura representa essa interação de forma explícita, materializando-a através dos conectores. Por conhecerem o fluxo interativo entre os componentes do sistema, essas entidades são capazes, entre outras coisas, de estabelecer protocolos de comunicação e de coordenar a execução dos serviços que envolvam mais de um componente do sistema.

Essa visão estrutural do sistema em um alto nível de abstração proporciona benefícios importantes, que são imprescindíveis para o desenvolvimento de sistemas de *software* complexos. Os principais benefícios são:

i A organização do sistema como uma composição de componentes lógicos;
ii A antecipação da definição das estruturas de controle globais;
iii A definição da forma de comunicação e composição dos elementos do projeto;
iv O auxílio na definição das funcionalidade de cada componente projetado.

Além disso, uma propriedade arquitetural representa uma decisão de projeto relacionada a algum requisito não-funcional do sistema, que quantifica determinados aspectos do seu comportamento, como confiabilidade, reusabilidade e modificabilidade [Sommerville 2001]. Esses conceitos são, de certa forma, melhor entendidos quando na seção 4.1 apresentamos a arquitetura do AIIP e colocamos em prática alguns destes conceitos aqui apresentados, bem como a explicação pela escolha de um determinado padrão de arquitetura e seus benefícios em nosso ambiente.

3. Trabalhos Correlatos

Os ambientes de programação são uma fonte dos obstáculos para os estudantes. Ambientes comerciais são normalmente projetados para serem usados por programadores profissionais. Eles geralmente não fornecem a simplicidade que os estudantes iniciantes precisam, pois normalmente usam apenas um conjunto reduzido das facilidades que eles provêm. [Gómez-Albarrán 2005]

Para enfrentar essas dificuldades, algumas pesquisas têm se concentrado no desenvolvimento de ferramentas com ambientes de desenvolvimento muito simples, que os alunos utilizam facilmente para escrever, compilar, executar e depurar os seus códigos. Essas ferramentas também costumam incluir algum tipo de maior capacidade de diagnóstico. [Gómez-Albarrán 2005]

A análise feita por [Gómez-Albarrán 2005] sintetiza diferentes direções e abordagens para a melhoria do processo de ensino/aprendizagem de programação. O autor

classifica as ferramentas apresentadas em seu trabalho em quatro categorias: (1) Ferramentas que incluem um simples e reduzido ambiente de desenvolvimento; (2) Ambientes baseados em exemplos; (3) Ferramentas baseadas em visualização e animação; e (4) Ambientes de simulação.

Observando os trabalhos citados por [Gómez-Albarrán 2005], e suas consequentes ferramentas e ambientes resultante de suas pesquisas, entendemos que a proposta apresentada neste artigo é bastante interessante por se tratar de uma ferramenta que se enquadra tanto na categoria (2), que incentiva a resolução de problemas de programação, como também por possuir funcionalidades características das categorias (1) e (3), dando mais flexibilidade ao processo de desenvolvimento das habilidades de programação dos alunos.

A proposta apresentada neste artigo se diferencia das muitas apresentadas no trabalho de [Gómez-Albarrán 2005], primeiramente, por usar o português como língua padrão do ambiente, liberando os alunos de qualquer dificuldade na interpretação de mensagens ou avisos. Além disso, utilizamos uma linguagem própria, próxima da linguagem Pascal para o desenvolvimento dos algoritmos, diminuindo a complexidade de aprendizado por parte dos alunos quando comparada ao uso de linguagens como LISP, C e Java, utilizadas em vários trabalhos citados por [Gómez-Albarrán 2005].

Outra contribuição importante da nossa proposta é a flexibilidade tanto na forma de avaliação dos estudantes, quanto nos conteúdos e problemas armazenados no ambiente, que podem sofrer alterações pelo professor a depender de aspectos como: o nível de conhecimento da turma, os conceitos estudados em um dado momento, dentre outros.

4. AIIP: Ambiente Inteligente para Iniciantes em Programação

O AIIP é um ambiente que vêm sendo desenvolvido com o objetivo de auxiliar o processo de ensino/aprendizado de programação. O AIIP tem parte de sua arquitetura (Seção 4.1) baseada no AMBAP [Almeida et al. 2002], uma vez que alguns componentes, como o editor e o interpretador foram adaptados deste ambiente.

Diferentemente do AMBAP, o AIIP possui mais funcionalidades educacionais, por disponibilizar além de um ambiente prático de ensino, a opção de aprendizado teórico onde o aluno pode aprender através de conceitos e exemplos sobre programação, dando também ao aluno a possibilidade de resolver os problemas armazenados em sua base de dados.

O AIIP se utiliza ainda de outros objetos educacionais de aprendizagem no intuito de ajudar o aluno em seu aprendizado, como, por exemplo, a exibição de dicas durante a resolução dos problemas, apresentação de *feedback* a cada problema resolvido, além de um sistema de estatísticas apresentadas ao usuário a cada unidade de conhecimento finalizada, com informações desde o número de exercícios resolvidos dentro da unidade, até uma média final calculada de acordo com parâmetros pré-estabelecidos do sistema, explicados na Seção 4.3.

Além de todas essas funcionalidades, o AIIP possui um assistente inteligente, explicado melhor na Seção 4.2, que acompanha o aluno em toda sua interação com o ambiente, monitorando-o e quando necessário interrompendo-o para informá-lo sobre algum aspecto relevante em seu aprendizado.

4.1. Arquitetura Proposta

A arquitetura proposta para o ambiente está estruturada através de quatro módulos principais, de acordo com o estilo arquitetural em camadas [Shaw and Garlan 1996]. Como citado anteriormente, alguns componentes foram adaptados do AMBAP, como o seu Interpretador e seu Editor, neste caso equivalente ao componente IHC do AIIP. Como pode ser observado detalhadamente na Figura 1, o usuário interage com o sistema através da camada de `Visão`. Essa camada, por sua vez, requisita as funcionalidades da camada de `Sistema`, que é responsável por implementar os casos de uso especificados. Para atender os requisitos especificados, essa camada requisita os serviços especializados da camada de `Negócio`, que contém serviços básicos relativos aos ambientes virtuais de ensino de programação. Finalmente, os dados são armazenados e acessados através da camada de `Dados`.

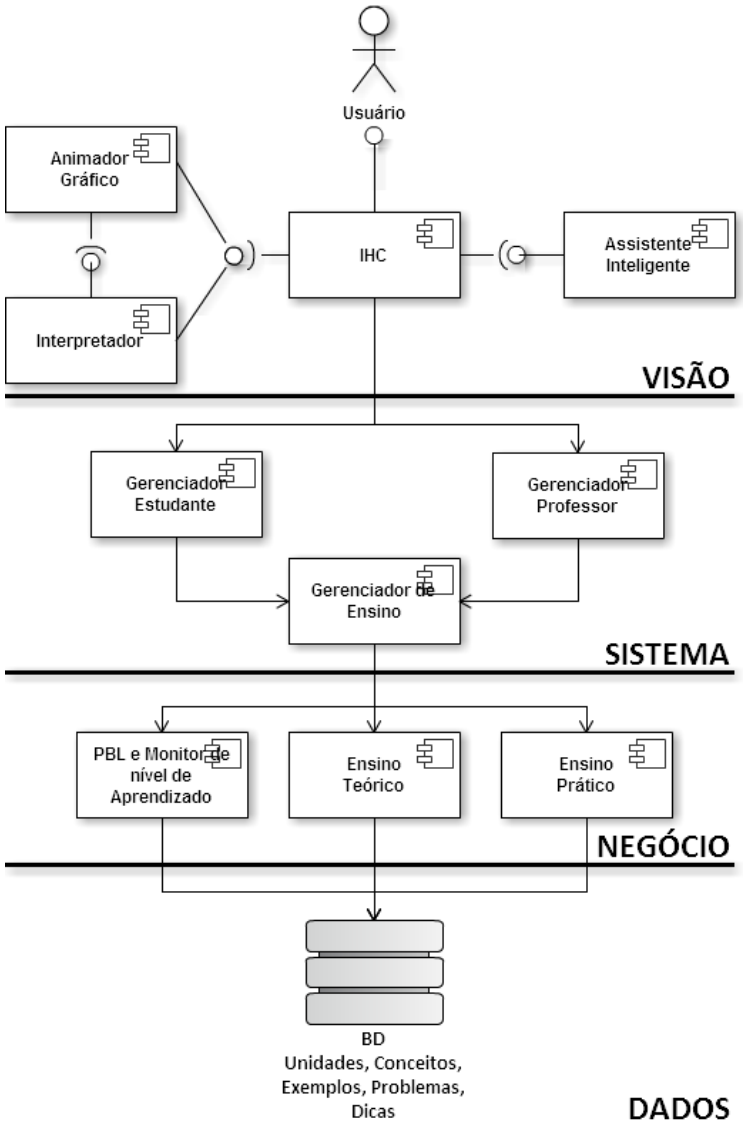


Figura 1. Arquitetura Proposta para o Ambiente

Cada umas dessas camadas está estruturada internamente de acordo com o estilo arquitetural de componentes independentes [Shaw and Garlan 1996], possuindo assim

seus respectivos componentes internos. Esses componentes podem se conectar tanto a outros componentes da mesma camada, quanto a componentes de camadas adjacentes à sua, construindo desta forma a aplicação principal.

Camada de Visão: A camada de `Visão` é responsável por interagir diretamente com os usuários, além de apresentar os diversos tipos de conteúdo que devem ser disponibilizados.

Camada do Sistema: A camada de `Sistema` é responsável pelo gerenciamento das informações dos usuários e dos conteúdos acessados e modificados pelos mesmos. Caso o usuário seja um aluno, ele poderá utilizar o sistema nos modos prático e teórico, tendo suas ações monitoradas pelo sistema. No caso do professor, além dos modos prático e teórico, ele ainda dispõe de um componete para gerenciar todo o conteúdo do sistema e os métodos de avaliação dos níveis de aprendizado dos estudantes.

Camada de Negócio: A camada de `Negócio` tem o objetivo de prover os modos de ensino aos quais o estudante é apresentado na camada de `Visão`. Além disso, esta camada possui um componente responsável por gerenciar a aprendizagem através da resolução de problemas e pelos cálculos e avaliações dos estudantes, de acordo com métricas pré-estabelecidas.

Camada de Dados: A camada mais baixa da arquitetura proposta é a camada de `Dados`, responsável pelo gerenciamento de todo o banco de dados do sistema.

4.2. O Assistente Inteligente

O assistente inteligente do AIIP tem o objetivo de auxiliar os alunos na resolução dos problemas do ambiente, bem como, quando eles estiverem desenvolvendo seus algoritmos, independente de estarem resolvendo um problema específico. Podemos então entender como objetivo principal desse assistente a observação das ações realizadas pelos alunos na resolução de problemas e construção de algoritmos e suas consequentes reações da maneira mais apropriada àquele aluno naquele determinado momento. A partir deste objetivo, suas principais funções são: o fornecimento de dicas e refinamentos das soluções, bem como *feedbacks* apropriados aos alunos, não somente sob demanda, mas também ao perceber a necessidade do aluno, a identificação de erros cometidos pelos alunos bem como os cálculos para avaliação dos algoritmos dos mesmoss.

É importante ressaltar que o assistente do AIIP não se trata de um Sistema Tutor Inteligente - STI [Wenger 1987], visto que não possui todas as características essenciais comuns à estes sistemas, como por exemplo, um módulo de estratégias pedagógicas bem definidas. Porém, a idéia de utilização de um assistente inteligente mais simples ao invés de um STI completo é bastante válida e interessante, uma vez que a construção de STI's em domínios de programação é uma tarefa complexa e que há bastante tempo vem sendo estudada, ainda sem os resultados significativos desejados pela comunidade.

Vale deixar claro aqui que o assistente inteligente do AIIP ainda encontra-se em fase de implementação e testes, ou seja, ainda pode sofrer alterações e melhorias nesta etapa. Consideramos nosso assistente como "inteligente" pelo fato do mesmo ter a capacidade de tomar ações de acordo com regras de produção pré-estabelecidas armazenadas no ambiente, tais como em sistemas especialistas, onde as condições das regras dependem do estado do usuário e as ações são as assistências oferecidas pelo assistente.

Quando o aluno está trabalhando no modo teórico, o assistente não reage às suas ações, porém todo seu progresso com relação aos conteúdos é monitorado pelo ambiente para que posteriormente, na hora de se trabalhar no modo prático, tais informações sirvam como base para que o assistente realize inferências sobre o aprendizado do aluno. Uma vez que o aluno começa a trabalhar no modo prático com objetivo de resolver problemas específicos armazenados no ambiente, o assistente começa realmente a atuar. A ativação do assistente é realizada automaticamente de quatro formas complementares: (1) após cada ação de pressionar a tecla *Enter*, (2) após cada ação de salvar o código, (3) após cada ação de compilar o programa e (4) após uma solicitação específica do aluno.

4.3. Avaliação da Solução dos Problemas dos Alunos

Para avaliar e atribuir uma nota à solução do aluno para os problemas propostos, quatro fatores são analisados pelo assistente inteligente: ***Algoritmo***, ***Dicas***, ***Refinamentos*** e ***Erros***.

Para a atribuição da nota do fator ***Algoritmo***, algumas métricas são levadas em consideração e ponderadas de acordo com sua importância para o código, com relação aos alunos iniciantes em programação, são elas: *Linhas de Código*, *Variáveis*, *Instruções*, *Funções*, *Blocos de Condição* e *Blocos de Repetição*. Cada uma dessas métricas possui um intervalo de aceitação que indica até que ponto a solução do aluno, com relação a cada métrica, está correta. Para um fácil entendimento, no caso da métrica *Linhas de Código* de um problema, onde neste a solução ideal para sua resolução seriam 20 linhas e a solução apresentada pelo aluno possui 30 linhas, a nota desta métrica seria calculada baseada em um índice de aceitação, que neste caso pode ser definida pelo professor de acordo com a especificidade do problema. Ao final de tudo, o assistente realiza um cálculo e atribui uma nota para a solução do aluno de acordo com cada uma dessas métricas, que serão ponderadas com seu peso, para então ser atribuída a nota final do aluno para o fator algoritmo.

Para o fator ***Dicas***, o assistente atribui uma nota levando em consideração os itens: nível de dificuldade do problema, quantidade de dicas da unidade e a quantidade de dicas solicitadas.

Com relação ao fator ***Refinamentos***, que são na verdade os passos necessários para se chegar à resolução de cada problema, o assistente realiza o cálculo da nota de forma semelhante à do fator dicas, sendo que os itens analisados são: nível de dificuldade do problema, quantidade de refinamentos do problema e a quantidade de refinamentos solicitados.

Por fim, o último fator levado em consideração são os ***Erros***. Os erros armazenados são reconhecidos pelo assistente e então, de acordo com o nível de dificuldade do problema e da quantidade de erros cometidos durante aquele problema é realizado um calculo e atribuída uma nota para esse fator.

Ao final da resolução de cada problema é calculada a nota final para o mesmo, de acordo com o peso de cada um dos fatores citados anteriormente, onde o somatório dos pesos dos fatores é igual a 1. Na Tabela 1 é apresentado um exemplo de nota atribuída a um problema solucionado pelo aluno, de acordo com a nota obtida em cada fator e seu peso para o cálculo da nota final.

Tabela 1. Exemplo de nota atribuída ao aluno para um problema

Fator	Nota do Aluno	Peso do Fator	Nota do Fator
Algoritmo	7,00	0,40	2,80
Refinamentos	8,00	0,25	2,00
Dicas	9,00	0,20	1,80
Erros	10,00	0,15	1,50
Nota Final	-	-	**8,10**

É importante ressaltar que, tanto os fatores analisados para compor a nota geral do aluno para um problema, quanto as métricas utilizadas para cada fator, foram elaboradas de acordo com experiências próprias de ensino e da observação e análise de problemas e das dificuldades dos alunos em aulas da disciplina Laboratório de Programação em nossa instituição nos últimos 3 semestres, com aproximadamente 40 alunos por turma/semestre. Também é importante destacar que, de acordo com a preferência do professor, os fatores e as métricas podem ter seus pesos alterados para compor a nota dos alunos, levando em consideração a turma, o problema ou outros aspectos.

Já com a nota para o problema calculada, o aluno então pode trabalhar em outro problema objetivando atingir a pontuação mínima para avançar no conteúdo e estudar conceitos das unidades mais avançadas, e, consequentemente resolver os problemas destas unidades. O aluno deve atingir, no mínimo, 70% da pontuação máxima esperada pela resolução de todos os problemas da unidade para que consiga avançá-la.

Na Tabela 2 esta idéia é exemplificada, onde temos para cada unidade a quantidade de problemas associados a ela, separados por níveis e por pesos, onde o calculo da pontuação máxima de uma unidade é realizado pela soma do resultado da multiplicação entre a quantidade de problemas de cada nível, o peso dos problemas desse nível e 10 (a nota máxima para cada problema). Após a resolução de cada problema, a nota do aluno é armazenada e o assistente verifica se o aluno já tem condições de avançar para unidades posteriores e resolver problemas dessas unidades.

Tabela 2. Exemplo do cálculo da pontuação máxima de cada unidade

Unidade	Qnt Problemas	**Problemas Nível 1**	**Peso Nível 1**	**Problemas Nível 2**	**Peso Nível 2**	**Problemas Nível 3**	**Peso Nível 3**	**Pontuação Máxima Unidade**
1	20	10	3	5	3	5	4	650
2	12	5	3	4	3	3	4	390
3	15	6	3	5	3	4	4	490

5. Considerações Finais e Trabalhos Futuros

Este artigo apresentou o AIIP, um ambiente que vem sendo desenvolvido para auxiliar o ensino de programação, no qual um assistente inteligente interfere no processo de ensino de acordo com o estado e comportamento dos alunos. Algumas métricas foram definidas para que o índice de aprendizagem dos alunos com relação às unidades de conhecimento e aos problemas de programação em que os mesmos são submetidos pudessem ser feitos de forma automática pelo assistente, levando em consideração alguns aspectos de aprendizagem.

O ambiente pode auxiliar não só os alunos, mas também aos professores, uma vez que ao acompanhar a evolução, oferecer feedbacks e dicas e avaliar as soluções dos alunos automaticamente no momento da interação com os mesmos, o AIIP libera, parcialmente,

o professor para dedicar mais atenção e um acompanhamento maior para alunos com mais dificuldades na turma. Outro ponto importante é a flexibilidade que o professor possui para alterar os índices das métricas de avaliação dos alunos, dependendo do nível de conhecimento da turma ou da dificuldade de alguns problemas.

Desta forma, acreditamos que do modo que nosso ambiente foi estruturado, a junção de objetos de ensino Teórico e Prático, apoiados por um assistente inteligente, estimulando a resolução de problemas, formam um conjunto de funcionalidades lúdicas que atendem, até certo ponto, aos requisitos esperados pela comunidade da área no processo de ensino/aprendizagem de programação.

Pretendemos, tão logo quanto possível e após a implementação total de todos os componentes, publicar os resultados de avaliações realizadas junto aos alunos ingressantes no curso de Ciência da Computação da nossa instituição com respeito à satisfação e níveis de dificuldade e aprendizagem dos conceitos de programação com a utilização do ambiente, bem como relatar as melhorias dos componentes que ainda estão em fase de implementação para que o mesmo possa ser avaliado de forma mais completa.

Referências

Almeida, E. S., Costa, E. B., Silva, K. S., Paes, R. B., Almeida, A. A. M., and Braga, J. D. H. (2002). Ambap: um ambiente de apoio ao aprendizado de programação. In *WEI 2002: X Workshop Sobre Educação em Computação*, Florianópolis, SC, Brasil.

Gómez-Albarrán, M. (2005). The teaching and learning of programming: a survey of supporting software tools. *The Computer Journal*, 48(2):130–144.

Han, K., Lee, E., and Lee, Y. (2010). The impact of a peer-learning agent based on pair programming in a programming course. *IEEE Transactions on Education*, 53(2):318 –327.

Júnior, J. C. R. P. and Rapkiewicz, C. E. (2006). O processo de ensino-aprendizagem de fundamentos de programação: uma visão crítica da pesquisa no brasil. In *WEI 2006: XIII Workshop sobre Educação em Computação*, Campo Grande, MS, Brasil.

Nobre, I. A. M. and Menezes, C. S. (2002). Suporte à cooperação em um ambiente de aprendizagem para programação (samba). In *SBIE'2002: XIII Simpósio Brasileiro de Informática na Educação*, São Leopoldo-RS, Brasil.

Pillay, N. (2003). Developing intelligent programming tutors for novice programmers. *ACM SIGCSE Bulletin*, 35(2):78–82.

Robins, A., Rountree, J., and Rountree, N. (2003). Learning and teaching programming: a review and discussion. *Computer Science Education*, 13:137–172.

Shaw, M. and Garlan, D. (1996). *Software architecture: perspectives on an emerging discipline*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Sommerville, I. (2001). *Software Engineering*. Addison-Wesley, 6th edition.

Wenger, E. (1987). *Artificial intelligence and tutoring systems: computational and cognitive approaches to the communication of knowledge*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Comparação de Grades Curriculares de Cursos de Computação Baseada em Agrupamento Hierárquico de Textos

Nils Murrugarra-Llerena, Fernando Alva-Manchego, Solange Oliveira Rezende

[1]Instituto de Ciências Matemáticas e de Computação – ICMC
Universidade de São Paulo – USP
Caixa Postal: 668 – CEP: 13560-970 – São Carlos – SP

{nineil,falva,solange}@icmc.usp.br

Abstract. *Comparing curricula allows an evaluation of the quality of the courses and programs in a university; however, this comparison is often done manually. In this paper, it is proposed an approach based on hierarchical clustering of texts to help in the task of comparing curricula. A case study was done using curricula from computing undergraduate courses from Peruvian and Brazilian universities. The results show that the approach proposed allows to discover hidden relations between curricula.*

Resumo. *A comparação de grades curriculares permite identificar e avaliar a qualidade dos cursos e programas em uma universidade; contudo, essa comparação é quase sempre feita manualmente. Neste trabalho, propõe-se uma abordagem baseada em agrupamento hierárquico de textos para auxiliar na tarefa de comparação de grades curriculares. Um estudo de caso foi realizado utilizando grades curriculares de cursos de graduação em computação de universidades peruanas e brasileiras. Os resultados obtidos mostram que a abordagem proposta permite descobrir relações ocultas entre grades curriculares.*

1. Introdução

A análise de grades curriculares é um ponto importante que permite avaliar a qualidade dos cursos correspondentes. Por exemplo, [Biddle and Tempero 1996] fazem uma comparação da grade curricular do curso de computação da Universidade de *Wellington* com recomendações da ACM/IEEE, especificamente do núcleo básico. Já no contexto brasileiro, os trabalhos de [Pereira et al. 2010] e [Prietch and Pazeto 2010] examinam grades curriculares para fazer uma análise de oferta e propostas de padronização de matriz curricular. Embora os objetivos desses trabalhos sejam distintos, eles possuem uma característica comum: o processo de comparação é manual.

Prém, nem sempre pode-se contar com o tempo necessário para realizar este tipo de estudo comparativo de maneira completamente manual e, portanto, seria útil contar com um recurso computacional que auxile à tarefa. Nesse contexto, as técnicas de mineração de textos se apresentam úteis pois permitem encontrar automaticamente relações ocultas entre distintos documentos.

A mineração de textos (MT) pode ser definida como a aplicação de métodos computacionais e técnicas sobre dados textuais para encontrar informação intrínseca e relevante, assim como conhecimento previamente desconhecido [Do Prado and Ferneda 2007]. Existem numerosas aplicações da MT, que incluem

pesquisa pioneira em análise e classificação de notícias, *email* e filtro de *spam*, extração hierárquica de tópicos de paginas web, extração e gestão automática de ontologias e inteligência competitiva [Srivastava and Sahami 2009].

O agrupamento (*clustering*) de textos é uma das técnicas usadas para MT, que é útil quando se quer agrupar documentos similares e torna a navegação entre eles mais fácil para usuários finais [Ebecken et al. 2003]. Existem dois tipos de agrupamento: particional e hierárquico. Este último, de interesse neste artigo, permite obter uma hierarquia, que é uma estrutura mais informativa do que o conjunto não estruturado de grupos obtidos pelos métodos particionais. Uma vantagem do agrupamento hierárquico é que não precisa da especificação do número de grupos e a maioria dos algoritmos hierárquicos que têm sido usados em recuperação de informação são determinísticos [Manning et al. 2008].

Para auxiliar na tarefa de comparação e análise de grades curriculares, neste artigo propõe-se empregar uma abordagem baseada em métodos de agrupamento de documentos. Mais especificamente, foi escolhido o agrupamento hierárquico aglomerativo para permitir uma comparação em diferentes níveis de agrupamentos. Para testar o método, foi realizado um estudo de caso comparando as grades curriculares de 40 cursos de graduação de computação peruanos e 33 brasileiros. Foram gerados arquivos de textos contendo informações de cada grade curricular em português.

Os resultados obtidos demonstram que é possível encontrar relações ocultas entre as grades curriculares usando o método proposto. Por exemplo, no caso do Peru, mostram-se agrupamentos entre cursos que têm como foco alguma área específica da computação (ciência da computação e/ou sistemas da informação). No caso das grades curriculares do Brasil, verificou-se que os cursos que pertencem a uma mesma instituição possuem uma maior relação entre eles do que com cursos de outras universidades ou institutos. E no caso das relações entre os documentos do Peru e do Brasil, apresentam-se semelhanças entre programas de computação peruanos e brasileiros com a mesma orientação na área de computação.

Na Seção 2 são descritos alguns conceitos de agrupamento de textos. O corpus de documentos relacionados à grades curriculares de cursos de computação, processo de pré-processamento e a extração dos padrões são descritos na Seção 3. A Seção 4 apresenta a discussão dos resultados. Na Seção 5 são descritas algumas situações nas quais uma análise comparativa deste tipo poderia ajudar na toma de decisões. Por fim, na Seção 6 descrevem-se as conclusões e trabalhos futuros.

2. Agrupamento de Textos

A mineração de textos refere-se ao processo de descobrir conhecimento em grandes bases de dados textuais e pode ser considerada como uma especialização do processo de mineração de dados. Enquanto a mineração de dados trabalha com dados com estrutura definida, a mineração de textos trabalha com dados não estruturados.

O agrupamento de textos, uma das técnicas de mineração de textos, consiste na organização de um conjunto de documentos, baseada em uma medida de similaridade, na qual os documentos de um mesmo grupo são altamente similares entre si, mas dissimilares em relação aos documentos de outros grupos [Manning et al. 2008].

Dentre os métodos de agrupamento consideram-se os métodos hierárquicos, os

quais geram um conjunto de grupos aninhados que são organizados em uma árvore. Cada vértice (grupo) na árvore (exceto os vértices folha) é a união de seus filhos (sub-grupos), e a raiz da árvore é o grupo contendo todos os objetos [Tan et al. 2005].

Para uma representação visual do agrupamento hierárquico é utilizado o dendrograma. Esta estrutura é uma árvore com N folhas e altura $N - 1$, na qual os documentos são dispostos no eixo horizontal, enquanto que o eixo vertical indica a distância (ou a similaridade) com que os agrupamentos são criados.

O nó raiz do dendrograma representa todo o conjunto de dados, e cada nó folha é considerado um ponto de dados. Os nós intermediários, portanto, indicam quão próximos estão os objetos uns dos outros e a altura do dendrograma usualmente expressa a distância entre cada par de pontos de dados ou grupos, ou entre um ponto de dados e um grupo [Xu and Wunsch 2008]. Um exemplo de dendrograma é apresentado na Figura 1.

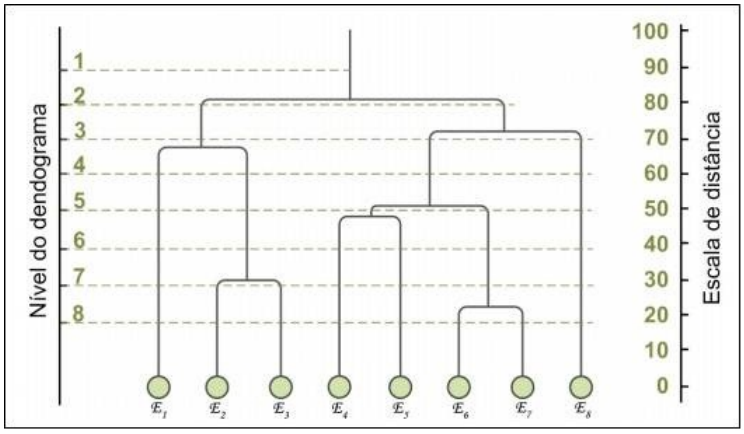


Figura 1. Clustering Hierárquico representado num dendrograma. Fonte [Metz 2006].

De acordo com [Zheng et al. 2006], o agrupamento de textos pode-se dividir em três categorias: agrupamentos baseado em palavras, em conhecimento e em informação. O agrupamento baseado em palavras representa os documentos pelas suas palavras-chaves. O agrupamento baseado em conhecimento considera um conhecimento manualmente criado. Finalmente, o agrupamento baseado em informação é sensível ao contexto, este considera frases, segmentos de texto e a semântica.

Os métodos mais representativos para obter agrupamentos hierárquicos são: *single-link*, *complete-link* e *average-link* [Tan et al. 2005]. O *single-link* define a proximidade de *clusters* como a proximidade entre os dois pontos mais próximos que estão em diferentes *clusters*. Enquanto, o *complete-link* considera a proximidade entre os pontos mais afastados em diferentes *clusters* como a proximidade de *clusters*. Finalmente, o *average-link* define a proximidade de *clusters* como a proximidade média de pares de pontos de *clusters* diferentes. Uma ilustração das proximidades desses métodos é apresentada na Figura 2.

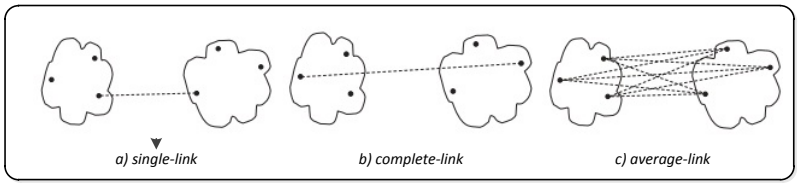


Figura 2. Critérios de proximidade de *clustering*. Fonte [Tan et al. 2005].

3. Abordagem Proposta para Comparação Automática de Grades Curriculares e Estudo de Caso

Para identificar similaridades entre grades curriculares e, assim, realizar uma análise entre elas de forma quase automática, propõe-se utilizar agrupamento hierárquico de textos. Nesta seção, apresenta-se a aplicação dessa técnica para um conjunto de grades curriculares de cursos de computação correspondentes a dois países: Peru e Brasil. Na Figura 3 apresenta-se um diagrama geral do processo seguido. Os detalhes do corpus de documentos utilizado, assim como os passos envolvidos na abordagem proposta para esse processo são descritos a seguir.

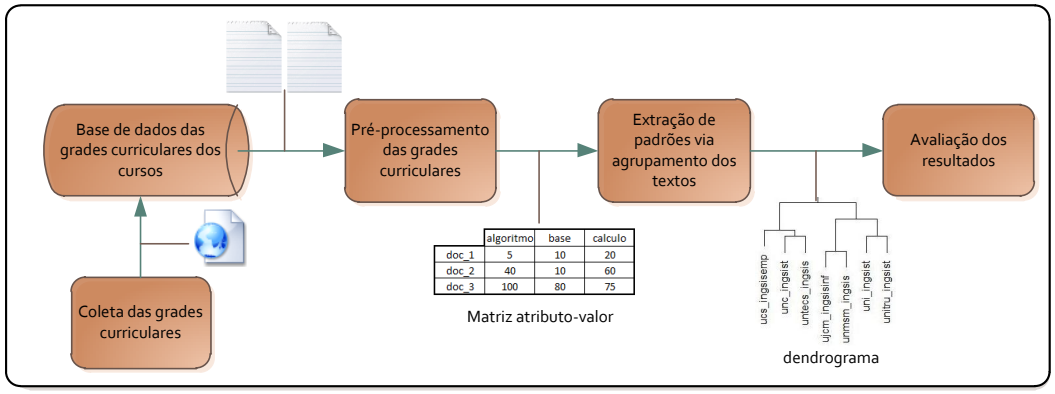


Figura 3. Abordagem Proposta para Comparação Automática de Grades Curriculares

3.1. Corpus de documentos

O corpus está conformado por 73 grades curriculares correspondentes a cursos de graduação em computação nas áreas de: ciências da computação, sistemas de informação e engenharia de software. De todas elas, 40 correspondem a universidades peruanas (totalidade de cursos com informação disponível na web) e 33 a universidades brasileiras (instituições com cursos de pós-graduação em computação - avaliação CAPES ≥ 3.

Para cada grade curricular foi criado um arquivo de texto com os nomes das disciplinas oferecidas (obrigatórias e optativas). Cada arquivo foi nomeado considerando

a sigla da universidade e/ou instituto do curso e a sigla do nome de curso. Por exemplo, no arquivo usp-icmc_cc, usp-icmc é a sigla do Instituto de Ciências Matemáticas e de Computação (ICMC) da Universidade de São Paulo (USP) e cc é a sigla do curso de Ciência da Computação.

Adicionalmente, considerando as diferenças entre as línguas dos países, foi necessário traduzir os nomes das disciplinas peruanas para o português. A tradução foi feita manualmente, pois a tradução automática realizada de forma direta (palavra por palavra) em muitos casos não permitiria uma comparação apropriada. Por exemplo, a disciplina de Trabalho de Conclusão de Curso no Brasil, possui diferentes nomes no Peru, como: Trabajo de Tesis, Trabajo de Graduación, Tesis e Proyecto de Fin de Carrera. Para poder obter concordância na tradução manual, foi criada uma tabela de equivalências entre os nomes de várias disciplinas que apresentam esse tipo de problema. Na Tabela 1 são apresentadas algumas dessas equivalências.

Nome em português	**Nome em espanhol**
Laboratório	Taller Laboratorio
Pesquisa Operacional	Investigación Operacional Investigación de Operaciones
Trabalho de Conclusão de Curso	Trabajo de Graduación Trabajo de Tesis Tesis Proyecto de Fin de Carrera
Estagio Pré-profissional	Práctica Pre profesional

Tabela 1. Algumas equivalências entre nomes de disciplinas

3.2. Pré-processamento das Grades Curriculares

Um dos passos de maior importância na mineração de textos consiste em processar os documentos para obter uma matriz atributo-valor e, assim, deixá-los no formato adequado para o processo de extração de padrões.

Para transformar texto não estruturado numa tabela atributo-valor, foi utilizada a abordagem *bag-of-words*, na qual cada palavra é considerada como um atributo e a frequência da palavra no texto é o valor do atributo. Para este processo, primeiro realizou-se a *tokenização* do conteúdo de cada grade curricular para separá-lo em palavras isoladas (sem considerar as sinais de pontuação). Depois disso, a técnica de *stemming* permitiu transformar cada palavra dos textos para o radical que o originou, através da remoção de sufixos, seguindo umas regras linguísticas pré-estabelecidas na ferramenta.

Um problema com a abordagem *bag-of-words* é que ao considerar cada palavra como um atributo a dimensionalidade (tamanho) da matriz atributo-valor é alta. Para lidar com esse problema primeiro realizou-se a *remoção de stopwords* para remover aquelas palavras que são muito comuns (preposições, artigos, etc.) e, portanto, não significativas para o algoritmo de aprendizado quando consideradas isoladamente. Finalmente, para só empregar na análise aquelas palavras (ou *stems*) mais representativas dentre as existentes e reduzir ainda mais a dimensionalidade, foi realizada uma seleção de palavras baseada

em frequência. Só aquelas palavras na matriz atributo-valor com uma frequência maior a 10 e menor do que 100 foram selecionadas. Neste processo, foi utilizada a ferramenta PreTexT [Soares et al. 2008], que implementa as funcionalidades descritas previamente.

3.3. Extração de Padrões via Agrupamento de Textos

A comparação de textos é considerada como uma atividade descritiva já que se deseja identificar conjuntos de grades curriculares similares. Por este motivo, foi empregado um algoritmo de agrupamento para identificar e analisar as similaridades entre os cursos.

Foi selecionado um algoritmo de *clustering* hierárquico para poder visualizar os diferentes conjuntos de grades curriculares em 2, 3,...,k grupos. O algoritmo empregado foi o *average link* (implementado no R [R Development Core Team 2011]), pois, em dados textuais, avaliações experimentais têm mostrado que ele é uma boa opção entre os algoritmos que adotam estratégias aglomerativas [Zhao et al. 2005].

Empregando o algoritmo selecionado, foram obtidos os dendrogramas correspondentes aos experimentos planejados. Na Figura 4, são apresentados os grupos gerados considerando só as grades curriculares dos cursos peruanos; na Figura 5 foram considerados só os cursos brasileiros e, finalmente, na Figura 6 são apresentadas as relações obtidas de todo o conjunto de documentos considerado no estudo de caso.

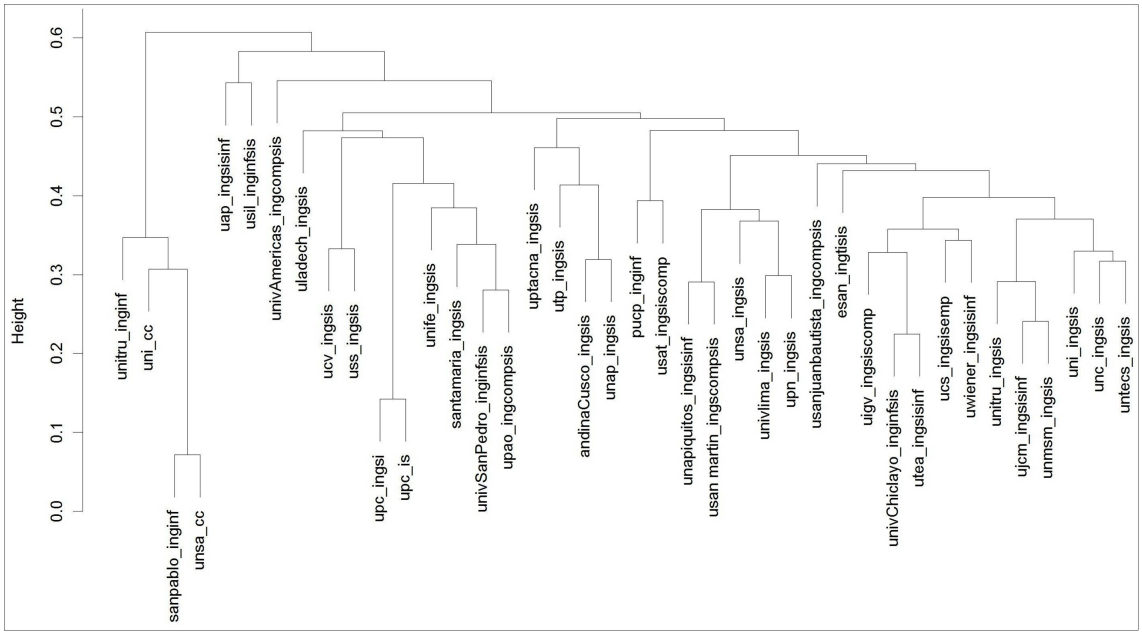


Figura 4. Dendrograma das grades curriculares do Perú

4. Análise e Avaliação dos Resultados

Na Figura 4 são apresentados dois grupos de grades curriculares: o primeiro (da esquerda) formado pelos programas unitru_inginf, uni_cc, sanpablo_inginf e unsa_cc, e o segundo (da direita) formado por todos os demais. Depois da revisão manual das disciplinas oferecidas nos programas de cada grupo, pode-se observar que o primeiro tem uma maior porcentagem de disciplinas orientadas a Ciências da Computação, e o segundo está mais focado em Sistemas de Informação.

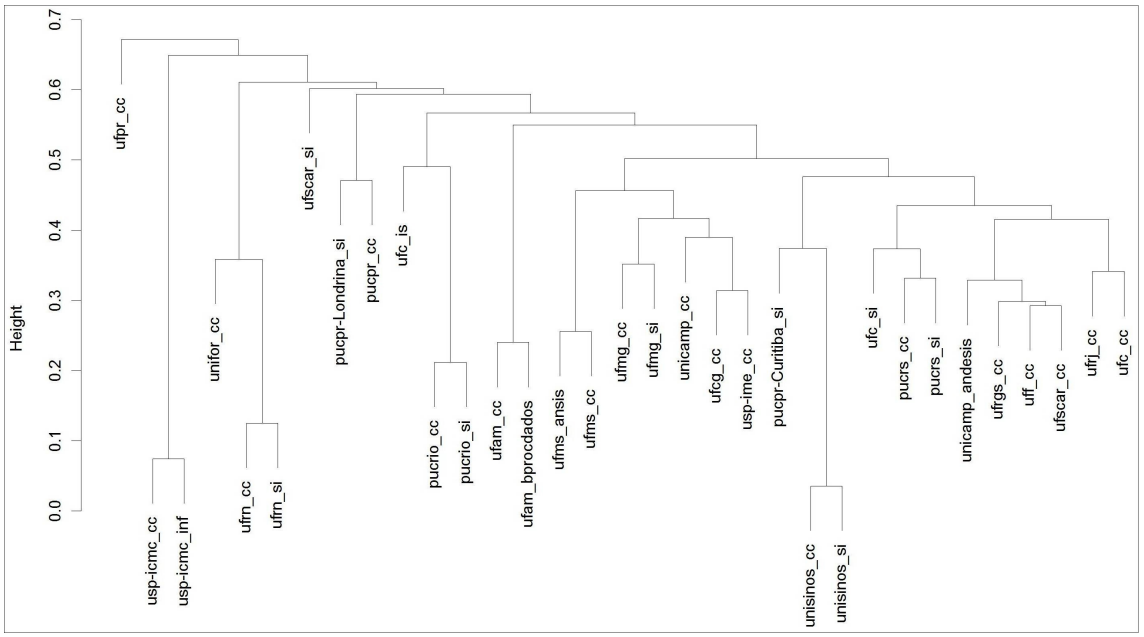


Figura 5. Dendrograma das grades curriculares do Brasil

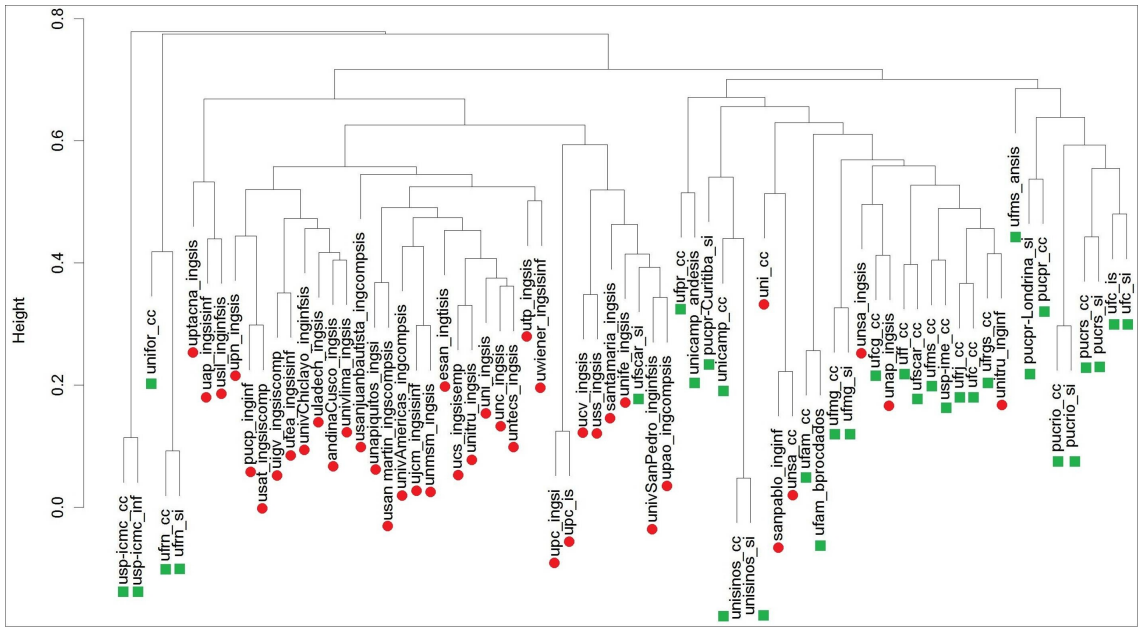


Figura 6. Dendrograma das grades curriculares do Perú (•) e do Brasil (■)

Na Figura 5, os grupos formados pelas grades curriculares brasileiras possuem a característica que os cursos que pertencem à mesma universidade ou instituto estão fortemente relacionados. Por exemplo: usp-icmc_cc com usp-icmc_inf, pucrio_cc com pucrio_si, entre outras. Quando for buscado o mesmo padrão na Figura 4, isso só seria possível com as grades curriculares da universidade *upc*, mas os cursos das universidade *unitru* e *uni* não possuem esse mesmo relacionamento. Analisando as disciplinas dos pares de cursos que apresentam esta característica, é apreciado que eles compartilham várias disciplinas dos primeiros anos e, depois, os estudantes recebem matérias de

especialização na área escolhida. Isto não acontece na *unitru* ou na *uni* e por isso essa separação entre os seus cursos.

Finalmente, na Figura 6, as grades curriculares formam quatro grandes grupos: o primeiro formado pelos dois cursos de computação do *icmc-usp*; o segundo, pelo curso de computação da *unifor* e os dois cursos da *ufrn*; no terceiro, 34 dos 40 cursos de universidades peruanas; e finalmente, no extremo direito, têm-se 27 dos 33 cursos brasileiros.

Considerando só o grupo do extremo direito, na Figura 6, pode ser observado que somente seis cursos de universidades peruanas estão relacionadas com os brasileiros: uni_cc, sanpablo_inginf, unsa_cc, unsa_ingsis, unap_ingsis e unitru_inginf. As quatro primeiras, voltando para a Figura 4, mantém uma forte relação, evidenciada pela sua proximidade no primeiro grupo, como foi descrito previamente. A característica que as grades curriculares destas universidades apresentam é que as disciplinas estão completamente orientadas à área de Ciências de Computação.

5. Como Utilizar o Conhecimento Extraído das Grades Curriculares

De modo geral, o conhecimento extraído automaticamente a partir das grades curriculares dos cursos pode ser utilizado para analisar situações e apoiar tomadas de decisões.

Particularmente, o conhecimento extraído pode ser usado para testar modificações nas grades curriculares atuais ou avaliar novas propostas de grades. Os encarregados em projetar a nova grade curricular (ou propor as modificações de uma existente) de um determinado curso poderia usar a comparação da sua proposta com as existentes atualmente. Uma análise como apresentada neste artigo permite contrastar o "novo" modelo com o "antigo", assim como as diferenças com outros cursos mais consagrados.

Outro uso pode ser a comparação das grades curriculares de um país com as de cursos de referência internacional, o que permite avaliar a situação desses cursos e, talvez, a qualidade dos mesmos. No estudo de caso realizado com cursos de universidades peruanas e brasileiras, por exemplo, pode-se observar que só quatro universidades peruanas possuem cursos de computação que têm muita semelhança com os cursos brasileiros, e que a grande maioria dos cursos peruanos é muito distante dos brasileiros em termos de grades curriculares. Uma possível interpretação deste resultado é que o perfil do egresso desses cursos é diferente do perfil brasileiro, i.e., pode estar mais relacionado com a realidade do mercado de trabalho peruano. Estes resultados podem servir, por exemplo, para evidenciar problemas no projeto das grades curriculares e propor mudanças nestes cursos. Comparações deste tipo também poderiam ser realizadas considerando propostas de grades curriculares de instituições amplamente renomadas, como o caso da ACM/IEEE.

Adicionalmente, este tipo de análise também pode ser aproveitado para realizar um estudo de evolução dos cursos na linha do tempo. Gráficos como os aqui apresentados, poderiam ser gerados periodicamente para analisar a progressão dos cursos e apoiar a tomada de decisões estratégicas.

6. Conclusões e Trabalhos Futuros

Neste trabalho foi aplicado um método de agrupamento hierárquico de textos para comparar grades curriculares em cursos de computação. O método foi aplicado para descobrir relações ocultas entre as grades curriculares de cursos de graduação, correspondentes a

universidades do Peru e do Brasil. Os resultados obtidos evidenciam que este tipo de análise ajuda na exploração das relações embutidas nas grades curriculares.

Foram descritas algumas situações nas quais este tipo de análise poderia ser aproveitada como suporte na toma de decisões sobre as características de uma grade curricular. O método proposto permite comparar quase automaticamente uma grade contra outras, o que poderia ser utilizado na estimação da qualidade de um curso. Contudo, sempre é necessário o conhecimento de um especialista na área para poder interpretar apropriadamente os resultados obtidos.

É sabido que nem sempre o nome de uma disciplina corresponde ao conteúdo ensinado na mesma. Por isso, para melhorar o processo de comparação, seria melhor considerar o conteúdo total das disciplinas, i.e., as suas ementas. Infelizmente, nem todas as universidades e/ou institutos disponibilizam este tipo de informação publicamente na web para que possa ser aproveitada. Construir um corpus com estas características e aplicar o método descrito para descobrir novas relações constitui-se num trabalho futuro.

Os métodos hierárquicos aglomerativos usados não são incrementais e isso indica que ao ingressar um novo documento é necessário realizar todo o processo novamente. Quando o corpus de documentos é pequeno isso não é um grande problema, mas se o corpus fosse de um tamanho maior apresentariam-se dificuldades no tempo de processamento. Utilizar uma abordagem hierárquica incremental para o auto-agrupamento de novas grades curriculares sem a necessidade de processar todo o conjunto de textos de novo, apresenta-se como um caminho a ser explorado.

Agradecimentos

Este trabalho foi realizado com apoio financeiro da CAPES e CNPq.

Referências

Biddle, R. L. and Tempero, E. D. (1996). Comparing a Computing Curriculum with the ACM/IEEE-CS Recommendations. In *Proceedings of the 1996 International Conference on Software Engineering: Education and Practice*, SEEP '96, pages 263–270, Washington, DC, USA. IEEE Computer Society.

Do Prado, H. A. and Ferneda, E. (2007). *Emerging Technologies of Text Mining: Techniques and Applications*. Information Science Reference (an imprint of IGI Global), Hershey, PA.

Ebecken, N. F. F., Lopes, M. C. S., and de Aragão Costa, M. C. (2003). Mineração de Textos. In Rezende, S. O., editor, *Sistemas Inteligentes - Fundamentos e Aplicações*, chapter 13, pages 338–370. Manole.

Manning, C. D., Raghavan, P., and Schtze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.

Metz, J. (2006). Interpretação de clusters gerados por algoritmos de clustering hierárquico. Master's thesis, Instituto de Ciências Matemáticas e de Computação - USP - São Carlos.

Pereira, L. Z., de Albuquerque, J. P., and de S. Coelho, F. (2010). Uma Análise da Oferta e Abordagem Curricular dos Cursos de Bacharelado em Sistemas de Informação no

Brasil. In *XVIII Workshop de Educação em Computação (WEI 2010), Anais do XXX Congresso da Sociedade Brasileira de Computação - CSBC 2010*, pages 897–906.

Prietch, S. S. and Pazeto, T. A. (2010). Mapeamento de Cursos de Licenciatura em Computação seguido de Proposta de Padronização de Matriz Curricular. In *XVIII Workshop de Educação em Computação (WEI 2010), Anais do XXX Congresso da Sociedade Brasileira de Computação - CSBC 2010*, pages 921–930.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Soares, M. V., Prati, R. C., and Monard, M. C. (2008). PreTexT II: Descrição da Reestruturação da Ferramenta de Pre-Processamento de Textos. Technical Report 333, ICMC-USP, São Carlos - SP.

Srivastava, A. and Sahami, M. (2009). *Text Mining: Classification, Clustering, and Applications*. Chapman & Hall/CRC, 1st edition.

Tan, P.-N., Steinbach, M., and Kumar, V. (2005). Cluster Analysis: Basic Concepts and Algorithms. In *Introduction to Data Mining*, chapter 8, pages 487–568. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition.

Xu, R. and Wunsch, D. C. (2008). Hierarchical Clustering. In *Clustering*, chapter 3, pages 31–62. John Wiley & Sons, Inc.

Zhao, Y., Karypis, G., and Fayyad, U. (2005). Hierarchical Clustering Algorithms for Document Datasets. *Data Mining Knowledge Discovery*, 10:141–168.

Zheng, Y., Cheng, X., Huang, R., and Man, Y. (2006). A Comparative Study on Text Clustering Methods. In *Proceedings of the Second International Conference on Advanced Data Mining and Applications*, ADMA 2006, pages 644–651, Xi'An, China.

Licenciatura em Computação no Estado de São Paulo: uma Análise Contextualizada e um Estudo de Caso

Luis Naito Mendes Bezerra, Ismar Frango Silveira

Universidade Cruzeiro do Sul – Av. Dr. Ussiel Cirilo, 225 – 08060-070 São Paulo – SP

{luis.naito, ismar.silveira}@cruzeirodosul.edu.br

Resumo. *As Licenciaturas em Computação surgiram com o objetivo principal de suprir a necessidade de profissionais qualificados para atuar no ensino de Computação na Educação Básica, assim como atuar nas mais diversas aplicações da área da Informática na Educação. Entretanto, o que tem se observado é uma redução tanto na oferta quanto na procura por tal curso em âmbito nacional. Considerando a Computação como uma ciência no mesmo patamar das ciências "clássicas", o propósito deste artigo é apresentar um panorama do curso de Licenciatura em Computação no estado de São Paulo, considerando o contexto nacional e analisando um caso específico de um curso de uma IES situada na capital do estado.*

Abstract. *Computer Science Teaching Degree undergraduate courses emerged having as their main goal the need for qualified professionals to work in Computer teaching at K-12 and High School level, as well as in several applications of Computers in Education. Nonetheless, it has been observed a constant reduction in courses offering and enrollments in a nation-wide context. Considering Computer Science as a science at the same level than the so-said classical sciences, the main goal of this paper is to present a overview of such course in São Paulo state, regarding national context while analyzing a specific case of a course offered by a University in state's capital.*

1. Introdução

As Licenciaturas em Computação têm em sua gênese a intenção de formar profissionais qualificados para atuar no ensino de Computação e Informática no nível médio, além de serem aptos a atuarem no segmento de mercado que envolve a Informática na Educação de maneira plena. Isto está bastante claro em Cabral et al. (2008) e Dahmer et al. (2001), que ressaltam que os cursos de Licenciatura em Computação visam formar recursos humanos para projetar sistemas de software para EAD e educacional e formar educadores para o ensino de Computação em instituições que introduzirem Computação em seus currículos, como matéria de formação.

Nesse aspecto, os mesmos autores mencionam que não há consenso sobre a forma correta de introduzir a computação no Ensino Médio. Isso reflete diretamente na criação de um perfil não muito claro para o egresso de tais cursos, os quais não raramente se vêem disputando vagas no mercado com profissionais de diversas outras formações, muitas vezes não adequadas. Junte-se isso ao fato dos setores de mercado, quer na esfera pública ou privada, em geral desconhecerem o perfil de um egresso em Licenciatura em Computação, os objetivos do curso, ou mesmo a sua própria existência,

chega-se em um cenário onde o egresso não encontra espaço adequado de atuação. O desconhecimento não se limita aos empregadores, como também se estende aos vestibulandos, que não raro desconhecem por completo ou têm concepções errôneas a respeito desse curso, isso quando são cônscios de sua existência.

Assim o objetivo específico deste artigo é o de delinear um perfil dos cursos de Licenciatura em Computação no Estado de São Paulo, contextualizando-os no panorama nacional, ao mesmo tempo em que se analisa um caso específico da oferta desse curso, hoje em processo de extinção, em uma IES privada na capital do estado.

2. A Computação como ciência básica

Apesar da Computação estar bem estabelecida e com o "status" de Ciência nos âmbitos da Graduação e Pós-Graduação, o mesmo não ocorre na Educação Básica. Denning (2005) ressalta o caráter eminente de Ciência existente na Computação, sendo classificada no clássico trabalho de Newell e Simon (1976) como uma ciência natural. Desta maneira, seria plausível dar à Computação o mesmo tratamento dispensado às Ciências ditas "clássicas" (Física, Química e Biologia), que, juntamente com a Matemática formam o corpus de conhecimento da área de Ciências Naturais tradicionalmente presente no currículo da Educação Básica. Entretanto, não é o que ocorre, o que acaba – entre outros aspectos – por afetar a procura por cursos de formação de professores nessa área. As Figuras 1 e 2 a seguir são esclarecedoras neste aspecto.

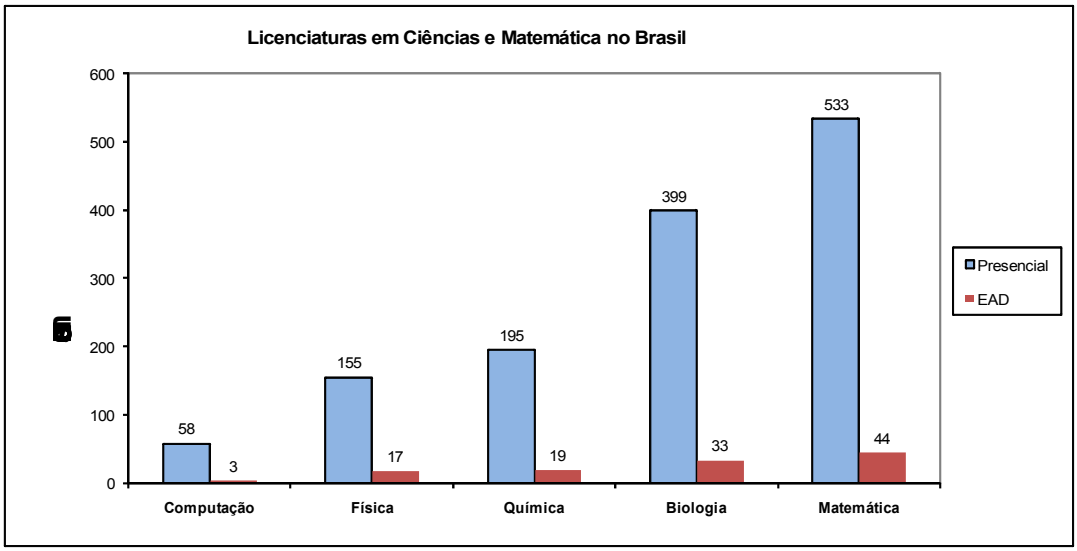


Figura 1. Cursos de Licenciatura em Ciências e Matemática no Brasil em 2009 (INEP, 2011)

A Figura 1 mostra que o número de cursos presenciais de Licenciatura da Computação no Brasil em 2009 era de 58 e somente 3 ofertados na modalidade a distância, de acordo com o Censo do INEP. Mais adiante, será mostrado que o número de cursos presenciais reduziu de 69 em 2008 para 49 em 2010, de acordo com o e-MEC. Isso representa aproximadamente 4% dos 1340 cursos presenciais existentes em 2009 na área das Ciências (incluindo Computação) e Matemática. Mesmo na modalidade a distância, a oferta de cursos de Licenciatura em Computação é reduzida em relação, por exemplo, à Licenciatura em Matemática (3 naquela e 44 nesta).

A Figura 2 mostra que em 2009 havia 15 vezes mais egressos em cursos presenciais de Licenciatura em Matemática do que em Computação. Os números de egressos da modalidade EAD não são comparáveis até o presente momento, dado que estes cursos têm históricos de criação diversos e são bastante recentes.

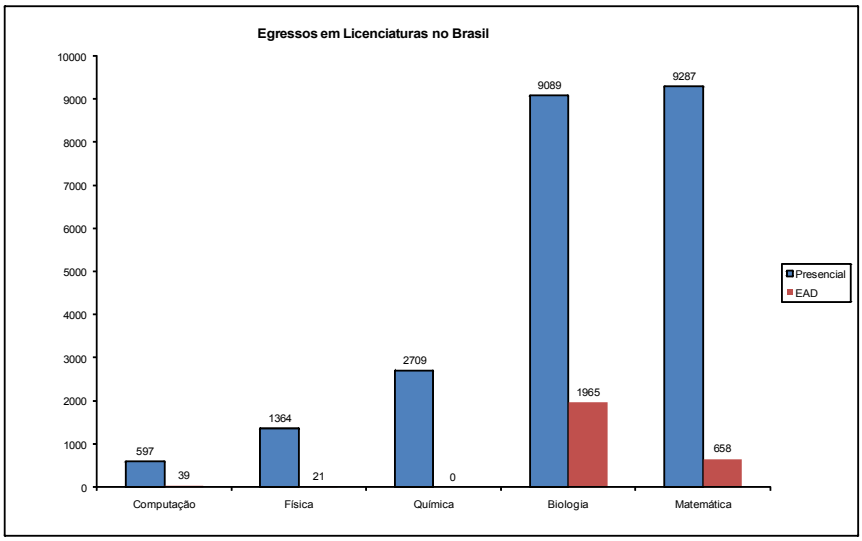


Figura 2. Egressos de Licenciaturas em Ciências e Matemática no Brasil (INEP, 2011)

Os dados apresentados levam à necessidade de reflexão sobre o papel da Computação na Educação Básica, em especial frente aos desafios impostos pelo milênio que há pouco se iniciou. O ACM Model Curriculum for K-12 Computer Science (Tucker, 2006) advoga a necessidade de desenvolvimento de habilidades computacionais na Educação Básica, no sentido da Ciência da Computação ser importante intelectualmente, levar à possibilidade de múltiplos caminhos profissionais futuros, desenvolver capacidades de resolução de problemas, dar suporte e se relacionar com outras ciências e motivar os estudantes. Tal currículo de referência sugere a formação em Computação de alunos da Educação Básica de acordo com uma abordagem tri-axial, que contempla conceitos, habilidades e competências, conforme a Figura 3.

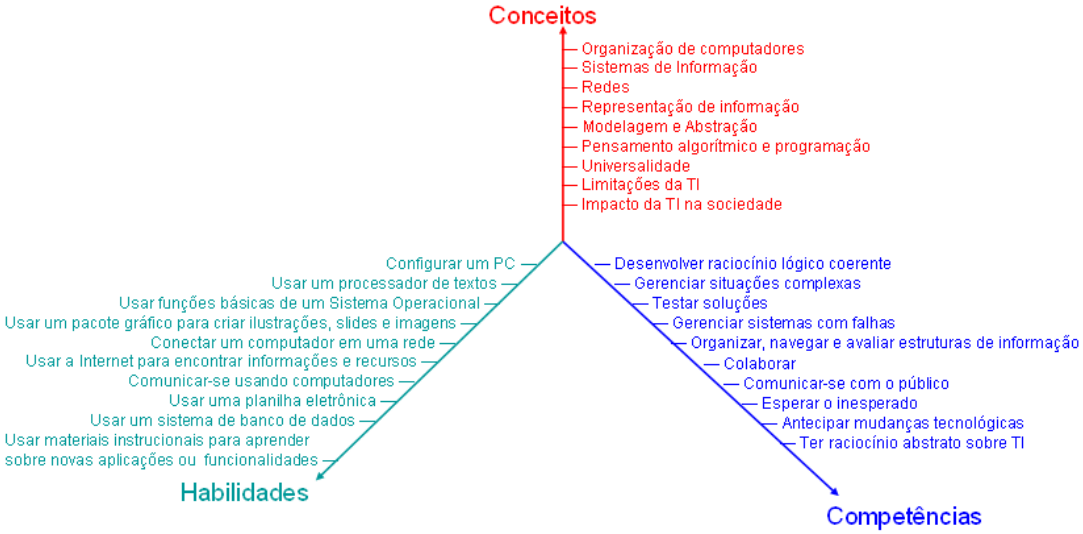


Figura 3. Conceitos, Habilidades e Competências da Ciência da Computação para Educação Básica – adaptado de Tucker (2006)

Dado esse conjunto de conceitos, habilidades e competências esperados de um aluno da Educação Básica, é patente a necessidade de que a formação de educadores que venham a atuar nesse nível de ensino propicie uma capacitação adequada a estes, de maneira a permiti-los uma prática docente que proporcione a seus alunos condições de aprendizagem que os permitam desenvolver tais habilidades e competências, dominando os conceitos apresentados.

O que se verifica na literatura é o fato de que docentes com formação em Ciências e Matemática (considerando o espectro tradicional do termo "Ciências") não têm, durante a graduação, elementos curriculares suficientes para atingir os objetivos acima citados. Fernandes e Silveira (2009), ao analisar matrizes curriculares das Licenciaturas em Matemática oferecidas no Estado de São Paulo, apontam graves deficiências no que diz respeito aos conteúdos de informática na formação de professores de Matemática. Neste trabalho, que corrobora Wilges (2009), os autores apontam que os conteúdos curriculares levam quase sempre a uma abordagem utilitária de softwares, raramente tratando do assunto no contexto da Informática na Educação. Um estudo mais amplo a respeito, envolvendo por completo a área de formação de professores em Ciências e Matemática pode ser visto em Gatti (2009). Dados estes estudos, pode-se concluir que a formação de professores em Ciências clássicas e Matemática não atende ao perfil esperado pelo currículo de referência da ACM, exposto acima. Assim, uma formação sólida em Computação voltada para atuação docente no nível da Educação Básica faz-se necessária, o que reforça a justificativa para a atuação de egressos de Licenciaturas em Computação nesse nível de educação.

3. A Licenciatura em Computação no estado de São Paulo

Com base em levantamento efetuado no final de 2010 no Sistema e-MEC (http://emec.mec.gov.br), é possível observar que há 49 cursos de Licenciatura cadastrados no país, distribuídos de acordo com a Figura 4 abaixo.



Figura 4. Distribuição geográfica do atuais 49 cursos presenciais de Licenciatura em Computação no País (Adaptado a partir do Sistema e-MEC, 2010)

Entretanto, este levantamento pode não revelar a real situação dos cursos no país, já que um número significativo destes pode encontrar-se em estado de extinção, ou seja, não são atualmente ofertados nos vestibulares das IES, apesar de constar o seu registro junto ao Ministério da Educação. O que se verifica, na realidade, é um baixo número de egressos oriundos destes cursos, como assinalam Priecht e Pazeto (2009).

Este quadro pode ser visto e maior detalhe no estado de São Paulo, objeto de pesquisa deste artigo: de acordo com o Sistema e-MEC, há cadastrados doze cursos de Licenciatura em Computação no estado de São Paulo, sendo onze presenciais. Entretanto, ao se visitar os portais institucionais das IES que constam nesse cadastro, percebe-se que 6 delas não ofertaram vagas em seus processos seletivos de 2010 e/ou 2011, como pode ser visto na Tabela 1 a seguir.

Tabela 1. Oferta da Licenciatura em Computação no Estado de SP em 2010 (Fonte: e-MEC)

IES	Portal	Ofertado	Modalidade
UNASP	www.unasp.br	*não*	Presencial
CEUCLAR	www.claretiano.edu.br	sim	EAD
CUFSA	www.fsa.br	sim	Presencial
FAENAC	www.faenac.edu.br	*não*	N/A
FACCAT	www.faccat.br	*não*	N/A
UniFADRA	www.fundec.br	sim	Presencial
FADISC	www.fadisc.br	sim	Presencial
FATEA	www.fatea.br	sim	Presencial
COC	www.coc.com.br	sim	Presencial
UNICSUL	www.cruzeirodosul.edu.br	*não*	Presencial
UNIABC	www.uniabc.br	*não*	N/A
USC	www.usc.br	*não*	N/A

(N/A significa informação não disponível)

O que pode ser percebido através desta tabela é que, mesmo sendo o estado de São Paulo a ter o maior número de cursos de Licenciatura em Computação, o fato de apenas 50% destes estarem em funcionamento permite inferir que o número de cursos com oferta regular no Brasil é seguramente inferior ao número obtido no levantamento junto ao e-MEC.

4. Estudo de Caso na Universidade Cruzeiro do Sul

O Curso de Licenciatura em Computação da Universidade Cruzeiro do Sul, situada em São Paulo capital, foi ofertado a partir de 2001, sendo estruturado em três anos, com disciplinas anuais com carga horária de duas, três ou quatro horas semanais, distribuídas de segunda a sábado. O curso foi ofertado regularmente até 2004, ano em recebeu o reconhecimento do MEC, e coincidentemente o primeiro ano em que não recebeu número suficiente de ingressantes para a formação de turma. O ano de 2006, que representa o ano de conclusão da última turma regular, também foi o ano a partir do qual o curso deixou de ser oferecido no processo seletivo da IES.

A matriz curricular, que pode ser vista na Tabela 2 a seguir, pretendia oferecer ao aluno uma sólida formação básica e tecnológica em computação, além de formação nas áreas educacional e humanística.

Tabela 2. Estrutura da Matriz Curricular da Licenciatura em Computação da Universidade Cruzeiro do Sul

Núcleo	Disciplina
Fundamentos da Computação	Sistemas Operacionais Teoria da Computação e Linguagens Formais Algoritmos e Programação Linguagens e Técnicas de Programação Algoritmos e Estruturas de Dados
Tecnologia da Computação	Organização e Arquitetura de Computadores Banco de Dados Engenharia de Software Redes de Computadores Sistemas Multimídia e Interfaces Inteligência Artificial
Núcleo de Matemática	Fundamentos da Matemática Lógica Matemática Matemática Discreta Geometria Analítica e Álgebra Linear Probabilidade e Estatística
Núcleo de Contexto Social e Profissional	Introdução à Educação Psicologia da Educação Didática Informática e Sociedade Comunicação e Expressão Inglês Teoria e Prática Pedagógica em Informática I Organização da Educação Brasileira Métodos de Avaliação de Aprendizagem Teoria e Prática Pedagógica em Informática II

Procurou-se equilibrar a quantidade de disciplinas específicas do núcleo de Computação (Fundamentos e Tecnologia) com as de formação complementar (Matemática, Contexto Social e Profissional), como pode ser visto na Figura 5 a seguir.

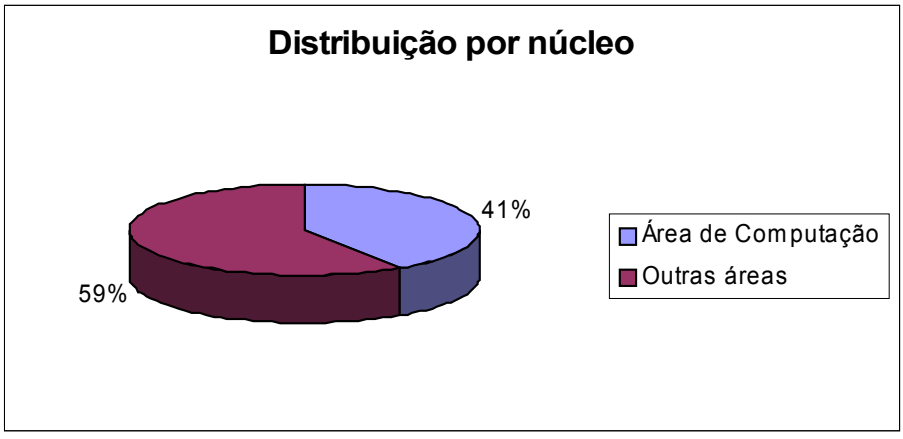


Figura 4. Distribuição das disciplinas por núcleo

As disciplinas propostas, acrescidas do estágio supervisionado e de uma metodologia focada no aprendizado tinham por objetivo capacitar o egresso a colaborar na disseminação da Informática na sociedade brasileira, a partir do exercício da docência e consultoria em escolas públicas e particulares, assim como a atuação na criação, desenvolvimento e implantação de software educativo.

Com esta matriz curricular e os objetivos listados, a IES propunha um curso com vocação multidisciplinar, integrando as áreas de Computação, Informática e Educação, que oferecesse ao aprendiz a apropriação e desenvolvimento dessas áreas em um ambiente dinâmico e criativo, caracterizado pela participação ativa do aprendiz em sala de aula.

O Projeto Pedagógico do curso previa uma sólida formação fundamental em Computação e Educação, que aliada ao desenvolvimento da iniciativa própria e à capacidade de enfrentar novas situações, deveriam ser outras importantes características do perfil do profissional a ser formado, obtidas por meio de um currículo abrangente, da prática em situações reais e da relação ensino-aprendizagem participativa. Desta forma, o profissional deveria ter autonomia na busca de informações e uma base conceitual que lhe permitira acompanhar a rápida evolução do conhecimento nessa área do saber.

Na primeira turma do curso de Licenciatura o principal desafio com relação ao corpo docente foi propiciar uma integração das diversas formações dos professores e os objetivos de um curso de Licenciatura da Computação. Este desafio possibilitaria uma transição de um curso multidisciplinar para um curso interdisciplinar com a adoção de estratégias e práticas de ensino e aprendizagem apropriadas ao perfil do curso, ingressantes e egressos. Entretanto, a queda gradativa do número dos ingressantes levou à IES ao encerramento de ofertas de vagas no curso em epígrafe.

A Figura 5 a seguir mostra a evolução do número total de alunos ativos no curso no período 2001-10.

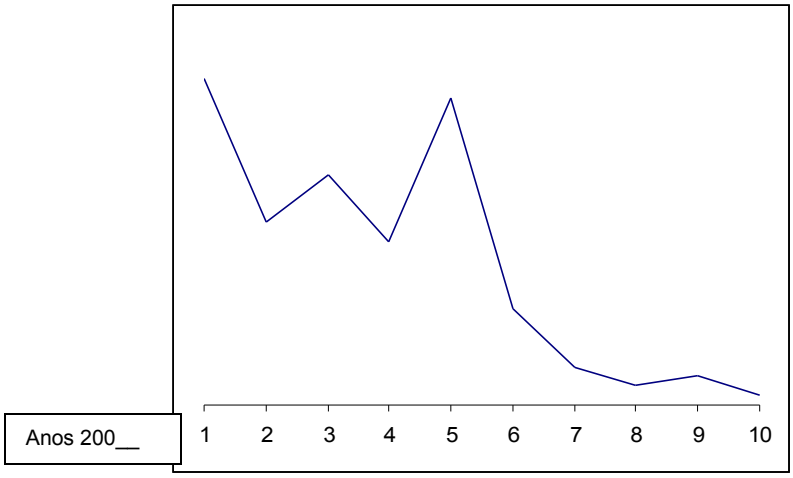


Figura 5: Evolução do número total de alunos matriculados no curso

Note-se que a queda obviamente acentuada no número total de matrículas a partir da cessão da oferta do curso nos processos seletivos da IES α, restando atualmente somente alunos antigos em processo de complementação de disciplinas.

Algumas considerações podem ser feitas que justifiquem, ou ao menos lancem luz sobre a queda vertiginosa na procura por este curso, entre as quais:

- A falta de clareza, em todos os níveis, do perfil esperado do egresso da Licenciatura em Computação. Ao não compreender exatamente os objetivos do curso, os alunos tendiam a enxergá-lo como um Bacharelado em Ciência da Computação, com pitadas de pedagogia e com menor tempo de conclusão, ou,

em um viés diametralmente oposto, como um curso de Padagogia com ênfase no uso da Informática, porém com uma visão totalmene utilitarista desta.

- A inexistência de políticas públicas de absorção do egresso em escolas municipais, estaduais ou mesmo particulares. O egresso, ao mesmo tempo em que era preterido em vagas para a docência das "outras" ciências, concorria em pé de igualdade com os egresso de licenciaturas outras.
- No caso específico deste curso na IES estudada, verifica-se que uma divulgação do mesmo feita de maneira direcionada poderia ter atenuado a queda na procura ou mesmo retardado a extinção do curso. Entretanto, essa ação não teria sido suficiente, a longo prazo, para suplantar os problemas existentes no cenário exposto nos itens acima.

5. Conclusões

Os cursos de Licenciatura em Computação são cada vez mais necessários para a formação de profissionais qualificados para atuar no campo de Informática na Educação, especialmente em se considerando os desafios do ensino de Computação para a Educação Básica nesta nova realidade imposta pela importância da área, cada vez mais pervasiva no cotidiano da sociedade pós-industrial. Dada esta demanda, seria de se esperar um crescimento na oferta e na procura de tais cursos.

Entretanto, o cenário que se delineia é exatamente o oposto: verifica-se uma queda acentuada nas ofertas de cursos na IES do Brasil, em particular no Estado de São Paulo. Tal queda justifica-se diretamente pela diminuição de ingressantes, o que pode ser uma conseqüência visível da falta de uma definição mais clara do perfil de egresso – definição esta que poderia ser mais bem trabalhada no âmbito da SBC, o que vem ocorrendo, bem como de políticas públicas de incentivo à formação específica de professores de Computação para atuação na Educação Básica – situação esta que vem sendo trabalhada em alguns estados da Federação, que vêm estudando criar uma legislação específica para garantir que os professores de computação na Educação Básica sejam egressos de cursos de Licenciatura em Computação.

Não se trata aqui de defesa de reserva de mercado ou necessariamente de regulamentação da profissão: ao considerar-se a Computação como uma ciência, conforme colocado no início desse artigo, é plausível considerar que o ensino desta ciência na Educação Básica seja tarefa de professores qualificados para tal. Ao se observar que a realidade de hoje apresenta professores com formação em outras ciências sendo responsáveis pelo ensino de Computação, a ausência de ações que reafirmem o curso de Licenciatura em Computação – quer na esfera legal, profissional ou acadêmica – irá forçosamente fazer com que se acentue a queda do número de cursos no país, levando a uma perpetuação do status da Computação como vista hoje, uma mera ferramenta, de caráter utilitário. Este, certamente, não é o cenário desejado pela comunidade de Computação no Brasil.

Referências

Cabral, M. I. C. et al. (2008) A Trajetória dos Cursos de Graduação da Área de Computação e Informática. Rio de Janeiro: SBC.

Dahmer, A.; Santos, B. S.; Ogiba, S. e Kist, T. (2001) Uma Proposta de Plano Pedagógico para o Curso de Licenciatura em Computação. Anais do Curso de Qualidade da SBC de 2001. Disponível na Internet em <http://www.sbc.org.br/bibliotecadigital/download.php?paper=221> Acesso em 0 de setembro de 2010.

Denning, P. J (2005) Is Computer science science? Communications of the ACM, v. 48, 4, p. 27-31.

Fernandes, M. L. ; Silveira, I. F. (2009) Informática na Formação Inicial de Professores: um Estudo das Licenciaturas em Matemática do Estado de São Paulo. In: Norma Suely Gomes Allevato; Edda Curi. (Org.). Pesquisas e Práticas em Educação: Matemática e Ciências. 1 ed. São Paulo: Terracota, v. 1, p. 85-101.

Gatti, B. A. (2009) Formação de professores para o ensino fundamental: estudo de currículo das licenciaturas em pedagogia, língua portuguesa, matemática e ciências biológicas. In: NUNES, M. M. R. (Org.). Formação de professores para o ensino fundamental. São Paulo: Fundação Carlos Chagas, DPE

INEP (2011) Sinopses Estatísticas da Educação Superior. Disponível na Internet em: < http://www.inep.gov.br/superior/censosuperior/sinopse/default.asp> Acesso em 30 de março de 2011.

Newell, A. e Simon, H. A. (1976) Computer science as empirical inquiry: symbols and search. Communications of the ACM, v. 19, 3, p. 113-126

Priecht, S. S. e Pazeto, T. A. (2009) Análise, Sugestões e Perspectivas de um Curso de Licenciatura em Informática. Anais do XVII Workshop sobre Educação em Informática. Disponível na Internet em: < http://www.sbc.org.br/bibliotecadigital/ download.php?paper=1353> Acesso em 20 de setmbro de 2010.

Tuker, A. (2006) A Model Curriculum for K–12 Computer Science. Final Report of the ACM K–12 Task Force Curriculum Committee. Disponível na Internet em <http://csta.acm.org>. Acesso em 16 de setembro de 2010.

Wilges, A. M. (2006) Uma investigação acerca das práticas docentes no ensino superior de matemática envolvendo o uso de softwares educacionais, 2006. 115 f. Dissertação (Mestrado em Educação em Ciências e Matemática). - Faculdade de Física, Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, 2006. Disponível na Internet em http://tede.pucrs.br/tde_busca/ arquivo.php? codArquivo=678. Acesso em 20 de julho de 2009.

Um jogo educativo voltado ao ensino de Modelagem de Dados

Jarbele Cássia da Silva[1], Alisson V. Brito[1,2]
Departamento de Ciências Exatas – DCE [1]
Programa de Pós-Graduação em Informática – PPGI [2]
Universidade Federal da Paraiba - UFPB
{jarbele.cassia, alissonbrito}@dce.ufpb.br

Abstract. *This article presents an analysis of the Data Modeling lecture offered in a Degree in Computer Science. It was noted, therefore, a need for practical lessons that accurately reflect what is shown in theory. From this motivation, this study proposes the use of an educational game to teach this discipline.*

Resumo. *Este artigo apresenta o relato de uma análise realizada na disciplina Modelagem de Dados oferecida em um curso de Licenciatura em Ciências da Computação. Foi observada uma carência em relação a aulas práticas que reflitam com exatidão o que é demonstrado na teoria. A partir dessa motivação, este estudo propõe a utilização de um jogo educativo no ensino desta disciplina.*

1. Introdução

Segundo Costa (2008) a modelagem é uma das principais atividades no desenvolvimento de software, sendo ela uma ponte que visa encurtar a distância da abstração e entendimento da lógica de negócio entre os *stakeholders* e os profissionais da engenharia de software. É na disciplina Modelagem de Dados que alunos de tecnologia tem a oportunidade de adentrar no mundo da abstração.

O objetivo deste artigo é apresentar o relato de um estudo baseado nas estratégias de ensino utilizadas na disciplina Modelagem de Dados oferecida no curso de Licenciatura em Ciências da Computação (LCC) da Universidade XXX. Os dados aduzidos são relativos ao semestre 2010.1 e referem-se fielmente às opiniões e experiências relatadas por alunos e professores que já experimentaram desta fonte. Após toda análise realizada propomos um modelo de Jogo Educativo voltado ao ensino desta disciplina, baseado nas dificuldades relatadas pelos discentes e docentes do curso.

2. Modelagem de Dados em um curso de LCC

Boa parte dos alunos que cursam a disciplina de Modelagem de Dados pela primeira vez sentem um grande impacto no que diz respeito a compreensão das técnicas utilizadas pela Engenharia de Software para se elaborar um sistema, pois não estão habituados a pensar abstratamente. Criar modelos relacionais, definir classes, extrair e distinguir requisitos, e elaborar diagramas pode não parecer uma tarefa tão trivial em um primeiro momento. O aluno se perde em meio a tantos conceitos e informações. Como afirma Melo (2006), o aluno entende a teoria, mas não consegue sair do ponto zero, não consegue começar.

Para constatar estas informações foi realizado um estudo com alunos que cursam ou já cursaram a disciplina Modelagem de Dados, e com professores que já a ministraram no curso de Licenciatura em Computação. Sobre os alunos que já cursaram a disciplina foi relatado que nenhum deles reprovou a disciplina, entretanto boa parte foi submetida a um exercício de 'prova final' para poder alcançar aprovação na mesma; dentre as principais dificuldades ao estudar a Modelagem de Dados foram identificadas como as mais críticas, a ausência de materiais didáticos e o exercício prático da teoria visto em aula; as avaliações eram realizadas por meio de mini-testes, provas, apresentação de seminários e/ou elaboração de projetos de software que estimularam a prática do estudo; dentre os conteúdos estudados, os alunos afirmaram ter domínio no que trata de Projeto de

Software utilizando diagramas - principalmente o Diagrama *Use Case*; e, como sugestão ao aperfeiçoamento da prática pedagógica na disciplina, boa parte dos alunos optaram por mais aulas práticas e pela confecção de materiais que facilitem o estudo, ou seja, com fins mais didáticos (ver Figura 1).

Foi constatado ainda, que a principal dificuldade ao estudar a Modelagem de Dados está na escassa prática da teoria vista durante as aulas; os alunos preferem ser avaliados por meio de projetos de software ou seminários; e, como sugestão ao aperfeiçoamento da prática pedagógica na disciplina, boa parte dos alunos optou por mais aulas práticas (ver Figura 2).

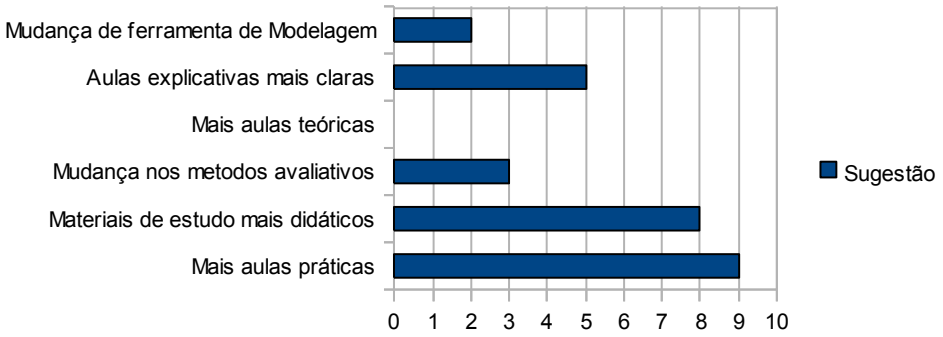


Figura 1 Sugestão para o ensino da disciplina

Sobre os alunos que estão cursando a disciplina Modelagem de Dados foi observado que nenhum deles está cursando-a por mais de uma vez; para aprender seu conteúdo boa parte dos alunos estudam sozinhos por meio de materiais disponibilizados pelo professor, para isto dedicam-se em média menos de cinco horas semanais; dentre os conteúdos estudados na disciplina foi denominado como mais fácil de aprender os conceitos iniciais de Orientação a Objetos e como mais difícil os conhecimentos a cerca da Engenharia de Requisitos.

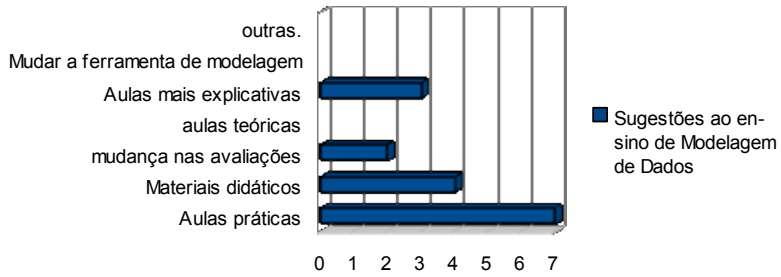


Figura 2 Sugestão para o aperfeiçoamento do ensino de Modelagem de Dados

Os professores responsáveis por esta disciplina possuem uma vasta experiência acadêmica na área, visto que já a ministraram por mais de cinco vezes. Eles relatam que as principais dificuldades que os alunos possuem ao estudar a Modelagem de Dados está no exercício da teoria aprendida durante as aulas, na compreensão dos conceitos da Engenharia de Requisitos e no pensar abstratamente. Este quesitos se manifestam durante o estudo de conteúdos, como: Conceitos de Análise OO (Orientada a Objetos), Projetos de Software utilizando diagramas e alguns conceitos da UML. Para estimulá-los e auxiliá-los na prática estudantil, os docentes disponibilizam materiais de pesquisa (a citar: referências bibliográficas, material da aula, entre outros.), realizam exercícios práticos e aplicam atividades avaliativas. O exercício de avaliação na disciplina se dá por meio de mini-testes, apresentação de seminários, elaboração de projeto de software e provas escritas. Segundo os professores, o êxito maior é na realização de provas escritas, onde os alunos conseguem melhor se expressar. Vale ressaltar ainda que para modelar os sistemas os alunos dispõem de aulas laboratoriais e usufruem da ferramenta de modelagem JUDE Community.

3. Trabalhos relacionados

Utilizados frequentemente para fins educativos, os jogos são dotados de características que estimulam e aguçam o aprendizado. Segundo Monsalve (2010, p. 32), em muitas áreas da tecnologia, os jogos são usados como uma ferramenta de ensino, mas isso é raro no campo da Modelagem de Dados. Dispomos de alguns jogos educativos referentes ao ensino da Engenharia de Software, mas nada focado no processo de Modelagem de Dados.

Costa (2008) relata uma experiência baseada no ensino de seis semestres letivos consecutivos da disciplina de Modelagem e Implementação de Software oferecida em um curso de graduação em Ciência da Computação. Todos os dados da pesquisa foram adquiridos em seminários e estudo de caso feitos pelos alunos, e através de métodos avaliativos. A proposta foi mostrar o desempenho da disciplina, o índice de maturidade dos alunos em relação a aprovação/reprovação e realizar uma análise sobre a prática de ensino adotada na disciplina.

Costa (2008 apud Soares, 2004) descreve uma experiência de ensino da disciplina Engenharia de Software em um curso de graduação em Ciência da Computação. Já Monsalve (2010, p. 06) apresenta uma abordagem de modelagem intencional apoiada em conceitos de transparência para a implementação do jogo educacional SimulES, um jogo para apoiar o ensino de Engenharia de Software que, além da visão dos modelos, utiliza estratégias de elicitação de requisitos para melhorar o entendimento do jogo e sua interação com os jogadores.

Velasquez (2009, p. 05) descreve uma possível metodologia de Engenharia de Software para o desenvolvimento de jogos de computador. Segundo ele, faltam mais cursos dedicados ao ramo, e também uma metodologia para ajudar no desenvolvimento dos jogos. Velasquez (2009, p. 76) propõe um modelo de software direcionado exclusivamente para a criação de jogos de computador.

Estes trabalhos relacionados apresentam grande preocupação em ensinar conceitos aliados a práticas da Engenharia de Software. Como mencionado anteriormente, infelizmente, são poucos os estudos direcionados ao campo da Modelagem de Dados.

4. Proposta ao ensino de Modelagem de Dados

Vislumbrando o perfil dos alunos que estudam a Modelagem de Dados, foi elaborada uma proposta/solução em auxílio as estratégias metodológicas que se referem ao ensino de Modelagem de Dados.

"Modelando" é um jogo para o aprimoramento dos conhecimentos obtidos a cerca do estudo da Modelagem de Dados. Sua vantagem em relação aos métodos tradicionais de ensino é que torna o processo divertido e competitivo, promovendo a integração entre os estudantes. Ele propõe ainda uma disputa entre os participantes para saber quem melhor e mais rapidamente realiza as tarefas que são propostas. Pode ser jogado em equipes de duas ou três pessoas, buscando melhorar o tempo e o aperfeiçoamento dos artefatos.

Na Figura 3 é ilustrado o processo que é seguido durante uma partida. Na *jogada de início* o jogador deverá responder três questionamentos sobre a Engenharia de Software; estas questões estarão em fichas distintas sorteadas aleatoriamente. Na *jogada de ações* é escolhido o cenário que irá ser tratado durante todo o jogo; o jogador deverá extrair os requisitos do cenário e, em seguida, criar um artefato (um diagrama use case, de classes, de atividades, outros). A medida que estas ações forem sendo executadas, o jogador poderá submeter o que gerou para o juiz do jogo, sabendo que cada erro encontrado no artefato repercutirá em uma penalidade (decremento de um ponto na pontuação geral) e cada acerto culminará em uma bonificação (três pontos a cada acerto). Na *jogada de validação* os artefatos criados são submetidos ao juiz: se aprovado, a equipe vence o jogo; se não, ele deverá corrigir seu erro para chegar na etapa final.

Durante a *Jogada de Ações* o jogador poderá solicitar quatro dicas ao juiz que o auxiliarão no cumprimento das tarefas. A função de *juiz* deverá ser exercida pelo professor ou monitor da disciplina, ou ainda por uma equipe de pessoas (máximo duas) que possuam experiência na área. Não é permitido o uso de materiais ou ferramentas que auxiliem na resolução das propostas.

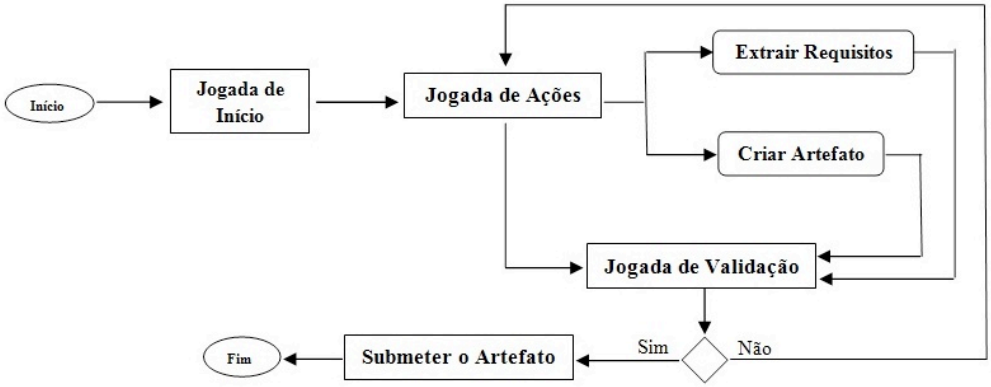


Figura 3 Processo de atividades no Jogo

Cada equipe tem no máximo quatro minutos para responder as indagações da *Jogada de Início;* dez minutos para extrair os artefatos e quinze minutos para gerar o artefato, na *Jogada de Ações*. Sabe-se para tanto que se o jogador antecipar a entrega das tarefas ele será bonificado ou penalizado na sua pontuação, assim como descrito anteriormente. Vence a equipe que obtiver mais pontos e em menos tempo.

5. Considerações Finais

Nesse trabalho foi proposto um exercício diferente, o jogo Modelando, para exercitar os conhecimentos a cerca de conteúdos referentes ao estudo da disciplina Modelagem de Dados, de modo a tornar esta prática mais prazerosa. O jogo é focado no auxílio ao aprendizado desta disciplina e colabora na avaliação do professor. Através de cada fase o jogador é estimulado a exercitar seu raciocínio em busca da solução mais apropriada ao problema em questão. Pretende-se dar continuidade a este estudo considerando as seguintes vias: *i)* aplicá-lo numa turma de Modelagem de Dados a fim de validar a contribuição e a eficiência do jogo no processo de aprendizagem dos estudantes; *ii)* ampliar o jogo, acrescentando novas tarefas e/ou regras para deixá-lo mais consistente; *iii)* abranger outros conteúdos específicos para possibilitar a estudantes mais alternativas durante seu estudo; *iv)* desenvolvê-lo como um software educacional.

Referências

Booch, Grady; Rumbaugh, James; Jacobson, Ivar. (2005) “UML: Guia do Usuário” , 2. ed. 6. reimp. Rio de Janeiro: Elsevier.

Costa, Heitor A. X.; Resende, Antonio M. P. de; Silveira, Fábio Fagundes. (2008) “Relato de experiência de ensino de modelagem e implementação de software em um Curso de Graduação em Ciências da Computação”, Agosto.

Guedes, Gilleanes T. A. (2006) “UML: Uma abordagem prática”, 2. ed. São Paulo: Novatec.

Marconi, Marina de Andrade; Lakatos, Eva Maria. (2006) “Técnicas de Pesquisa”, 6. ed. São Paulo: Atlas.

Melo, Ana Cristina. (2006) “Exercitando Modelagem em UML”,Rio de Janeiro: Brasport.

Monsalve, Elizabeth Suescún. (2010) “Construindo um Jogo Educacional com Modelagem Intencional Apoiado em Princípios de Transparência”, Dissertação (Mestrado em Informática) – Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro.

Velasquez, Carlos Eduardo Lé. (2009) “Modelo de Engenharia de Software para o Desenvolvimento de Jogos e Simulações Interactivas”, Dissertação de Mestrado – Universidade Fernando Pessoa, Porto.

Ambiente de Simulação Gráfica 3D para Ensino da Arquitetura de Processadores

Eduardo de Lucena Falcão[1], Erick Vagner Cabral de Lima Borges[1], Igor Lucena Peixoto Andrezza[1], Glauco Sousa e Silva[1], Kívio Grangeiro Wanderley[1], Berg Élisson Sampaio Cavalcante[1], Hamilton Soares da Silva[1]

[1]Departamento de Informática – Universidade Federal da Paraíba (UFPB)
Cidade Universitária 58.059-900 – João Pessoa – PB – Brasil

{eduardolfalcao, erickvagnerr, igorlpa90, glauco.sousa6, wkivio, bergelisson01}@gmail.com, hamilton@di.ufpb.br

Abstract. *In Computer Architecture, one of the challenges is getting the students to understand the workings of a processor correctly. To assist in this task, applications are used to simulate these architectures. From these barriers of education, is proposed the development of an application that visually simulates the instructions of an implementable and didactic architecture. Such instructions can be visualized, both with and without pipeline, which facilitates the comprehension of this concept, and of the operation of a processor while executing an instruction.*

Resumo. *Em arquitetura de computadores, um dos desafios é conseguir que os alunos compreendam o funcionamento de um processador corretamente. Para auxiliar esta tarefa são utilizadas aplicações que simulam essas arquiteturas. A partir dessas barreiras do ensino, se propôs desenvolver uma aplicação que simula visualmente as instruções de uma arquitetura didática e implementável. Tais instruções podem ser visualizadas, tanto com quanto sem pipeline, o que facilita a compreensão desse conceito, e do funcionamento de um processador enquanto realiza uma instrução.*

1. Introdução

A informática tem tido uma relevância cada vez maior nos dias de hoje no cenário educacional. Aplicações para o ensino facilitam a transmissão do conhecimento de professores para alunos através de métodos dinâmicos, massificando uma aula para múltiplos ramos sem haver grande esforço ou repetições desnecessárias.

O uso da computação gráfica tem se mostrado útil nos métodos de aprendizagem em diferentes áreas. O uso de CAD, sistemas computacionais que auxiliam na construção de desenhos técnicos, na geologia, engenharia e arquitetura, por exemplo, mostra o quanto uma interface gráfica com 3D facilita no desenvolvimento de um projeto, no seu entendimento e posterior ensino.

Facilitar a aprendizagem do correto funcionamento de um processador para os alunos e proporcionar uma visão comparativa sobre algumas possibilidades arquiteturais são os principais problemas encontrados no ensino de arquitetura de computadores. Um desafio ainda maior é tornar o conteúdo atrativo visualmente. Várias caixas e fios com letras pequenas tornam os circuitos um assunto de difícil entendimento a partir de uma simples leitura.

Os simuladores atuais que estão disponíveis para o ensino apresentam uma interface de usuário pouco elaborada e com poucos recursos operacionais, e mesmo considerando o uso de arquiteturas mais simples, é usual por parte dos alunos certa dificuldade de trabalhar com eles [Borges e Silva 2006]. A grande maioria destes simuladores exige dos alunos codificarem o sistema, ou seja, é necessário conhecimento em programação de linguagem de máquina ou rodam no DOS apenas exibindo os valores presentes nos registradores, o que torna a aprendizagem do funcionamento da arquitetura do processador uma tarefa longa e exigente.

Para auxílio a esta tarefa, está sendo desenvolvida uma aplicação com objetivo de prover maior motivação e interesse por parte dos alunos em participar das aulas de arquitetura de computadores. Recursos como animações gráficas tridimensionais estão sendo utilizados para facilitar o entendimento do funcionamento dos processadores, já que na prática não é possível ver o caminho percorrido pela informação processada. Este simulador não exige do aluno nenhum grau de programação para que a arquitetura seja demonstrada, Dessa forma é possível efetuar um estudo detalhado dos elementos presentes na arquitetura.

2. Trabalhos Relacionados

Existem muitas ferramentas que simulam processadores, como exemplos temos o Abacus, WinMIPS64 [Scott 2006], SATSim [Wolf e Wills 2000] e o GNUSim8085 [Sridhar 2002].

O Abacus (desenvolvido em Windows) e o GNUSim8085 (desenvolvido para plataformas Unix) simulam o microprocessador 8085 da Intel, o qual tem sido base do ensino de arquitetura de computadores, devido a sua simplicidade, seguindo o modelo de Von Neumann e sem possuir *pipeline*. Ambos apresentam os valores dos registradores no decorrer da execução das instruções, além de exibir o conteúdo presente na memória do processador.

O simulador WinMIPS64 simula a arquitetura do processador MIPS64, que utiliza o *pipeline.* Essa ferramenta possibilita a visualização das instruções passando pelos *pipelines* e os valores presentes nos registradores.

Para o ensino de arquiteturas *superescalares* há o SATSim que não se baseia em nenhum processador real, dessa maneira ele possibilita a configuração da quantidade de unidades funcionais por tipo de instrução, o número de elementos por *buffer* de reordenação, o número de instruções buscadas por ciclo e etc.

3. Fundamentação Teórica

3.1. Arquitetura de Processadores

É necessário conhecer a arquitetura interna de um processador para facilitar a compreensão de como ele executa programas. Em um processador existem vários circuitos chamados de registradores que funcionam como posições de memória onde o seu acesso é extremamente rápido.

Todos os processadores usam uma linguagem baseada em códigos numéricos na memória, onde cada código significa uma instrução no qual chamamos de linguagem de máquina. Os códigos numéricos são representados por siglas que são os nomes das

instruções, e os operandos dessas instruções são os registradores, valores existentes na memória e valores constantes.

Alguns processadores são baseados no modelo arquitetural de Von Neumann que considera uma unidade central de processamento (CPU) com apenas uma unidade funcional, podendo buscar apenas uma instrução por ciclo de execução como demonstrado na Figura 1.

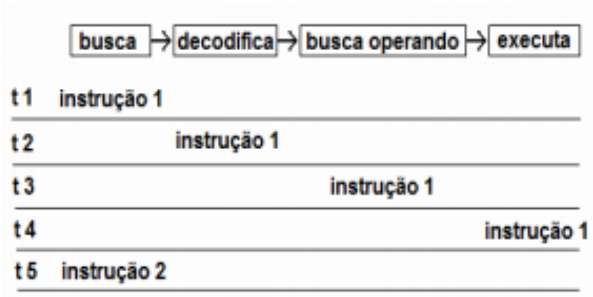


Figura 1 – Ilustração do modelo de Von Neumann

Para o ensino de arquitetura de computadores, é interessante demonstrar o conceito de *pipeline*, técnica de hardware que permite que a unidade central de processamento realize a busca de uma ou mais instruções além da próxima a ser executada. Estas instruções são colocadas em uma fila de memória dentro do processador (CPU) onde aguardam o momento de serem executadas, esta técnica permite que o tempo de processamento de uma instrução seja menor do que o tempo que seria gasto caso o *pipeline* não fosse utilizado. A estratégia do *pipeline* é demonstrada na Figura 2.

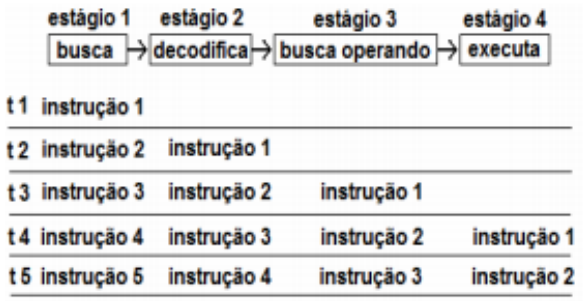


Figura 2 – Ilustração da Estratégia do pipeline

3.2. X3D

O X3D (*Extensible 3D*) é um padrão aberto adotado internacionalmente para disponibilização de conteúdos tridimensionais na Web [Brutzman e Daly 2007]. Ele permite descrever em um arquivo, formas e comportamentos de ambientes virtuais tridimensionais complexos e interativos.

A visualização destas cenas descritas em X3D é realizada através de *browsers* específicos que consistem em aplicações capazes de interpretar e processar os arquivos X3D, apresentando os modelos tridimensionais, animados ou não, e permitindo interações do usuário com os objetos. Os *browsers* X3D (também chamados de navegadores ou *players*) podem se apresentar como *plugins* ou *applets* em navegadores

Web como o *Mozilla Firefox* ou *Google Chrome*, ou como aplicações independentes [Web3D Consortium 2011].

Para a formação gráfica da cena são utilizados os nós X3D, que por sua vez compõem o grafo de cena. Cada nó é formado por uma seqüência específica de campos. Os nós descrevem as funcionalidades disponibilizadas pelo X3D, e os campos servem para especificar os atributos e características dos nós X3D utilizadas para descrição tridimensional do ambiente. A interatividade de uma cena X3D pode ser caracterizada por mudanças de posição, orientação, tamanho, cores ou outras características pertinentes aos campos do nó X3D, resultando em uma animação. Eventos e rotas são as funcionalidades do X3D que permitem adicionar tais comportamentos a objetos do ambiente virtual de forma simples. Para interações mais complexas são utilizados o nó *Script* e a API SAI (*Scene Access Interface*) que permite o controle do grafo da cena através de linguagens de programação [Web3D – *Part 2: Scene Access Interface*].

Para este trabalho, foi utilizado o *browser* Xj3D, a API SAI (*Scene Access Interface*) que permite manipulação do grafo de cena X3D, e a linguagem de programação Java, para programar toda interatividade do aplicativo.

4. Simulador desenvolvido

O simulador produzido demonstra o funcionamento do processador didático apresentado em [Tanenbaum 2006], que apesar de não ser comercial é implementável.

O nível de microarquitetura que será descrito tem a função de simular o nível ISA (*Instruction Set Architecture* – Arquitetura do Conjunto de Instruções). O Caminho de Dados utilizado para a simulação é a parte da CPU que contém a ULA, suas entradas e saídas. Ele contém vários registradores aos quais foram atribuídos nomes simbólicos como PC, SP, MDR e MBR, onde MBR foi dividido em MBR1 com 8 bits e MBR2 com 16 bits. É importante ressaltar que esses registradores são acessíveis apenas no nível de microarquitetura (pelo microprograma).

Com o objetivo de que o comprimento do caminho de execução seja menor, foram utilizados dois barramentos completos de entrada para a ULA (barramentos A e B) e um barramento de saída. Todos (ou ao menos a maioria) dos registradores devem ter acesso a ambos os barramentos de entrada. A vantagem de ter dois barramentos de entrada é que não é possível adicionar um registrador a outro registrador em um só ciclo.

O simulador é composto por uma interface gráfica que contém uma lista das instruções disponíveis, uma breve descrição do objetivo da instrução escolhida e um espaço reservado para o visualizador da simulação. Na lista fornecida, o usuário escolhe qual instrução do processador deseja visualizar como ilustrado na Figura 3.

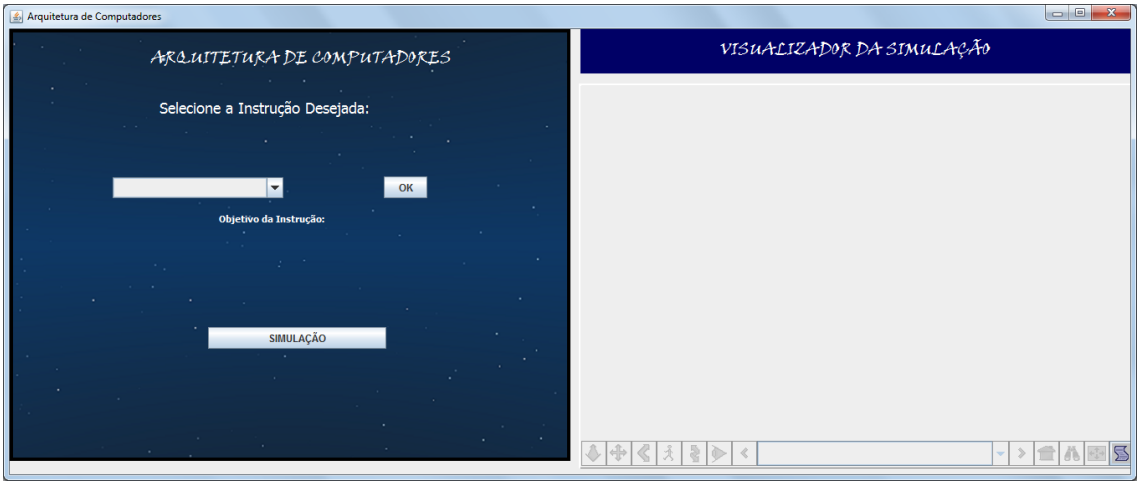


Figura 3 – Tela Inicial

Após a escolha da instrução o botão "OK" deve ser pressionado. Então uma breve descrição com o objetivo da instrução é mostrada logo após, como se observa na Figura 4.

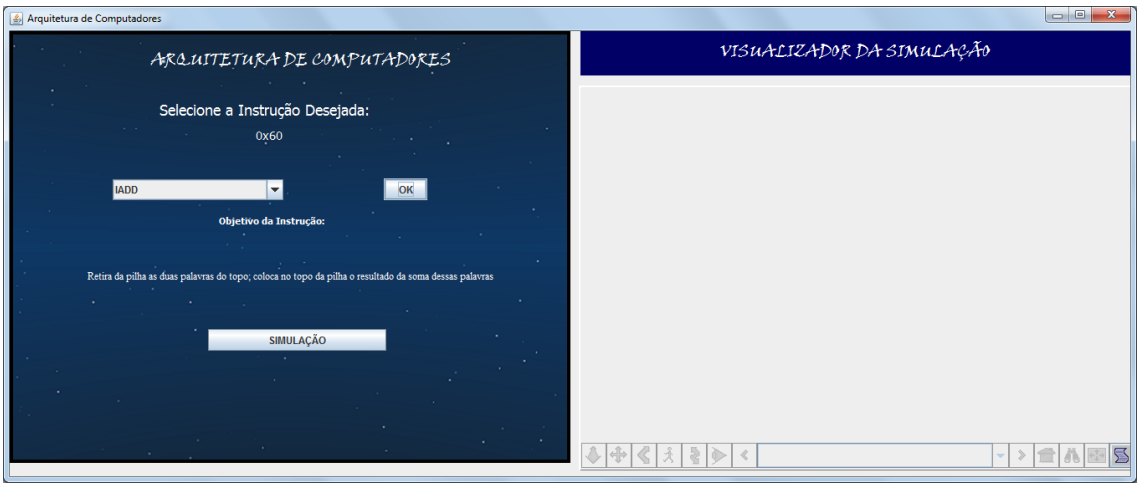


Figura 4 – Descrição do Objetivo da Instrução

Ao pressionar o botão "SIMULAÇÃO" ela é iniciada de acordo com a instrução escolhida. A posição das telas lado a lado facilita a compreensão, já que o usuário pode observar a descrição do objetivo e o mesmo em ação.



Figura 5 – Simulação da Instrução

A memória e todos os outros elementos do processador são representados em um contexto tridimensional. A simulação demonstra a seqüência de passos da execução de uma instrução na Figura 5, desde sua saída da memória até seu retorno, caso ocorra. A exibição de cada passo é feita através da interpolação das cores ou posição dos elementos que estiverem ativos. Cada instrução exibida contém um conjunto de micro-instruções que são demonstradas pela exibição passo-a-passo a cada *clock*, de acordo com o conceito de *pipeline* visto anteriormente, e são marcadas com uma cor diferente com o objetivo de demonstrar o *clock* e o *pipeline* como demonstrado na Figura 6. Para melhor visualização, a câmera acompanha o fluxo de passos da execução da instrução, focando nos blocos que estão sendo utilizados em determinado momento da simulação.



Figura 6 – Ilustração da Simulação do Pipeline

5. Conclusões e Trabalhos Futuros

O entendimento de como funciona uma determinada arquitetura é uma tarefa difícil de ser compreendida quando se estuda apenas a teoria. Isto evidencia a importância do uso de simuladores neste contexto.

A partir dessa constatação, o simulador apresentado neste trabalho visa facilitar o entendimento da arquitetura baseada no modelo de Von Neumann. A facilitação da compreensão ocorre devido à apresentação intuitiva das informações ao usuário pelo simulador. Os trabalhos correlatos encontrados na literatura apresentam apenas informações textuais dos valores de registradores, *flags* e memória, e algumas estatísticas.

O presente trabalho propõe que o fluxo de informações do processador seja ilustrado através de um ambiente 3D, que torna simples a visualização de informações sendo transferidas de um objeto como o barramento, registrador, ou ULA, para outros por intermédio da mudança de cores dos mesmos. O principal objetivo desse simulador é facilitar o entendimento dos alunos de cursos da Ciência da Computação e similares no estudo da arquitetura dos computadores. Outra característica positiva, que o diferencia em relação aos demais simuladores citados nesse trabalho, é a possibilidade de visualizar uma mesma instrução sendo executado em uma mesma arquitetura de processador, com e sem o uso do *pipeline,* facilitando de forma substancial a compreensão dessa importante evolução ocorrida na área.

Para trabalhos futuros propõe-se a realização de uma avaliação do uso do simulador por estudantes para coletar a impressão destes e verificar a facilidade de uso, e grau de aprendizado adquirido, para posterior aperfeiçoamento da aplicação. Propõe-se também a disponibilização via Web do aplicativo, promovendo o acesso dos alunos à ferramenta e possibilitando uma facilidade para a prática da Educação à Distância.

Referências

Borges, José Antonio S. Silva, Gabriel P. (2006) “NeanderWin - Um Simulador Didático para uma Arquitetura do Tipo Acumulador”.

Brutzman, D., Daly, L., (2007) “X3D: 3D Graphics for Web Authors”. Morgan Kaufmann Publishers.

Moreira, Angélica A., Martins, Carlos Augusto Paiva da Silva. (2009) “R2DSim: Simulador Didático do RISC Reconfigurável”

Scott, M. (2006) “WinMips64, version 1.5, School of Computing, Dublin City University, Ireland”, http://www.computing.dcu.ie/~mike/winmips64.html, 07 2009, 20.

Sridhar, Z, (2002) “GNUSim8085, versão 1.3”, http://gnusim8085.sourceforge.net/index.php/Main_Page, 06 2009, 23.

Tanenbaum, Andrew S. (2006) “Organização Estruturada de Computadores”, Editora Prentice-Hall.

Verona, André B., Martini, João A., Gonçalves, Tiago Lopes (2009) “SIMAEAC: Um Simulador Acadêmico para Ensino de Arquitetura de Computadores”.

Web3D Consortium, “X3D – FAQ (Frequented Asked Questions)”. Online: http://www.lsi.usp.br/~lsoares/x3d/faq.html. Acesso em: março/2011.

Web3D – Part 2: Scene Access Interface, “Extensible 3D (X3D). Part 2: Scene access interface (SAI). ISO/IEC 197752.2: 2009”. Online: http://www.web3d.org/x3d/specifications/ISOIECFDIS197752.2X3DSceneAccessInterface/index.html. Acesso em março/2010.

Wolff, M. e Wills, L. (2000) “SATSim: A Superscalar Architecture Trace Simulator Using Interactive Animation”, In: WCAE: Workshop On Computer Architecture Education, Vancouver, Canada.

Zelenovsky, Ricardo. (2002) “PC: um guia Prático de Hardware e Interfaceamento”, Editora MZ Produções Culturais.

IO Simulator: Um Simulador de Dispositivos de Entrada e Saída para Auxiliar o Ensino de Sistemas Operacionais

Thiago R. Medeiros[2], Cleyton C. Souza[5], Tiago D. N. de Sousa[4], Renê N. S. Gadelha[1], Edilson Leite da Silva[2], João B. Delfino Júnior[3]

[1]Centro de Informática – Universidade Federal de Pernambuco
(CIN-UFPE) Recife, PE – Brasil

[2]Departamento de Ciência da Computação – Centro Universitário de João Pessoa
(DCC – UNIPE) João Pessoa, PB – Brasil

[3]Departamento de Ciência da Informação – Universidade Federal da Paraíba
(DCI - UFPB) João Pessoa, PB – Brasil

[4]Departamento de Informática – Universidade Federal da Paraíba
(DI-UFPB) João Pessoa, PB – Brasil

[5]Departamento de Sistemas e Computação – Universidade Federal de Campina Grande
(DSC-UFCG) Campina Grande, PB – Brasil

{thiagorodrigues.medeiros,cleyton.caetano.souza,
renegadelha,edilson.leite, jbdj2006}@gmail.com, tiagodvneves@yahoo.com

Abstract. *Most of the didactic innovations added in classrooms is provided by application of computer technology. This paper presents IO Simulator, a simulator for teaching and learning the discipline of operating systems, characterized as indispensable in any course curriculum for Computer Science. The paper describes the architecture of the simulator and its use in academic environments and also presents the reports obtained in the experiment for validation of this proposal.*

Resumo. *Grande parte das inovações didáticas acrescidas às salas de aula é proporcionada pela informática aplicada ao contexto. Assim, este artigo apresenta o IO Simulator, um simulador educacional para ensino/aprendizagem da disciplina de Sistemas Operacionais, caracterizada como imprescindível em qualquer grade curricular do curso de Ciência da Computação. O artigo descreve a arquitetura do simulador e sua utilização nos ambientes acadêmicos, como também apresenta os relatos obtidos no experimento para validação desta proposta.*

1. Introdução

O ensino de disciplinas vem sendo aperfeiçoado através de novas técnicas, conceitos metodológicos, diretrizes, padrões e ferramentas, expandindo o conhecimento de forma acentuada. Com o advento da tecnologia, a prática do ensino foi assessorada por diversas ferramentas, auxiliando consideravelmente os professores e os alunos. Apesar de todas essas facilidades, o trabalho de ministrar aulas ainda enfrenta alguns problemas, principalmente em relação a disciplinas de base teórica aprofundada, como

Sistemas Operacionais (SO), uma disciplina referente ao curso de Ciência da Computação.

Na totalidade das instituições de ensino que oferecem o curso de Ciência da Computação, a disciplina de Sistemas Operacionais é oferecida na grade curricular para os alunos, sendo esta obrigatória segundo as diretrizes da Sociedade Brasileira de Computação e tendo a carga horária preenchida por aulas práticas, na sua maioria exercícios em sala de aula, e teóricas. Estas últimas são predominantes na carga horária e, de forma geral, sua demasia resulta no acumulo de dúvidas por parte dos alunos, ocorrendo também a desmotivação desses diante do imenso embasamento teórico. Desta maneira, evidencia-se a necessidade de intervir nesse processo com abordagens práticas em sala de aula, a fim de sanar ou pelo menos minimizar a dificuldade do aluno em absorver tais conhecimentos [Gadelha et al. 2010].

Uma alternativa para a solução do problema citado é a utilização de simuladores educacionais. Estas ferramentas se consolidam por possuírem características interativas de ensino, além disso, a simulação possibilita ao aluno maior absorção e entendimento do conteúdo em sala de aula [Jones e Newman 2001].

Este trabalho apresenta o IO Simulator, um ambiente de simulação que mostra de maneira prática os conceitos teóricos introduzidos pelo professor em sala de aula, exibindo detalhadamente os processos e peculiaridades do comportamento do sistema de gerenciamento de dispositivos de entrada e saída. A interatividade da ferramenta fornece aos alunos um ambiente agradável de aprendizado, detalhando o fluxo operacional deste módulo do SO por meio de animações dinâmicas.

O presente artigo é composto da seguinte forma: na Seção 2 são apresentados e discutidos alguns trabalhos relacionados a esta proposta; na Seção 3 é fornecido um referencial teórico sobre Sistemas Operacionais, com foco principal em Dispositivos de Entrada e Saída; na Seção 4 é relatada a arquitetura, o desenvolvimento e todas as funcionalidade do IO Simulator; na Seção 5 são apresentados e comentados os dados fornecidos pelos experimentos realizados em sala de aula com o uso da ferramenta aqui proposta. Por fim, são expostas as considerações finais na Seção 6, assim como os trabalhos futuros.

2. Trabalhos Relacionados

A construção de simuladores educacionais tem sido intensificada nos últimos anos, no entanto a introdução dessas ferramentas em salas de aula data mais de três décadas. Este fato motivou um levantamento bibliográfico com o propósito de encontrar objetos similares que pudessem incorporar novos conceitos à proposta apresentada neste trabalho. Após esta revisão, elencaram-se diversos trabalhos correlacionados, sendo estes posteriormente avaliados e justapostos com o IO Simulator. Posteriormente, foram incorporadas diversas características ao IO Simulator, a fim de beneficiar esta proposta, porém constataram-se algumas deficiências que existiam em alguns dos trabalhos similares, sendo estas apresentadas a seguir:

- Cruz (2007): proporciona uma simulação de todo um sistema computacional, desde a compilação até os artefatos de entrada e saída de dados (IO). A ferramenta possui código aberto, permitindo alteração de sua estrutura, além de já contemplar todos os módulos de um SO. A interface gráfica do sistema foi avaliada como de difícil entendimento por possuir diversos detalhes mal

estruturados no simulador. Identificou-se ainda a ausência de algumas informações pertinentes ao usuário, além da abordagem pouco complexa e detalhista do módulo IO.

- Isotani (2009): relata uma ferramenta para interação virtual por meio do gerenciamento de processos, tendo como principal oferta ao aluno a captura de processos em tempo real pelo usuário e a exibição destes em forma de gráficos. A ferramenta gera simulações com algoritmos específicos de escalonamento e apresenta os resultados através de interfaces gráficas amigáveis. O trabalho não aborda o gerenciamento dos dispositivos de entrada e saída, além de ser uma ferramenta voltada especificamente para a família de SOs da Microsoft[1].
- Reis (2009): apresenta um software para auxilio ao ensino de políticas de escalonamento de processos e de alocação de memória em SOs. A ferramenta está disponível na Internet e pode ser acessada sem necessidade de instalação de demais softwares. Sua interface é amigável e bem elaborada, porém o software foi avaliado como pouco compreensível na simulação do escalonamento de processos, fato este atribuído a didática pouco usual adotada. A ausência de tutoriais é perceptível, dificultando o entendimento dos alunos. Esses também não contemplam a gerência de dispositivos de entrada e saída, pois este simulador não incorpora esta área do SO.
- Maia (2001): é um simulador *standalone* para ensino de gerência de processos e de memória. Durante pesquisa e aplicação, foram encontrados diversos erros técnicos ao ser utilizado em sala de aula, sendo este o principal empecilho para a adoção do mesmo.

É possível, após apresentação e análise de alguns simuladores, perceber que o IO Simulator, trabalho aqui proposto e apresentado na Seção 4, introduz novos conceitos e aperfeiçoa alguns equívocos encontrados durante a análise de outros simuladores educacionais correlatos. Outro fator que enaltece o IO Simulator é a baixa quantidade de ferramentas educacionais sobre a supracitada área da disciplina de SO, proporcionando uma alternativa para alunos e professores neste processo de ensino/aprendizagem.

3. Sistema de Entrada e Saída (E/S)

O SO é um dos mais complexos sistemas computacionais, devido a sua grande quantidade de rotinas para suportar funcionalidades, a nível de hardware e software, com o intuito de facilitar o trabalho do usuário. As principais funções dos sistemas operacionais são oferecer uma interface para o usuário utilizar os dispositivos de hardware e atuar como gerenciador de recursos do equipamento, como a memória, o processador, os arquivos e os dispositivos [Tanenbaum 2003].

Como gerenciador, para [Maia 2001], os SOs controlam a utilização dos recursos fornecidos pelo *hardware* (processador, memória e dispositivos de E/S*)* e sua distribuição entre os diversos programas/aplicativos que os compartilham, possuindo assim uma divisão central em relação aos processos de gerenciamento. Entre os diversos tipos de gerência, este artigo aborda apenas a simulação da gerência de dispositivos de E/S, que é o propósito deste trabalho.

[1] Microsoft é uma empresa americana que produz diversas soluções em forma de produtos de software.

A gerência de dispositivos de E/S está estruturada em camadas, onde as de mais baixo nível ocultam a complexidade dos dispositivos para as camadas superiores, oferecendo uma interface simples e confiável para os usuários e suas aplicações [Tanenbaum 2003]. Com relação a gerencia desses dispositivos, o SO possui duas camadas: uma chamada subsistema de E/S que fornece um interface uniforme entre os dispositivos e os usuários e outra composta pelos *drivers* que tratam as peculiaridades de cada dispositivo.

O subsistema de E/S possui um conjunto de rotinas que permite a comunicação dos dispositivos que são ligados ao computador, possibilitando aos usuários realizar operações de E/S sem entender os detalhes do dispositivo que está usando [Machado 2007]. Dessa forma, as operações de E/S são realizadas através de chamadas a essas rotinas o que torna as aplicações independentes em relação aos diferentes dispositivos de E/S.

Já os *drivers* realizam a comunicação entre o subsistema de E/S e os seus dispositivos. Eles recebem comandos gerais para acessarem os dispositivos e os traduzem em comandos específicos que serão executados pelos controladores dos respectivos dispositivos. O *driver* é capaz de reconhecer características específicas dos dispositivos, como por exemplo, o número de registradores dos seus controladores [Machado 2007].

Outro componente muito importante envolvido nas operações de E/S são os controladores. Eles são responsáveis por manipular os dispositivos diretamente, fazendo a ligação entre estes com seus respectivos *driver*s. Os controladores possuem memórias e registradores próprios usados na execução de instruções enviadas pelo *driver* e , da mesma forma que os de discos, utilizam técnicas de *cache* para melhorar as operações de entrada e saída [Tanenbaum 2003].

Já os dispositivos de E/S, propriamente ditos, são responsáveis diretos pela comunicação entre o sistema computacional e os seus usuários e podem ser classificados como de entrada de dados (teclado e mouse), saída de dados (monitor e impressora) ou entrada e saída de dados (modem e disco). A transferência de dados feita pelos controladores dos dispositivos pode ser através de blocos ou caracteres de informação. A transferência em blocos acontece em dispositivos como discos ópticos e magnéticos e a transferência por caracteres acontece em dispositivos como impressoras e interface de rede [Machado 2007].

Na próxima seção deste trabalho é apresentado o IO Simulator, que permite visualizar a simulação da seqüência de interações entre os componentes envolvidos nas operações de entrada e saída, e o que acontece em cada um desses componentes.

4. Proposta

A disciplina de Sistemas Operacionais (SO) possui um conteúdo predominantemente teórico, diversificado e de difícil aprendizagem, pois envolve assuntos em grande parte abstratos e correlatos [Perez-Davilla 1995]. Essas características tornam possível lecionar cada conteúdo de SO sobre diferentes aspectos. Em relação ao sistema de entrada e saída (E/S) de um SO, é possível abordar o assunto detalhando cada componente presente nesse sistema, podendo ainda direcionar o foco para o fluxo de E/S onde esses componentes estão inseridos. Entretanto, sem o auxilio de ferramentas

que retenham a atenção do aluno, dificilmente esse conteúdo é compreendido da maneira esperada por um professor [Gadelha et. al. 2010].

Nesse contexto, foi desenvolvido o IO Simulator, um simulador que apresenta os conceitos do Sistema de E/S de maneira organizada e dinâmica, integrando em uma única ferramenta uma vasta carga teórica de conteúdo e prática interativa, sendo estes, de acordo com Maia (2001), fatores preponderantes para lecionar de forma satisfatória a disciplina de SO. O IO Simulator foi desenvolvido utilizando uma tecnologia para criação de animações, Flash, funcionando embutida em um navegador *web* e podendo ser acessada através do site da aplicação (http://wrco.ccsa.ufpb.br:8080/SimuladorSO/).

O projeto do IO Simulator teve como objetivo principal apoiar o processo de ensino/aprendizagem da disciplina de SO, através da simulação do sistema de E/S. Esse módulo da disciplina de SO possui certas peculiaridades, tornando-o um dos mais difíceis de serem lecionados. Essa dificuldade se deve principalmente pelo fato do aluno se deparar com um conteúdo repleto de abstrações, a nível de arquitetura do SO, além de uma grande quantidade de novos termos.

Através do IO Simulator, é possível interagir com um computador virtual e verificar como funciona o fluxo de E/S, observando onde cada componente do computador está inserido nesse fluxo, que é apresentado ao usuário através de uma animação. Ainda há a possibilidade de consultar os conceitos relacionados ao tema por meio do próprio simulador. Dessa forma, o conhecimento tende a ser absorvido de forma satisfatória, pois permite ao aluno relacionar a teoria com os dispositivos de entrada e saída de dados encontrados no seu cotidiano. Na Figura 1, é representada a tela inicial do simulador, sendo esta dividia em duas áreas.

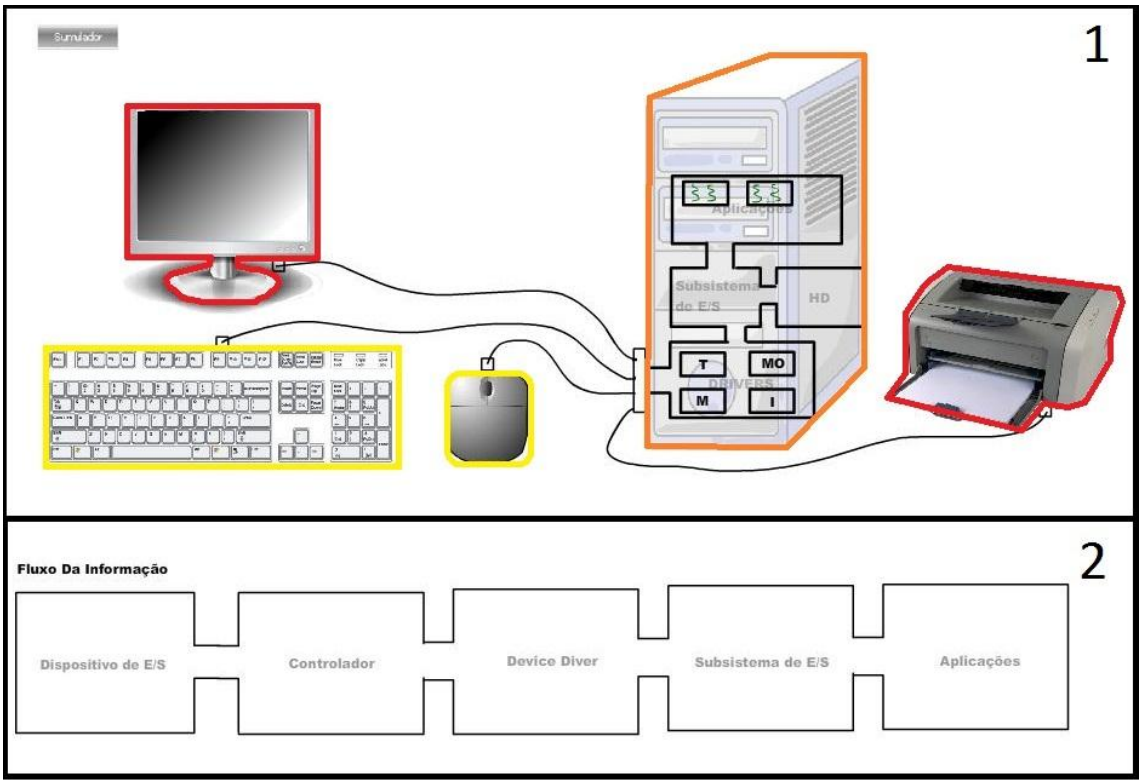


Figura 1: Tela principal do sistema

Na primeira área da Figura 1, é ilustrado um modelo de computador pessoal. Todos os componentes contidos nesta máquina participam da simulação. Esses componentes estão divididos em três conjuntos: os dispositivos de entrada (teclado e mouse), destacados pela cor amarela, os dispositivos de saída (impressora e monitor), destacados pela cor vermelha, e os dispositivos de entrada e saída (gabinete), que está destacado pela cor laranja. A interação com o simulador se inicia quando o usuário escolhe qual dispositivo quer acionar, fazendo isto por meio de cliques. Assim, a simulação tem inicio por meio da exibição do fluxo operacional de cada dispositivo e sua interação com os demais recursos do SO. Para facilitar a compreensão do fluxo, foram adicionados botões que controlam a animação e permitem ao usuário parar ou rever a animação. Durante a exibição do fluxo de E/S, aparecem textos explicativos no canto superior direito da área 1, contendo a descrição do processo que está sendo simulado e qual o papel de cada componente no fluxo.

Na área 2, é esboçada uma abstração do Sistema de E/S, representando uma visão lógica deste sistema, disponível em grande parte da literatura de SO e composta por cinco componentes: o dispositivo de E/S, o controlador, o *driver*, o subsistema de E/S e a aplicação [Machado 2007] [Tanenbaum 2003]. A animação acontece simultaneamente na área 1 e 2. Assim, o aluno consegue relacionar o imagem do computador pessoal com a estrutura lógica de E/S.

Por meio dessa divisão de fluxos operacionais para cada dispositivo, seja ele de entrada, saída ou entrada/saída, o simulador associa o fluxo exposto na área 1 com a estrutura lógica exposta na área 2. Também é perceptível no simulador a existência de ciclos no gerenciamento de E/S. Dessa forma, assim que o fluxo de saída tem inicio na imagem da área 1, a estrutura montada no fluxo de entrada é exposta na área 2, sendo esta transição acompanhada por textos auto-explicativos. Além disso, o usuário pode obter o significado de cada componente da imagem do computador pessoal ou também de cada parte da estrutura lógica apresentada.

Na Figura 2, é representada uma montagem contendo algumas partes da simulação disponível pelo IO Simulator. Para melhor exemplificar, é perceptível na Figura 2, o fluxo que este sinal percorre ao se usar o teclado, desde o pressionamento de uma tecla até a exibição da sua respectiva saída no monitor. Inicialmente, é apresentado o sinal analógico sendo transmitido no teclado (A). Em seguida, após passar pelo controlador do teclado, o sinal se desloca pelo fio (B). Ao chegar ao gabinete, o sinal é tratado respectivamente pelo *driver* do teclado, pelo subsistema de E/S e pelo processo responsável por iniciar o fluxo de saída(C). No fluxo de saída, após sair do gabinete, o sinal é transmitido pelo fio (D) e se encerra quando chega ao monitor.

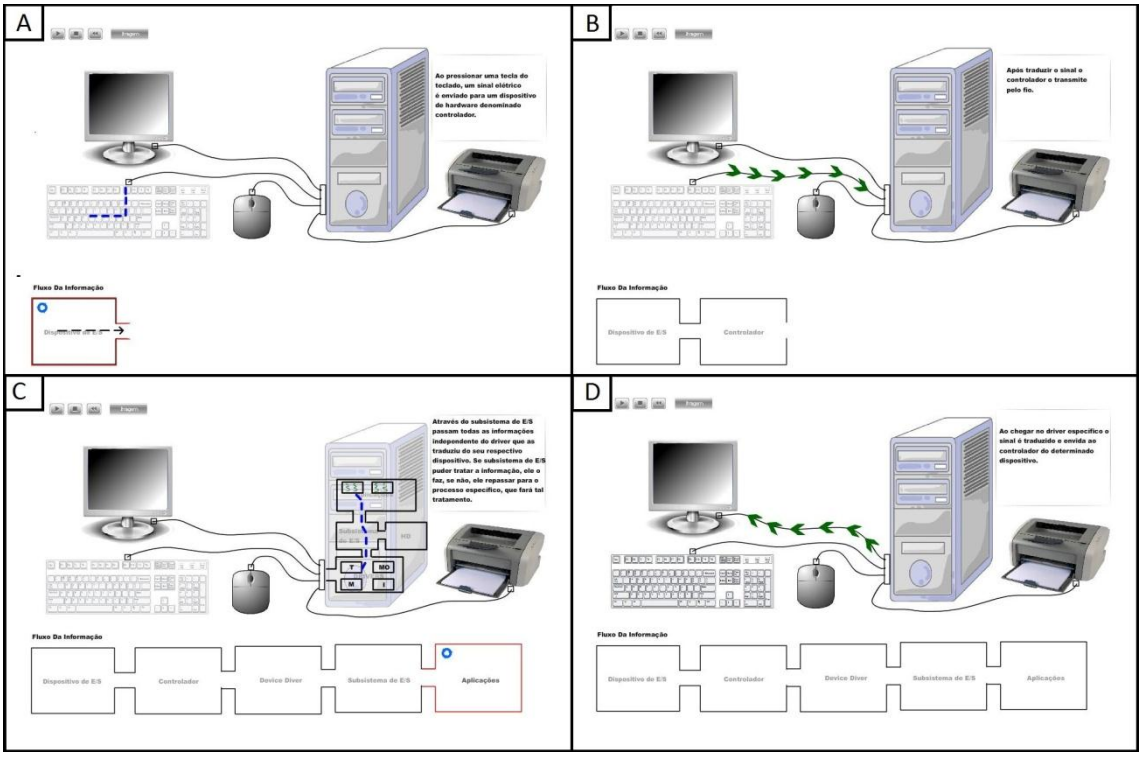


Figura 2: Detalhamento do Fluxo Operacional

Durante o processo apresentado na Figura 2, o aluno tem a possibilidade de interagir com qualquer componente presente na animação, permitindo consultar seu conceito ou entender sua função dentro do fluxo de E/S. Dessa forma, o IO Simulator agrega ao ensino de SOs a carga teórica e prática necessárias para um entendimento pleno no conteúdo da disciplina.

5. Resultados e Experimentos

Nesta seção, são apresentados os experimentos realizados para avaliar o trabalho proposto e os resultados obtidos do uso do IO Simulator em sala de aula. No experimento realizado com 2 turmas do curso de bacharelado em Ciência da Computação no Centro Universitário de João Pessoa (UNIPE), apresentou-se a ferramenta aos alunos e docentes, detalhando seu funcionamento como forma de demonstração prática da teoria que havia sido estudada em sala de aula na disciplina de SO.

Após a utilização da ferramenta, 38 (trinta e oito) alunos presentes responderam a um questionário dividido em três grupos de questões, sendo estes respectivamente: (1) a usabilidade da ferramenta; (2) o auxílio no processo de ensino/aprendizagem e (3) a plena abordagem do conteúdo programático. Na última questão, foi pedido aos alunos que fornecessem sugestões para a melhoria da ferramenta, caso eles tivessem alguma.

O resultado da análise das respostas obtidas para as três primeiras perguntas é exibido na Tabela 1. Para cada critério avaliado, nessa tabela é mostrada a porcentagem dos alunos que o avaliaram de forma positiva e negativa.

Tabela 1: Resultado dos experimentos em sala de aula

Critério avaliado	Avaliação Positiva	Avaliação Negativa
Usabilidade da ferramenta	**86,84 %**	**13,16 %**
Auxílio no processo de ensino/aprendizagem	**92,1 %**	**7,9 %**
Plena abordagem do conteúdo programático	**81,58 %**	**18,42 %**

Nota-se, a partir desse resultado, que cada critério foi avaliado positivamente por mais de 80% (oitenta por cento) dos alunos. Sendo assim, os benefícios trazidos pela ferramenta aos alunos foram significativos durante todo o andamento da disciplina.

É possível perceber, com esses resultados, que O IO Simulator agradou os alunos e atendeu às suas expectativas como uma ferramenta de apoio ao ensino de sistema de dispositivos de entrada e saída, o que demonstra a boa aceitação da ferramenta por parte tanto dos alunos quanto do professor presente durante o experimento. Para a maior parte desses alunos, a ferramenta cumpriu o seu papel didático, pois os auxiliou no entendimento do assunto explicado em sala de aula por mesclar a teoria à prática e por ser uma ferramenta de utilidade fácil e intuitiva.

A porcentagem de alunos que responderam as últimas perguntas do questionário, relativas às sugestões para a melhoria da ferramenta, foi de 39% (trinta e nove por cento). Dentre esses, alguns alunos sugeriram que fossem colocados mais dispositivos e funcionalidades de entrada e saída, criticaram a velocidade das animações e a qualidade das imagens, sendo todos estes detalhes incrementados na proposta.

Com os resultados desses experimentos, foi atestada a capacidade que IO Simulator tem de ser uma ferramenta de grande utilidade para o ensino de professores e para a aprendizagem de alunos, já que desperta o interesse desses por ser uma alternativa mais agradável visualmente ao ilustrar componentes e seus relacionamentos de acordo com os conceitos teóricos e por ser um ambiente que propicia de forma dinâmica a participação interativa dos alunos em todo esse processo de ensino e aprendizagem.

6. Conclusão

Soluções para auxilio educacional vêm sendo difundidas no meio acadêmico, motivando o desenvolvimento de novas propostas a fim de acrescentar maior qualidade ao ensino e aprendizagem. Diante deste panorama, apresentamos neste trabalho o IO Simulator, um simulador educacional para auxiliar respectivamente professores e alunos no ensino e aprendizagem da disciplina de SO.

Após a utilização do IO Simulator em sala de aula, atestou-se a capacidade desta ferramenta em agregar qualidade ao ensino de professores, facilitando a aprendizagem de alunos. Esta acontece principalmente pelo fato do simulador proposto neste trabalho ser uma alternativa mais agradável visualmente se comparado com outros trabalhos

similares, por ilustrar componentes e seus relacionamentos de acordo com os conceitos teóricos, sendo também um ambiente que propicia de forma dinâmica a participação interativa dos alunos em todo esse processo de aprendizagem.

Como pesquisa futura, propõe-se a integração deste simulador com propostas que acobertem as demais áreas contidas na disciplina de SO, originando na criação de uma ferramenta que contemple todas as necessidades dos alunos durante o andamento desta supracitada disciplina.

Referências

Cruz, E. H. M., Silva, V. P., Gonçalves, R. A. L. "Sistema Operacional Integrado Simulado: Módulo de Entrada e Saída". 14o Escola Regional de Informática-Paraná, 2007.

Gadelha, R. N.; Azevedo, R. R.; Oliveira, H. T. A.; Sousa, T. D. N.; Souza, C. C.; Silva, E. L. 2010. OS Simulator: Um Simulador de Sistemas de Arquivos para o Ensino/Aprendizagem de Sistemas Operacionais. 21º Simpósio Brasileiro de Informáica na Educação, 2010.

Isotani, S., Jorge, C. H., Junior, N. M. Q., Silva, F. S., Isotani, Seiji. "Uma Ferramenta de Apoio à Aprendizagem de Sistemas Operacionais". 29o Congresso da Sociedade Brasileira de Computação, 2009.

Jones, D. and Newman, A. 2001. "RCOS.java: a simulated operating system with animations". Proceedings of the Computer-Based Learning in Science Conference, Brno, Rep. Tcheca.

Machado, F. B. 2007 "Arquitetura de sistemas operacionais". Rio de Janeiro: LTC.

Maia, L. P. 2001. "SOsim: Simulador para o Ensino de Sistemas Operacionais". Dissertação de Mestrado, Núcleo de Computação Eletrônica, UFRJ, Rio de Janeiro.

Perez-Davilla, A. 1995. OS – bridge between academia and reality. ACM SIGCSE Bulletin,27(1):146–148.

Reis, F. P., Júnior, P. A. P., Costa, H. A. X. "TBC-SO/WEB: Um Software Educacional para o Ensino de Políticas de Escalonamento de Processos e de Alocação de Memória em Sistemas Operacionais". 20o Simpósio Brasileiro de Informática na Educação, 2009.

Tanenbaum, A. S. 2003. Sistemas Operacionais Modernos, 2ª. Edição, São Paulo: Prentice. Hall.

O uso do MERLOT por Alunos de Teoria da Computação para a Criação de Materiais de Ensino-Aprendizagem

Carolina Evaristo Scarton, Sandra Aluisio

Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo (USP)
Caixa Postal 668 – 13.566-590 – São Carlos – SP – Brazil

{carol,sandra}@icmc.usp.br

Abstract. *This article presents an approach to teaching and learning used by a course in formal language theory and the theory of computation offered by the Bachelor of Computer Science program at the Institute of Mathematics and Computer Science, University of São Paulo. With this approach students become the protagonists of the teaching-learning process using learning objects as objects of study for the formal tests and also as means of dissemination of the content of the course. Since the course is strongly theoretical, it presented itself as a good choice to change the traditional teaching approach used in previous classes, based on the widespread use of lectures and written tests, aiming at greater effectiveness in the teaching-learning process.*

Resumo. *Neste artigo é apresentada uma abordagem de ensino-aprendizagem para a disciplina de Teoria da Computação e Linguagens Formais, oferecida ao curso de Bacharelado em Ciências da Computação do Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo. Com esta abordagem, os alunos se tornaram protagonistas do processo ensino-aprendizagem, utilizando learning objects como objeto de divulgação de conteúdo e estudo para as provas formais. A disciplina, por ser de caráter teórico, apresentou-se como uma boa escolha para alterar a abordagem tradicional de ensino utilizada nos cursos anteriores, baseada fortemente em aulas teóricas e provas escritas, visando maior efetividade no processo ensino-aprendizagem.*

1. Introdução

A disciplina de graduação em Cursos de Bacharelado em Ciências da Computação chamada Teoria da Computação (TOC), ou também Teoria da Computação e Linguagens Formais e Autômatos, apresenta os fundamentos matemáticos da Ciência da Computação, analisa quais problemas podem ser computados por um dado modelo teórico de computação e, de uma maneira geral, responde quais são as capacidades e as limitações dos computadores. É geralmente dividida em três partes, que tratam de três grandes teorias: a Teoria das Linguagens Formais e dos Autômatos, a Teoria da Computabilidade e a Teoria da Complexidade (Hopcroft, Motwani, and Ullman, 2006; Sipser, 2005). A primeira parte trata das definições e propriedades de modelos matemáticos de computação que têm um papel fundamental em várias áreas da Computação, como o processamento de textos, compiladores, definição de linguagens de programação, dentre outras. A segunda parte é centralizada na Tese de Church-Turing e nas evidências dela: se um problema algorítmico não pode ser resolvido por uma Máquina de Turing (MT), então não existe nenhuma solução computável para ele. Como nem todos os problemas algorítmicos, que podem ser resolvidos em princípio, podem ser resolvidos na prática, pois os recursos computacionais requeridos (tempo

ou espaço) podem ser proibitivos, o estudo da complexidade computacional que classifica problemas de acordo com a dificuldade computacional é explorado na terceira parte.

Com esta ementa, a disciplina se constitui de conceitos, teorias e provas de teoremas, fazendo com que o modelo clássico de ensino, baseado em aulas teóricas e avaliação via provas escritas, seja o mais cogitado para ensinar grande parte da ementa, no qual o professor explica um determinado tópico, levando os alunos a pensar e a questionar o professor com as dúvidas suscitadas. Este modelo formalista no qual o aluno é um receptor não é totalmente adequado, dado que não alcança o potencial esperado dos alunos que precisam adquirir habilidades para poderem aplicar as teorias e conceitos aprendidos em áreas como implementação de linguagens de programação, reconhecimento de padrões, modelagem de sistemas, análise da complexidade de algoritmos, dentre outras.

Não é possível dizer com certeza qual o principal fator responsável por esta antiga tendência de ensino ainda ser empregada nos dias de hoje. Porém, fatores históricos podem afetar a decisão dos professores de continuar neste padrão, simplesmente segue-se o modelo com o qual foi educado o que dificulta a mudança de paradigma. Fiorentini (1995) apresenta as principais vertentes do ensino da Matemática no Brasil e, com uma rápida análise, pode-se observar que, atualmente, a tendência pedagógica aplicada no ensino superior brasileiro na área de exatas em geral (não somente na Matemática) ainda é predominantemente baseada no modelo formalista. Didaticamente centrado no professor, esta tendência tem como características salas de aulas com muitos alunos, docentes focados na "transmissão/repasse do conteúdo em aula expositiva" (Anastasiou, 1998) e alunos com aprendizagem passiva e com foco na memorização/reprodução dos conteúdos apresentados pelo professor (Fiorentini, 1995).

Apesar de ser uma área de conhecimento relativamente nova, o ensino da computação no Brasil segue também os antigos padrões que foram herdados de seus fundadores que vieram de áreas tradicionais como, por exemplo, a Matemática. Talvez as duas únicas diferenças claras do padrão formalista são: (a) uso predominante de tecnologias multimídias para apresentação do conteúdo (projetor multimídia nas aulas e sites para disponibilização do conteúdo); (b) uma abordagem de ensino que valoriza o trabalho prático dos alunos, aceitando soluções propostas por eles. Porém, especificamente para a disciplina de interesse neste trabalho, a Teoria da Computação, no ambiente da sala de aula, a estrutura continua centrada no professor, como mostra uma pesquisa das ementas dos cursos em Universidades Federais e Estaduais no Brasil, facilmente verificada pelas ementas expostas na Web. Contudo, os ingressantes das universidades, a partir do ano de 2008, são, predominantemente, jovens nascidos e criados na década de 90 que, possivelmente, conviveram com tecnologias multimídia desde sua infância. Estes jovens têm dificuldade de se enquadrar no padrão de ensino atual, pois o contato constante com a tecnologia, em que eles são muitas vezes protagonistas, faz o padrão de ensino centrado no professor perder o sentido. A partir de meados de 2015, o cenário se tornará mais crítico. Os ingressantes serão jovens nascidos no final da década de 90 e com infância vivida nos anos 2000 e estes, com certeza, conviveram (e viveram) com tecnologias multimídia durante toda a infância e adolescência. Ramal (2002) prevê que "na *cibercultura*, o computador vai substituir o professor" no quesito transmissão de conteúdos. Os alunos não serão mais "pessoas que não sabiam quase nada" às quais o professor apresentava "repetidamente conteúdos prontos". Anastasiou (2007) também comenta o fato de que os alunos atuais entram na "universidade trazendo novas e diferenciadas experiências em sua história de vida". Portanto, é evidente que uma nova geração de alunos está próxima e os professores deverão mudar o paradigma para a

efetividade do ensino. Cordiolli (2009) argumenta que pode-se promover os alunos a condição de sujeito participante da própria aprendizagem. Ele aponta que para isso é necessário, entre outros fatores, (a) estimular a colaboração entre os alunos, (b) estimular a aprendizagem através da pesquisa, (c) propor usos do conteúdo no mundo real e (d) compartilhar com o aluno a responsabilidade pela aprendizagem.

É fato que já existem algumas iniciativas na área de computação que podem ser citadas como pioneiras nessa evolução. É caso de tecnologias de ensino a distância (EAD) que vem sendo utilizadas em disciplinas do ensino superior de computação, não só para disponibilização de conteúdo, mas também, para interação professor-aluno e aluno-aluno através de fóruns (Ishitani, 2009; Segura et al. 2009). Outras iniciativas abordam o uso de blogs (Marques et al., 2010) e da Web 2.0 (Molin, 2010). No caso da disciplina abordada neste artigo, existem ferramentas que fornecem apoio gráfico ao ensino dos tópicos. Dentre elas é possível citar: Ginux Abstract Machine (Jukemura et al., 2005) que aborda autômatos finitos; Visual Automata Simulator[1] que aborda autômatos finitos e MT's; Sistema de Criação e Teste de Modelos Formais - SCTMF (Costa et al., 2008) que aborda autômatos finitos, MT's, gramáticas regulares, expressões regulares e autômatos de pilhas; o JFLAP[2] que aborda todos os tópicos do SCTMF (e possivelmente é a ferramenta mais utilizada); e, por fim, o LabLF (Aguiar e Oeiras, 2010) que engloba o JFLAP em um sistema de apoio a aprendizagem que permite a elaboração de listas de exercícios ou avaliação e a correção automática ou semiautomática das mesmas. Porém, iniciativas como esta nem sempre acompanham uma mudança de paradigma dentro da sala de aula, aparecendo somente como uma ferramenta de apoio extraclasse.

Neste trabalho, é apresentada uma abordagem experimental para a disciplina de Teoria da Computação e Linguagens Formais, ministrada no curso de Bacharelado em Ciências de Computação, do Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (ICMC/USP). Esta abordagem propõe uma quebra do paradigma clássico incentivando os alunos: (a) a elaborarem conteúdos de ensino-aprendizagem que estendem ou aprofundam os tópicos abordados em aula, (b) a disponibilizarem este conteúdo na forma de *learning objects* (LO's), via ferramenta MERLOT[3], (c) a utilizarem tecnologias multimídia para explicação do conteúdo, para a prática e exercícios propostos, e (d) a apresentarem o conteúdo para a turma em uma aula, utilizando somente os learning objects criados com os conteúdos multimídia propostos. Na Seção 2, é apresentada a ferramenta MERLOT e a definição de *learning objects* utilizada. Na Seção 3, é apresentada de forma detalhada a abordagem utilizada e a forma de avaliação dos grupos pelo docente. Na Seção 4, são apresentados alguns exemplos dos trabalhos dos alunos. Na Seção 5 é apresentado o resultado da avaliação da abordagem feita via questionários para os alunos. Por fim, na Seção 6, são apresentadas as considerações finais.

2. MERLOT e *Learning Objects*

O Learning Technology Standards Committee (IEEE LTSC, 2002) define um *learning object* (LO) como "qualquer entidade, digital ou não, que pode ser usada, reusada ou referenciada em aprendizado com suporte tecnológico". Para que possam ser indexados, recuperados e reusados em grandes repositórios são usados metadados com diversos tipos de informações.

[1] http://www.cs.usfca.edu/~jbovet/vas.html
[2] http://www.jflap.org/
[3] http://www.merlot.org/

MERLOT (Multimedia Educational Resource for Learning and Online Teaching) é um conjunto de recursos e experiências de ensino, ferramenta para a criação de sites (Merlot *Content Builder*), de livre acesso, sob a licença Creative Commons, e que permite a criação de *learning objects* (LO's) para serem reusados no ensino de diversas matérias, pois são classificados para fácil recuperação. Os recursos são criados por professores e estudantes de nível superior do mundo todo, que compartilham estes materiais e abordagens de ensino. No MERLOT, os LO's podem ser classificados em 18 categorias que descrevem o conteúdo destes LO's (como por exemplo, a categoria de *Quiz/Test* que contém LO's com qualquer dispositivo de avaliação que seja um teste ou um quiz). Após a criação do LO, é possível divulgar esse conteúdo na comunidade do MERLOT, preenchendo um formulário com informações sobre o conteúdo do LO e sobre o autor. Assim, o LO fica visível nas buscas realizadas no site do MERLOT.

Um fator decisivo da escolha do MERLOT foi a ferramenta *Content Builder* ter uma grande facilidade de uso: tanto no design do site quanto na inserção de conteúdo (texto, imagem ou vídeo), pois é fornecida ao usuário uma interface de edição clara e amigável. Portanto, em poucos minutos era possível que os alunos construíssem um LO simples com a ferramenta. Além disso, a escolha do ambiente MERLOT se justificou pelo acesso livre dos sites criados e foco na divulgação de conteúdo. Também, pretendeu-se colocar os alunos em contato com uma nova tendência: o ensino a distância através da Web, que este ambiente também viabiliza. O ambiente é divulgado via revista publicada quatro vezes ao ano (MERLOT Journal of Online Learning and Teaching[4]) e via conferências, sendo a de 2011 realizada em julho (2011 MERLOT/SLOAN Conference[5]). A manutenção do ambiente é apoiada por desenvolvedores atentos e atenciosos com problemas de usuários.

Adotamos no nosso curso a definição de LO's do projeto seminal da Cisco System em 1999 (Barritt, Lewis, and Wieseler) que foi usada para desenvolver a metodologia chamada de *Reusable Information Object (RIO) Strategy:* um conjunto de itens de conteúdo, prática e avaliação que são combinados com um único objetivo de aprendizagem. Esta definição foi preferida, pois indica especificamente o conteúdo a ser criado pelos alunos, direcionando a montagem do recurso de ensino-aprendizagem.

3. Uso do MERLOT na disciplina de Teoria da Computação e Linguagens Formais

No ICMC/USP a disciplina de Teoria da Computação e Linguagens Formais é oferecida aos alunos do 6° período (3° ano). Os 45 alunos que participaram deste estudo são, em sua maioria, ingressantes do ano de 2008[6]. Primeiramente, os alunos foram divididos em duplas ou trios. Depois, para cada grupo sorteou-se um tópico bem específico da disciplina de Teoria da Computação e Linguagens Formais – muitos dos quais também foram abordados pelo professor em aula expositiva - para ser seu tema de trabalho. O material disponível nos LO's deveria conter: (a) **Instrução**: definição/explicação do tema; exemplificação; relação do tema com problemas da vida real, quando fosse o caso. Se houvesse mais de uma definição, deveriam trazê-las, com referência completa dos livros texto. Uso de tecnologias avançadas de computação como jogos, simulações, programação, vídeos, etc. Lista de links (cursos na Internet, por exemplo) ou bibliografias de livros de onde os alunos poderão pesquisar mais

[4] http://jolt.merlot.org/
[5] http://taste.merlot.org/mic.html
[6] Os ingressantes do ano de 2008 são os alunos que cursaram a disciplina em seu período ideal.

sobre o tema (indicação pontual do capítulo e página a pesquisar); (b) **Prática**: exemplos resolvidos para fixar o conceito que não foram utilizados nem em sala nem nas listas de exercícios e (c) **Avaliação**: exercícios sobre o tema que não estivessem nas listas de exercícios fornecidas pela professora, com soluções na forma de testes objetivos na Web.

As apresentações dos temas ocorreram em aulas antes da prova em que estes conteúdos seriam abordados. Antes da apresentação, os alunos enviavam o endereço do *LO's* para avaliação do docente que corrigia o conteúdo e enviava as correções para os alunos, que deveriam corrigir o *LO* antes da apresentação. O tempo máximo para as apresentações era de 20 minutos. Além da avaliação anterior à apresentação, os *LO's* eram avaliados durante a apresentação, para verificar se os alunos tinham realizado as correções recomendadas e se os conceitos estavam bem consolidados. Os alunos foram, então, avaliados considerando três principais aspectos: (a) conteúdo dos *LO's* enviados ao docente antes da apresentação; (b) apresentação dos *LO's*; e (c) domínio do assunto, adequação da apresentação das várias partes dos LO's ao tempo de apresentação e motivação da classe durante a apresentação.

4. Exemplos de trabalhos com o MERLOT

Nesta seção são apresentados alguns exemplos de trabalhos com o MERLOT (Seção 4.1) e algumas iniciativas inovadoras (Seção 4.2).

4.1. Apresentação do Conteúdo utilizando o MERLOT

O MERLOT possui uma interface de fácil uso para a criação de sites com conteúdo (*LO's*). Porém, é possível construir tanto páginas bem estruturadas (com boa organização visual do conteúdo e uma estrutura de tópicos coerente) como páginas mal estruturadas, nas quais não se tem uma apresentação prévia dos tópicos, pouco ou nenhum uso de subtítulos, ou quadros para agrupar conteúdos relacionados. Nesta seção, apresentam-se alguns casos de sucesso na organização do material, via o ambiente *Content Builder* do MERLOT.

Na Figura 1 é apresentada a estrutura proposta por um grupo que utilizou uma abordagem semelhante aos sites comuns, com menus de navegação, que permitem fácil acesso aos tópicos apresentados.

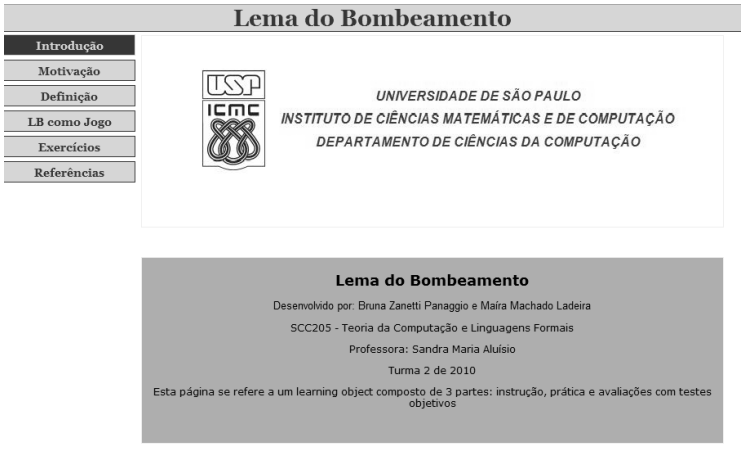


Figura 1: Exemplo de *learning object* seguindo modelo de site

Na Figura 2 apresenta-se outra estrutura de LO que segue um estilo semelhante ao utilizado em blogs, com uso de título, subtítulos e itens para facilitar a leitura



Figura 2: Exemplo de *learning object* seguindo estilo de blog

Outros grupos se utilizaram de cores para separar instrução, prática e avaliação e assim melhor apresentar o conteúdo.

4.2. Iniciativas inovadoras

Em geral, todos os alunos utilizaram alguma tecnologia alternativa para apresentação e/ou fixação do conteúdo. Porém, algumas iniciativas merecem ser citadas. Na Figura 3 são apresentadas duas cenas do filme *O Vidente* com o ator Nicolas Cage.



(a) (b)

Figura 3: (a) Cena do filme *O Vidente* com o ator Nicolas Cage "se dividindo" e (b) Cena do filme *O Vidente* com o ator Nicolas Cage procurando em vários locais (analogia com autômatos finitos não-determinísticos)

As cenas ilustradas na Figura 3 foram utilizadas pelos alunos para fazerem analogia com a capacidade de autômatos finitos não-determinísticos (AFND) dividirem seus recursos para explorarem todas as transições possíveis a partir de um nó, para um dado elemento do vocabulário

O mesmo grupo que trabalhou com AFND também criou uma paródia da música *Um minuto* do cantor D'Black para tratar do tema (Figura 4).

Um Minuto para AFND (paródia da música Um Minuto – D'Black)	
Por onde quer que eu vá vou chegar no final *Às vezes dividindo* *Os caminhos não são tão simples, mas eu vou seguir* *Viagem entre estados* *Uma estrada imaginaria que eu procuro dentro do meu Rangel* *Toda vez que eu for mudar eu posso escolher* *A distância não importa, eu posso me mover* *Pra qualquer lugar vizinho, vou pelo meu caminho* *Um minuto eh muito pouco pra poder entender* *Mas a gente treinou bem e sabe o que fazer* *E no final vocês vão aprender*	*Por onde quer que eu vá num AFND* *É mais intuitivo* *É mais fácil de compreender, vocês vão perceber* *O não determinismo é uma forma de pensar sem ter que definir* *A estrada a seguir* *Toda vez que eu for mudar eu posso escolher* *A distância não importa, eu posso me mover* *Pra qualquer lugar vizinho, vou pelo meu caminho* *Um minuto eh muito pouco pra poder entender* *Mas a gente treinou bem e sabe o que fazer* *E no final vocês vão aprender*

Figura 4: *Um Minuto para AFND* paródia da música *Um minuto - D'Black*

O grupo que trabalhou com Gramáticas Formais criou um jogo da memória para fixação do conteúdo, com a definição de cada uma delas (Figura 5).



Figura 5: Jogo da Memória sobre Gramáticas Formais

Outros grupos também criaram tecnologias ou aproveitaram-se de tecnologias existentes (como é o caso do Jogo da Vida no contexto de Máquinas de Turing) para apresentação/fixação do conteúdo e o JFLAP, via uso de jogo de dois participantes para provas de lema do bombeamento.

Uma iniciativa que muitos grupos utilizaram fui o uso de quiz nos exercícios propostos. Já um fato que ocorreu durante uma das apresentações e que merece ser citado é que, quando um dos grupos foi apresentar o quiz, os integrantes desafiaram os alunos a participar, concorrendo a um chocolate caso conseguissem uma boa pontuação.

5. Avaliação do uso do MERLOT pelos alunos

Ao final da disciplina, realizou-se uma pesquisa com os alunos, buscando saber qual foi a experiência deles em relação ao uso do MERLOT. A avaliação teve dois focos: (1) uso do MERLOT como ferramenta de divulgação de conteúdo; e (2) uso do MERLOT como ferramenta de estudo. A seguir são apresentadas as questões para (1) e (2):

1. MERLOT como ferramenta de divulgação:
    a. Como você considera sua experiência com a interface do MERLOT para a criação do *LO*? (muito baixa) 0 1 2 3 4 5 6 7 8 9 10 (muito alta)
    b. Você teve alguma dificuldade para usar o ambiente do MERLOT?
    c. O que você acha do uso do *LO* para divulgação do conteúdo de seu trabalho? (muito ruim) 0 1 2 3 4 5 6 7 8 9 10 (muito bom)
    d. O que você acha do uso de *LO's* para a apresentação do trabalho em sala de aula? (muito ruim) 0 1 2 3 4 5 6 7 8 9 10 (muito bom)
    e. Você acha que se pudesse usar outra tecnologia (Power Point, por exemplo) para apresentação do trabalho, este seria melhor apresentado?
2. MERLOT como ferramenta de estudo:
    a. Você utilizou os *LO's* dos outros alunos para estudar para a prova?

b. Você considera que os *LO's* foram bons materiais de apoio para seu estudo fora da sala de aula?

c. Na média, qual era a qualidade do conteúdo dos *LO's* apresentados em sala? (muito ruim) 0 1 2 3 4 5 6 7 8 9 10 (muito bom)

d. Na média, qual era a qualidade visual dos *LO's* apresentados em sala? (muito ruim) 0 1 2 3 4 5 6 7 8 9 10 (muito bom)

e. Você gostaria que fosse empregada tecnologia semelhante em outras disciplinas?

Na Figura 6 são apresentados os resultados para as perguntas relacionadas a (1) (6a) e a (2) (6b). Para as questões com respostas quantitativas ordinais (1a, 1c, 1d, 2c e 2d) foram calculadas as frequências nos intervalos [0-3], [4-7] e [8-10] (em porcentagem), a mediana, e o coeficiente de variação quartil (CVQ)[7] (em porcentagem). Para as demais variáveis (1b, 1e, 2a, 2b e 2e) foram gerados gráficos com a porcentagem de ocorrência de "sim" ou "não". Vale comentar que dos 45 alunos que concluíram o curso, 43 responderam ao questionário.

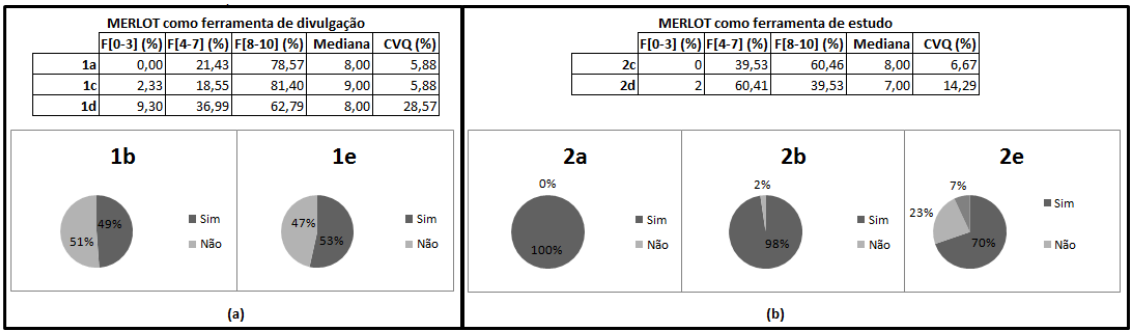


Figura 6: Resultados da avaliação do MERLOT: (a) como ferramenta de divulgação e (b) como ferramenta de estudo

Na Figura 6(a) é possível observar que avaliação dos alunos foi positiva no uso do MERLOT para divulgação do conteúdo (as questões 1a e 1c obtiveram medianas superiores a 7, a frequência no intervalo [8-10] foi superior a 75% e CVQ igual 5,88, em ambos). Porém, a maioria dos alunos (53% na questão 1e) preferiria utilizar outra tecnologia para apresentação (somente 62,79% dos alunos deram notas no intervalo [8-10] para a questão 1d – uso do MERLOT para apresentação – com CVQ de 28,57%) e 51% dos alunos teve dificuldade no uso do MERLOT (questão 1b). Já na Figura 6(b) é possível observar que os alunos avaliaram positivamente o uso do MERLOT como ferramenta de estudo (questões 2a, 2b e 2e – a maioria dos alunos respondeu "sim" a essas questões). Quanto a qualidade do conteúdo dos LO's (questão 2c) e a qualidade visual dos LO's (questão 2d) pouco se pode afirmar, pois apesar dos valores da mediana ser relativamente alto (8 e 7 respectivamente), somente 60,46% dos alunos responderam a questão 2c com um valor no intervalo [8-10] e somente 39,53% dos alunos responderam a questão 2d com um valor no intervalo [8-10] (o valor de CVQ também foi alto nesse último caso – 14,29%). Estes resultados podem indicar que talvez seja necessária uma avaliação mais efetiva dos LO's pelo professor, incentivando os alunos a efetuar todas as correções pedidas e que o uso de outra tecnologia para a apresentação em sala deve ser considerado.

[7] Coeficiente de variação/dispersão quartil é uma medida estatística de dispersão para fazer comparações entre os elementos de um conjunto de dados ou entre conjuntos de dados. É utilizada em variáveis quantitativas ordinais, para as quais não se pode calcular medidas como média ou desvio padrão.

Para reforçar os resultados obtidos, alguns comentários dos alunos também merecem ser citados:

"O uso do MERLOT foi importante tanto para pesquisa como para aprendizado. Um método de estudo iterativo e dinâmico auxilia o processo de fixação do conteúdo na memória."

"A tentativa de inovação é boa e bem-vinda. Entretanto minha opinião é de que as apresentações não foram muito produtivas para os que assistem. Talvez seria necessário reformular a dinâmica dos seminários para algo mais fácil de se acompanhar."

"... Creio que com o professor checando o conteúdo deles, é uma ferramenta de grande ajuda para os alunos. Somente para a apresentação em sala não me agradou muito por ser necessário ter de abaixar e levantar a página para mostrar o conteúdo."

"O uso do MERLOT, na minha opinião, foi positivo no sentido de que os alunos que os desenvolveram tiveram de compreender o assunto, e o repassaram de foram que a compreensão do leitor fosse facilitada. Como a matéria é muito teórica, esta disciplina necessitava de algo como isto."

Como pode-se observar pelos comentários dos alunos, eles gostaram da experiência e alguns deles fizeram boas observações, críticas ou sugestões sobre o uso do MERLOT. Também é possível observar que, alguns comentários traduzem os resultados das questões apresentadas anteriormente, como é o caso do descontentamento dos alunos no uso do MERLOT para a apresentação.

6. Considerações Finais

Este trabalho apresentou uma experiência para trazer os alunos a participar do processo de ensino-aprendizagem de forma ativa e criativa. A escolha de uma disciplina bastante teórica como a TOC mostrou que com uma tecnologia adequada e abordagem colaborativa é possível motivar os alunos a construírem o próprio conhecimento, com a facilitação de um professor e monitor assistente de ensino. Vale destacar que este trabalho teve também como objetivo divulgar abordagens e ambientes computacionais como os repositórios de LO's que privilegiam novas formas de ensinar e aprender, trazendo os alunos para participar do processo de ensino-aprendizagem de forma ativa e criativa. No nosso conhecimento, a abordagem é inovadora, pois embora se encaixe nas estratégias de ensino propostas no trabalho de Anastasiou (2007) de seminários e ensino com pesquisa, tem um diferencial de dividir o seminário em instrução, prática e avaliação, e incentivar o ensino com a pesquisa, reforçando a exploração de tecnologias multimídia para apresentação do conteúdo, com vistas ao reuso. Também, permitiu um ciclo de discussão do material com o professor da disciplina como uma forma de corrigir conceitos inválidos ou incompletos. Este ciclo foi usado em um cenário de motivação, pois visava à correção do material a ser ensinado pelo grupo aos alunos. Como trabalhos futuros, pretendemos indexar os LO's criados, e propor uma forma de incrementar a lista de tópicos de pesquisa com assuntos cada vez mais avançados, dado que os LO's dos anos anteriores podem servir de base para a construção de novo conhecimento. Uma possibilidade pode ser a inclusão de artigos científicos da própria área de TOC como área de pesquisa como temas de pesquisa, uma das estratégias usadas por Goyal and Sachdeva (2009) para a criação de um curso de TOC mais vivo e ativo.

Como esta experiência também conseguiu divulgar abordagens e ambientes computacionais como os repositórios de LO's, novos repositórios e ambientes devem ser

pesquisados e comparados, como uma alternativa ao MERLOT, de forma que se consiga o melhor e mais fácil ambiente de construção de conhecimento via LO's.

Referências

Aguiar, R. da S. e Oeiras, J. Y. Y. (2010) "Laboratório de Linguagens Formais: LabLF". Revista Brasileira de Informática na Educação, v. 18, n. 1, p. 106-118.

Anastasiou, L. das G. C. (2007) "Processos de Ensinagem na Universidade: Pressupostos para as Estratégias de Trabalho em Aula". Univille, 3ª edição, Santa Catarina, SC, 144p.

Barritt, C., Lewis, D., Wieseler, W. (1999). Cisco Systems Reusable Information Object Strategy. Definition, Creation Overview, and Guidelines Version 3.0. [Internet] Disponível em: http://www.cisco.com/warp/public/779/ibs/solutions/learning/whitepapers/el_cisco_rio.pdf Acesso em: 30/03/2011.

Cordiolli, M. (2009) "Formas de gestão do trabalho pedagógico, em sala de aula, na educação superior". A Casa de Astérion, Curitiba, PR, 28p.

Costa, Y. M. G., Meneses, R. C., Uber, F. R. (2008) Uma Ferramenta para Auxílio Didático no Ensino de Teoria da Computação. Workshop sobre Educação em Computação (WEI). Anais do XXVIII Congresso da Sociedade Brasileira de Computação, Belém, PA, p. 208-217.

Fiorentini, D. (1995) "Alguns modos de ver e conceber o ensino de matemática no Brasil". Zetetiké, Campinas, SP, Ano 3, n.4, p. 1-37.

Goyal, M. and Sachdeva, S. (2009) "Enhancing Theory of Computation Teaching Through Integration with other Courses". International Journal of Recent Trends in Engineering, Vol. 1, No. 2, p. 137-140.

Hopcroft, J. E., Motwani, R., Ullman, J.D. (2006) Introduction to Automata Theory, Languages, and Computation, 3rd Edition, Addison Wesley.

Ishitani, L. (2009) "Aprendizagem colaborativa no ensino presencial: TICs ajudam ou atrapalham?". Workshop sobre Educação em Computação (WEI). Anais do XXIX CSBC, Bento Gonçalves, RS, p. 419-428.

IEEE LTSC (2002). "Learning Technology Standards Committee". [Internet] Disponível em: http://ltsc.ieee.org/wg12/ Acesso em: 30/03/2011.

Jukemura, A. S., Hugo, A. D., Uchoa. J. Q. (2005) GAM – Um Simulador para Auxiliar o Ensino de Teoria da Computação. Workshop sobre Educação em Computação (WEI). Anais do XXV Congresso da Sociedade Brasileira de Computação, São Leopoldo, RS, p. 2432-2443.

Marques, A. de M., Pimentel, M. e Siqueira, S. (2010) "Dinâmicas Educacionais com o Uso de Blogs: Requisitos a partir de Experiências". Workshop sobre Educação em Computação (WEI). Anais do XXX CSBC, Belo Horizonte, MG, p. 1177--1186.

Molin, S. I. L. (2010) "Projeto de Aprendizagem e as Ferramentas da WEB 2.0: Uma experiência na sala de aula". Workshop sobre Educação em Computação (WEI). Anais do XXX CSBC, Belo Horizonte, MG, p. 1344--1353.

Ramal, A. C. (2002) "Educação na cibercultura hipertextualidade, leitura, escrita e aprendizagem". Artmed, Porto Alegre, RS, 268 p.

Segura, R. de A., Araújo Jr, C. F. e Silveira, I. F. (2009) "Aprendizagem Baseada em Problemas Apoiada por Ambientes Virtuais: um Estudo de Caso em Banco de Dados". Workshop sobre Educação em Computação (WEI). Anais do XXIX CSBC, Bento Gonçalves, RS, p. 447--450.

Sipser, M. (2005) "Introduction to the Theory of Computation". Course Technology, 2nd Edition, 431p.

Programa de Residência em Software

Luiz Ricardo Begosso[1], Luiz Carlos Begosso[2,3], Alex Sandro Romeo Souza Poletto[2], Douglas Sanches da Cunha[1], Fernando Cesar de Lima[2]

[1]Centro de Pesquisas em Informática - Fundação Educacional Município Assis (FEMA)
Av. Getúlio Vargas, 1200 - CEP 19.807-634 – Assis - SP – Brasil

[2]Coordenadoria de Informática - Instituto Municipal Ensino Superior Assis (IMESA)
Av. Getúlio Vargas, 1200 - CEP 19.807-634 – Assis - SP – Brasil

[3]Faculdade de Tecnologia de Ourinhos – FATEC
Av. Vitalina Marcusso, 1400 - CEP 19.910-206 – Ourinhos - SP – Brasil

{begosso,lbegosso,apoletto,douglas.cunha}@femanet.com.br,
limasoftware@hotmail.com

Abstract. *Software industry has needed more and more professionals with experience in the aspects of software development process. This paper aims to describe the project of implementation of Software Residency at an undergraduation course, allowing students to participate in the development of software projects for real companies, experiencing teamwork personal relations and strengthening theoretical aspects of students.*

Resumo. *A indústria de software tem necessitado cada vez mais de profissionais com experiência nos aspectos relacionados ao processo de desenvolvimento de software. Este trabalho tem o objetivo de descrever o projeto de implantação do Programa de Residência em Software em um ambiente de graduação, permitindo aos estudantes a participação no desenvolvimento de projetos de software para empresas reais, vivenciando relações interpessoais de trabalho em equipes e reforçando os aspectos teóricos na formação dos alunos.*

1. Introdução

Preparar e qualificar o egresso de cursos de computação, para atender as necessidades do mercado de trabalho, é um desafio cada vez maior para as instituições de ensino superior. Essa preparação diz respeito a municiar, motivar e qualificar o futuro profissional com as boas práticas de análise e desenvolvimento de software.

Geralmente, as aulas práticas oferecidas nos cursos de graduação em computação, não conseguem traduzir um ambiente que aproxime ao máximo do dia a dia de uma empresa de produção de software. Por mais que o docente empenhe em usar laboratórios de informática com ferramentas propícias às aulas práticas, exemplos de softwares desenvolvidos, estes não são suficientes para reproduzir o referido ambiente. Isso faz com que os egressos dos cursos de computação saiam para o mercado de trabalho sem a maturidade mínima que a indústria gostaria que tivessem.

O presente trabalho tem por objetivo descrever o processo de implementação de um programa de residência de software para cursos de graduação de Análise e

Desenvolvimento de Sistemas e Ciência da Computação. Os alunos, alvos dessa pesquisa, são estudantes da Fundação Educacional do Município de Assis – FEMA. A pesquisa resulta de uma ação que trata do programa denominado de "Programa de Residência em Software". Esse programa proporciona a imersão tecnológica de alunos das últimas séries dos cursos de computação e ex-alunos, recém-graduados, para que recebam treinamento intensivo baseado no desenvolvimento de projetos reais nas áreas de análise e desenvolvimento de software. O programa também objetiva, a partir de uma abordagem teórico-metodológica, fornecer aos estudantes: uma experiência de desenvolvimento de software para um cliente real; vivenciar as relações pessoais na equipe de desenvolvimento; reforçar aspectos teóricos na formação desses estudantes; e facilitar a inclusão desses novos profissionais no mercado de trabalho.

Este artigo está organizado da seguinte forma: na Seção 2 é descrita a Residência; na Seção 3 são apresentados os trabalhos correlatos a presente proposta; na Seção 4 é descrito o "Programa de Residência em Software", foco principal desse trabalho e, finalmente, na 5ª seção são relatados os resultados parciais obtidos.

2. Residência

No Brasil, a primeira iniciativa para a complementação da formação do estudante foi a modalidade Estágio Supervisionado utilizada nos cursos de formação de professores com o objetivo de propiciar ao futuro professor uma aproximação e vivência prática da sala de aula.

Outra iniciativa importante promovida no final da década de 70, instituiu a modalidade de residência médica caracterizando-a como curso de pós-graduação destinada à capacitação profissional de estudantes de medicina.

De acordo com Peckham e Batson (2004), a universidade encontra dificuldades para fornecer uma experiência balanceada entre a teoria aprendida e a prática da modelagem e produção de software. É nesta linha que o presente trabalho está circunscrito. A proposta aqui apresentada alinha aspectos teóricos da formação do estudante com atividades práticas, empregadas pelas empresas produtoras de software.

3. Trabalhos Correlatos

Way (2005) apresentou um trabalho que consistia em organizar estudantes de graduação do curso de Engenharia de Software, num ambiente que simulava uma empresa da área de desenvolvimento de software. A proposta de Way tinha por objetivo delegar responsabilidades aos alunos nas questões de condução e tomada de decisão nas etapas do projeto, desenvolvimento e distribuição do produto elaborado.

A principal diferença entre o trabalho de Way (2005) e o "Programa de Residência em Software" é que, este último foi conduzido em estreita relação com uma empresa real de desenvolvimento de software, enfatizando a prática de mercado.

Em outra pesquisa relacionada ao presente trabalho, Sampaio et al (2005) propôs um programa de residência em software denominado Programa de Teste de Software. O público alvo, participante do programa, foi composto por estudantes de último ano ou recém-formados em cursos da área de computação. Os estudantes obtiveram treinamento intensivo durante 12 meses, em diversas áreas da produção de software,

porém o foco principal foi a formação de profissionais para atuarem na área de testes de software.

Diferentemente do Programa de Teste de Software, o Programa de Residência de Software, objeto desse trabalho, manteve esforços para a formação dos estudantes na área de desenvolvimento e implementação de sistemas.

Um interessante estudo foi apresentado por Shackleford (2000), no qual um grupo de alunos, de cursos de graduação da área de Tecnologia da Informação (TI), foi treinado para promover manutenções, instalações, reparos e configurações de computadores e de infraestrutura de redes da Universidade de West Indies. De acordo com Shackleford (2000), 93% dos alunos que participaram do programa consideraram-se melhor preparados para o mercado de trabalho e 97% atribuíram seu bom desempenho profissional ao programa de estágio da Universidade.

O "Programa de Residência em Software" difere do trabalho desenvolvido por Schackleford (2000) no domínio de aplicação e na abordagem. Enquanto Schackleford concentra o treinamento na área da infraestrutura de redes e equipamentos da Universidade e posterior prestação de serviços, por parte dos estudantes, à Administração da Universidade, o "Programa de Residência em Software" mantem o foco na formação intensiva do Engenheiro de Software objetivando, ao final de um ano de imersão no programa, inseri-lo no mercado de trabalho.

4. Descrição do Programa de Residência em Software

Para atingir o objetivo proposto nesse trabalho, foram estabelecidas algumas diretrizes para a construção do programa de residência. Tais diretrizes, executadas a partir do início de 2010 contemplaram seis etapas.

A primeira etapa foi fundamentada em firmar parceria com empresa de recursos humanos que atuasse na área de contratação de profissionais para empresas desenvolvedoras de software.

A segunda etapa consistiu em selecionar um grupo de alunos para o programa de residência. A seleção foi realizada pela empresa de Recursos Humanos que atua há mais de 10 anos nessa área. Os critérios utilizados para a seleção dos alunos foram estabelecidos pela empresa que fez a seleção, a partir de sua experiência nesse mercado. Dos 35 alunos inscritos, 20 foram selecionados para o programa.

A terceira etapa, realizada por um grupo de cinco professores, consistiu em definir o projeto de software a ser desenvolvido pelos participantes do programa de residência. Após reuniões com o setor acadêmico da FEMA, definiu-se que os estudantes residentes deveriam desenvolver o sistema que gerencia as atividades relacionadas à disciplina de Trabalho de Conclusão de Curso (TCC). O sistema foi denominado de Sistema de Acompanhamento de TCC. Essa etapa foi importante para que os docentes tivessem a dimensão e o controle de algumas características e recursos do sistema a ser desenvolvido.

Na quarta etapa iniciou-se o processo de treinamento dos estudantes. Para essa etapa foram destinadas 100 horas para a condução do programa de residência, assim divididas e ministradas pelos docentes envolvidos no programa: Metodologia de Desenvolvimento de Software, Garantia da Qualidade de Software, Banco de Dados, e

Ferramentas de Desenvolvimento de Software para Web, sendo 52h destinadas ao aprimoramento conceitual e 48h ao desenvolvimento do projeto de software. O grupo de residentes reunia-se semanalmente durante 12 horas, sob a supervisão de um docente.

Na quinta etapa do programa de residência em software, os estudantes, divididos em grupos, iniciaram o processo de levantamento dos requisitos, modelagem do Banco de Dados e o desenvolvimento do software definido na 3ª etapa.

A última etapa, que atualmente encontra-se em curso, envolve a participação dos alunos no desenvolvimento de um projeto real para uma empresa de desenvolvimento de software. O projeto foi definido pela empresa e envolve um módulo da área de Recursos Humanos. Esta etapa iniciou-se no mês de fevereiro de 2011, com a participação dos responsáveis pelo projeto na empresa, que conduziram duas reuniões com o grupo de alunos e estabeleceram as tarefas a serem executadas, e está prevista para ser concluída no mês de maio de 2011, finalizando o processo com avaliação dos resultados obtidos.

5. Conclusões e Resultados Parciais

O presente trabalho apresentou os aspectos relacionados à implementação do Programa de Residência em Software em um ambiente de graduação. Esta experiência evidenciou diversos aspectos positivos, especialmente ao que se refere ao aumento da maturidade dos alunos e à aproximação da academia com o mercado de trabalho.

Destaca-se que, nos moldes apresentados, o programa é inovador, gerando diversos resultados positivos para a formação dos alunos, tais como: a aquisição de experiência no desenvolvimento prático de projetos reais; a geração de mão de obra ainda mais qualificada para a indústria de software.

Ao final da sexta etapa de implantação do processo, uma avaliação formal será realizada para analisar os resultados obtidos e propor melhorias para continuidade do projeto.

Referências

Peckham, J., Batson, T. (2004) “Web Development Group: an Enterprising Campus-Based Internship Program for CS Majors”, Journal of Computing Sciences in Colleges, Volume 19, Issue 5.

Sampaio, A. et. al. (2005) “Software Test Program: a Software Residency Experience”. In International Conference on Software Engineering. Proceedings of 27th ICSE (Educational Track). St. Louis, USA, p. 611-612.

Shackleford, K. (2000) “Assessing the benefit of in-house work experience for university students”, Proceedings of the 28th annual ACM SIGUCCS conference on User services: Building the future, p. 354-358.

Way, T. P. (2005) “A Company-based Framework for a Software Engineering Course”. In Technical Symposium on Computer Science Education. Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education. St. Louis, USA, p. 132-136.

Uma experiência de ensino de Interação Humano-Computador para alunas do ensino médio

Sílvia Amélia Bim

Dept. de Ciência da Computação – UNICENTRO, Guarapuava – PR – Brasil

sabim@unicentro.br

Abstract. *This paper describes the first experience of teaching the Semiotic Engineering evaluation methods for female students from 1st year of high school. The purpose of the short course was to present the discipline of Human-Computer Interaction (HCI), promoting girls to a little-known profile of the course of Computer Science, in an attempt to stimulate their interest in this area. The results, although satisfactory, indicates that reformulations are necessary so that we can increase girls' participation in the short course.*

Resumo. *Este artigo descreve a primeira experiência de ensino dos métodos de avaliação da Engenharia Semiótica para alunas do 1º ano do ensino médio. O objetivo do minicurso foi apresentar a disciplina de Interação Humano-Computador (IHC), divulgando para meninas um perfil pouco conhecido do curso de Ciência da Computação, em uma tentativa de estimular o interesse delas por esta área de atuação. Os resultados, embora satisfatórios, indicam que são necessárias reformulações na proposta para que se possa aumentar a participação das meninas no minicurso.*

1. Introdução

Nos últimos anos se constata uma queda na procura por cursos de Ciência da Computação e áreas afins. A diminuição do número de mulheres interessadas na carreira computacional é ainda mais significativa [Cabral 2007, Coder et al. 2009, Cheryan et al. 2009].

Claudia Maria Bauzer Medeiros, presidente da Sociedade Brasileira de Computação (SBC) durante os anos de 2003 a 2007, informa que em 2005, no Brasil, dos estudantes de pós-graduação em Ciência da Computação, somente 25% eram mulheres, e entre os docentes, de 25% a 30%. Porém, nos cursos de graduação em Ciência e Engenharia da Computação têm ocorrido uma diminuição do número de mulheres nos últimos 15 anos, passando de 30% para 5 a 10% [Alves Filho 2005].

No caso específico da Universidade Estadual do Centro-Oeste (UNICENTRO) nos últimos anos o número de alunas que se inscrevem e que são aprovadas no curso de Ciência da Computação é bastante reduzido, conforme ilustram as Figuras 1 e 2.

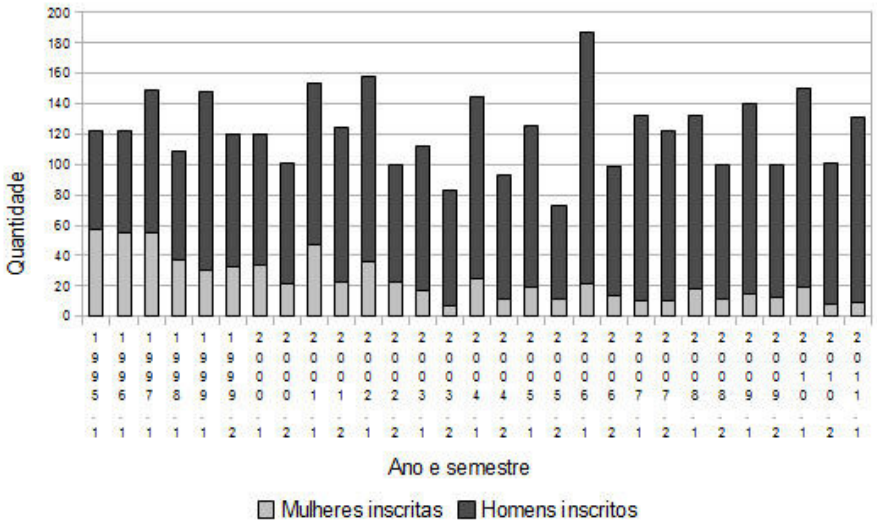


Figura 1. Relação de inscritos no vestibular de Ciência da Computação

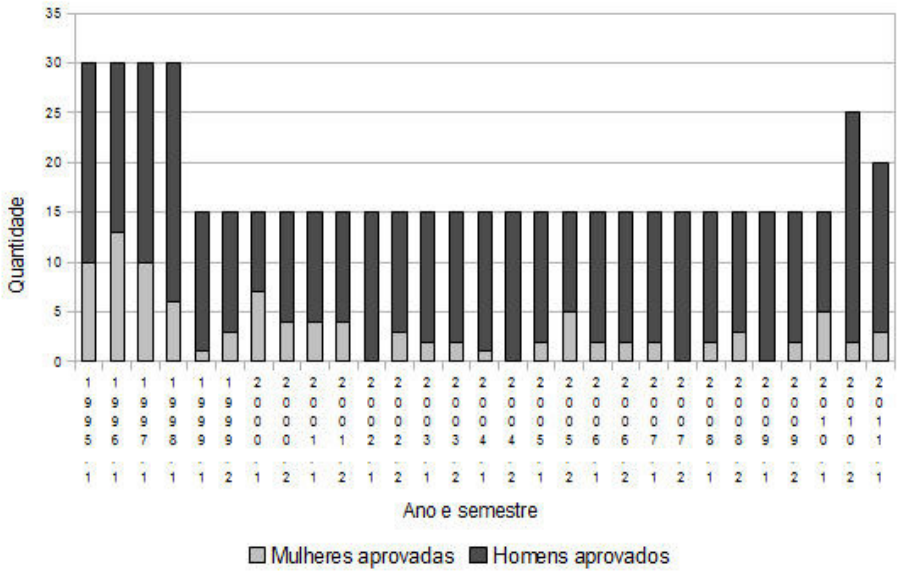


Figura 2. Relação de aprovados no vestibular de Ciência da Computação

Com a baixa procura de mulheres pela área da Computação elas perdem a oportunidade de atuar em uma das melhores carreiras profissionais e a área perde a perspectiva feminina no desenvolvimento de soluções computacionais [Cheryan *et al.* 2009]. É necessário que as equipes de desenvolvimento de tecnologia sejam formadas por homens e mulheres pois desta forma é possível aumentar os pontos de vista e a diversidade em um diálogo. Assim, a inovação e as possibilidades de criatividade também aumentam [Isaak 2010, Klawe et al. 2009].

Um dos desafios é entender porque as mulheres se desinteressam pela área. Uma hipótese é que há um preconceito de que a profissão é voltada apenas para quem se interessa por máquinas e que não existe o aspecto humano. Psicologicamente, mulheres

trabalham melhor em atividades multidisciplinares, que envolvem o aspecto social [Bauzer 2006].

A partir deste cenário, foi ministrado um minicurso de Interação Humano-Computador (IHC) para meninas do primeiro ano do ensino médio de uma escola pública de Guarapuava. A área de IHC é multidisciplinar e o aspecto humano faz parte dos objetos de estudo. Desta forma, um dos objetivos do minicurso era apresentar para as meninas uma disciplina da área de Computação que não lida apenas com o aspecto puramente tecnológico.

Há vários exemplos de oferta de cursos de diversas áreas de Ciência da Computação para alunos do ensino médio e fundamental [Guha, et al. 2005; Bruckman et al. 2007; Garcia et al. 2008]. Algumas iniciativas são exclusivamente dedicadas às meninas [Kelleher et al. 2007; Margolis e Fischer 2002].

Dentre as iniciativas para estimular o interesse pela área de computação destacamos o trabalho de Yardi et al. [2008] que descreve a implementação de um curso de seis semanas para ensinar IHC para alunos do ensino médio (de 11 a 15 anos). O programa do curso contemplava a área de *design* de IHC, seguindo a abordagem da Engenharia Cognitiva. No minicurso descrito neste artigo, optou-se por trabalhar o conteúdo da avaliação de IHC, utilizando os métodos propostos pela Engenharia Semiótica [de Souza 2005]. Uma das razões da escolha dos métodos é o fato de terem sidos propostos no Brasil por uma mulher, demonstrando assim uma contribuição brasileira e feminina para a área de Ciência da Computação.

Este artigo descreve uma iniciativa de ensino de IHC para alunas do 1º ano do ensino médio com o objetivo de despertar o interesse destas meninas pela área de Ciência da Computação. Na seção 2 a Engenharia Semiótica e seus métodos de avaliação, escolhidos como conteúdo do minicurso de IHC, são brevemente descritos. Em seguida, na seção 3 o conteúdo do curso é apresentado. Os resultados são citados na seção 4. As considerações finais são feitas na seção 5.

2. Engenharia Semiótica e seus métodos de avaliação de IHC

A Engenharia Semiótica é uma teoria brasileira de IHC que caracteriza a interação humano-computador como um processo de comunicação humana mediada por sistemas computacionais [de Souza 2005]. Os interlocutores deste processo de comunicação são os *designer*, os usuários e o sistema. A comunicação acontece em dois níveis: a comunicação direta usuário-sistema e a metacomunicação do *designer* para o usuário mediada pelo sistema, através da sua interface [Barbosa e da Silva 2010].

Esta teoria propõe dois métodos de avaliação de IHC: o Método de Inspeção Semiótica (MIS) [de Souza et al. 2006] e o Método de Avaliação de Comunicabilidade (MAC) [Prates et al. 2000]. O MIS é um método de inspeção onde o avaliador percorre a interface inspecionando os signos metalinguísticos, estáticos e dinâmicos buscando reconstruir a metamensagem do *designer* que pode ser parafraseado no seguinte esquema [de Souza 2005]:

"Esta é a minha interpretação sobre quem você é, o que eu entendi que você quer ou precisa fazer, de que formas prefere fazê-lo e por quê. Este é portanto o sistema

que projetei para você, e esta é a forma como você pode ou deve usá-lo para atingir os objetivos incorporados na minha visão."

Por ser um método de inspeção, sem a participação do usuário, o MIS enfoca a emissão da metamensagem. Suas etapas são: (1) Inspeção dos signos metalinguísticos; (2) Inspeção dos signos estáticos; (3) Inspeção dos signos dinâmicos; (4) Contraste entre as metamensagens (geradas nas etapas anteriores) e; (5) Apreciação da metacomunicação.

O MAC por sua vez, envolve a participação direta do usuário e consequentemente enfoca a recepção da metamensagem do *designer*. A partir dos registros gerados no teste com o usuário o avaliador analisa o vídeo da interação procurando por rupturas na comunicação. As rupturas identificadas são caracterizadas por uma das trezes etiquetas de comunicabilidade [Prates et al 2000]. Suas etapas são: (1) Etiquetagem; (2) Interpretação; (3) Perfil Semiótico.

3. O curso de IHC

3.1. Seleção das alunas

A partir do contato com a Secretaria de Estado da Educação (SEED) para identificar as escolas estaduais em Guarapuava que ofertam o ensino médio o Colégio Estadual Visconde de Guarapuava foi escolhido como local para a seleção das alunas.

A escolha se deu a partir da sugestão da SEED buscando satisfazer um pré-requisito do curso de IHC que pressupõe que as alunas tenham experiência com o uso frequente do computador (uso com editores de texto, buscas na internet, envio e recebimento de emails, por exemplo). Segundo a SEED os alunos do Colégio Visconde possuem um nível sócio-econômico que indica que a maioria deles possui computadores em suas residências.

O colégio selecionado oferta três turmas do primeiro ano do ensino médio no período diurno, segundo informou a supervisora pedagógica. A Tabela 1 indica a quantidade de alunos por turma.

Tabela 1. Relação de turmas e alunos do primeiro ano do ensino médio no Colégio Estadual Visconde de Guarapuava

Turma	Total de alunos	Meninas	Meninos
1º A	37	17	20
1º B	40	24	16
1º Normal	40	38	02

Todas as alunas das três turmas do primeiro ano do ensino médio foram convidadas a assistir uma rápida palestra durante o recreio, que tem duração de 20 minutos. A maioria das alunas das turmas do 1º A e do 1º Normal assistiu a palestra. A turma do 1º B foi representada por apenas duas alunas.

Durante a apresentação, que teve duração de dez minutos, foi feito o convite para um minicurso para conhecer um pouco mais sobre o curso de Ciência da Computação. Explicou-se que o assunto seria sobre IHC, uma das disciplinas do currículo de Ciência

da Computação, e que o foco seria avaliação de IHC. Por fim, se explicou a razão pela qual apenas as meninas estavam sendo convidadas.

Ao final da apresentação uma ficha foi entregue para cada uma das meninas. Aquelas interessadas em participar do minicurso deveriam informar nome, email, telefone e assinalar em quais das opções de horários oferecidas teriam disponibilidade para frequentar o minicurso.

Vinte alunas preencheram a ficha de interesse entretanto apenas dez participaram do minicurso. Dentre elas, uma aluna da 8ª série que assistiu a apresentação acompanhando a irmã que era de uma das turmas do 1º ano. Como o objetivo do minicurso é despertar o interesse de meninas pelo curso de Ciência da Computação através do ensino de IHC, o genuíno interesse da aluna da 8ª série não poderia ser menosprezado.

Para cada aluna foi entregue um Termo de Consentimento para que os responsáveis tomassem conhecimento sobre o projeto e dessem ou não o seu consentimento para a participação das meninas. Os Termos de Consentimento deveriam ser entregues no primeiro dia do minicurso.

3.2. Duração e conteúdo

O minicurso teve duração de seis horas, divididas em quatro encontros, realizados no ambiente da própria universidade. A Tabela 2 apresenta o conteúdo ministrado e a quantidade de alunas em cada encontro.

Tabela 2. Conteúdo do minicurso

Encontro	Quantidade de alunas	Atividades / Conteúdo trabalhado
1	4	Ambientação no campus universitário **Introdução**: Objetivo do minicurso; Explicação sobre o objetivo do curso de Ciência da Computação; Explicação sobre a diferença entre os cursos de Ciência da Computação, Engenharia da Computação e Sistemas de Informação; Apresentação do site do curso de Ciência da Computação da UNICENTRO; Apresentação do site da Sociedade Brasileira de Computação; Explicação sobre a área de Interação Humano-Computador e a disciplina de IHC ofertada no curso de Ciência da Computação da UNICENTRO; Importância da participação de mulheres na área de Computação (leitura do artigo "A Barbie é Engenheira da Computação" [Isaak 2010]). **Conceitos**: Signo, semiose ilimitada, signos explicativos, signos estáticos, signos dinâmicos. IHC na perspectiva da Engenharia Semiótica; Metamensagem do *designer*. Exemplo: Metamensagem do *designer* do Bloco de Notas.
2	10	Revisão dos conceitos Inspeção do Google Maps.
3	2	Inspeção da página inicial da ICDL[1] – *International Children´s Digital Library*. Conceito: cenário. Inspeção do Google Maps através de um cenário proposto.
4	4	Etiquetagem de uma interação no Google Maps.

[1] http://en.childrenslibrary.org

4. Resultados

4.1. Sobre o conteúdo trabalhado

Para que as alunas pudessem avaliar as interfaces utilizando os métodos de IHC propostos pela Engenharia Semiótica foram apresentados os conceitos de signo, semiose ilimitada e signos metalinguísticos, estáticos e dinâmicos. Os signos metalinguísticos foram apresentados como signos explicativos, pois se considera que o termo "explicativos" é mais intuitivo que "metalinguísticos".

O conceito de semiose ilimitada foi explicado através da seguinte atividade: A partir da imagem de um balão as alunas deveriam escrever a primeira ideia que lhes ocorresse ao ver a imagem do balão. A partir desta primeira ideia as alunas deveriam então escrever a primeira ideia que lhes ocorresse. Repetiram este processo quatro vezes, conforme ilustra a Figura 3.

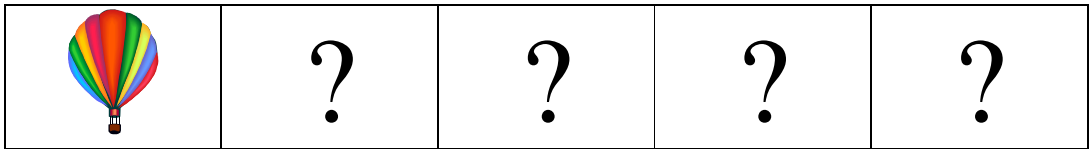


Figura 3 – Sequência do exercício de semiose ilimitada

Cada aluna apresentou a sua lista de palavras que foram escritas no quadro do laboratório. Elas então perceberam que cada uma criou uma lista diferente, embora com conceitos semelhantes. A partir da constatação das alunas explorou-se então o conceito da semiose ilimitada. Definiu-se que a semiose ilimitada é uma cadeia de interpretações e que cada pessoa irá construir a sua própria cadeia. Dependendo da experiência de cada pessoa as cadeias podem ser semelhantes ou não. Fez-se então considerações sobre o contexto computacional. As interfaces de sistemas computacionais também são interpretadas por seus usuários. A experiência prévia dos usuários irá influenciar na interpretação que eles farão sobre cada elemento da interface e sobre a interação. Desta forma o *designer* deve escolher cuidadosamente os elementos que irão compor a interface que estão criando, sempre atentos ao perfil dos usuários que querem atingir, para que a interpretação dos usuários seja mais semelhante possível a que o *designer* imaginou.

Nenhum dos dois métodos foi apresentado formalmente para as alunas. O MIS foi apresentado somente como uma atividade de inspeção onde as alunas deveriam explorar a interface do Google Maps de modo segmentado, analisando os signos explicativos, estáticos e dinâmicos, mas elaborando uma instância geral da metamensagem ao invés de uma instância para cada tipo de signo. Ou seja, apenas as etapas 1, 2 e 3 do MIS[2] foram praticadas, ainda que de modo adaptado.

Esta adaptação foi feita pois a análise segmentada dos signos de uma interface exige uma grande capacidade de interpretação sistemática e de abstração, conforme pesquisa realizada por Bim [2009]. Uma vez que os alunos do ensino superior

[2] (1) Inspeção dos signos metalinguísticos; (2) Inspeção dos signos estáticos; (3) Inspeção dos signos dinâmicos

apresentam deficiência nestas capacidades e consequentemente consideram as primeiras etapas do MIS difíceis, decidiu-se por fazer uma adaptação do método.

Com relação ao MAC somente a etapa de etiquetagem foi trabalhada pois no contexto do ensino deste método no ensino superior, esta é a etapa mais explorada por professores [Bim 2009]. Um vídeo com uma possível interação para o cenário proposto foi compartilhado com as alunas. A atividade consistia na observação do vídeo em busca das rupturas de comunicação que deveriam ser identificadas com uma das treze etiquetas do MAC. Uma lista com o nome das etiquetas e uma explicação para cada uma delas foi entregue às alunas que deveriam indicar em que momento do vídeo tais etiquetas eram identificadas.

4.2. Sobre o desempenho das alunas

Os resultados das inspeções e da etiquetagem realizadas pelas alunas foram satisfatórios. Mesmo sem profundos conhecimentos de informática e de IHC elas conseguiram identificar a metamensagem do *designer*, ainda que em um nível não tão detalhado quanto o MIS espera.

A identificação das rupturas teve um resultado melhor do que o da inspeção da metamensagem do *designer*. A interação do usuário com o Google Maps, registrada no vídeo que foi analisado, parece fornecer mais elementos para que as alunas, como avaliadoras, possam identificar os problemas da interação do que a inspeção realizada no MIS.

4.3. Sobre a participação exclusiva de meninas no minicurso

Uma das alunas fez o seguinte comentário sobre a turma ser exclusivamente de meninas: "*É melhor assim. Quando os meninos estão na sala não podemos fazer nenhum tipo de pergunta que eles já falam: "Nossa! Como você não sabe isto?""* Segundo Santos et al. [2006] há vários estudos que mostram que a educação mista pode desvalorizar ou ocultar o feminino. Na Irlanda, onde o ensino não é misto, meninos e meninas têm a mesma representatividade na área de Computação, com desempenhos também semelhantes [Bauzer 2006]. Este pode ser um indício de que devemos proporcionar mais atividades separadas para desenvolver nas meninas a auto-estima profissional na área tecnológica. E ao mesmo tempo diminuir o pré-conceito dos meninos com relação a capacidade das meninas nas atividades que envolvam o uso da tecnologia.

4.4. A impressão das alunas sobre o minicurso

Foi solicitado que as alunas escrevessem, sem identificação, suas impressões sobre o minicurso. A seguir apresentamos o depoimento das quatro alunas que participaram do último encontro do minicurso. Uma das alunas fez o seguinte relato:

"Eu achei que o curso foi bom. É bom aprender coisas diferentes, mas realmente me mostrou que essa não é a minha área. (...) não gostei das coisas nos mínimos detalhes. Prefiro coisas que vão direto ao ponto. Mas se fosse ter outro curso de informática eu faria para adquirir mais conhecimento. Mas eu gosto de coisas diferentes, mais na área de Direito. Não é uma faculdade que eu faria. Mas o curso

deve continuar para dar essa oportunidade de conhecimento as meninas dos colégios públicos." – A1

A avaliação de IHC é uma atividade que exige uma intensa atividade de interpretação sistemática [Bim 2009]. Em geral, mulheres são mais observadoras e detalhistas do que os homens. É interessante, entretanto, observar no depoimento da A1 seu desinteresse pela análise detalhada das interfaces e interações exploradas durante o minicurso. Apesar disto a aluna afirma ter interesse pela área de Direito. Tal afirmação parece contraditória, uma vez que a área de Direito também exige um trabalho detalhado na análise dos casos. O desinteresse pela análise detalhada de uma determinada situação nos parece ser mais característico da própria personalidade desta faixa etária, que é conhecida como Geração Z [Geek 2006].

No relato a seguir, a aluna comenta sobre a sua interpretação com relação ao processo de interação. É possível perceber que a perspectiva comunicativa apresentada pela Engenharia Semiótica foi compreendida pela aluna:

"Eu achei legal. Eu aprendi coisas interessantes. E [o minicurso] abriu os meus olhos sobre a questão de saber o que o programa está querendo nos dizer." – A2

A terceira aluna enfoca a contribuição que as mulheres podem dar para a área de Ciência da Computação. O conteúdo apresentado no primeiro encontro (leitura do texto de [Isaak 2010]) foi mencionado pela aluna (referência a Barbie).

"Eu sempre gostei de informática. Acho legal, divertido. E creio realmente que as mulheres "podem" mais que os homens e devem se impor... A Barbie foi a primeira e que surjam muitas outras." – A3

Por fim, no último depoimento, podemos perceber explicitamente que o minicurso contribui para despertar o interesse da aluna pela área de Computação. Além disto, pode-se observar no relato que as atividades de IHC despertam o senso crítico da aluna que percebe também que a área pode ser divertida.

"Eu gostei bastante do curso. Aposto que a partir de agora todas nós veremos o mundo da computação com olhos mais críticos. Acho que poderia ter mais atividades para interagirmos com o computador. É bem divertido de fazer e realmente despertou o meu interesse pela computação." – A4

Todos os relatos fazem considerações positivas com relação ao minicurso. Entretanto, nem todas as alunas confirmam o seu interesse pela Ciência da Computação. Tal resultado já era esperado uma vez que não se pretende que todas as meninas escolham a área de Computação como carreira. Uma das contribuições do minicurso foi também esclarecer as características e os objetivos da área, auxiliando na escolha da carreira profissional das meninas, escolhendo ou não o curso.

5. Considerações finais

Esta foi a primeira experiência de ensino da Engenharia Semiótica para alunas do ensino médio. Os resultados obtidos são satisfatórios. Entretanto, outras experiências devem ser realizadas e analisadas para que a estratégia de ensino seja aperfeiçoada.

O ensino de Ciência da Computação é uma área ainda carente de pesquisas [Wilson e Guzdial, 2010]. É necessário que várias iniciativas sejam realizadas para

contribuir com o aperfeiçoamento de uma área que tem características muito particulares. O ensino de alunas no ensino médio pode colaborar com estas pesquisas.

Algumas correções devem ser feitas com relação ao recrutamento das alunas. Dado o alto índice de desistência das alunas acredita-se que não é necessário informar que há um limite de vagas e então selecionar um número maior de alunas. Outra estratégia para tentar aumentar a participação das alunas no minicurso é oferecê-lo na própria escola. Embora o ambiente universitário possa ser um incentivador para a escolha de um curso superior, o campus fica afastado da escola e eventualmente as alunas moram mais perto da escola do que do campus da universidade.

Agradecimentos

Ao acadêmico Vinícius Almeida Macedo pelo levantamento dos dados apresentados nos gráficos utilizados neste artigo e pela confecção dos mesmos.

Referências

Alves Filho, M. Por uma computação mais democrática (e feminina). Jornal da Unicamp, edição 298, 22 a 28 de agosto de 2005. Disponível em http://www.unicamp.br/unicamp/unicamp_hoje/ju/agosto2005/ju298pag02.html. Último acesso em fev/2011.

Barbosa, S. D. J.; Silva, B. S. Interação humano-computador. Campus, 2010.

Bauzer, C. SBC quer ampliar a participação feminina. Computação Brasil. Março/Abril/Maio 2006. Disponível em http://femininolivre.wordpress.com/ 2010/07/20/sbc-quer-ampliar-a-participacao-feminina/ Último acesso em mar/2011.

Bim, S. A. Obstáculos ao ensino dos métodos de avaliação da Engenharia Semiótica. Rio de Janeiro, 2009. 181p. Tese de Doutorado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro. Disponível em http://www.serg.inf.puc-rio.br/serg/wp-content/uploads/2010/03/SilviaBim-Doutorado2009.pdf Último acesso em fev/2011.

Bruckman, A.; Forte, A.; Bandlow, A. HCI for kids. Lawrence Erlbaum Associates, NJ. 2007

Cabral, M.I.C. et al. Perfil dos Cursos de Computação e Informática no Brasil. XXVII Congresso da SBC - XVI WEI, Rio de Janeiro. (2007)

Cheryan, S., Plaut, V.C., Davies, P.G., Steele, C.M. Ambient Belonging: How Stereotypical Cues Impact Gender Participation in Computer Science. Journal of Personality and Social Psychology. 2009, Vol. 97, No. 6, 1045–1060

Coder, L.; Rosenbloom, J.L.; Ash, R.A.; Dupont, B.R. Increasing Gender Diversity in the IT Work Force. Communications of the ACM. Vol. 52, n. 5. 2009. p. 25-27.

de Souza, C.S. The Semiotic Engineering of Human-Computer Interaction. Cambridge, MA: The MIT Press. 2005.

de Souza, C.S. et al. The Semiotic Inspection Method. In: Anais do VII Simpósio sobre Fatores Humanos em Sistemas Computacionais. SBC: Natal, RN. p. 148-157. 2006.

Garcia, R.E.; Correia, R.C.M.; Shimabukuro, M.H. Ensino de Lógica de Programação e Estrutura de Dados para Alunos do Ensino Médio. XVII WEI – Workshop sobre o Ensino de Computação. Belém do Pará – PA. 2008. p. 246-249.

Geek, C. The Generation Z Connection: Teaching Information Literacy to the Newest Net Generation. redOrbit. 19/fev/2006. Disponível em http://www.redorbit.com/news/technology/397034/the_generation_z_connection_teaching_information_literacy_to_the_newest/ Último acesso em fev/2011.

Guha, M.,; Druin, A.; Chipman, G.; Fails, J.; Simms, S.; Farber, A. Working with young children as technology design partners. Communication of the ACM. Vol. 48. 2005. p. 39–42.

Isaak, J. A Barbie é Engenheira da Computação. Tradução: Claudia Bauzer Medeiros. SBC Horizontes. V. 03, nº 01. Abril, 2010. Disponível em http://portal.sbc.org.br/horizontes/doku.php?id=v03n01:19 Último acesso em fev/2011.

Kelleher, C.; Pausch, R.; Kiesler, S. Storytelling alice motivates middle school girls to learn computer programming. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (SIGCHI'07). San Jose, CA. 2007.

Klawe, M., Whitney, T., Simard, C. Women in Computing – take 2. Communications of the ACM. V. 52, nº 2, fev/2009. p. 68-76.

Margolis, J.; Fisher, A. Unlocking the clubhouse: Women in computing. Cambridge, MA: MIT Press. 2002.

Prates, R.O.; Barbosa, S.D.J.; de Souza, C.S. A Case Study for Evaluating Interface Design through Communicability. In Proceedings of the ACM Designing Interactive Systems, DIS'2000. Brooklyn, NY, p. 308-316, ago. 2000a.

Santos, L.W. dos, Ichikawa, E.Y., Cargano, D.F. (org.) Ciência, tecnologia e gênero: desvelando o feminino na construção do conhecimento. Londrina, IAPAR, 2006.

Wilson, C., Guzdial, M. How to make progress in Computing Education. Communications of the ACM. V. 53, nº 5, mai 2010. p. 35-37.

Yardi, S.; Krolikowski, P.; Marshall, T.; Bruckman, A. An HCI Approach to Computing in the Real World. ACM Journal on Educational Resources in Computing. Vol. 8, n. 3. 2008.

APIN: Uma Ferramenta Para Aprendizagem de Lógicas e Estímulo do Raciocínio e da Habilidade de Resolução de Problemas em um Contexto Computacional no Ensino Médio

Cleyton Aparecido Dim[1], Francisco Edson Lopes da Rocha[1]

[1]Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará(UFPA)
66.075-110 – Belém – PA – Brazil

cleytondim@gmail.com, felr@ufpa.br

Abstract. *This article discusses learning difficulties in the early years of the computer courses in disciplines involving logic. Considering that provision of appropriate educational tools can substantially improve the performance of students, it was developed at the Educational Computing Laboratory, Federal University of Pará, the tool called APIN – Agência Planetária de Inteligência (Planetary Intelligence Agency), as a possible contribution to solve the problem. It works out the idea of developing a logical-mathematical basis of high school students aiming at supporting the acquisition of new knowledge, according to the constructivist idea known as assimilation theory.*

Resumo*. Este artigo discute as dificuldades de aprendizagem nos primeiros anos dos cursos de computação em disciplinas em que a lógica está envolvida. Considerando que a oferta de ferramentas pedagógicas adequadas pode melhorar substancialmente o desempenho dos estudantes, foi desenvolvida no âmbito do Laboratório de Informática Educativa da Universidade Federal do Pará a ferramenta APIN – Agência Planetária de Inteligência, como uma possível solução do problema. Trabalha-se com a idéia de desenvolvimento de uma base lógico-matemática em alunos do Ensino Médio visando dar suporte à aquisição de novos conhecimentos, de acordo com a idéia construtivista conhecida por teoria da assimilação.*

1. Introdução

Disciplinas onde o raciocínio lógico é um fator de grande importância, como as que tratam de algoritmos, por exemplo, são encontradas desde os primeiros semestres em cursos superiores de computação, e cursá-las com bom aproveitamento tem sido um desafio para os acadêmicos, que enfrentam dificuldades de aprendizagem.

De acordo com Raabe e Silva (2005) estas dificuldades são, em parte, responsáveis pelas reprovações e desistência nestes cursos. Isto pode ser observado, por exemplo, no curso de Ciência da Computação na Universidade Federal do Pará, onde no ano de 2010 na disciplina de algoritmos constaram 21 alunos aprovados e 15 reprovados e na disciplina de algoritmos em um curso de Licenciatura em Computação na Universidade do Estado do Mato Grosso onde se registrou 26 alunos aprovados e 35 reprovados no primeiro semestre de 2010.

Uma das causas deste cenário é apontada por Santos e Costa (2005) e Pereira Júnior *et al*. (2005) como sendo a falta de uma base lógico-matemática que ajude a desenvolver o raciocínio lógico do aluno. Existem estudos realizados, como o de Rocha *et al*. (2010), que buscam solucionar o problema da dificuldade de aprendizagem através

de diferentes metodologias de ensino ou ferramentas de auxílio à aprendizagem, mas a maioria em um contexto remediador dentro do ambiente da universidade, não servindo para solucionar o problema da ausência de base lógico-matemática.

Segundo as Diretrizes Curriculares Nacionais para o Ensino Médio elaboradas pelo Mec (1998), o desenvolvimento da base lógico-matemática deve ter início ainda no Ensino Médio, sendo estabelecido expressamente que as escolas devem organizar seus currículos de modo a "*adotar metodologias de ensino diversificadas, que estimulem a reconstrução do conhecimento e mobilizem o raciocínio, a experimentação, a solução de problemas e outras competências cognitivas superiores*". Dessa forma, se cumprido este pressuposto, os alunos interessados em um curso superior na área da computação ingressarão na Universidade com base lógica e raciocínio desenvolvido.

Para auxiliar neste processo de desenvolvimento do raciocínio, podem ser utilizadas ferramentas de auxílio à aprendizagem, com recursos gráficos e animações que tornem a aprendizagem mais atrativa e ofereçam possibilidades de estímulo do raciocínio lógico. Como exemplo de uma ferramenta com esta característica pode-se citar o sistema Alice (2010), uma ferramenta desenvolvida na *Carnegie Melon University* para ensinar programação em escolas do ensino básico. Este sistema conta com recursos gráficos que permitem ao aluno criar animações ao escrever códigos de programação e alterar o estado de personagens e objetos. Na mesma linha de aplicativo aparece o Scratch (2010). Este sistema conta com essas mesmas características, aparentando uma interface de uso mais simples. Embora sejam bons sistemas de ensino de programação em um contexto iniciante, parecem não contribuir para o desenvolvimento de uma base lógico-matemática integral, uma vez que estes sistemas tratam apenas da lógica de programação deixando de fora outros tipos de lógica importantes para o raciocínio, como a lógica das proposições e a lógica de predicados, por exemplo.

Estão sendo desenvolvidos no Laboratório de Informática Educativa da Universidade Federal do Pará (http://labie.ufpa.br) diversos projetos que têm o objetivo de estimular este desenvolvimento lógico-matemático em candidatos aos cursos superiores de computação. Este artigo apresenta um destes projetos, a APIN – Agência Planetária de Inteligência, ferramenta construída com a finalidade de introduzir, no Ensino Médio, a aprendizagem de alguns tipos de lógica, como a proposicional e a de predicados, estimular o raciocínio lógico e a capacidade de solução de problemas em um contexto computacional. Para isso, a APIN conta com três módulos distintos: i) **Academia Apin**, onde o aluno tem lições de lógica e programação, organizado em forma de tutoriais; ii) **Missões Apin**, programado em forma de jogos e destinado à aplicação dos conhecimentos adquiridos na Academia; iii) **Operações de Treinamento de Inteligência**, para realizar olimpíadas e competições de lógica entre alunos e escolas. Um protótipo da ferramenta está disponível em http://labie.ufpa.br/apin.

A segunda seção deste artigo é dedicada a uma visão geral sobre o construtivismo e a teoria da Aprendizagem Significativa, onde pode estar a resposta do problema da aprendizagem nos cursos de computação. Na terceira seção são demonstradas algumas ferramentas educativas que poderiam ser utilizadas para estimular o raciocínio lógico no Ensino Médio, e na quarta seção é apresentada a ferramenta APIN. As Considerações Finais concluem o artigo na Seção 5.

2. Fundamentação Teórica: O Construtivismo e a Aprendizagem Significativa

De acordo com Castañon (2005), o termo construtivismo tem sido utilizado em diferentes ocasiões, sendo que uma das aplicações na aprendizagem se dá pela teoria de Piaget (1973) sobre a construção da estrutura cognitiva do sujeito, a Epistemologia Genética, que trata da modificação das estruturas mentais sob influência da interação do indivíduo com o meio em que vive.

Piaget apresenta em seu trabalho dois conceitos que são os fundamentos da sua idéia de construtivismo: a assimilação e a acomodação. Todo conhecimento novo recebido pelo sujeito será ou agregado a conhecimentos prévios compatíveis na sua estrutura cognitiva, na chamada assimilação, ou será alocado um novo esquema de conhecimento, caracterizando-se a acomodação, em ambos os casos alterando a estrutura cognitiva, que se adapta às mudanças. A Figura 1 exemplifica este processo.

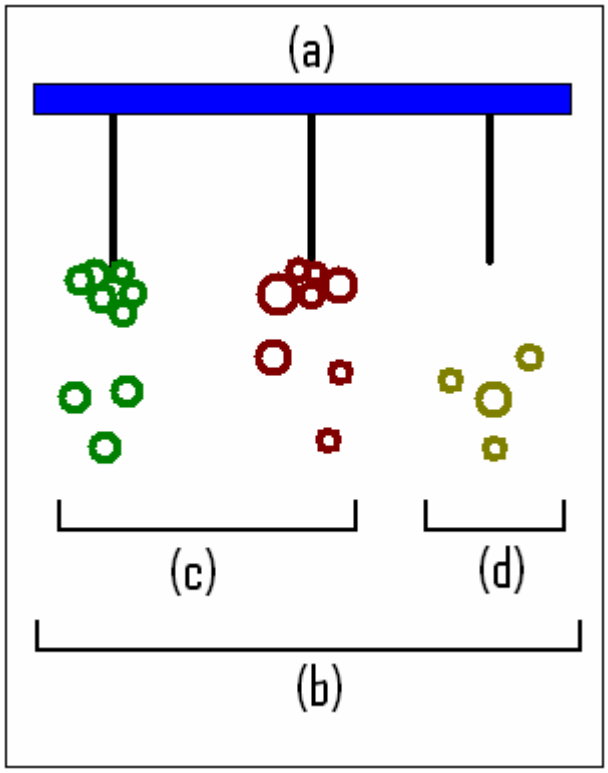


Figura 1 - Aprendizagem no Construtivismo: (a) Estrutura Cognitiva, (b) Novos Conhecimentos, (c) Assimilação, (d) Acomodação.

Aprendizagem Significativa é um conceito de aprendizagem baseado no construtivismo, definida como o centro do mecanismo de assimilação, referindo-se ao processo de enlace entre a estrutura cognitiva do sujeito com novas informações adquiridas por este (Ausubel, 1962 *apud* Rocha, 2007).

Na teoria de Ausubel (1962) observa-se que para que ocorra a aprendizagem significativa, o que consequentemente inclui a assimilação, alguns requisitos são necessários, dentre eles a existência de conhecimento prévio que possa se relacionar com as novas informações sobre um determinado tema.

Percebe-se então o porquê da necessidade de uma base lógico-matemática. Sem conhecimentos prévios que sirvam de ancoradouros na estrutura cognitiva para os novos conhecimentos que envolvem lógica, estes serão acomodados em um novo esquema

cognitivo e a aprendizagem neste caso será mais lenta, uma vez que o acadêmico terá de começar a aprender algo totalmente novo, ao invés de receber o conhecimento e rapidamente saber do que se trata e com o que associar.

3. Softwares Educativos

Atualmente tem-se utilizado tecnologias que favorecem o processo de ensino-aprendizagem nos variados níveis de educação. Dentre essas tecnologias podemos citar os softwares ou ferramentas educativas, que podem contribuir com a aprendizagem dos alunos, conforme expresso por Jucá (2006).

De acordo com Bertoldi e Ramos (1999), os softwares educativos podem ser divididos em diferentes categorias ou modalidades, das quais as mais difundidas são os de Exercício e Prática, Tutoriais, Simulações/Modelagem e Jogos.

Os softwares de Exercício e Prática são os mais simples e tradicionais. Consistem na exibição de exercícios em formato digital, com adição ou não de recursos audiovisuais para estimular o aluno a resolvê-los. Também tem a característica de mostrar os erros e acertos.

Os Tutoriais são mais complexos que os Exercícios por terem o objetivo de transmitir um conhecimento ao aluno, ou complementar um conhecimento previamente adquirido, contando com recursos gráficos e sonoros para despertar o interesse do usuário.

Os softwares Simuladores e de Modelagem replicam objetos ou ambientes reais no meio digital. Através deles é possível realizar simulações em um computador, e o aluno pode interagir com este ambiente digital, e observar resultados que seriam semelhantes às mesmas ações em um ambiente real. O ambiente pode inclusive ser criado pelo próprio aluno.

Jogos em um contexto educativo são fontes de recreação, aprendizagem e desenvolvimento de habilidades. Neles podem ser inseridos desafios cuja solução necessita de conhecimentos em alguma área de conhecimento, raciocínio lógico e coordenação motora.

Sendo a ausência de base lógico-matemática prévia uma das causas da dificuldade de aprendizagem nas disciplinas fundamentais da computação, é importante o seu desenvolvimento para estimular o raciocínio lógico nos alunos de Ensino Médio, de maneira que estejam aptos a ingressar em um curso superior em computação. Softwares educativos que estimulem o raciocínio lógico e a habilidade de resolução de problemas podem ter um impacto positivo neste desenvolvimento.

Nos parágrafos que seguem estão listados alguns softwares educativos que podem estimular o raciocínio lógico ou tem a finalidade específica de amenizar os problemas de aprendizagem na computação.

Alice (Alice, 2010): É um software educativo de simulação onde o usuário modela um ambiente com objetos tridimensionais existentes no sistema. Cada objeto tem atributos e métodos que são utilizados pelo usuário através de comandos sequenciais. Seu objetivo é de introduzir alunos da educação básica na lógica de programação de uma forma atrativa e divertida.

InLogic (Zeni, 2007): Este é um sistema de apoio ao ensino de lógica, com tabelas, gráficos e calculadoras. Foi desenvolvido para utilização no Ensino Superior de alunos de Lógica Formal, abordando um amplo conteúdo da lógica proposicional. É um exemplo de ferramenta desenvolvida em uma abordagem metodológica do problema de aprendizagem nos cursos superiores em computação.

JogosLogica (Sebben et al., 2009): Um sistema com jogos em três modos distintos: um de raciocínio lógico com a alteração de um jogo conhecido como Sudoku, um com desafios de lógica de proposições e um com associação e classificação de objetos e trabalho da coordenação motora. Foi desenvolvido para indivíduos da terceira idade e está disponível em plataforma WEB.

Scratch (Scratch, 2010): Similar ao sistema Alice, também tem o objetivo de introduzir conhecimentos de lógica de programação através da combinação de métodos e edição de atributos de objetos, com uma interface e linguagem mais simples, e gráficos bidimensionais.

Embora úteis em seus objetivos, estas ferramentas não aparentam contribuir com o desenvolvimento de uma base lógico-matemática integral, seja por ausência de um material pedagógico instrutivo sobre lógica, característica dos softwares tutoriais, ou por abordagem metodológica do problema, o que parece ser incompatível com a teoria da ausência de uma base prévia.

Em razão disto, foi desenvolvida a ferramenta Apin – Agência Planetária de Inteligência, que tem o objetivo específico de desenvolver uma base lógico-matemática em alunos do Ensino Médio, contando para isso com um sistema de tutoriais e exercícios que contém material pedagógico sobre diferentes tipos de lógica, como a proposicional e a de predicados, e um sistema de jogos onde há desafios de lógica que são resolvidos através da combinação de métodos de um personagem que interage com diversos ambientes em diferentes missões. Na seção que segue apresentam-se os detalhes da ferramenta Apin, seus módulos e abrangência didática.

4. APIN: Agência Planetária de Inteligência

A ferramenta educativa APIN tem o objetivo de desenvolver uma base lógico-matemática em alunos do Ensino Médio através da aprendizagem de alguns tipos de lógica, estímulo do raciocínio lógico e da capacidade de solução de problemas em um contexto computacional. Para tanto sua estrutura combina o esquema de software tutorial-exercício, contendo material didático para ensino de diferentes lógicas e exercícios relativos aos tópicos estudados, com o esquema de jogos, fornecendo em um ambiente lúdico e didático diversas possibilidades de aplicação de conhecimentos de lógica para estímulo do raciocínio e solução de problemas.

A APIN representa uma agência fictícia responsável pela segurança do planeta, na qual o aluno ingressa como um aprendiz na subdivisão Academia APIN; nesta o aluno tem aulas de diferentes tipos de lógica, com cada lógica sendo uma etapa deste sistema tutorial. Após o aluno se "graduar" na Academia APIN, ele se torna um Agente Oficial da Agência.

Uma vez que o aluno está na posição de Agente Oficial, ele passa a "trabalhar" para a APIN recebendo missões que representam desafios lógicos.

Há também um módulo para competições de lógica, cujo objetivo é a realização de olimpíadas de lógica em escolas. A idéia é abstraída para a ferramenta como operações de treinamento conjunto de inteligência entre as diversas filiais da APIN pelo universo.

4.1. Módulos da APIN

Conforme mencionado na seção anterior, são três os módulos que compõem a APIN: Academia APIN, Missões da APIN e Operações de Treinamento de Inteligência. Nas subseções que seguem são explicados com maiores detalhes a didática de cada módulo.

4.1.1. Academia APIN – Sistema Tutorial

O sistema tutorial da APIN constitui-se de páginas com material didático e questões objetivas sobre cinco tipos de lógica em um contexto iniciante e inclusivo, organizadas na ordem que segue:

- Lógica Proposicional: Trata de proposições e inferências em linguagem natural. Também são apresentadas técnicas para provar argumentos. Os exercícios incluem proposições para o aluno inferir qual a resposta de determinada pergunta, bem como há proposições com inferências incorretas, onde o aluno deve descobrir onde está o erro.

- Lógica de Predicados: Aqui são utilizados conceitos da lógica proposicional, adicionando um elemento importante da lógica de predicados: quantificadores. Os exercícios seguem o mesmo molde dos da lógica proposicional, com a inclusão dos conceitos de quantificadores.

- Lógica Difusa: São transmitidos os conceitos de múltiplos valores para argumentos, ao invés de restringi-los a verdadeiro e falso como na Lógica Formal. Exercícios incluem a inferência e a detecção de erros.

- Lógica Binária: Contém material com conceitos de Negação, Conjunção, Disjunção e Disjunção Exclusiva de argumentos em um formato de linguagem natural. Há exercícios de detecção do valor lógico de sentenças que incluem os conectivos lógicos, bem como a averiguação de erros em resultados fornecidos para análise.

- Lógica de Programação: São trabalhados os conceitos de sequência de comandos, variáveis, parâmetros, estruturas condicionais, conectivos lógicos e estruturas de repetição. O aluno toma conhecimento destes conceitos através de textos e animações, e os pratica através das chamadas “aulas iniciais de controle” na agência, que representam um pré-treinamento do agente para o controle de um robô utilizado para completar as missões solicitadas quando adentrar na APIN. Nas aulas iniciais de controle, o aluno altera os atributos do robô através de comandos sequenciais que criam animações, podendo condicionar e repetir a execução destes comandos. A lógica de programação é uma etapa muito importante para o objetivo da ferramenta, sendo as outras lógicas mencionadas, em parte, um preparatório formador de base para esta lógica.

4.1.2. Missões da APIN – Sistema de Jogo

Após passar pela etapa de tutoriais-exercícios o aluno entra na APIN como um agente e recebe diversas missões preventivas ou reativas em nome da segurança do planeta. Nessas missões o aluno faz uso dos conhecimentos adquiridos na Academia APIN, tendo que lidar com enigmas lógicos e problemas a serem resolvidos.

Dado o elevado risco pessoal que as atividades em campo representam, a APIN fornece um robô para cada agente, que deve ser controlado remotamente através de um painel onde são inseridos comandos em bloco. Após a inserção dos comandos, eles são enviados ao robô que executa as ações correspondentes.

Nas missões o aluno interpreta os enigmas lógicos que podem estar presentes nos anúncios das missões e/ou no cenário e pensa em como usar as funções disponíveis do robô para resolvê-los. O anúncio da missão aparece no cenário quando a mesma é iniciada, logo após solicitação do agente. A solicitação é feita por meio do acionamento de um botão específico para esta finalidade.

Após receber a missão, o agente decide que ações o robô precisa realizar para levá-la a cabo. Ele escolhe as ações na botoeira do robô, na ordem em que elas devem ser executadas. No cenário apresentado na Figura 2, o agente já fez as escolhas necessárias para cumprir uma missão que solicita o acionamento de uma máquina, estando estas ações registradas no painel do robô. Para fazer o robô se movimentar segundo a coleção de ordens registradas, o agente pressiona o botão "Acionar IBOT".



Figura 2 - Exemplo de uma missão

4.1.3. Operações de Treinamento de Inteligência

Com alguma freqüência, a APIN realiza operações de treinamento de inteligência entre as várias filiais da APIN espalhadas pelo universo, que são uma adaptação do módulo de missões para competições, onde será vitorioso o agente que finalizar as missões e conseguir a maior pontuação. Também podem ser comparados os resultados entre as filiais APIN para verificar qual delas tem os agentes mais promissores.

É um modulo que objetiva a realização de olimpíadas de lógica entre escolas. Cada escola é tida como uma filial APIN e os agentes como seus alunos.

4.2. Sistema de Pontuação

A APIN mantém um sistema de pontuação interna, que é um medidor do desempenho do agente em suas atividades na Academia, na Agência e nos Treinamentos.

Para as atividades na Academia APIN a pontuação é calculada levando-se em consideração a quantidade de erros e o tempo para resolução dos exercícios. Já para medir o desempenho do agente em suas missões na agência, a pontuação é calculada pelo tempo de conclusão da missão, o gerenciamento de recursos do robô e a quantidade de falhas e reinício de missões. O módulo das Operações de Treinamento de Inteligência segue o mesmo critério de pontuação das missões na agência.

Esta pontuação pode servir como apoio para o professor realizar avaliações de desempenho do aluno, ou para definir vencedores em olimpíadas de lógica. Como sugestão, professores podem também analisar os comandos utilizados pelos alunos para a animação do robô na etapa de lógica de programação do módulo tutorial e para completar as missões do módulo de jogo, considerando a clareza, simplicidade e criatividade nos códigos.

5. Considerações Finais e Trabalhos Futuros

A dificuldade de aprendizagem em disciplinas que envolvem lógica está presente nos cursos superiores de computação. Destaca-se que entre as causas para este problema está a ausência de uma base lógico-matemática prévia. Esta base está ligada ao processo de assimilação na teoria do construtivismo, facilitando o processo de aprendizagem por servir de ancoradouro aos novos conhecimentos na estrutura cognitiva do indivíduo.

Por ser justificável a utilização de metodologias diversificadas no ensino médio para estimular o raciocínio e a habilidade de resolução de problemas, e em razão da aparente incompatibilidade com a teoria da base lógico-matemática nos softwares com a finalidade de amenizar os problemas de aprendizagem nos cursos superiores de computação e de estimular o raciocínio lógico que aqui foram analisados, foi apresentada a ferramenta APIN, que trabalha com a idéia de desenvolver uma base lógico-matemática e conta com recursos gráficos, tutoriais explicativos, exercícios e missões em formato de jogo para despertar o interesse do aluno e motivá-lo a querer aprender.

A ferramenta APIN pretende auxiliar os alunos do ensino médio a descobrir o mundo da lógica e expandir o pensamento por meio das lições sobre Lógica Formal e Difusa. Complementarmente, os usuários são introduzidos na Lógica de Programação, onde podem aprender um pouco do mundo da computação, de forma que em um possível ambiente universitário as aprendizagens conceitual e prática sejam facilitadas e as desistências reduzidas.

Está disponível em http://labie.ufpa.br/apin um protótipo da ferramenta, demonstrando uma aula de lógica do módulo de tutoriais/exercícios, e uma missão do módulo de jogo.

Considera-se que a pesquisa apresentada neste artigo pode e deve prosseguir por meio de melhorias, das quais destacam-se as seguintes:

- Construção de um *framework* para criação e edição de missões: permitiria gerar arquivos de missões trabalhando em alto nível, fora do contexto da programação do sistema, sendo indispensável o desenvolvimento do código de interpretação destes arquivos de missões;
- Criação de um repositório *on-line* para arquivos de missões: permitiria criar desafios *peer-to-peer*, de alunos para outros alunos, ou de escolas para escolas, por exemplo;
- Desenvolvimento da ferramenta APIN para dispositivos móveis: adequação da ferramenta para trabalhar em aparelhos como celulares e outros pda´s.

Referências Bibliográficas

Alice. (2010) “Projeto Alice”, http://www.alice.org, Outubro.

Ausubel, P. D. (1962) “A Subsumption Theory of Meaningful Learning and Retention”. Journal of General Psychology, (66):213–224.

Bertoldi, S.; Ramos, E. M. F. (1999) “Avaliação de Software Educacional – Impressões e Reflexões”. Dissertação. Programa de Pós-Graduação em Ciências da Computação. Universidade Federal de Santa Catarina.

Castañon, G. A. (2005) "Construtivismo e Ciências Humanas”. In: Revista Ciências e Cognição, Vol 5: 36-49.

Jucá, S. C. S. (2006) “A Relevância dos Softwares Educativos na Educação Profissional.” In: Revista Ciências e Cognição, Vol 8: 22-28.

Mec. (1998) Ministério da Educação. Conselho Nacional de Educação. Diretrizes Curriculares Nacionais para o Ensino Médio - Resolução CEB/CNE Nº 3, de 26 de junho de 1998.

Pereira Júnior, J. C. R.; Rapkiewicz, C. L.; Delgado, C.; Xexeo, J. A. M. (2005) “Ensino de Algoritmos e Programação: Uma Experiência no Ensino Médio.” In: XIII Workshop sobre Educação em Computação, São Leopoldo/RS.

Piaget, J. (1973) “Psicologia e Epistemologia: Por uma Teoria do Conhecimento” (trad. Agnes Cretella). São Paulo: Ed. Forense (original publicado em 1966).

Raabe, A. L. A.; Silva, J. M. C. (2005) “Um Ambiente para Atendimento às Dificuldades de Aprendizagem de Algoritmos.” In: XIII Workshop sobre Educação em Computação, São Leopoldo/RS.

Rocha, P. S.; Ferreira, B.; Monteiro, D.; Nunes, D. S. C.; Goés, H. C. N. (2010) “Ensino e Aprendizagem de Programação: Análise da Aplicação de Proposta Metodológica Baseada no Sistema Personalizado de Ensino.” In: RENOTE – Revista Novas Tecnologias na Educação, Vol 8, N 3.

Rocha, F. E. L.; (2007) “Avaliação da Aprendizagem: Uma Abordagem Qualitativa Baseada em Mapas Conceituais, Ontologias e Algoritmos Genéticos”. Tese de Doutorado. Programa de Pós-Graduação em Engenharia Elétrica. Universidade Federal do Pará.

Santos, R. P.; Costa, H. A. X. (2005) “TBC-AED e TBC-AED/WEB: Um Desafio no Ensino de Algoritmos, Estruturas de Dados e Programação.” In: IV WEIMIG - Workshop de Educação em Computação e Informática do Estado de Minas Gerais, Varginha/MG.

Scratch. (2010) “Projeto Scratch”, http://scratch.mit.edu, Outubro.

Sebben, N.; Guedes, A. L.; Stahlhofeer, M. M.; Guedes, F. L. (2009) “Desenvolvendo Jogos para a Terceira Idade”. In: Simpósio SC Games, Florianópolis/SC.

Zeni, J. R. R. (2007) “INLOGIC: Uma Ferramenta para o Ensino e Aprendizagem de Lógica.” In: XVIII Simpósio Brasileiro de Informática na Educação, São Paulo/SP.

Inclusão digital na cidade de Pombal

Adriana Damasceno[1], **Ítala Fernandes**[1],
Flavia Moreira[1], **Tamires Santos**[1], **Clotildes Alvino**[1]

[1]Centro de Ciência e Tecnologia Alimentar (CCTA)
Universidade Federal de Campina Grande (UFCG)
Rua Jario Vieira Feitosa, Bairro dos Pereiros, Pombal – Paraíba – Brasil

adrianacarla@ccta.ufcg.edu.br,

{italamayara17, flavia_izabely, tamiressantos_tata, cloalvino}@hotmail.com

Abstract. *Digital inclusion is rather important to decrease unemployment. We need young people to be employed, so they must have access to courses which can increase their qualification concerning the use of informatics. However, some parts of our country do not offer chances to qualify these students. The project "Inclusão digital na cidade de Pombal"intends to allow students to have free informatics classes. This work was developed during a year and it majored 11 tutors in this field and 74 students from public schools of a city in the countryside of Paraíba (Brazil). As part of this initiative, 305 students from high school answered a questionnaire to characterize a profile of these students.*

Resumo. *A inclusão digital é de fundamental importância para o combate ao desemprego. Para que os jovens sejam inseridos no mercado de trabalho, é preciso que eles tenham acesso a uma melhor qualificação onde o computador possa ser usado. No entanto, em algumas regiões do país há escassez de programas de atendimento a essa população. O projeto "Inclusão digital na cidade de Pombal"visa suprir essa necessidade em âmbito local, oferecendo aulas de Informática Básica gratuitas para alunos da rede pública de ensino em uma cidade do Alto Sertão paraibano. O projeto conseguiu formar 74 alunos de escolas públicas e 11 tutores de conhecimento na área. Ademais, foram aplicados 305 questionários para caracterizar o perfil desse público-alvo.*

1. Introdução

A exclusão sócio-econômica é um dos maiores males da atualidade, principalmente no nordeste brasileiro. Um dos tipos de exclusão atual e que resulta do uso de diversas tecnologias é a exclusão digital. É necessária uma iniciativa de inclusão eficaz para que seu combate possa ser realizado. Essa é uma das crescentes causas dos altos índices de desemprego presente no país.

Dentro desse contexto, o Brasil busca desenvolver ações de inclusão digital que permitam a melhoria da capacitação da população. Uma dessas iniciativas são os telecentros. Estes são unidades criadas pelo poder público para permitir o acesso à Internet. Segundo dados do Observatório Nacional de inclusão digital (ONID) [ONI 2011], apesar de o nordeste ocupar o segundo lugar em número de telecentros no Brasil, essa região detém somente 24,87% dessas unidades cadastradas, contrastando com a região

sudeste, que ocupa o primeiro lugar e possui 44,44%, praticamente o dobro. Observando a classificação no número dessas unidades por estado, percebe-se uma situação mais alarmante, onde o estado da Paraíba ocupa o 19º lugar, detendo somente 1,11% dos telecentros, e na cidade de Pombal, há somente um telecentro cadastrado na ONID.

Infelizmente essas ações de nada adiantam sem uma capacitação efetiva dos usuários no que diz respeito ao uso dos computadores. A inclusão digital, para acontecer, precisa de três instrumentos básicos que são: computador, acesso à rede e o domínio dessas ferramentas. Não basta apenas o cidadão possuir um simples computador conectado à internet para considerarmos este um incluído digitalmente. Ele precisa saber o que fazer com esses recursos.

Ademais, além do baixo número de telecentros, vários dados estatísticos da Paraíba comprovam que o estado carece de programas de capacitação eficazes. De acordo com dados dos indicadores básicos do Datasus em 2009 [DAT 2011], a taxa de analfabetismo no estado da Paraíba para pessoas com menos de 15 anos foi de 23,43%, ao passo que a taxa de desemprego é de 7,53%. Segundo dados do IBGE [IBG 2011], a incidência de pobreza na cidade de Pombal é de 53,30%, e somente 22,75% da população está matriculada em escolas de ensino médio ou fundamental, o que demonstra a baixa capacitação de mão de obra local.

Diante desse panomama, o projeto "Inclusão digital na cidade de Pombal"objetiva diminuir, em âmbito local, a problemática da exclusão digital, oferecendo formas de capacitação profissional gratuitas e direcionadas aos jovens que buscam o primeiro emprego, promovendo o desenvolvimento educacional e econômico para a comunidade de Pombal, situada na região do Alto Sertão do estado da Paraíba.

No ano de 2010, o projeto conseguiu oferecer aulas gratuitas de informática básica a 74 alunos da rede pública de ensino e formou 11 tutores multiplicadores de conhecimento na área. Com isso, vários estudantes de comunidades pobres da cidade puderam receber uma melhor qualificação para o mercado de trabalho. Paralelamente, 305 questionários foram aplicados na rede de ensino de segundo grau da cidade de Pombal com o objetivo de caracterizar o perfil do público-alvo no que diz respeito aos conhecimentos de informática.

O objetivo desse artigo é apresentar esse projeto, mostrando as contribuições oferecidas à comunidade de Pombal e iniciar o processo de caracterização desse público de atendimento. A Seção 3.2 mostra sua dinâmica de funcionamento e os mecanismos de avaliação usado tanto para os componentes do projeto como para os alunos atendidos no curso. A apresentação desses resultados é mostrada na Seção 3.1. A Seção 2 mostra alguns conceitos relacionados à problemática da inclusão digital e projetos de objetivo similar a esse. As conclusões (Seção 4) mostram alguns dos direcionamentos atuais do projeto.

2. Referencial Teórico

A implantação de computadores nas escolas é de extrema importância para que ocorra a inclusão não só digital, mas também social. É muito difícil para a escola manter sua biblioteca atualizada com jornais, revistas, livros recentes e incluir seus alunos no universo cultural como: teatros, cinemas, bares, restaurantes e galerias de artes. Nesse ponto, o

computador com acesso à internet acaba sendo fator primordial para a inclusão. Esta não se dá simplesmente através do acesso ao computador. É preciso capacitar as pessoas para o uso das novas tecnologias de forma que elas tenham acesso à informação e aos meios de criação e produção [Silva 2004].

O computador não resolverá o problema da exclusão, mas poderá contribuir para que esta diminua. Ele deve ser utilizado como um mundo virtual ao alcance do aluno e discussões e respostas de questões podem ser realizadas. A utilização do computador como ferramenta pedagógica tende a desenvolver nos alunos habilidades como autonomia na aprendizagem de conteúdos, liberdade de expressão, comprometimento social, respeito ao diferente e busca de soluções rápidas e imediatas, entre outros [Silva et al. 2005].

A exclusão digital é a mais nova ferida social, e acirra ainda as já gritantes desigualdades existentes. A questão da conectividade, ou a falta dela, é uma questão real e problemática, pois as pessoas que não possuem acesso às Tecnologias de Informação e Comunicação apresentam uma fragilidade cada vez mais considerável no mercado de trabalho [Rocha 2010].

Incluir digitalmente não significa apenas ensinar uma pessoa a usar um computador para acessar a internet, pesquisar ou elaborar um texto. É também ensinar como melhorar os quadros sociais, utilizando-se dos recursos que um computador oferece. Além disso, essa ação permite a melhoria de vida e a qualificação profissional, entre outros benefícios que a tecnologia traz. Contudo, a internet é um ambiente de informação complexo para quem não tem familiaridade ou capacitação na busca e recuperação da informação.

Os desafios para a inclusão digital são muitos tendo em vista as diferenças sociais gritantes. Portanto, no Brasil as estratégias para a promoção da inclusão digital devem estar alinhadas com as estratégias que promovam a inclusão social. É preciso ultrapassar as barreiras sociais, culturais e econômicas e promover a inclusão digital. Segundo Guedes e Sorj [Sorj 2005], a inclusão digital é geralmente definida num país pela relação entre a porcentagem de pessoas com acesso a computador e/ou a internet no domicílio e o total da população. Essa relação é reducionista, pois caracteriza o simples ato de acessar como inclusivo. A partir desta noção, as políticas de inclusão digital não podem estar desatreladas das de Educação, haja vista o potencial emancipador dado através do conhecer aguçado de um modelo pedagógico mais sensível às necessidades da população [Abreu 2010].

Por isso, um processo de inclusão digital na maioria das vezes é complexo, envolvendo diversas ações como a criação de um projeto para um bairro, município, cidade, estado ou país. Adicionalmente, ele deverá envolver a população local, tornando-os agentes participativos de forma a contribuir para o alcance do objetivo. É preciso que os projetos promovam a inclusão digital no Brasil como um todo e não fique apenas no Sudeste, onde cidades mais desenvolvidas como São Paulo e Rio de Janeiro se destacam [ONI 2011].

Projetos de inclusão digital circundam uma possível vitória a partir do momento que tiram as pessoas do analfabetismo digital para o início de sua busca pela não estagnação e mostram a tamanha diversidade cultural e social do país, o poder aquisitivo e a luta pela busca do conhecimento, determinando o grau de instrução dos indivíduos. Frente a isto, as inovações tecnológicas a nível mundial não tendem a parar, mas o acompanhamento do Brasil é um jogo de interesses individuais, ou seja, cada indivíduo tem a consciência das atitudes a serem tomadas em busca da informação, onde os

que detêm o poder tornam-se manipuladores da grande massa, visando lucros e visibilidade [Sorj 2005].

O posicionamento atual da Ciência da Informação, como facilitadora da comunicação do conhecimento, principalmente em países em desenvolvimento como o Brasil, é indispensável quando se defronta com a realidade: O que está a definir o aumento ou redução da desigualdade social é justamente o nível de utilização do conhecimento e sua aplicação através desse tipo de tecnologia [Silva et al. 2005].

Portanto, entende-se que acessibilidade e inclusão digital não dizem respeito apenas ao uso do computador e ao acesso a rede de informações, mas à promoção de capacidades intelectuais de acesso a esse processo. Desta forma, tem-se inclusão digital como um caminho para se chegar à tão discutida inclusão social. Nesse contexto, serão tratados alguns projetos que promovem a inclusão digital no Brasil e na Paraíba. São eles o Projeto Casa Brasil, o Projeto Cidadão Conectado, o Projeto *KHouse* Profissionalizante, o Programa Escola Solidária, o Projeto das Estações Digitais e iniciativas próximas à cidade de Pombal.

O projeto Casa Brasil [Dodt 2010] leva conectividade e computadores a comunidades de baixo desenvolvimento humano. O objetivo principal do projeto é promover a apropriação do equipamento público e das ferramentas de inclusão digital para gerar o desenvolvimento social. Trata-se um projeto do Governo Federal que existe desde 2003.

O projeto Cidadão Conectado tem o objetivo de possibilitar que aqueles que não têm acesso a um computador possam adquirir um equipamento. Além de disponibilizar o acesso às tecnologias, o projeto também permite que toda uma cadeia produtiva venha a ser reforçada no Brasil, inibindo a ação do mercado que não paga impostos nem contrata mão de obra com garantias trabalhistas.

O modelo *KHouse* Profissionalizante começou a funcionar em fevereiro de 2003 com o objetivo de qualificar os jovens carentes para a inserção no mercado de trabalho. Uma pesquisa realizada em 2006 mostrou que o que faz esses alunos procurarem o curso é um reflexo da sociedade atual, que exige uma capacitação em Informática como requisito básico. A pesquisa mostrou ainda que a maioria dos estudantes, do projeto *KHouse* Profissionalizante são do ensino médio, estudam em escolas públicas e têm renda familiar média de dois salários mínimos.

No que diz respeito à Paraíba, existem poucos programas de inclusão digital sendo executados. Mais especificamente na cidade de João Pessoa, o Centro Universitário de João Pessoa (UNIPe), através do Programa Escola Solidária [ECS 2011], capacita alunos carentes de escolas públicas através do uso do computador e já completou 10 anos de existência, formando mais de mil jovens de baixa renda. Todos os alunos que assistem a essas aulas também recebem lanche, ajuda em transporte, material escolar e fardamento.

O Projeto das Estações Digitais [EST 2011], uma iniciativa da Prefeitura de João Pessoa (PMJP), por meio de convênio com o Ministério da Ciência e Tecnologia, tem proporcionado oportunidades de aprendizado e lazer para a população pessoense. Visando combater a exclusão digital, o projeto conta atualmente com 920 pessoas matriculadas em cursos de Informática. No ano passado 2.366 alunos foram capacitados nos cursos básicos de informática com carga horária de 60 horas/aula. Cada unidade das Estações Digitais possui 11 computadores conectados à internet banda larga, uma impressora e um

scanner, com o objetivo de facilitar o acesso da população às tecnologias da Informação e da Comunicação. No local as pessoas aprendem a usar o computador, acessar a internet para mandar e receber mensagens, fazer pesquisas e usar os diversos serviços e facilidades disponíveis.

Nas adjacências da cidade de Pombal, o Campus Sousa da Universidade Federal de Campina Grande oferece vagas para três turmas de jovens carentes que estejam interessados em aprender noções básicas de Informática [Cavalcanti 2006]. Além desses cursos, também são oferecidas vagas para cursos nas áreas de Contabilidade, Administração, Vendas e Economia.

3. Inclusão Digital na Cidade de Pombal

O objetivo do projeto "Inclusão digital na cidade de Pombal"é oferecer curso de Informática Básica a alunos de escolas públicas de Pombal para que eles possam ser inseridos no mercado de trabalho. A Seção 3.1 aborda as principais características do público atendido pelo projeto e a Seção 3.2 explica a metodologia de trabalho utilizada. Várias dessas ações foram baseadas na dinâmica de projeto adotada pelo Projeto Escola de Computação Solidária [ECS 2011].

3.1. Caracterização do Público-Alvo

Durante o início da execução do projeto, os dados disponíveis sobre o público-alvo visado eram os presentes nos sites do IBGE [IBG 2011], DATASUS [DAT 2011] e CETIC (Centro de Estudos sobre as Tecnologias de Informação e da Comunicação) [CET 2011]. No entanto, era necessário conhecer melhor esse setor da comunidade para que as demandas às quais o projeto se propunha a atender fossem atendidas.

A partir dessa necessidade, foi criado um questionário (mostrado na Figura 1) para medir conhecimentos em Informática básica a ser respondido pelos alunos das quatro escolas de segundo grau da cidade de Pombal. Este é composto de 12 questões de natureza objetiva e aborda desde a posse de computadores em suas residências ao uso da suíte de escritório *Office* e o sistema operacional *Windows*, mais comumente utilizados. Os questionários foram aplicados pelos tutores do projeto, previamente treinados quanto à maneira de se proceder.

Foram aplicados 305 questionários, tendo como critério de inclusão o entrevistado ser aluno de segundo grau da rede de ensino da cidade de Pombal, o mesmo estar presente no momento da coleta de dados e concordar com a realização do mesmo. Um gráfico com o resumo dos dados obtidos durante essa etapa é mostrado na Figura 2. Seu eixo horizontal mostra os número das questões exibidas na Figura 1 e o eixo vertical representa o percentual de respostas positivas de cada pergunta.

Tendo como base os resultados obtidos através do questionário, pode-se constatar que 52,46% dos alunos possuem computador em casa, o que se aproxima dos dados da pesquisa do CETIC sobre o local de uso individual do computador, que afirma que 54% dos alunos do ensino médio entrevistados possuíam computador em casa. Apesar de quase a metade deles não possuir computador, 83,93% têm acesso à internet e 84,92% sabem usá-la para realizar as pesquisas que se fazem necessárias no cotidiano e no ambiente escolar. Isso supera as estatísticas da pesquisa realizada pelo CETIC, que informa que apenas 60% dos alunos de ensino médio da região nordeste sabem navegar na internet.

1. Você possui computador em casa?
2. Você tem acesso à internet?
3. Você sabe fazer uma pesquisa na internet?
4. Você sabe digitar um texto no *Word*?
5. Você sabe elaborar uma tabela no *Excel*?
6. Você sabe elaborar um gráfico no *Excel*?
7. Você sabe preparar uma apresentação no *PowerPoint*?
8. Você sabe salvar um arquivo em mídia (CD, disquete e outros) ?
9. Você sabe anexar um arquivo ou foto para ser enviado em um e-mail?
10. Você sabe instalar um hardware no *Windows*?
11. Você sabe instalar um software no *Windows*?
12. Você utiliza outro sistema operacional além do *Windows*?

Figura 1. Questionário de habilidades em Informática

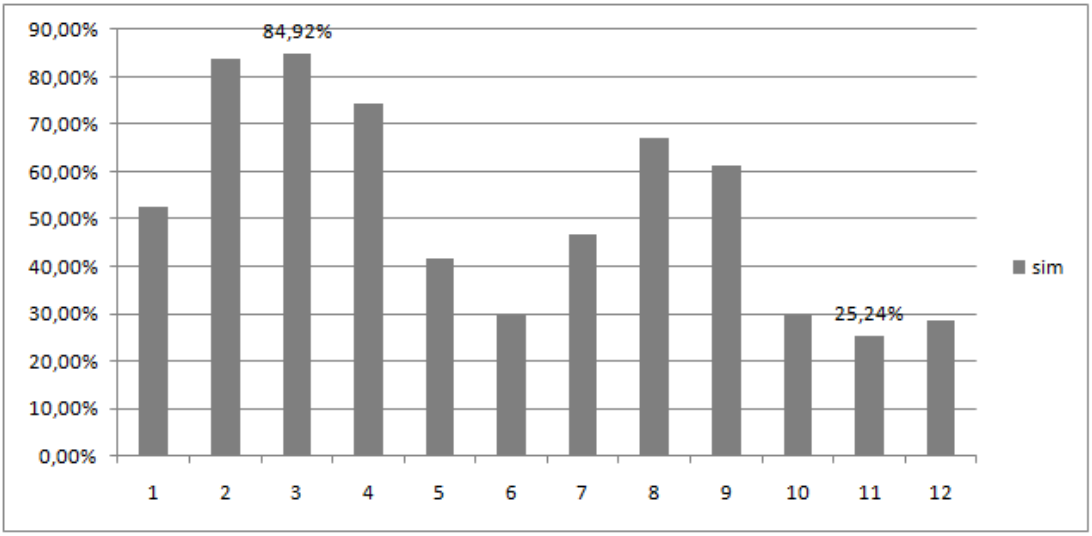


Figura 2. Avaliação do conhecimento em Informática dos alunos de Pombal

Quanto às ferramentas de trabalho ensinadas no curso, 74,43% dos entrevistados afirmou saber utilizar um editor de texto, o que novamente está acima do índice de 60% mostrado no CETIC. A percentagem de alunos que afirmou saber usar o editor de planilhas para elaboração de tabelas é de apenas 41,64% e para a elaboração de gráficos é de 23,90%, ficando abaixo dos 43% estimados pelo CETIC para o uso global desta ferramenta. Quanto ao uso de editor de *slides*, 46,56% dos alunos de Pombal afirmou saber elaborar uma apresentação, provavelmente devido aos recursos visuais presentes nessa ferramenta estimularem o aprendizado da mesma.

Já em relação à habilidade de salvar um arquivo em mídia externa, 66,88% dos alunos afirmou saber realizar tal procedimento. Estima-se que isso aconteça devido à fácil acessibilidade à tecnologia fonográfica, que leva a salvar arquivos de áudio e vídeo nesses meios de armazenamento de dados. Na sequência, em relação a anexar um arquivo ou foto para envio através de e-mail, 61,31% dos alunos afirmou conhecer esse procedimento.

Esse elevado índice pode ser ocasionado pelo crescimento da necessidade de comunicação através deste recurso.

Quando perguntados sobre saber instalar *hardwares* e *softwares* no *Windows*, as respostas afirmativas foram respectivamente 29,83% e 25,24%. Esse baixo índice pode ser justificado devido ao pequeno interesse nesse tipo de recurso, deixando este a encargo de profissionais na área. Finalmente, 29,18% dos alunos informaram não saber usar outro sistema operacional além do *Windows*, como o Linux ou o MAC-OS. Esse fato pode estar ocasionado pela escassez de cursos que utilizem outros sistemas operacionais na cidade de Pombal.

3.2. Projeto de Inclusão Digital

No ano de 2010, o projeto visou beneficiar alunos carentes que estivessem cursando o ensino médio em escolas públicas nas cidades de Pombal. Os conteúdos abordados foram o uso dos recursos básicos do computador e seus conceitos de funcionamento, internet e mecanismos básicos de busca, *Windows* XP, *Word* 2007, *Excel* 2007 e *PowerPoint* 2007.

Foram ministradas aulas teóricas e práticas no laboratório de Informática do Centro de Ciência e Tecnologia Agroalimentar, pertencente à Universidade Federal de Campina Grande. Todos os programas usados para o curso estavam instalados nos 20 computadores existentes e as aulas puderem ser ministradas com um aluno por computador. Como foi planejado oferecer dois cursos por semestre, 80 vagas foram disponibilizadas à comunidade no ano de 2010. Um total de 74 alunos que desejavam adquirir conhecimentos de Informática básica foram formados. Os cursos tiveram carga horária total de 56 h/aula e ocorreram durante os sábados em duas turmas por semestre, das 7 às 11 horas da manhã e das 13 às 17 horas da tarde, totalizando a realização de quatro turmas nesse ano.

Os critérios de seleção dos alunos de segundo grau foram baseados no rendimento acadêmico e renda familiar do aluno. Houve divulgação dos cursos nas escolas públicas e cartazes e formulários de inscrição foram distribuídos nas escolas a serem atendidas para que o máximo possível de estudantes pudesse ser beneficiado.

A dinâmica de interação entre os tutores e a coordenadora era composta de reuniões convocadas sob demanda. Um site[1] foi criado contendo informações de gerenciamento do projeto tais como dados e alocação dos tutores, planejamento do curso e arquivos usados. Para cada sábado, três dos dez tutores disponíveis eram alocados, sendo que somente um era responsável pela integridade dos equipamentos do laboratório e realização da frequência dos alunos.

Com relação a mecanismos de avaliação, estes foram aplicados tanto para a coordenadora como para os tutores e alunos participantes do projeto. A avaliação dos tutores foi realizada de forma contínua ao longo do curso pela coordenadora e pelos alunos participantes. Esta, ao designar atividades para cada tutor, observou continuamente a pró-atividade, desempenho e responsabilidade no cumprimento das tarefas. Para cada tarefa não cumprida, e-mails, contatos telefônicos ou reuniões foram realizados como forma de acompanhar o desenvolvimento do tutor. Várias dessas mensagens postadas do projeto estão disponíveis na lista de discussão do GoogleGroups[2].

[1]Disponível em https://sites.google.com/site/inclusaodigitalufcg/

[2]Disponível em http://groups.google.com/group/informatica-probex-ufcg

Os alunos da rede municipal de ensino realizaram suas avaliações através de questionários cujas perguntas estão mostradas na Figura 3. Estes foram aplicados ao final de cada semestre como forma de que a coordenação do projeto tomasse conhecimento das críticas, sugestões ou elogios. A aplicação dos mesmos foi realizada através de formulário impresso no primeiro semestre e de forma digital através de uma página específica na internet durante o segundo semestre, obtendo grande número de respostas positivas para cada questão.

1. As aulas começam pontualmente?
2. A estrutura do laboratório de Informática é adequada para a realização do curso?
3. Você acha que está entendendo o conteúdo do curso?
4. Você acha que o conteúdo vai ser usado na sua vida profissional?
5. Os monitores são prestativos?

Figura 3. Questionário avaliador dos tutores do projeto

Quando perguntados se a estrutura do laboratório era adequada ao curso e se os monitores eram prestativos, 100% dos alunos responderam positivamente, 88,58% informaram que as aulas começaram pontualmente ao longo do curso e 97,14% deles entenderam que o conteúdo apresentado será importante em suas vidas profissionais. O resumo desses dados é mostrado na Figura 4. Seu eixo horizontal é composto pelos números das questões exibidas na Figura 3 e o eixo vertical exibe a porcentagem de respostas positivasobtidas em cada pergunta.

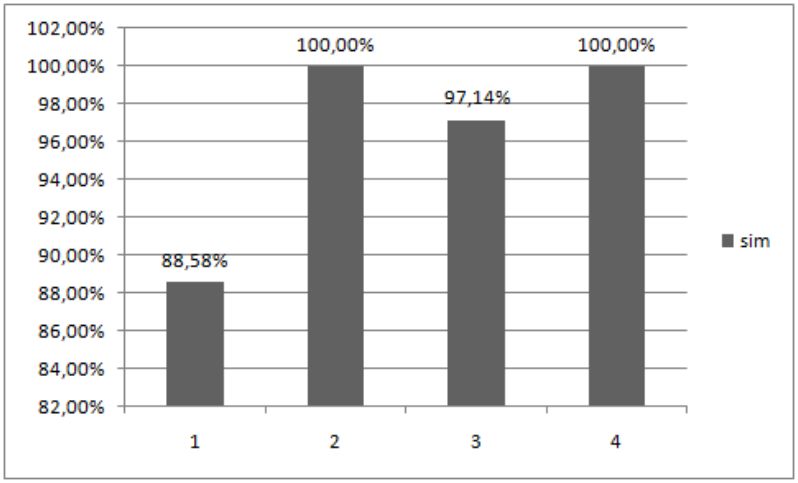


Figura 4. Resumo da avaliação do curso pelos alunos participantes

Os tutores foram os responsáveis pela avaliação dos alunos carentes selecionados para o programa, que foi realizada de forma continuada ao longo do curso. À medida que trabalhos foram realizados em sala de aula, que os alunos mostraram sua participação e que estavam absorvendo o conteúdo mostrado em cada encontro, pontos de participação foram atribuídos a cada um. O aluno carente que obteve frequência superior a 70% recebeu o certificado de participação do curso.

A coordenadora também foi avaliada constantemente pelos tutores através das reuniões periódicas. Ademais, esta também foi avaliada anualmente através de relatórios de status enviados à Universidade a qual ela é vinculada.

4. Conclusões

Cursos que proporcionem a inclusão digital de setores carentes da sociedade no nordeste do Brasil são muito necessários e ao mesmo tempo escassos. O projeto "Inclusão digital na cidade de Pombal"tem o objetivo de proporcionar o conhecimento de Informática Básica gratuitamente aos alunos de escolas públicas de segundo grau da cidade de Pombal, situada no Alto Sertão da Paraíba. São oferecidas aulas de *Windows*, *Word*, *Excel*, *PowerPoint* e internet. Durante o ano de 2010, foram formados 74 alunos de escolas públicas e 11 tutores na área.

Além disso, o projeto também iniciou a identificação do público-alvo que se propôs a atender. Um total de 305 questionários foram aplicados em todas as escolas de ensino da rede de segundo grau da cidade. Com isso, pôde-se constatar que 52,46% dos alunos têm computador em casa, 83,93% têm acesso à internet, 74,43% afirmam saber usar um editor de textos e 70,82% dos alunos não sabem usar outro sistema operacional além do *Windows*. Além disso, também conseguiu-se identificar o nível de conhecimento deles em ferramentas como o editor de textos *Word*, a planilha eletrônica *Excel*, o editor de slides *PowerPoint* e no uso da internet.

No ano de 2011, o projeto continuará a usar o laboratório de informática do Centro de Ciência e Tecnologia Agroalimentar da Universidade Federal de Campina Grande, que passará a ter 50 computadores. Com isso, uma nova expansão poderá ser realizada e será possível oferecer 100 vagas por semestre distribuídas em duas turmas, e consequentemente 200 vagas anuais. A área de atuação passará a ser não só Pombal, mas também os municípios de São Bentinho e São Domingos, que distam até 10 Km desta cidade. Contatos com as prefeituras dessas duas cidades já foram realizados para que estas forneçam transporte para que os alunos possam se deslocar para o Campus de Pombal.

Uma nova aplicação dos questionários deve ser realizada nestas 3 cidades para identificar o conhecimento de Informática básica para os alunos de escolas de segundo grau. A partir dos dados dessa amostra, tratamentos estatísticos devem ser realizados para que uma definição mais completa desse público seja alcançada.

Ademais, uma proposta do projeto com o valor de R$ 27.064,00 também foi submetida ao CNPq sob o edital MCT/CNPq nº 49/2010 (Inclusão Digital e Social) e teve aprovação. Conseguiu-se melhorar a capacitação da mão de obra na cidade de Pombal, e planeja-se oferecer os mesmos benefícios para as três cidades. Além disso, também contribuí-se de forma significativa para o desenvolvimento dos tutores participantes, uma vez que estes praticam continuamente sua didática e oratória durante as aulas. A comunidade local também conhece a Universidade com esse projeto, e há um estreitamento das relações entre ambas as partes.

No que diz respeito à repercussão na comunidade local, esta foi bastante válida. Uma parceria com o SEBRAE (Pombal) foi consolidada, e este se dispôs a fornecer pastas e material de escritório a ser distribuído com as apostilas. A assessoria de imprensa da Universidade a qual a coordenadora é vinculada também estabeleceu contato e divulgou

uma reportagem em seu site [MENDES 2010]. Vários dos resultados alcançados foram citados na reportagem, como o planejamento de expansão do mesmo. Na sequência, o Diário da Borborema [BOR 2010], além de outras fontes, como Paraíba Online, IParaiba e Liberdade 96 FM também divulgaram a notícia.

5. Agradecimentos

Os autores agradecem suas entidades parceiras, em especial ao SEBRAE (Pombal), pela doação de materiais de escritório, ao CNPq por custear o projeto (processo 563489/2010-3) e ao Centro de Ciência e Tecnologia Agroalimentar da Universidade Federal de Campina Grande por permitir o uso do laboratório de informática.

Referências

(2010). Ufcg vai levar inclusão digital a moradores de pombal. http://www.diariodaborborema.com.br/2011/01/02/educacao1_0.php.

(2011). Centro de Estudos sobre as Tecnologias da Informação e da Comunicação. http://www.cetic.br/.

(2011). DATASUS. http://tabnet.datasus.gov.br/cgi/idb2009/matriz.htm.

(2011). Escola de Computação Solidária. http://www.computacaosolidaria.org/.

(2011). Estações digitais na capital capacitam 2 mil pessoas. http://www.joaopessoa.pb.gov.br/noticias/?n=12970.

(2011). Ibge cidades. http://www.ibge.gov.br/cidadesat/.

(2011). Observatório Nacional de Inclusão Digital. http://visualizacoes.onid.org.br/.

Abreu, A. e Nunes, I. (2010). Políticas públicas para acesso às tecnologias de informação: Uma visão sobre a desigualdade social. *Encontro Nacional de Estudantes de Biblioteconomia, Documentação, Gestão e Ciência da Informação*, 33.

Cavalcanti, J. e Santos, S. (2006). Infoinclusão na cidade de Sousa/PB: Uma história de sucesso. *IV Encontro de extensão da UFCG*, 4.

Dodt, L. e Freire, L. (2010). Inclusão digital e e acessibilidade no Brasil. *XXXIII Congresso Brasileiro de Ciências da Comunicação*, 34.

MENDES, G. (2010). UFCG vai levar inclusão digital a moradores de Pombal. http://www.ufcg.edu.br/prt_ufcg/assessoria_imprensa/mostra_noticia.php?codigo=11582.

Rocha, S. (2010). Promovendo a Inclusão Sócio-Digital na Escola Pública: o Projeto Minha Escola, Minha Vida e Suas Implicações no Cotidiano Discente. 2.

Silva, H., Jambeiro, O. e Lima, J., and Brandão, M. (2005). Inclusão digital e educação para a competência informacional: uma questão de ética e cidadania e cidadania. *Ciência da Informação*, 34(1):28–36.

Silva, R. e Meira, W. (2004). Rede de Inclusão e letramento digital. In *Congresso Brasileiro de Extensão Universitária*, volume 2.

Sorj, B. e Guedes, L. (2005). Exclusão digital: problemas conceituais, evidências empíricas e políticas públicas. *Novos Estudos - CEBRAP*, (72):101–117.

Ensino de Algoritmos a Nível Médio Utilizando Música e Robótica: Uma Abordagem Lúdica

Tatyane S. C. da Silva[1], Amanda S. C. da Silva[1], Jeane C. B. de Melo[1]

[1]Universidade Federal Rural de Pernambuco (UFRPE)
Rua D. Manoel de Medeiros, s/n, Dois Irmãos - CEP: 52171-900 – Recife – PE – Brasil

{taty.calixto,amanda.calixto2,jeane.ufrpe}@gmail.com

Abstract. *The computer teaching in elementary and high schools explores mainly the use of computers as a teaching resource for other disciplines. The relevance of this approach is recognized, however, does not explore the teaching of theoretical aspects of Computer Science in schools. This article aims to report a teaching experience that introduce concepts of computer science in a public high school adopting an contextualized way, introducing algorithms notions using music and introduction to programming with the Lego Mindstorms NXT robot.*

Resumo. *O ensino de informática nas escolas a nível médio e fundamental explora principalmente a utilização do computador como recurso didático para outras disciplinas. A relevância dessa abordagem é reconhecida, contudo não explora o ensino de aspectos teóricos de Ciência da Computação nas escolas. Este artigo tem por objetivo relatar uma experiência de ensino que introduz os conceitos de Ciência da Computação no ensino médio de uma escola pública de modo contextualizado, introduzindo noções de algoritmos usando música e introdução a programação com o robô Lego Mindstorms NXT.*

1.Introdução

O papel da informática na educação está cada vez mais difundido nas escolas de nível médio e fundamental. A utilização da computação como ferramenta de apoio ao aprendizado de outras disciplinas através de jogos, *softwares* educacionais e meios digitais de comunicação amplia as possibilidades de metodologias de ensino que permitem explorar os mais diversos assuntos de forma lúdica e atrativa. No entanto, mesmo reconhecendo a sua importância, tal abordagem não costuma promover o ensino da Ciência da Computação por si. A utilização de tais ferramentas explora a visão de usuário do sistema e, conceitos básicos sobre algoritmos e mesmo sobre o modelo computacional, na maioria das vezes, não são tratados, levando grande parte do alunato a possuir uma visão mística do que vem a ser Ciência da Computação [Bell et al 2011].

Portanto, a busca por metodologias inovadoras que permitam trabalhar conceitos de Ciência da Computação no ensino médio é uma constante. Em metodologias de ensino tradicionais, a definição formal de algoritmos é apresentada, seguida da resolução de problemas em pseudocódigo e/ou em uma linguagem de programação. Nestas, observa-se que os discentes apresentam dificuldade de abstração, no uso da sintaxe e se mostram desmotivados a prosseguir os estudos. Na metodologia aqui apresentada, a contextualização dos algoritmos e sua presença em atividades cotidianas são bastante exploradas, levando os alunos a perceberem os algoritmos como uma

forma sistemática de resolver problemas usando um conjunto finito de instruções, buscando utilizar abordagens lúdicas. Descrever esta experiência metodológica, realizada através de uma *Oficina de Algoritmos*, é o objetivo do presente trabalho.

Introduzir conceitos da área básica da Ciência da Computação no ensino médio apresenta um desafio inicial: a denominação das atividades propostas de modo a torná-las atrativas. Embora os algoritmos estejam presentes no cotidiano, não se costuma designá-las através do termo Algoritmo. "AlgoRitmo - Passo a Passo" além de utilizar palavras presentes no vocabulário dos educandos, remete a conceitos musicais, motivando a introdução de tais conceitos e seus algoritmos na oficina proposta.

A utilização da linguagem musical é capaz de promover situações positivas de aprendizagem, pois, além de provocar emoções, a música atua desenvolvendo o raciocínio, a criatividade, a memória, a receptividade sensorial entre outros processos cognitivos. A música incentiva ainda à participação, à cooperação e a socialização, atitudes imprescindíveis no processo de ensino-aprendizagem [Correia 2010].

A resolução sistemática de problemas, utilizando um conjunto finito de instruções, associadas à programação propriamente dita, também foi explorada na oficina. Visando minimizar o impacto que a introdução de uma linguagem de programação provoca, optou-se por aplicar de forma concreta os conceitos ensinados e desenvolver o pensamento algorítmico programando através da robótica.

A robótica é um instrumento para o desenvolvimento do raciocínio reflexivo, da criatividade e do trabalho em grupo sendo escolhida, portanto, como um recurso para ensino de algoritmos [Castilho 2002]. Através da robótica educativa os estudantes podem descobrir e explorar novos caminhos na resolução de problemas, elaborar hipóteses, estabelecer relações e chegar a conclusões [Benitti *et al* 2009].

Esta abordagem para introdução de conceitos de algoritmos no ensino médio é descrita neste trabalho. A metodologia e resultados obtidos com a oficina "AlgoRitmo Passo a Passo", destinada a alunos do 3º ano do ensino médio da Escola Estadual Joaquim Xavier de Brito, localizada em Recife e que participa do Programa Institucional de Bolsas de Iniciação a Docência (PIBID) da UFRPE, são apresentados nas seções subsequentes.

2.Metodologia

Com a finalidade de introduzir os fundamentos e disseminar o ensino de Ciência da Computação a nível médio, a oficina *AlgoRitmo - Passo a Passo* buscou trabalhar o conceito e contextualizar o ensino dos fundamentos de algoritmos com uma abordagem lúdica. A oficina ocorreu no período de dois dias no contra-turno, com duração de três horas. A cada dia buscava-se tratar o conceito de algoritmos, a identificação dos mesmos em atividades cotidianas e então explorar algoritmos computacionais, através da resolução de problemas e programando o robô *Lego Mindstorms NXT*.

Inicialmente, o conceito de algoritmo foi apresentado para os alunos. Então, foi solicitado aos discentes identificar os algoritmos que julgassem se adequar ao conceito apresentado. Dentre os exemplos citados espontaneamente surgiu a receita de bolo, exemplo clássico em cursos introdutórios de algoritmos.

Então, os alunos foram incentivados a resolver problemas algoritmicamente utilizando noções de teoria dos grafos através da atividade: A Cidade Enlameada, a qual

consiste em encontrar uma solução para pavimentar uma cidade utilizando um menor caminho possível para diminuir o custo dessa obra. Qualquer morador precisa ir de sua casa para a casa do vizinho, utilizando apenas estradas pavimentadas [Bell et al 2011]. Essa atividade foi realizada em duplas incentivando o trabalho colaborativo. Ao finalizarem a atividade foi pedido que cada dupla explicasse sua resolução para os colegas. Com isso, novas soluções para o mesmo problema foram compartilhadas, contribuindo com a construção do conhecimento de todos.

No segundo momento da oficina foram introduzidas noções de conceitos musicais, associando a idéia de executar um conjunto de instruções para produzir uma saída desejada. O instrumento escolhido foi o pandeiro, instrumento percussivo melódico, que compreende quatro golpes fundamentais denominados: *open tone*, *finger*, *heel* e *slap*, tratados como o conjunto de instruções. A execução desses golpes no pandeiro produz diferentes sons e o conjunto da execução de golpes de maneira ordenada produz os diversos ritmos. Para cada ritmo há uma ordenação diferente, ou seja, um conjunto de instruções a serem seguidas a fim de produzir a saída esperada.

Para facilitar o aprendizado dos fundamentos, os nomes em inglês foram substituídos por números, simplificando a compreensão do ritmo proposto. O ritmo escolhido foi o samba-enredo, por ser popular e por utilizar os mais elementares fundamentos do pandeiro. Uma vez apresentado o algoritmo do pandeiro, os alunos foram incentivados a tocar o ritmo escolhido, buscando explorar o fato de que nos algoritmos do cotidiano ou na execução de um ritmo é necessário seguir um conjunto de instruções para realizar uma tarefa.

A associação dos algoritmos do cotidiano com algoritmos computacionais foi feita utilizando Robótica. Assim, no segundo dia da oficina foi proposta a utilização do robô *Lego Mindstorms NXT*. Os alunos aprenderam os comandos básicos para programar o robô. Além dos princípios básicos de programação, foi ensinado o que é robótica, pseudocódigo e noções de programação para fazer o robô andar em todas as direções e mostrar mensagens no display.

3.Resultados

Os discentes foram receptivos com a proposta e na realização de todas as atividades. Mesmo sendo realizada no contra-turno e com participação espontânea, a oficina contou com cerca de 20% da turma. Os educandos desconheciam o termo algoritmo e sua definição formal. Após a oficina, eles puderam identificar que os algoritmos estão presentes em seu cotidiano e sua utilidade na resolução sistemática de problemas.

Esta compreensão se refletiu na resolução do problema da Cidade Enlameada. Além de encontrarem a resposta correta, eles conseguiram descrever o modo como a solução foi encontrada. Ao serem questionados sobre a resposta obtida, uma das duplas, mesmo não conhecendo o conceito formal de grafos, resolveu a atividade de forma equivalente à proposta no material. Posteriormente, o conceito abordado no problema da Cidade Enlameada foi contextualizado, buscando mostrar que o pensamento algorítmico é relevante e pode ser aplicado em atividades diárias, como encontrar a melhor rota para percorrer diversos pontos da cidade de Recife, gastando menos tempo e dinheiro.

Os discentes demonstraram bastante aceitação com a utilização de instrumentos percussivos na oficina. Apesar de conhecerem o pandeiro, os alunos não tinham experiência em tocá-lo, e ao serem instruídos a tocar o instrumento percussivo, os

mesmos se sentiram desafiados, vivenciando uma nova experiência que contribuiu para o aprendizado e colaborou para consubstanciar o conceito e aplicações de algoritmos.

Ao programarem o robô *Lego Mindstorms NXT*, os alunos ficaram muito entusiasmados ao ver que o robô executava as tarefas de acordo com o código programado por eles. Além dos exercícios propostos para essa atividade, os discentes, estimulados pela curiosidade e interesse em conhecer mais sobre as funcionalidades do robô, exploraram novas maneiras de programá-lo.

4.Conclusões

Este trabalho consistiu em descrever uma experiência de ensino de Ciência da Computação no ensino médio, utilizando recursos de robótica e conceitos musicais. Os alunos se mostraram bastante receptivos com as atividades propostas, compreendendo o conceito de algoritmos, percebendo sua aplicação na resolução de problemas e contextualizando os mesmos em suas atividades cotidianas.

A utilização de conceitos musicais, relacionando-os com o conceito de algoritmos, corroborou para uma melhor compreensão da definição de um conjunto de instruções para resolver um problema ou executar uma tarefa. A robótica computacional permitiu a introdução de algoritmos computacionais e noções de programação de uma forma lúdica, conforme relato dos próprios educandos, mostrando-se adequada para o ensino de algoritmos. Uma diretiva futura é utilizar a programação na robótica para trabalhar problemas clássicos da Ciência da Computação. A proposta é utilizar os aspectos lúdicos relacionados às atividades que envolvem informática de modo a favorecer o ensino desta ciência cada vez mais presente no contexto social.

5.Agradecimentos

Este trabalho foi realizado com o apoio do Programa Institucional de Bolsa de Iniciação à Docência - PIBID, da CAPES – Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil.

Referências

Bell,T.,Witten,I.H. e Fellows,M. (2011) “Computer Science Unplugged. Ensinando Ciência da Computação sem o uso do computador”. Tradução por: Luciano Porto Barreto. Disponível em: http://csunplugged.org/books. Acesso: Março de 2011.

Castilho, M.I.(2002) “Robótica na Educação: Com que objetivos?” (Monografia de Especialização em Informática na Educação) - Universidade Federal do Rio Grande do Sul. Disponível em: http://www.pgie.ufrgs.br/alunos_espie/espie/mariac/public_html/robot_edu.html. Acesso: Março de 2011.

Benitti, F. B. V., Vahldick,A., Urban, D. L., Krueger, M. L. e Halm, A. (2009). “Experimentação com Robótica Educativa no Ensino Médio: ambiente, atividades e resultados”. WIE - XV Workshop sobre Informática na Escola. Disponível em: http://www.inf.furb.br/dsc/download/ciesc2010_submission_16.pdf. Acesso: Maio de 2011

Correia,M.A.(2010) “A Função Didático-Pedagógica da Linguagem Musical: Uma Possibilidade na Educação”. Educar. Rev. n.36. Paraná Editora UFPR.

WIM - XI Workshop de Informática Médica

PREFÁCIO

O Workshop de Informática Médica (WIM) reúne pesquisadores, estudantes, professores, empresários e profissionais interessados em pesquisas originais e aplicações inovadoras de computação na área da saúde. A difusão das novas tecnologias no âmbito da saúde incentivou o crescimento da interdisciplinaridade entre pesquisadores da ciência da computação e áreas afins com profissionais da área da saúde. Esta integração abre novas perspectivas de pesquisas com impactos sociais significativos, principalmente, no domínio educacional, nas práticas terapêuticas e no acesso e transmissão de informações.

O WIM 2011, 11ª edição do evento, ocorre de 19 a 20 de julho no Centro de Convenções de Natal-RN. O WIM é um evento-satélite patrocinado, organizado e apoiado pela SBC (Sociedade Brasileira de Computação) no CSBC 2011 e apoiado pela Sociedade Brasileira de Informática em Saúde (SBIS). As atividades no WIM compreendem apresentação de artigos completos e artigos resumidos relatando trabalhos em andamento, além de palestras convidadas e mesas-redondas, distribuídos ao longo de dois dias do CSBC 2011.

Os trabalhos aceitos para publicação no WIM 2011 foram divididos em duas categorias: completos e resumidos. O volume dos Anais reúne 30 artigos selecionados a partir de 105 submissões que contam com a participação de praticamente todos os estados brasileiros e alguns trabalhos submetidos por pesquisadores do exterior. Deve ser salientado o enorme esforço do Comitê de Programa do WIM 2011 para realizar a seleção dos trabalhos a serem apresentados e publicados dentre tantas submissões de excelente qualidade. Agradecemos a todos os autores que submeteram seus artigos e congratulamos aqueles que tiveram seus trabalhos aceitos.

Por último, gostaríamos de agradecer aos membros dos comitês de organização e de programa do CSBC, bem como aos revisores, pelo trabalho voluntário de fundamental importância para a realização do WIM 2011. Desejamos que o evento traga frutos para o trabalho de todos.

Abraços !!!

Alessandra Alaniz Macedo, FFCLRP-USP (Coordenadora Geral)
Saulo Bortolon, DI-UFES (Coordenador de Comitê de Programa)
Selan Rodrigues dos Santos, DIMAp-UFRN (Coordenador Local)

COMISSÕES

Coordenação Geral

Alessandra Alaniz Macedo, FFCLRP-USP (coordenadora geral)
Saulo Bortolon, DI-UFES (coordenador de Comitê de Programa)
Selan Rodrigues dos Santos, DIMAp-UFRN (coordenador local)

Comitê de Programa

Adriano Andrade - UFU
Alessandra Macedo - FFCLRP-USP
Alexandra Monteiro - UERJ
Andreia Malucelli - PUCPR
Antônio Tadeu Azevedo Gomes - LNCC
Aristófanes Silva - UFAM
Arlindo da Conceição - ICT-UNIFESP
Artur Ziviani - LNCC
Carine Webber - Universidade de Caxias do Sul
Carla Freitas - UFRGS
Claudia Varassin - UFES/UFF
Claudio Giulliano - SBIS
Clever Farias - FFCLRP/USP
Crediné de Menezes - UFES
Cristine Gusmão - UFPE
Daniel Weingaertner - UFPR
Daniela Trevisan - UFF
Edilson Ferneda - Universidade Católica de Brasília
Edson Scalabrin - PUC-PR
Eduardo Albuquerque - UFG
Evandro O. T. Salles - UFES
Evandro Ruiz - USP
Fatima Nunes - EACH-USP
Felipe Bergo - Unicamp
Fernanda Campos - Universidade Federal de Juiz de Fora
Fernando Koch - University of Utrecht
Giancarlo Guizzardi - UFES
Hélio Guardia - UFSCar
Iara Augustin - UFSM
Ildeberto Rodello - FEARP-USP
Iwens Sene Jr - UFG
Joaquim Felipe - USP
José Augusto Baranauskas - USP
Jose Brega - UNESP Bauru
Jose Camacho-Guerrero - Innolution
Julius Leite - UFF
Klaus Côco - UFES
Leandro Oliveira - UFG
Leila Silva - Universidade Federal de Sergipe

Liliane Machado - UFPB
Lourdes Brasil - UNB
Lucas Veronese - UFES
Luciana Cavalini - UFF
Luciano Romero Soares de Lima - Rede Sarah de Hospitais de Reabilitação
Luciano Silva - UFPR
Luis Murta Jr. - USP
Marcelo CostaOliveira - UFAL
Marcelo Zuffo - USP
Márcia Ito - LaPCiS/POS-CEETEPS
Marco Aurelio Benedetti - UFPE
Marco Goldbarg - UFRN
Maria Luiza Campos - UFF
Mariella Berger - UFES
Mario Dantas - UFSC
Orlando Loques - UFF
Patricia Dockhorn Costa - UFES
Paulo de Azevedo-Marques - USP
Paulo Eduardo Ambrósio - Universidade Estadual de Santa Cruz
Regina Braga - Universidade Federal de Juiz de Fora
Renato Bulcão Neto - UFG
Renato Tinós - USP
Rodrigo Andreão - IFES
Rosa Maria Costa - UERJ
Rosalie Belian - UFPE
Saulo Bortolon - UFES
Selan dos Santos - UFRN
Sergio Freire - UERJ
Sílvio Cazella - Unisinos
Tatiana Tavares - UFPB
Thomas Rauber - UFES
Vera Werneck - UERJ
Wanderley Lopes de Souza - UFSCar

ARTIGOS

Tortuosidade da Rede do Osso Trabecular a partir da Reconstrução Geodésica de Imagens Binárias Tridimensionais

Waldir L. Roque[1], **Katia Arcaro**[1] **e Iuri Freytag**[1]

[1]Programa de Pós-Graduação em Matemática Aplicada
Instituto de Matemática
Universidade Federal do Rio Grande do Sul (UFRGS)
Porto Alegre – RS – Brazil

roque@mat.ufrgs.br

Abstract. *Osteoporosis is a disease causing bone mass loss and quality deterioration. The trabecular bone plays a very important role for the bone quality and mechanical competence of the scaffold. It is known that the trabecular microarchitecture can improve the fracture risk prediction up to around 90%. Several parameters seem to be correlated providing structural details, however, the tortuosity of the trabecular network has not yet been seriously taken into account and its contribution has not been fully investigated and understood. Here we discuss and estimate the tortuosity based on the geodesic length and geodesic reconstruction for radio-distal bone samples. It is shown that the trabecular network tortuosity provides evidence of structural anisotropy and of its connectivity level.*

Resumo. *A osteoporose é uma doença que causa perda de massa e deterioração óssea. O osso trabecular é muito importante para a qualidade e competência mecânica do esqueleto. Sabe-se que a microarquitetura trabecular pode elevar a predição do risco de fratura para cerca de 90%. Vários parâmetros parecem estar correlacionados, estabelecendo detalhes estruturais, entretanto, a tortuosidade da rede trabecular ainda não foi seriamente levada em consideração e sua contribuição não foi completamente investigada e compreendida. Aqui é discutida e estimada a tortuosidade baseada no cálculo da distância e na reconstrução geodésica para amostras do osso rádio-distal. Os resultados mostram que a tortuosidade da rede trabecular provê evidências da anisotropia estrutural e do nível de conectividade.*

1. Introdução

Com o aumento da longevidade, a incidência da osteoporose vem se tornando um problema não apenas de natureza clínica mas, também, de natureza socioeconômica. A principal consequência da osteoporose é a ocorrência de fraturas e fissuras ósseas que, além de causar desconforto, muitas vezes necessitam de tratamentos e internações hospitalares, contribuindo, muitas vezes, para a degradação da saúde do paciente idoso, aumentando a morbidade e podendo levá-lo a óbito. Todo este processo envolve custos elevados, quer do setor público ou privado, sendo estimado em cerca de 25 bilhões de dólares os custos com a osteoporose nos EUA para 2025 [Burge et al. 2007]. No Brasil, dados indicam que um em cada três pacientes com fratura no osso ilíaco são diagnosticados como tendo osteoporose e, entre estes, apenas um em cada cinco recebem algum

tipo de tratamento. Os custos para o sistema de saúde privado no Brasil, apenas com a fratura osteoporótica do fêmur, foi da ordem de 12 milhões de reais no período de um ano, entre julho de 2003 e junho de 2004 [Araújo et al. 2005]. Neste sentido, torna-se, de fato, muito importante que se desenvolvam técnicas que possam auxiliar a predição do risco de fratura de um paciente de forma não invasiva e a um custo moderado.

A osteoporose é uma doença caracterizada pela perda de massa óssea e deterioração da estrutura interna do osso canceloso. O osso canceloso é essencialmente formado por uma rede de trabéculas e pela medula, sendo envolvido por uma parede densa chamada osso cortical; por ser metabolicamente mais ativo [Wesarg et al. 2010], o osso trabecular é mais vulnerável do que o osso cortical. O padrão ouro para o diagnóstico da osteoporose é a Densitometria Mineral Óssea (DMO), mas ela é responsável por apenas 70-80% da variação da resistência óssea [Homminga et al. 2003]; por outro lado, estudos recentes indicam que a microarquitetura trabecular eleva este valor para cerca de 90% [Carbonare et al. 2005, Accardo et al. 2005]. Em geral, valores baixos de DMO sinalizam risco de fratura, porém alguns pacientes com valores considerados moderados de DMO apresentam quadros de traumas ósseos graves [Wesarg et al. 2010, Homminga et al. 2003]. Como a DMO é uma medida que engloba toda a estrutura óssea (osso cortical e osso canceloso), esta não é capaz de detectar as deficiências e fragilidades na estrutura do osso trabecular.

Com a evolução de recursos tecnológicos para exames por imagem, características histomorfométricas e geométricas da estrutura trabecular, como a fração volumétrica, espessura trabecular, conectividade, propriedades elásticas, etc, vêm sendo amplamente exploradas e a microtomografia computadorizada (μCT) expandiu ainda mais este horizonte. Em um trabalho recente Roque *et al.* [Roque et al. 2010] mostram uma correlação positiva entre a fração volumétrica de osso trabecular, a Característica de Euler-Poincaré, que estima o grau de conectividade da estrutura, e o módulo de Young, para sua elasticidade; estes parâmetros são importantes para a competência mecânica do osso. Por outro lado, a estrutura trabecular pode ser vista como uma rede de vigas e pilares, que proporcionam a resistência mecânica da estrutura quando sujeita a cargas e tensões. Porém estas vigas e pilares apresentam-se de forma sinuosa, devendo tal característica ser levada em consideração quando da estimativa da competência mecânica do osso trabecular.

Em meios porosos, a tortuosidade τ é uma medida do grau de sinuosidade de um poro, porém a tortuosidade geométrica é um conceito definido para uma estrutura filamentosa qualquer. A rede trabecular apresenta-se de forma sinuosa e, portanto, parece existir uma relação entre sua tortuosidade e demais parâmetros, o que deve ser melhor investigado. A literatura da área ainda não dispõe de um estudo sistemático sobre a tortuosidade trabecular a partir de imagens e sua contribuição para a competência mecânica ainda é pouco conhecida. Alguns trabalhos recentes discutem a importância da tortuosidade da cavidade medular para a estrutura do osso canceloso, utilizando técnicas de ultrassom para estimá-la em algumas amostras *in vitro* de ossos bovino e humano [Attenborough et al. 2005, Hughes et al. 2007, Aygün et al. 2009].

Neste trabalho é apresentada e discutida a aplicação de um algoritmo para estimar a tortuosidade da estrutura trabecular, o qual foi empregado em imagens 2D e 3D, obtidas por microtomografia computadorizada. O trabalho está organizado da seguinte forma: a Seção 2 traz subsídios teóricos: define-se o conceito de tortuosidade e descreve-se o al-

goritmo da reconstrução geodésica; na Seção 3, são apresentados os materiais e métodos; na Seção 4, estima-se a tortuosidade para o conjunto de amostras do osso rádio-distal e discute-se os resultados obtidos; finalmente, na Seção 5, apresenta-se a conclusão do trabalho, mencionando-se alguns desafios futuros.

2. Embasamento Teórico

Esta seção destina-se a prover subsídios teóricos para o embasamento dos conceitos e técnicas utilizados.

2.1. Tortuosidade

Em um meio poroso a *tortuosidade* τ desempenha um papel importante na descrição da estrutura e apresenta diferentes conceitos [Clennell 1997]. Entretanto, matematicamente, a tortuosidade caracteriza a sinuosidade de um filamento. Neste trabalho, este conceito será aplicado às trabéculas. Geometricamente, a tortuosidade τ é definida como

$$\tau = \frac{L_G}{L_E}, \tag{1}$$

onde L_G é a distância geodésica entre dois pontos conexos na rede trabecular sem transpassar outras fases (cavidade medular), e L_E é a distância Euclideana. Na literatura, alguns autores consideram a distância Euclideana na equação (1) como sendo a distância em linha reta entre dois pontos do poro, independente da geometria das fases. No entanto, outros consideram a distância Euclideana como a distância entre dois planos paralelos que delimitam a estrutura [Wua et al. 2006]. A Figura 1 ilustra ambos os casos.

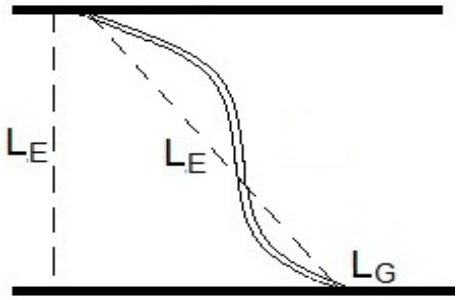


Figura 1. Ilustração da distância geodésica L_G e da distância Euclideana L_E, tomada entre planos e entre pontos.

O cálculo da tortuosidade sobre o espaço imagem requer o desenvolvimento de algoritmos que calculem a tortuosidade de um poro com base nos pixels que o representam. A técnica adotada neste trabalho está fundamentada no princípio da reconstrução geodésica proposta em Gommes et al. [Gommes et al. 2009], que busca associar um valor de tortuosidade para toda a rede de poros na estrutura de rochas.

Para descrever a técnica, considere-se uma imagem 3D da estrutura trabecular e um sentido de referência na direção positiva do eixo z, denominada $z+$. A estrutura é varrida por um plano normal ao sentido $z+$ e, utilizando-se o procedimento da reconstrução geodésica (ver Subseções 2.2 e 2.3), estima-se a distância geodésica L_G de uma trabécula, a partir do plano varredor inicial até uma fatia m, como sendo o número

n de reconstruções geodésicas necessárias para alcançar a fatia m. As reconstruções geodésicas recobrem todos os voxels que correspondem às trabéculas conexas às da primeira fatia. Por outro lado, a distância Euclideana L_E é determinada pelo número de fatias obtidas pela propagação paralela do plano varredor. O número n de reconstruções geodésicas necessárias para recobrir todas as trabéculas conexas até uma fatia m é tal que $m \leq n$, com a igualdade ocorrendo apenas quando a rede trabecular é perpendicular ao plano varredor.

Como para uma certa distância Euclideana $L_E(m)$ tem-se diversos valores de distâncias geodésicas, já que são necessárias mais reconstruções do que o número de fatias para recobrir a rede trabecular até a fatia m, tem-se uma distribuição $\rho(L_E, L_G)$ de distâncias Euclideana e geodésica, onde $\rho\, dL_E\, dL_G$ é proporcional ao número de pontos com distância Euclideana $[L_E, L_E + dL_E]$ e distância geodésica $[L_G, L_G + dL_G]$. A Figura 2**a)** mostra a distribuição para toda extensão na direção $z+$ para a amostra 266. Para estimar a tortuosidade da rede trabecular, define-se $\langle L_G(m) \rangle$ como a distância geodésica média para uma certa distância Euclideana $L_E(m)$. O gráfico de dispersão dos pontos $(L_E, \langle L_G \rangle)$ pode ser facilmente obtido a partir das imagens e, assim, a tortuosidade pode ser estimada como a inclinação da reta de melhor ajuste para a distribuição das distâncias, como é apresentado na Figura 2**b)**.

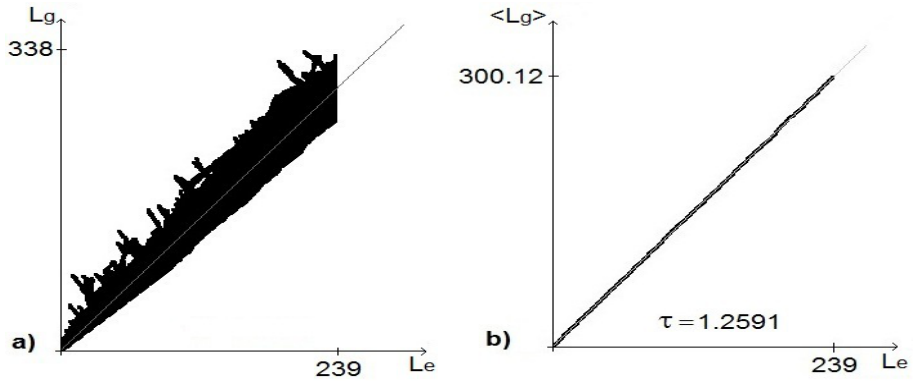


Figura 2. a) Gráfico da distribuição de distâncias e b) gráfico de $L_E \times \langle L_G \rangle$, com o valor da tortuosidade para a amostra 266.

2.2. Reconstrução Geodésica em Imagens Bidimensionais

A técnica da reconstrução geodésica (RG), que pode ser aplicada a imagens bi ou tridimensionais, é um procedimento que reconstrói a imagem do espaço de poros pixel a pixel, crescendo sempre na direção ao longo do poro. Sendo assim, para reconstruí-lo geodesicamente, o custo computacional depende do quão sinuoso ele é; quanto mais sinuoso o poro, mais reconstruções são necessárias para recobrir a imagem. A Figura 3 delineia o princípio da reconstrução geodésica para uma imagem binária I quadrada com 16 pixels, utilizando um elemento estruturante de 4-vizinhos (EE_4).

Como mencionado anteriormente, as trabéculas estão sendo consideradas como se fossem poros, sendo representadas por pixels brancos, enquanto seu complemento está ilustrado em cinza, para melhor visualização. O algoritmo inicia copiando-se a primeira coluna com pixels brancos, denotada por r_0. Em seguida, a operação usual de dilatação é realizada com o elemento estruturante ($\oplus EE_4$), o que resulta na imagem d_1. A imagem

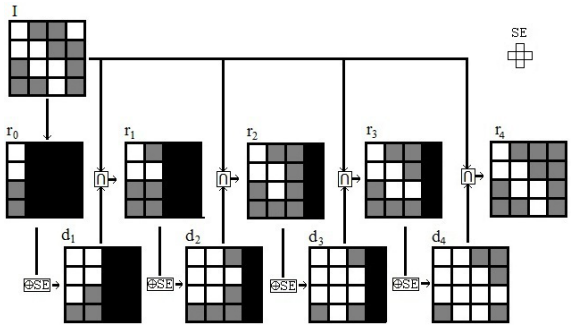


Figura 3. Esquematização do funcionamento do algoritmo de RG para uma imagem bidimensional 4×4 pixels. Note que o pixel isolado no canto superior direito é eliminado durante o procedimento.

d_1 é, então, intersecionada com a imagem inicial, I, resultando em r_1. Aplicando-se sobre a imagem r_1 a operação de dilatação, resulta em d_2, que é novamente intersecionada com I, obtendo-se r_2. O procedimento é repetido até quando uma imagem dilatada resultar em interseção vazia com I. A operação interseção de imagens, $\cap$, é realizada do modo usual para imagens binárias, isto é, um pixel assume valor 1 (branco) se e somente se em ambas as imagens ele tiver valor 1.

A RG 2D ilustrada na Figura 4**a)** é realizada tomando-se uma linha de varredura partindo de $x = 0$ e varrendo a imagem na direção de x positivo. Em **b)**, a linha de varredura parte de $x = m$, onde m é o número de colunas, e varre na direção negativa de x. Analogamente para a direção y.

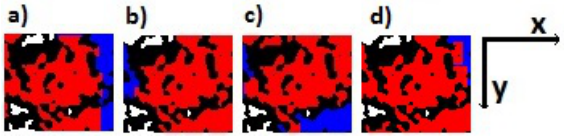


Figura 4. RG de uma amostra trabecular 2D (68×60) nas direções a) $x+$, b) $x-$, c) $y+$ e d) $y-$. Em vermelho, o que foi reconstruído até a reconstrução de número 60. Em branco, trabéculas isoladas, ignoradas pelo algoritmo por não terem conectividade com a linha de varredura inicial. Em azul, a RG após a 60ª reconstrução. Em c) fica claro que o algoritmo volta na direção $y-$ para recobrir toda a trabécula.

2.3. Reconstrução Geodésica em Imagens Tridimensionais

Para imagens tridimensionais, o procedimento da RG segue exatamente os mesmos passos, substituindo-se agora a linha de varredura por um plano varredor, a coluna inicial por um plano inicial e contando-se com um elemento estruturante do tipo 6-vizinhos (EE_6). A Figura 5 mostra um exemplo tridimensional de rede trabecular (amostra 266) que está sendo geodesicamente reconstruída. As trabéculas que já o foram, aparecem em destaque (vermelho) na figura.

3. Materiais e Métodos

Nesta seção são apresentados os materiais e métodos utilizados para o desenvolvimento do trabalho. O estudo de caso está baseado em imagens 3D geradas por μCT

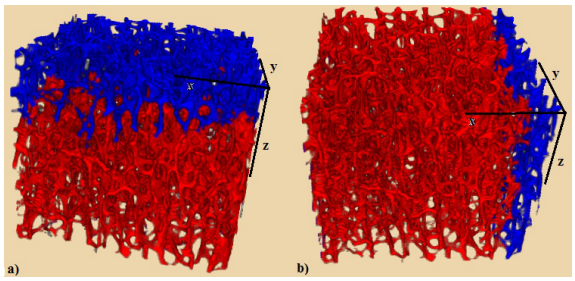


Figura 5. RG da amostra 266 nas direções a) $z-$ e b) $x-$. Em vermelho, as RG até o número 240 e, em azul, as RG após a 240ª reconstrução.

do osso trabecular do rádio (pulso), as quais foram obtidas *in vitro* de 15 indivíduos. As amostras, com tamanho lateral de 12mm, foram coletadas a uma distância média de $9,75$mm a partir da extremidade distal do rádio, tiveram selecionadas regiões de interesse (ROI) com tamanhos que variam de acordo com a análise clínica do material, e foram imageadas com o scanner microCT-20 (Scanco Medical, Brüttisellen, Switzerland) com um voxel isotrópico de 34μm de lado. Para remoção de ruído, as imagens 3D μCT foram filtradas com uso de um filtro Gaussiano 3D. Em cada caso, os histogramas em escalas de cinza das imagens filtradas continham dois picos, correspondendo à medula e ao osso; as imagens foram, então, binarizadas usando um limiar global igual ao mínimo entre os dois picos. A direção crânio-caudal foi identificada com o eixo z de cada amostra. Detalhes adicionais com respeito à preparação das amostras e protocolos de aquisição das imagens estão descritos no trabalho de Laib et al. [Laib et al. 2001].

Para o estudo da tortuosidade, foram considerados os 15 conjuntos de imagens μCT, tomando-se 239 fatias em cada conjunto, com ROIs formadas por 212×212, 237×237, 242×242, 252×252 e 257×257 pixels; as 10 outras possuem 268×268 pixels. As matrizes binárias das imagens foram geradas com o programa *OsteoImage* [Roque et al. 2005], o qual vem sendo desenvolvido pelos autores especialmente para análise de imagens trabeculares. As matrizes binárias são utilizadas para a reconstrução geodésica (Subseções 2.2 e 2.3).

A RG foi efetuada nos seis sentidos principais do volume do osso trabecular das 15 amostras, ou seja, nas direções de varredura $z+$, $z-$, $y+$, $y-$, $x+$ e $x-$. Primeiramente, são calculados os valores das distâncias Euclideana $L_E(m)$ e geodésica $L_G(m)$, para $m = 1, \ldots\ldots, N$, onde N corresponde ao número total de fatias; em seguida os valores médios $\langle L_G(m) \rangle$ são computados e o ajuste linear dos valores $(L_E, \langle L_G \rangle)$ é considerado, obtendo-se, portanto, o valor de tortuosidade (ver Subseção 2.1) referente a cada sentido de cada amostra.

As imagens tridimensionais presentes neste trabalho foram geradas com o software livre *ImageJ* (http://rsbweb.nih.gov/ij/).

4. Resultados e Discussões

A Tabela 1 apresenta os valores obtidos para as tortuosidades nos seis sentidos, para cada uma das 15 amostras. Vê-se que, em alguns casos, as tortuosidades diferem significativamente em uma mesma direção para uma mesma amostra. Isso decorre do fato do algoritmo reconstruir algumas trabéculas distintas em cada sentido, que depende da conectivi-

Tabela 1. Tortuosidades estimadas nos seis sentidos das 15 amostras.

Amostra	τ_{z+}	τ_{z-}	τ_{y+}	τ_{y-}	τ_{x+}	τ_{x-}
254	1.2895	1.2779	1.4726	1.4569	1.6514	1.6847
255	1.2618	1.2749	1.4693	1.4556	1.6534	1.6137
256	1.2711	1.2807	1.5870	1.5317	1.6582	1.6973
262	1.2574	1.2486	1.5057	1.4857	1.6496	1.5766
263	1.3159	1.3120	1.4512	1.4046	1.6659	1.7408
264	1.2686	1.2722	1.4776	1.4271	1.5043	1.5046
265	1.1716	1.1668	1.4345	1.4118	1.4608	1.4664
266	1.2591	1.2500	1.4459	1.4126	1.6284	1.6075
267	1.2452	1.2478	1.4881	1.4676	1.5739	1.5183
268	1.4453	1.4403	1.5571	1.5316	1.8812	1.8237
269	1.3108	1.2853	1.6766	1.5668	2.0818	2.2224
270	1.3207	1.2792	1.6012	1.5632	1.7564	1.6771
271	1.2541	1.2384	1.5541	1.5222	1.5471	1.6465
272	1.3377	1.3026	1.5686	1.5575	1.7688	1.8355
273	1.4291	1.4290	1.6344	1.5358	2.1542	2.0394

dade trabecular ocorrer da primeira fatia até a última, mas também da forma da estrutura trabecular, que pode apresentar um maior ou menor número de discos ou filamentos na direção de propagação do plano varredor. Definindo-se o desvio das tortuosidades direcionais (DTD) como $\Delta\tau_k = |\tau_{k+} - \tau_{k-}|$, onde $k = x, y, z$ e calculando este desvio para a amostra 269 na direção x, por exemplo, obtém-se $\Delta\tau_x = |\tau_{x+} - \tau_{x-}| = 0.1406$, enquanto para a amostra 273, na direção z, tem-se $\Delta\tau_z = |\tau_{z+} - \tau_{z-}| = 0.0001$, que representam o maior e o menor valor de DTD entre todas as amostras.

Os DTDs indicam também que a rede trabecular apresenta uma certa anisotropia na sua distribuição, ou seja a rede trabecular apresenta-se com uma orientação preferencial que difere da direção de propagação do plano de varredura. Notou-se que os menores DTDs ocorrem na direção z, que para as amostras significam a direção distal-proximal do rádio. Esta baixa anisotropia no DTD é um indicativo de que a rede trabecular procura alinhar-se na direção na qual está submetida às maiores frequências de tensões e compressões [Tabor 2009]. Isto vem ao encontro da nossa conjectura de que a tortuosidade e a anisotropia devem desempenhar um papel importante na competência mecânica da estrutura trabecular e, por isso, devem ser detalhadamente investigadas.

Ainda observando os resultados na Tabela 1 e as representações volumétricas das amostras 266, 268 e 269 (Figura 6 **a)**, **b)**, **c)**, respectivamente), nota-se que a amostra 266 apresenta uma boa conectividade trabecular e baixo DTD; a amostra 268 apresenta conectividade e DTD intermediários. Já a amostra 269 apresenta uma baixa conectividade em uma região da rede trabecular e valores elevados dos DTDs, em consequência das trabéculas que ainda estão conectadas apresentarem grandes distâncias geodésicas, fato decorrente da existência de uma grande cavidade medular (porosa).

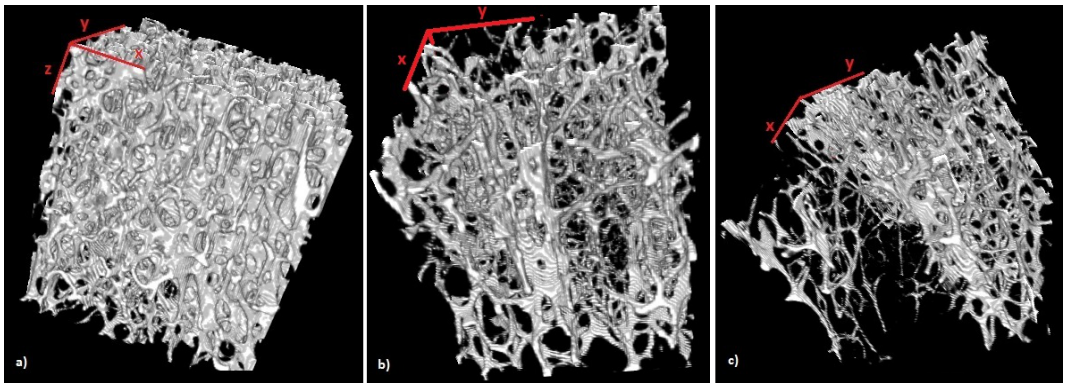


Figura 6. a) Visualização 3D da amostra 266, a qual apresenta uma boa conectividade trabecular e baixo DTD. b) Visualização 3D da amostra 268, a qual apresenta conectividade e DTD intermediários da estrutura trabecular. c) Visualização 3D da amostra 269, a qual apresenta grande cavidade medular, baixa conectividade e acentuado DTD.

5. Conclusões

O aumento da longevidade, se por um lado traz benefícios, por outro traz consigo doenças como a osteoporose, que é associada ao envelhecimento. A osteoporose causa a perda de massa óssea e a degradação da estrutura do osso trabecular, comprometendo, assim, a resistência mecânica da estrutura. Por ser uma doença prevalente e com elevado custo de tratamento, especialmente nos casos de fraturas ósseas, a mesma tem chamado a atenção não apenas da comunidade científica, mas também dos sistemas público e privado de saúde.

O osso trabecular desempenha um papel importante para a competência mecânica da estrutura, por isso o interesse em estudar e compreender sua microarquitetura é fundamental. Neste trabalho, a tortuosidade da rede trabecular foi estimada através da técnica da reconstrução geodésica aplicada à fase sólida do osso canceloso. A aplicação da reconstrução geodésica, a partir de imagens μCT, permitiu computar a distância geodésica da rede trabecular e, desta forma, estimar a tortuosidade τ nos seis sentidos do volume. A aplicação da técnica foi feita para 15 amostras do rádio-distal, cujos valores das tortuosidades estão na Tabela 1. Os resultados obtidos deixam claro que há uma variação no grau de sinuosidade da rede trabecular com relação à direção do plano varredor. Notadamente, através da análise dos desvios da tortuosidade direcional, pôde-se perceber a existência de uma anisotropia na tortuosidade da rede em relação às direções de propagação do plano varredor. Em particular, vê-se que as menores anisotropias na tortuosidade para todas as amostras ocorrem na direção z, ou seja, na direção distal-proximal do rádio. Este resultado corrobora o fato de que as trabéculas tendem a alinhar-se na direção de maiores tensões e compressões; em outras palavras: as tensões/compressões tendem a diminuir a sinuosidade das trabéculas nessas direções. Além disso, a partir da visualização 3D das estruturas, pôde-se conjecturar a existência de uma relação entre os DTDs, em cada amostra, e o grau de conectividade apresentado pela rede trabecular. Isso abre espaço para uma investigação mais detalhada da relação entre medidas de conectividade (Característica de Euler-Poincaré, por exemplo) e DTD.

O desenvolvimento de técnicas para estimar a tortuosidade do osso canceloso

ainda é algo bastante recente. Em [Attenborough et al. 2005], a tortuosidade trabecular foi obtida para amostras de réplicas do ilíaco, fêmur, vétebras lombares e calcâneo, utilizando pulsos de audiofrequência. No entanto, a estimativa da tortuosidade da rede trabecular a partir de imagens, como realizada neste trabalho, é algo inédito. Embora as metodologias e os sítios das amostras tenham sido distintos, os valores obtidos aqui estão consistentes com o intervalo de valores apresentados em [Attenborough et al. 2005].

O próximo passo será investigar as correlações entre a tortuosidade, a Característica de Euler-Poincaré, que estima a conectividade da estrutura, o Módulo de Young, que estima sua elasticidade, e a fração volumétrica, que estima a densidade mineral trabecular por unidade de volume. Todos estes parâmetros podem contribuir de forma significativa para a competência mecânica da estrutra óssea e, por conseguinte, para o estabelecimento de critérios que viabilizem uma melhor predição do risco de fratura, possibilitando um diagnóstico mais acurado e tratamento preventivo, diminuindo os transtornos e custos inerentes aos traumas ósseos.

Agradecimentos

W. L. Roque agradece ao Dr. Z. Tabor pela concessão das imagens μCT das amostras e por inúmeros debates técnicos. K. Arcaro agradece à CAPES pela bolsa de doutorado e Iuri Freytag pela bolsa do programa PICME/CNPq. Agradecemos ainda aos revisores pelos comentários e sugestões.

Referências

Accardo, A. P., Strolka, I., Toffanin, R., and Vittur, F. (2005). Medical imaging analysis of the three dimensional (3D) architecture of trabecular bone: techniques and their applications. In Leondes, C. T., editor, *Medical Imaging Systems Technology - Methods in General Anatomy*. World Scientific Publishing Co. Pte. Ltd., Singapura.

Araújo, D. V., Oliveira, J. H., and Bracco, O. L. (2005). Custo da fratura osteoporótica de fêmur no sistema suplementar de saúde brasileiro. *Arq. Bras. Endocrinol Metabol*, 49:897–901.

Attenborough, K., Shin, H.-C., Fagan, M. J., and Langton, C. M. (2005). Measurements of tortuosity in stereolithographical bone replicas using audiofrequency pulses (L). *J. Acoust. Soc. Am.*, 118:2779–2782.

Aygün, H., Attenborough, K., Postema, M., Lauriks, W., and Langton, C. M. (2009). Prediction of angle dependent tortuosity and elasticity effects on sound propagation in cancellous bone. *J. Acoust. Soc. Am.*, 126:3286–3290.

Burge, R., Dawson-Hughes, B., Solomon, D. H., Wong, J. B., Kin, A., and Tosteson, A. (2007). Incidence and economic burden of osteoporosis-related fractures in the United States: 2005 - 2025. *Journal of Bone and Mineral Research*, 22:465–475.

Carbonare, L. D., Valenti, M., Bertoldo, F., Zanatta, M., Zenari, S., Realdi, G., Cascio, V. L., and Giannini, S. (2005). Bone microarchitecture evaluated by histomorphometry. *Micron*, 36:609–616.

Clennell, M. B. (1997). Tortuosity: a guide through the maze. In *Developments in Petrophysics*, volume 122, pages 299–344. Geological Society, London.

Gommes, C. J., Bons, A.-J., Blacher, S., Dunsmuir, J. H., and Tsou, A. H. (2009). Practical methods for measuring the tortuosity of porous materials from binary or gray-tone tomographic reconstructions. *AIChE*, 55(8):2000–2012.

Homminga, J., Mccreadie, B. R., Weinans, H., and Huiskes, R. (2003). The dependence of the elastic properties of osteoporotic cancellous bone on volume fraction and fabric. *J. Biomechanics*, 36:1461–1467.

Hughes, E. R., Leighton, T. G., and White, P. R. (2007). Investigation of an anisotropic trotuosity in a Biot model of ultrasonic propagation in cancelloius bone. *J. Acoust. Soc. Am.*, 121:568–574.

Laib, A., Beuf, O., Issever, A., Newitt, D. C., and Majumdar, S. (2001). Direct measures of trabecular bone architecture from MR images. *Adv Exp Med Biol*, 496:37–46.

Roque, W. L., Arcaro, K., and Tabor, Z. (2010). An investigation of the mechanical competence of the trabecular bone. In Dvorkin, E., Goldschmit, M., and Storti, M., editors, *Mecánica Computacional*, volume XXIX, pages 2001–2009. AMCA.

Roque, W. L., Souza, A. C. A., Barbieri, D. X., and Rodrigues, F. C. (2005). Um sistema computacional baseado no processamento de imagens tomográficas para estudo da estrutura trabecular. *Anais do VII Workshop de Informática Médica*, pages 156–165.

Tabor, Z. (2009). On the equivalence of two methods of determining fabric tensor. *Medical Engineering and Physics*, 31:1313–1322.

Wesarg, S., Erdt, M., Kafchitsas, K., and Khan, M. F. (2010). Direct visualization of regions with lowered bone mineral density in dual-energy CT images of vertebrae. In Summers, R. M. and van Ginneken, M. B., editors, *Medical Imaging 2011: Computer-Aided Diagnosis*. SPIE Proceedings.

Wua, Y. S., van Vliet, L. J., Frijlink, H. W., and van der Voort Maarschalka, K. (2006). The determination of relative path length as a measure for tortuosity in compacts using image analysis. *European Journal of Pharmaceutical Sciences*, 28:433–440.

Detecção de regiões assimétricas em mamografias digitais utilizando índice de Getis-Ord.

Antônio Marcos V. Sales[1], Aristófanes Corrêa Silva[1], Anselmo Cardoso de Paiva[1]

[1]Núcleo de Computação Aplicada (NCA), Universidade Federal do Maranhão (UFMA)

marcossales@nca.ufma.br, ari@dee.ufma.br, paiva@deinf.ufma.br

Abstract. *Breast cancer accounts for 22% of new cancer cases worldwide each year, according to the National Cancer Institute. Early detection greatly increases the chances of cure. This paper presents the use of the Getis-Ord index in its general form in the search for structural asymmetries between pairs of regions of left and right mammograms, and subsequent training and placement in areas asymmetric and symmetric with a Support Vector Machine (SVM). Regions with lesions were classified as asymmetric with performance ranging from 52% to 100% efficacy. The cases that did not obtain the desired results suggest improvement in feature extraction and increasing training base of SVM.*

Resumo. *O câncer de mama representa 22% dos novos casos de câncer no mundo a cada ano, segundo o Instituto Nacional do Câncer. O diagnóstico precoce aumenta bastante as chances de cura. Este trabalho apresenta o uso do Índice de Getis-Ord em sua forma geral na busca de assimetrias estruturais entre pares de regiões de mamografias esquerdas e direitas, e posteriores treinamento e classificação em regiões assimétricas e simétricas com uma Máquina de Vetores de Suporte (MVS). Regiões com lesão foram classificadas como assimétricas com desempenho variando de 52% a 100% de acerto. Os casos que não obtiveram o resultado desejado sugerem a melhoria na extração de características e o aumento da base de treinamento da MVS.*

1. Introdução

Câncer é o nome dado ao crescimento desordenado de células que ataca tecidos e órgãos, podendo se tornar muito agressivo e causando o surgimento de tumores malignos. O câncer de mama, segundo tipo mais frequente no mundo, corresponde a 22% dos novos casos a cada ano e é o mais comum no sexo feminino, de acordo com o Instituto Nacional do Câncer (INCA) [Instituto Nacional do Câncer 2010]. O diagnóstico precoce aumenta as chances de cura. A mamografia, o exame mais usado na prevenção e diagnóstico do câncer de mama, permite a descoberta de lesões em fase inicial. A análise de imagens mamográficas pode ser melhorada com o auxílio de processamento digital de imagens e visão computacional.

Estudos anteriores mostraram que imagens de mamografia esquerda e direita da mesma paciente tendem a ter elevado grau de simetria em sua estrutura, sendo observada uma ligação entre o alto grau de assimetria e o risco de câncer [Scutt et al. 2006]. A análise de textura em mamografias [Braz Junior et al. 2006] pode ser auxiliada pelo uso de descritores espaciais [Braz Junior 2008]. Além disso, vêm sendo propostos métodos de análise de simetria entre as mamas em conjunto com a estatística espacial

para uma melhor extração das características [Ericeira et al. 2010].

Este trabalho apresenta o uso do descritor espacial conhecido como Índice de Getis-Ord, em sua forma geral, na detecção de regiões assimétricas em mamografias digitalizadas. Com a detecção das assimetrias alerta-se aos médicos para regiões que merecem melhor observação.

2. Índice de Getis-Ord

O índice de Getis-Ord é uma medida de associação espacial que se apresenta em duas formas, local e geral. A sua forma local mede a associação a partir da concentração dos pontos que se encontram dentro de uma distância *d* do ponto original. A sua forma geral considera todos os pares de pontos possíveis a uma distância *d* entre si [Getis and Ord 1992]. Esta última forma é utilizada neste trabalho e se apresenta como segue:

$$G(d) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}(d)x_i x_j}{\sum_{i=1}^{n}\sum_{j=1}^{n} x_i x_j}, j \neq i \qquad \textbf{(1)}.$$

Na Equação 1, *G(d)* representa o quociente do somatório de todos os pares de valores (x_i, x_j) que estejam a uma distância menor ou igual a *d* um do outro pelo somatório de todos os pares (x_i, x_j) possíveis. w_{ij} é a matriz de vizinhança e assume o valor 1 caso o par esteja dentro da distância *d* e 0 caso contrário.

Esta medida fornece um índice que indica o grau de associação entre os pixels para toda a região considerada, segundo a distância *d*. Isto indica *G(d)* como uma medida a ser usada na verificação de simetria entre regiões esquerda e direita, pois a comparação é feita pelo grau de associação dos pixels pertencentes a cada região, o que é diferente de, por exemplo, uma simples subtração entre os valores de intensidade dos pixels. Sendo assim pode-se medir o grau de simetria entre as regiões através da diferença entre seus *G(d)*s.

3. Metodologia

Foram utilizadas imagens do banco DDSM *(Digital Database for Screening Mammography*) que contém 2620 casos adquiridos a partir de hospitais e instituições [Heath et al. 2001].

Os pares de imagens selecionados foram submetidos à etapa de pré-processamento: redimensionamento para 512x512 pixels para tornar o processo do cálculo de *G(d)* mais viável computacionalmente; segmentação das mamas (utilizando Watershed); e registros rígido e deformável (utilizando o algoritmo "Demons"), no intuito de tornar as imagens esquerda e direita de cada par mais próximas uma da outra. Maiores detalhes podem ser encontrados em [Ericeira et al. 2010].

Essas imagens foram divididas em 256 regiões de tamanho 32x32 pixels para calcular os valores de *G(d)* para cada região selecionada correspondente nas mamas esquerda e direita do par. Foram calculados 31 valores de *G(d)* para cada região, *d* variando de 1 a 31 (31 é a distância máxima). Em seguida, para cada distância *d*, calculou-se o módulo da diferença entre os valores de *G(d)* da mama esquerda e da

direita, ou seja, | $G(d)_{esquerda} - G(d)_{direita}$ |. Assim obteve-se, para cada par de regiões, 31 valores de | $G(d)_{esquerda} - G(d)_{direita}$ |. Foram utilizados os pares sem lesão para gerar vetores (de 31 valores cada) pertencentes a regiões simétricas e, a partir dos demais, foram gerados vetores com valores de regiões assimétricas. Mantiveram-se os vetores de regiões simétricas e assimétricas em uma proporção aproximada de 2 para 1, respectivamente. Estes vetores foram usados para treinamento de uma Máquina de Vetores de Suporte (MVS) e posterior classificação de regiões em simétrica e assimétrica. O software utilizado para treinamento e classificação foi o LIBSVM [Chih-Chung and Chih-Jen 2001].

4. Testes e Resultados

Para validar a metodologia proposta selecionaram-se 62 pares (esquerda e direita) de mamografias, sendo 18 pares sem lesões, dos quais foram gerados 704 vetores referentes a regiões simétricas, e 44 pares apresentando algum tipo de lesão, dos quais foram gerados 352 vetores referentes a regiões assimétricas. Estes vetores foram usados exclusivamente no treinamento da MVS. Outros 12 novos pares de imagens foram selecionados para teste. Nesta fase foi usada a técnica *leave-one-out*.

O resultado de um dos testes realizados é mostrado na Figura 1. Corresponde ao par de imagens A1468 do banco DDSM. A MVS treinada classificou como assimétricas, neste teste, 100% das regiões marcadas como lesão no banco DDSM (destacadas em vermelho). Além destas, 35 regiões que não estavam marcadas no banco DDSM foram classificadas como assimétricas (destacadas em verde). A Figura 2 mostra uma comparação entre as médias de | $G(d)_{esquerda} - G(d)_{direita}$ | de todos os vetores calculados de regiões simétricas e assimétricas da amostra completa, para cada distância.

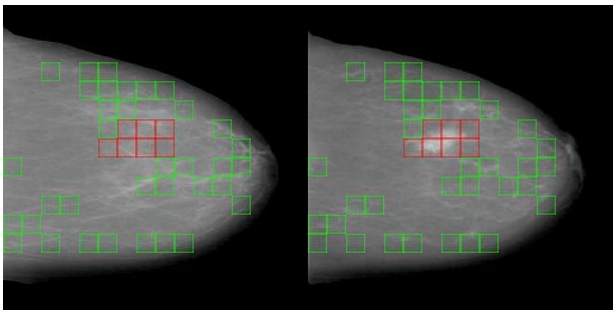


Figura 1. Par de mamogramas de uma paciente com regiões assimétricas detectadas marcadas em verde e vermelho. Regiões em vermelho correspondem à lesão.

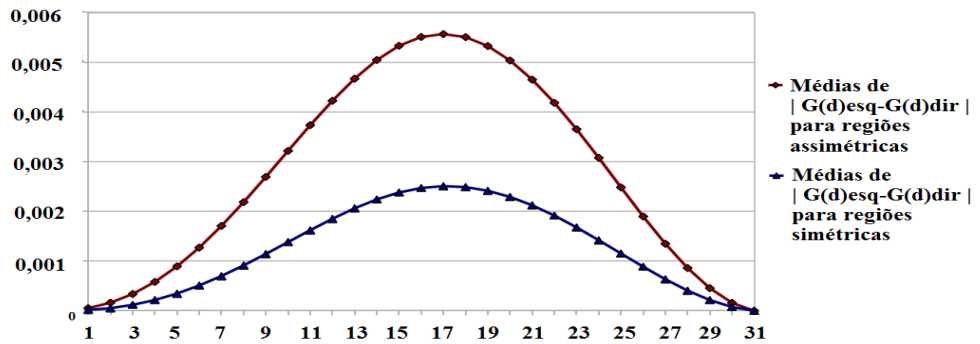


Figura 2. Gráfico comparando as médias de | $G(d)_{esquerda} - G(d)_{direita}$ | entre todas as regiões simétricas e assimétricas da amostra completa.

Em 6 dos 12 casos testados, 100% das regiões com algum tipo de lesão foram classificadas como assimétricas, e nos outros 6 casos, essa eficácia variou de 52% a 96%, isto é, a metodologia proposta conseguiu detectar apenas parte da lesão. É importante salientar que houve outras assimetrias detectadas, mas que não estavam marcadas como lesão no banco DDSM. Estas assimetrias merecem melhor observação.

5. Conclusão

Este trabalho apresentou a utilização do Índice de Getis-Ord na detecção de regiões assimétricas em pares de imagens de mamografia dentro de uma metodologia de extração de características, treino e classificação com uma MVS em regiões assimétricas e simétricas. Regiões indicadas no banco DDSM como possuidoras de lesão foram classificadas como assimétricas com um desempenho variando de 52% a 100% de acerto, além das assimetrias detectadas não marcadas como lesão no banco DDSM. Trabalhos futuros pretendem aumentar a base de treinamento e considerar o uso de outras medidas estatísticas, assim como o aperfeiçoamento da extração de características das regiões em estudo para uma classificação mais eficaz das mesmas e a distinção entre lesão e não-lesão nas regiões assimétricas.

Referências

Braz Junior, Geraldo (2008), "Classificação de Regiões de Mamografias em Massa e Não Massa usando Estatística Espacial e Máquina de Vetores de Suporte", Dissertação (Mestrado em Engenharia de Eletricidade), Centro de Ciências Exatas e Tecnologia, Universidade Federal do Maranhão.

Braz Junior, Geraldo., Corrêa Silva, Aristófanes., Cardoso de Paiva, Anselmo., Cesar Muniz de Oliveira, Alexandre. (2006), "Identificação de Massas em Mamografias usando Textura, Geometria e Algoritmos de Agrupamento e Classificação", VI Workshop de Informática Médica - WIM2006.

Chih-Chung, C., Chih-Jen L. (2001), "LIBSVM: a library for support vector machines", http://www.csie.ntu.edu.tw/~cjlin/libsvm/, Março de 2011.

Ericeira, D. R., Corrêa Silva, Aristófanes., Cardoso de Paiva, Anselmo. (2010), "Detecção de Regiões Suspeitas em Mamografias Digitais Utilizando Descrição Espacial com Variograma Cruzado", XII Congresso Brasileiro de Informática em Saúde – CBIS2010.

Getis, A., and Ord, JK. (1992), "The Analysis of Spatial Association by Use of Distance Statistics." Geographical Analysis 24 (July).

Heath, M., Bowyer, K., Kopans, D., Moore, R., and Kegelmeyer, WP., (2001), "The Digital Database for Screening Mammography," in *Proceedings of the Fifth International Workshop on Digital Mammography*, M.J. Yaffe, ed., 212-218, Medical Physics Publishing. ISBN 1-930524-00-5.

Instituto Nacional do Câncer (2010), "Estimativa 2010 - Incidência de Câncer no Brasil", http://www.inca.gov.br/estimativa/2010/, Março de 2011.

Scutt D., Lancaster G. A.& Manning J.T. Breast Cancer Res., 8. R14 (2006), "Breast Asymmetry and Predisposition to Breast Cancer", Breast Cancer Research, licensee BioMed Central Ltd.

Dimensão Fractal: Quantificação dos Comportamentos de Nódulos Mamários Malignos e Estruturas Adjacentes

Leandro A. Neves[1,4], Marcelo Z. do Nascimento[2,4], Moacir F. de Godoy[3,4]

[1]IGCE – Departamento de Estatística, Matemática Aplicada e Computação Universidade Estadual Paulista (UNESP) – Caixa Postal 178 – 13.506-900 – Rio Claro – SP – Brazil

[2]Centro de Matemática, Computação e Cognição – Universidade Federal do ABC (UFABC) – Santo André, SP – Brazil

[3]Faculdade de Medicina de São José do Rio Preto (FAMERP) – São José do Rio Preto, SP – Brazil

[4]Núcleo Transdisciplinar para Estudo do Caos e da Complexidade (NUTTECC) – São José do Rio Preto, SP – Brazil.

laneves@rc.unesp.br, marcelo.nascimento@ufabc.edu.br, mfgodoy@famerp.br

Abstract. *This paper presents a method for the quantification of normal mammogram image and malignant breast lump, as well as lesions adjacent structures. For the image processing, each region of interest was segmented using the semi-automatic multilevel method, which determines the threshold based on slope percentage analysis and comparing the entropies in different regions of the histogram of the image in question. The structures were quantified with the aid of the multi-scale fractal dimension for the identification of behavior patterns in structures of interest, providing information relevant to support Computer Aided Diagnosis (CAD).*

Resumo. *Neste trabalho é apresentado um método para quantificar imagem mamográfica com e sem nódulos mamários malignos, além das estruturas adjacentes aos nódulos. Regiões de interesse de cada imagem foram segmentadas com um método semi-automático e multinível, que determina o limiar a partir de análises dos percentuais de inclinação e comparações das entropias em diferentes regiões do histograma da imagem em questão. A quantificação das estruturas foi com a dimensão fractal multiescala, para identificar padrões de comportamentos das estruturas de interesse, fornecendo informações relevantes para embasar Sistemas de Apoio ao Diagnóstico (SAD).*

1. Introdução

O câncer de mama é o segundo tipo de câncer com maior incidência no mundo, Nishikawa (2007), INCA (2009). No Brasil, a projeção para o ano de 2010 foi de 50 mil novos casos da doença, com risco de 49 casos a cada 100 mil mulheres. Na região sudeste o número de casos estimado é maior, aproximadamente 65 para cada 100 mil mulheres, INCA (2009).

Neste cenário, a mamografia é uma das principais técnicas utilizadas para diagnosticar e apoiar em decisões terapêuticas. No entanto, a superposição de estruturas e efeitos físicos presentes no processo de aquisição das imagens influenciam a exatidão e confiabilidade do diagnóstico. Pesquisas mostram que 10 a 30% das anormalidades do tipo maligno visíveis em mamogramas não foram detectadas pelos especialistas durante exames de rotina, Gupta et al. (2006). Uma solução pode ser uma segunda leitura, realizada por outro especialista, para aumentar a sensibilidade do sistema, Wei et al. (2011). Essa estratégia eleva os custos administrativos dos hospitais e clínicas de radiodiagnóstico, sendo impraticável em muitos locais.

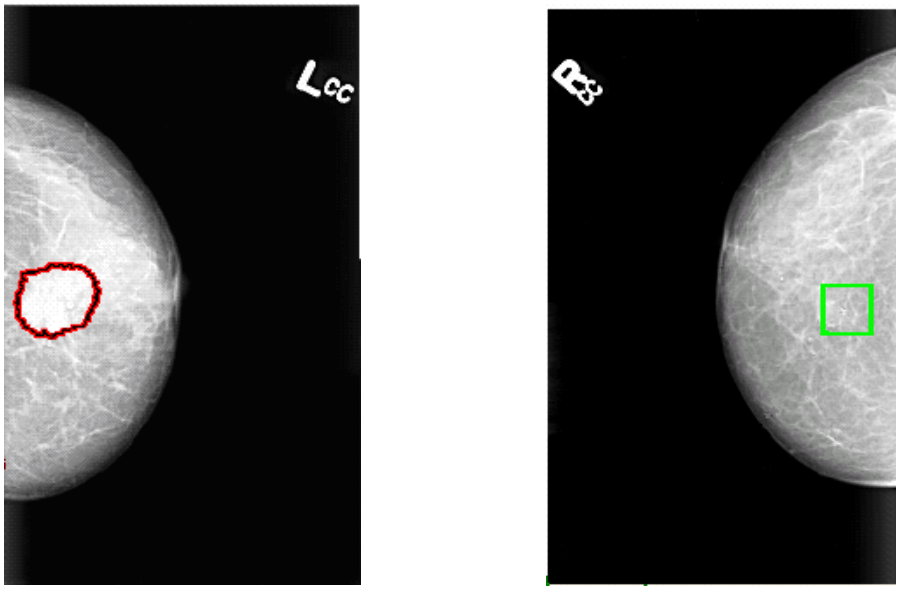
Na literatura especializada é possível encontrar uma série de trabalhos com propostas de modelos ou técnicas computacionais para apoiar o diagnóstico de câncer de mama. Os modelos construídos com técnicas capazes de tornar o conhecimento compreensível aos especialistas merecem destaque, pois as características analisadas permitem descrever o comportamento identificado em estruturas sadias ou doentes, Rangayyan and Nguyen (2005), Mavroforakis et al. (2006), Stojic et al. (2006), Pruess (2007), Li et al. (2007), Dua et al. (2010). Uma abordagem que ainda pode ser explorada é a caracterização de estruturas sadias e nodulares, com quantificações realizadas com o mínimo de influência das estruturas que circundam os nódulos. A representatividade desejada pode ser obtida com técnica capaz de caracterizar comportamentos mais próximos aos fenômenos naturais, como é o caso da dimensão fractal (*D*). A eficiência dessa técnica é potencializada quando observadas em escalas espaciais distintas. Essa associação é conhecida como assinatura fractal multiescala, Coelho and Costa (1995), Chaudhuri and Sarkar (1995), Plotnick et al. (1996).

Este trabalho apresenta a quantificação dos comportamentos das estruturas mamárias sadias, doentes e circunvizinhas (regiões com pixels parciais, entre tecido sadio e doente), por meio de assinaturas das texturas, obtidas com dimensão fractal multiescala. A separação dos pixels parciais foi realizada com um método de segmentação multinível, baseado em entropia máxima. Os comportamentos das estruturas de interesse e os níveis de cortes mais apropriados foram avaliados com gráficos ROC (*Receiver Operating Characteristic)*, obtidos a partir de um conjunto de imagens da base *DDSM (Digital Database for Screening Mammography)*. Os resultados fornecem informações significativas e podem embasar sistemas de apoio ao diagnóstico de nódulos mamários malignos.

2. Metodologia

A dimensão fractal (*D*) foi calculada para imagens da base *DDSM (Digital Database for Screening Mammography)* da Universidade do Sul da Flórida, EUA. Essa base é constituída por aproximadamente 2.500 casos, obtidos entre os anos de 1988 e 1999, por meio de projetos desenvolvidos por pesquisadores de quatro instituições dos Estados Unidos: *Massachusetts General Hospital* (MGH), *Wake Forest University School of Medicine* (WFUSM), *Sacred Heart Hospital* (SHH) e *Washington University* (WU). As imagens foram digitalizadas em diferentes equipamentos, tais como Howtek (WU e MGH), Lumisys (WFUSH e SHH) e DBA (MGH), sendo que para cada caso estão disponíveis as imagens obtidas nas visões Médio-Lateral Oblíquo (MLO) e Crânio-Caudal (CC), representadas em níveis de cinza com quantização de 12 ou 16 *bits*.

Para este trabalho, foram selecionadas 166 imagens nas visões CC e MLO, distribuídas em dois conjuntos: nódulo mamário maligno (NM), com 58 casos (29 CC e 29 MLO); e, o das estruturas sadias (ES), com as demais imagens (54 CC e 54 MLO). As imagens escolhidas foram obtidas com o equipamento LUMYSIS, com resolução 43,5 *microns* e 12 bits de quantização. Por questões de padronização de resolução, apenas as imagens obtidas pelo digitalizador LUMYSIS foram utilizadas nesses experimentos. Para tornar as quantificações mais precisas, os grupos foram redefinidos para conter apenas regiões de interesse (*Region of Interest* - ROI). O tamanho da janela foi de 65x65 pixels e foi suficiente para considerar as regiões nodulares para investigação. As ROI's para o grupo NM foram obtidas a partir do arquivo *.ics*, que contém o código em cadeia (c*ode-chain*) da posição da lesão, conforme disponível na base *DDSM*. Na Figura 1a, região marcada em vermelho, mostra um exemplo de região com a presença de um nódulo mamário selecionada por um especialista. Para o grupo ES, as ROI's foram extraídas de imagens sadias (considerando diferentes localizações). A seleção foi realizada manualmente por um especialista, Figura 1b.



(a) (b)

Figura 1. Exemplificação de regiões utilizadas para quantificação de nódulo mamário maligno (a), grupo NM, e normal (b), grupo ES.

Conforme descrito por Mancanttini et al. (2008), a região demarcada pelo código em cadeia (c*ode-chain*) da posição da lesão (65x65 pixels) seleciona uma região de forma insatisfatória, pois em muitos casos grande parte da região de fundo (região da mama que não possui anormalidade) é selecionada nesse processo e pode influenciar a quantificação dos mesmos. Por isso, essa situação foi minimizada com a segmentação do nódulo, por meio de um método de limiarização multinível, que determina um limiar comparando-se as entropias em diferentes regiões do histograma da imagem estudada, Neves et al. (2011). Esse método permite realizar ajustes dos parâmetros que são considerados para determinar limiares, o que motivou a aplicação do mesmo nas imagens investigadas neste trabalho.

A primeira etapa para segmentação das imagens contidas em NM foi determinar o histograma da ROI dada como entrada. O histograma foi dividido em classes (um ou

mais valores de intensidades de níveis de cinza). Para essa divisão, foi necessário pré-definir o tamanho da classe, recurso que permite avaliar histogramas com diferentes distribuições. A identificação das classes que possuem vales relevantes para a especificação dos limiares foi feita por meio da transição de sinal, analisando as seguintes características:

1.comparou-se a média de níveis de cinza na primeira metade da classe analisada com a média de níveis da segunda metade da mesma classe. Para primeira metade com média menor, a interpretação foi que os valores do histograma estavam aumentando, portanto em direção à um pico, conseqüentemente o sinal atribuído a classe foi positivo. Caso contrário, o sinal negativo foi atribuído a classe, indicando direção à um vale;

2.verificou-se o sinal da próxima classe. Se houve uma transição de sinal negativo para positivo, a classe foi indicada para especificar um limiar. Caso contrário, classes com o mesmo sinal foram concatenadas.

As comparações apresentadas podem ser influenciadas por regiões homogêneas do histograma (sem presença de vales ou vales não significativos). Para eliminar esse problema, uma classe indicada para especificar um limiar foi de fato considerada se possuía um percentual de inclinação maior que um valor pré-definido. Um percentual de inclinação é a diferença, em porcentagem, entre as médias de pontos da primeira e segunda metades da classe analisada, Neves et al. (2011). Para quantificar as ROI's, os valores utilizados como parâmetros de entrada para segmentação foram classes com tamanho 5 e percentual de inclinação definido em 15%. Esses valores foram suficientes para segmentar apropriadamente as regiões de interesse.

Para cada classe que atendeu os requisitos estabelecidos, um limiar foi calculado com base na entropia, considerando a probabilidade de uma intensidade segmentar corretamente um determinado grupo de objetos. Para isso, a imagem foi dada como resultado de um processo aleatório, em que a probabilidade p correspondeu à probabilidade de um pixel da imagem assumir um valor de intensidade i (i=1,..,n), Pun (1980), como mostrado nas Equações (1) e (2). A intensidade ou nível de cinza da classe com a maior entropia foi identificado como limiar.

$$H = -\sum_{i=1}^{n} p_i \log p_i \tag{1}$$

$$p_i = \frac{n_i}{N} \tag{2}$$

onde, H representa a entropia da imagem; n é o número total de saídas, ou seja, o número de níveis de cinza da imagem; p_i representa a probabilidade do nível de cinza i ser encontrado na imagem; n_i é a quantidade de pixels com a intensidade i e N é a quantidade total de pixels na imagem.

O resultado proporcionado pelo método de segmentação foi dois novos grupos de imagens, gerados a partir de NM. Um constituído somente com os nódulos malignos segmentados (NMS) e o outro com as estruturas remanescentes, ou seja, pixels parciais (PP). A caracterização foi realizada para cada imagem contida em NMS, PP e ES, por meio da dimensão fractal.

A dimensão fractal é útil como uma medida da complexidade de objetos e quando aplicada a texturas, permite quantificar a complexidade da organização de seus pixels (o quão preenchido está o espaço). A abordagem multiescala foi obtida sobrepondo uma malha de quadrados na imagem analisada, com o objetivo foi obter o número de quadrados necessários para cobri-la, Backes and Bruno (2006), Neves et al. (2011). Com isto, dada uma imagem I como entrada, o número de quadrados que contém parte da forma I, $N_r(I)$ depende do tamanho da caixa, r, Equação (3). A relação definida na Equação (3) permitiu estimar a dimensão fractal D, com base na Equação (4), Backes and Bruno (2006).

$$N_r(I) = \mu r^{-D} \tag{3}$$

$$D = -\lim_{r \to 0} \frac{\ln(N_r(I))}{\ln(r)} \tag{4}$$

A equação (4) impõe o cálculo de um limite, em que a contagem de quadrados, $N_r(I)$, é realizada para diferentes valores de r (r_0=max(altura, largura) e $r_{i+1}=r_i / 2$). A quantificação para diferentes valores de r caracteriza a abordagem multiescala. A aproximação de uma reta é obtida por meio da regressão (gráfico log-log) de $N_r(I)$ (número de caixas ocupadas) por r (tamanho do lado dessa caixa). Isso permite definir $D=\theta$ como sendo a dimensão fractal de I, Backes and Bruno (2006), Neves et al. (2011). A avaliação das ROI's foi com o método *BoxCounting*, aplicado como um descritor de textura, pois a luminância do pixel representou a altura do mesmo, Chaudhuri and Sarkar (1995), Coelho and Costa (1995), Neves et al. (2011). Esta modificação implica em dividir a imagem em cubos de lado r, ao invés de quadrados de lado r, consequentemente produzindo um novo $N_r(I)$, dependente do número de cubos que possuem parte da imagem. Com isso, os gráficos log-log representam assinaturas das texturas de interesse dos grupos NMS, PP e ES. A abordagem utilizada realizou uma transformação, que reduziu a dimensionalidade das informações analisadas para uma função unidimensional, capaz de quantificar os comportamentos das texturas investigadas, Backes and Bruno (2006).

Após o processamento das imagens e assinaturas obtidas, as comparações entre os grupos foram por meio de gráficos ROC. Essa abordagem foi adotada para quantificar as relações entre as taxas de acertos e erros dos grupos comparados, indicar o desempenho obtido com a dimensão fractal e definir os pontos de cortes mais apropriados, Fawcett (2006). As *matrizes de confusão* foram construídas para os grupos NMS versus ES, NMS versus PP e PP versus ES. Com isso, cada imagem foi classificada como uma das quatro saídas possíveis: verdadeiro positivo (*VP*); falso negativo (*FN*); verdadeiro negativo (*VN*); ou, falso positivo (*FP*). Os totais de imagens positivas (*P*) e negativas (*N*) obtidos nos grupos comparados permitiram calcular algumas informações estatísticas para avaliar a separabilidade dos grupos, Fawcett (2006), Dua et al. (2010), que foram:

- acurácia (*AC*), proporção de predições corretas em relação ao acerto total (Equação (5));
- precisão ou valor preditivo positivo (*Pr*), proporção de verdadeiros positivos entre todos os casos com teste positivo (Equação (6));

- revocação ou sensibilidade (*S*), probabilidade de um teste dar positivo na presença da doença (Equação (7));
- especificidade (*E*), probabilidade de um teste dar negativo na ausência da doença (Equação (8)).

$$AC - \frac{VP + VN}{P + N} \tag{5}$$

$$PR - \frac{VP}{VP + FP} \tag{6}$$

$$S = \frac{VP}{VP + FN} \tag{7}$$

$$E = \frac{VN}{VN + FP} \tag{8}$$

3. Resultados

O método de segmentação multinível, baseado em entropia máxima, foi aplicado no grupo NM e a exemplificação do resultado obtido com uma ROI em visão MLO está na Figura 2. As assinaturas das texturas dos grupos NMS, PP e ES estão na Figura 3, em que o gráfico relaciona o tamanho do cubo (eixo X) com o total de cubos (eixo Y) obtido no cálculo do método *BoxCounting*. Os gráficos ROC das dimensões fractais dos grupos NMS versus ES, NMS versus PP e PP versus ES estão na Figura 4, que representam o par sensibilidade (eixo Y) e 1-especificidade (eixo X).



(a) (b) (c)

Figura 2. Visualização de ROI's utilizadas para descrição dos comportamentos das texturas: (a) ROI em visão MLO dada como entrada e que pertence ao grupo NM; (b) estruturas segmentadas de (a) e que constituem uma ROI para o grupo PP; (c) nódulo mamário segmentado de (a) e que constitui uma ROI para o grupo NMS.

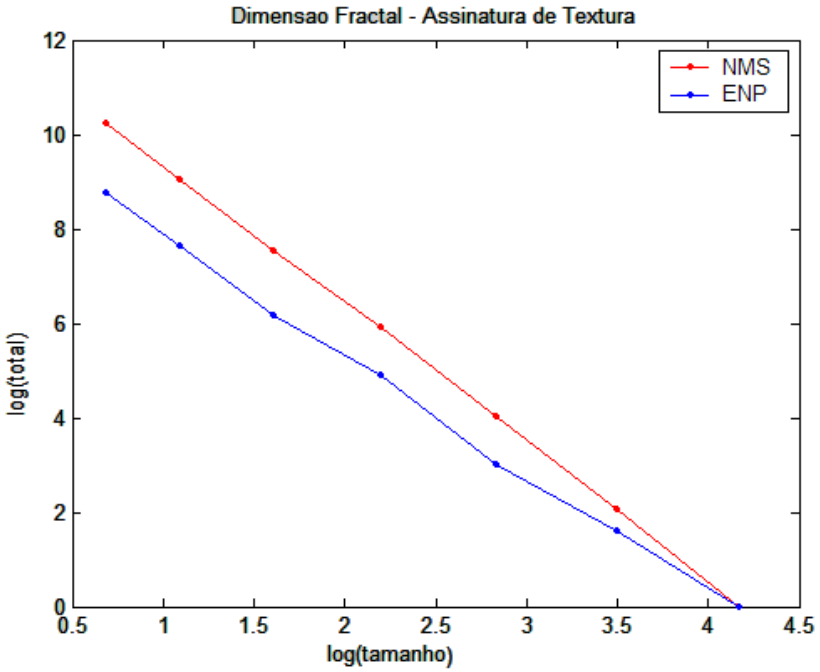


Figura 3. Assinaturas das texturas do tecido apresentado na Figura 2b e do nódulo mamário evidenciado na Figura 2c, representadas em gráficos log-log das dimensões fractais. O eixo X indica o tamanho do cubo e o eixo Y o total de cubos que foi contado.

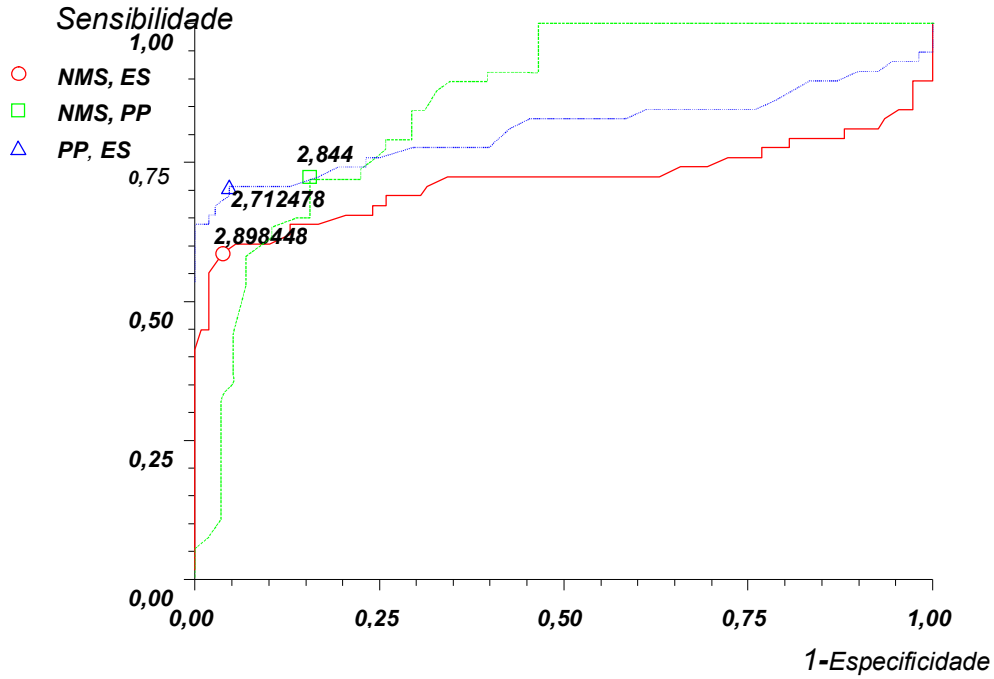


Figura 4. Gráficos ROC para diferenciação pela dimensão fractal entre os NMS versus ES, NMS versus PP e PP versus ES.

Os valores para acurácia (*AC*), com intervalos de confiança (IC 95%), pontos de corte (*PC*), precisão (*Pr*), sensibilidade (*S*) e especificidade (*E*) obtidos com as comparações dos gráficos ROC, evidenciados na Figura 4, podem ser visualizados na Tabela 1.

Tabela 1. Medidas de avaliação referentes aos grupos comparados e apresentados na Figura 4.

Grupos	Dados obtidos com o Atributo Dimensão Fractal					
	AC	IC	*PC*	*Pr*	*S*	*E*
NMS, ES	0,7169	0,6289 a 0,8049	2,8984	0,8947	0,5862	0,9629
NMS, PP	0,8635	0,7886 a 0,9385	2,844	0,8235	0,7241	0,8448
PP, ES	0,8109	0,7337 a 0,8882	2,7124	0,8913	0,7068	0,9537

4. Discussão

O método de segmentação proposto permitiu diferenciar as estruturas de interesse com características suficientes para aplicação da etapa de extração de características. A quantificação dos comportamentos das estruturas mamárias sadias e doentes foi realizada com a dimensão fractal multiescala. Essa afirmação é possível com base nos distintos comportamentos das assinaturas das texturas (Figura 3), bem como pelos gráficos ROC representativos dos grupos NMS versus ES, NMS versus PP e PP versus ES (Figura 4).

As assinaturas das texturas demonstraram padrões unidimensionais distintos para os grupos comparados, como por exemplo, NMS versus PP (Figura 3), uma vez que visualmente foi possível identificar cada grupo. Com isso, as informações contidas nas assinaturas são potencias vetores de características para distinguir texturas mamárias sadias das doentes, que podem ser explorados com análises pontuais ou globais.

Os comportamentos apresentados pelos gráficos ROC dos grupos NMS versus PP e PP versus ES evidenciaram tecidos com padrões diferentes (Figura 4). O grupo PP tem uma distinção de 81,09% do grupo ES e 86,35% do NMS, Tabela 1. Isso permite a inferência que o grupo PP possui tecido em processo de degeneração, com características diferentes dos tecidos sadios, mas com padrão que não o qualifica como parte efetiva do nódulo. Outra constatação é que a dimensão fractal discrimina o grupo NMS do ES, mas com taxas menores de acerto (71,69%) e sensibilidade (58,62%).

A relevância do atributo dimensão fractal para discriminar estruturas mamárias sadias das doentes foi demonstrada com níveis de cortes suficientes para obter valores significativos de precisão, sensibilidade e especificidade, em sua maioria acima de 70% (Tabela 1). Quando observamos os grupos NMS com ES e PP com ES, as precisões foram de aproximadamente 89%, enquanto NMS versus PP foi de 80%, valores estes que reforçam capacidade do atributo para distinguir e quantificar as estruturas investigadas.

5. Conclusões

Este trabalho apresentou uma metodologia capaz de quantificar os comportamentos das estruturas mamárias sadias e doentes, com uso de segmentação baseada em entropia,

dimensão fractal, assinatura de textura e gráficos ROC. As taxas de acertos foram relevantes, com a vantagem de quantificar o comportamento das regiões com pixels parciais. As contribuições citadas são significativas e úteis para embasar trabalhos futuros para associação das assinaturas das texturas nas visões CC e MLO, o que pode contribuir com um sistema de apoio ao diagnóstico mais próximo ao procedimento executado na prática clínica.

Agradecimentos

O primeiro autor agradece os apoios financeiros recebidos, PROPe (Pró-Reitoria de Pesquisa) e FUNDUNESP (Fundação para o Desenvolvimento da UNESP).

Referências

Nishikawa, R. (2007) Current status and future directions of computer-aided diagnosis in mammography, In *Computerized Medical Imaging and Graphics*, v. **31**, page 224–235

INCA - Ministério da Saúde. Secretaria de atenção à saúde. Inst. Nac. do Câncer. Coord. de prev. e vigilância; (2009). Estimativa 2010. Incidência de câncer no Brasil http://www1.inca.gov.br/estimativa/2010, 19/05/2011.

Gupta, S., Chyn, P. F., Markey, M. K. (2006) Breast cancer CADx based on BI-RADS descriptors from two mammographic views. In *Medical Physics*, pages 1810-1817.

Wei, J., Chan, H., Zhou, C., Wu, Y., Sahiner, B., Hadjiiski, L. M., et al. (2011). Computer-aided detection of breast masses: Four-view strategy for screening mammography. In *Medical Physics*, v. 38(4),pages 1867-1876.

Rangayyan, R., Nguyen, T., (2005) "Pattern classification of breast masses via fractal analysis of their contours", In: Proceedings of Computer Assisted radiology and Surgery (CARS), pages 1041–1046.

Mavroforakis, M., Georgiou, H., Dimitropoulos, N., Cavouras, D., Theodoridis, S. (2006) Mammographic masses characterization based on localized texture and dataset fractal analysis using linear, neural and support vector machine classifiers. In *Artificial Intelligence in Medicine*, pages 145–162.

Stojic, T., Reljin, I., Reljin, B. (2006) Adaptation of multifractal analysis to segmentation of microcalcications in digital mammograms. In *Physica A Statistical Mechanics and its Applications,* v. 367, pages 494–508.

Pruess, S. (2007) "Some remarks on the numerical estimation of fractal dimension", Fractals in the Earth Sciences, C.C. Barton and P.R. La Pointe, New-York, Plenum Press, Chap. 3, pages 65–75.

Li, H., Giger, M.L., Olopade, O.I., Lan, L. (2007) Fractal analysis of mammographic parenchymal patterns in breast cancer risk assessment. In *Academic Radiology*, v.14 (5), pages 513–521.

Dua, S., Singh, H., Thompson, H. W. (2010) Associative Classification of Mammograms using Weighted Rules. In *Expert Syst Appl*. v. 36 (5), pages 9250-9259.

Backes, A. R. and Bruno, O. M. (2006) Segmentação de texturas por análise de complexidade. In *INFOCOMP Journal of Computer Science*, v. 5 (1), pages 87–95.

Chaudhuri, B. B. and Sarkar, N. (1995) Texture Segmentation Using Fractal Dimension. In IEEE Transactions on Pattern Analysis and Machine Intelligence, v.17, pages 72-76.

Coelho, R. C. and Costa, L. da F. (1995) "The Box-Counting Fractal Dimension: Does it provide an Accurate Subsidy for Experimental Shape Characterization? If So, How to Use It?" In: Symp. on Comp. Graphics and Image Processing, São Carlos, Brazil, page. 183-191.

Plotnick, R. E, Gradner, R. H., Hargrove, W. W., Prestegaard, K. and Perlmutter, M. (1996) Lacunarity analysis: a general technique for the analysis of spatial patterns. In *Physical Review E*, v. 53, pages 5461-5468.

Mencattini, A., Salmeri, M., Lojacono, R., Frigerio, M., Caselli, F. (2008) Mammographic images enhancement and denoising for breast cancer detection using dyadic wavelet processing. In *IEEE Trans. on Instrumentation and Measurement*, pages 1422-1430.

Neves, L. A., Oliveira, F. R., Peres, F. A., Moreira, R. D., Moriel, A. R., de Godoy, M. F. and Murta Junior, L. O. (2011) Maximum entropy, fractal dimension and lacunarity in quantification of cellular rejection in myocardial biopsy of patients submitted to heart transplantation. In: *Journal of Physics: Conference Series*, v. 285 (1), pages 012032.

Pun, T., (1980) A New Method for Gray-Level Picture Thresholding Using the Entropy of the Histogram. In *Signal Processing*, v. 2, pages 223-237.

Fawcett, T. (2006) An introduction to ROC Analysis. In *Pattern Recognition Letters*, v. 27, pages 861-874.

InVesalius: Software Livre de Imagens Médicas

Paulo H. J. Amorim[1], Thiago F. de Moraes[1], Fábio de S. Azevedo[1], Jorge V. L. da Silva[1]

[1]Centro de Tecnologia da Informação Renato Archer - CTI
Rodovia Dom Pedro I (SP-65) Km 143,6 – Bairro Amarais
CEP:13069-901 – Campinas - SP

{paulo.amorim,thiago.moraes,fabio.azevedo,jorge.silva}@cti.gov.br

Abstract. *This work presents InVesalius, an open-source imaging software. Initially, the historical context and motivation for its development are discussed. Then, some software internals and development aspects are commented, as well as InVesalius main features and applications. At the end, the results of a survey realized within the user community are presented.*

Resumo. *Este trabalho apresenta o software livre para tratamento de imagens médicas InVesalius. Inicialmente, o contexto histórico e as motivações para o desenvolvimento do software são discutidos. Em seguida, comentam-se o desenvolvimento e aspectos internos do funcionamento do InVesalius, bem como seus principais recursos e aplicações. Por fim, são apresentados os resultados de uma pesquisa com seus usuários no Brasil e no exterior.*

1. Introdução

Imagens médicas têm como objetivo fornecer aos clínicos a possibilidade de visualizar internamente o corpo humano de forma não invasiva, além de tornar o diagnóstico mais preciso. Apesar de a técnica de raio-X existir desde o início do século XX, as imagens médicas em três dimensões surgiram somente em 1972 com o invento da tomografia computadorizada (TC). [Yoo 2004].

Além do raio-X, empregado na radiografia convencional e na tomografia computadorizada, outra forma de obter imagens do interior do corpo é a ressonância magnética (RM). A RM emprega um alto campo magnético em conjunto com ondas de rádiofrequência. Atualmente, as técnicas citadas geram imagens em formato digital. Em geral, essas imagens seguem o padrão DICOM (*Digital Imaging and Communications in Medicine*). Uma imagem DICOM, além dos voxels, contém metainformações, como nome do paciente, informações do equipamento, posição da imagem em relação ao espaço (no caso da TC e RM) [Pianykh 2007].

De acordo com o padrão DICOM, cada imagem da modalidade TC ou RM representa um ”corte”do corpo. A reconstrução 3D é feita por meio do empilhamento desses ”cortes”, interpolando os espaços entre eles, formando um volume.

A maioria dos fabricantes de equipamentos de TC e RM vendem separadamente os softwares que realizam a reconstrução 3D a um preço inacessível para a atual realidade da saúde pública brasileira. Mesmo já no início do século XXI, não se encontrava software livre ou gratuito que realizasse a reconstrução 3D. Hoje, embora já existam softwares

para esse fim que são gratuitos ou livres, eles ainda possuem algumas limitações. Alguns apresentam uma interface gráfica pouco amigável. Outros, apesar de serem software livre, só podem ser executados em plataforma específica.

O Centro de Tecnologia da Informação Renato Archer - CTI, há mais de uma década, já possui equipamentos de prototipagem rápida (impressoras 3D) e tem como um de seus objetivos a produção de modelos físicos para auxiliar profissionais de saúde em planejamento cirúrgico. Diante desse cenário, em 2001, iniciou-se, no CTI, o desenvolvimento do software InVesalius. Atualmente na terceira versão, esse software de visualização de imagens médicas e reconstrução 3D é multiplataforma e multi-idioma. Com foco em usabilidade, pode ser executado em computadores pessoais.

2. Desenvolvimento do InVesalius

O InVesalius é um software livre, cuja licença é a GNU General Public License versão 2. Isso permite que pesquisadores e desenvolvedores de universidades e centros de pesquisas de todo o mundo ajudem no desenvolvimento, e ainda apliquem seus temas de pesquisa e interesse neste desenvolvimento.

O desenvolvimento é centralizado com o uso da ferramenta de controle de versão Subversion [Subversion 2011]. Essa ferramenta, basicamente, permite o controle de alterações e seus autores. Emprega-se a política de que novas ferramentas e erros sejam antes documentados, através da ferramenta *Trac* [Trac 2011] e depois seja feito o desenvolvimento e isto é feito não apenas pelos desenvolvedores, mas também pelos usuários do software.

3. Aspectos internos e Funcionamento

O software é desenvolvido na linguagem de programação *Python*. Python é uma linguagem multi-paradigma, pois permite o uso dos paradigmas orientado a objeto, procedural e funcional. No caso do InVesalius, o paradigma orientado a objetos foi utilizado quase que totalmente. Apenas em certos casos os paradigmas funcional e procedural foram utilizados. Também são utilizadas muitas bibliotecas que serão mencionadas no decorrer do texto.

O início da operação do software se dá pelos dados de entrada. O usuário do software seleciona os arquivos de imagens médicas, que podem ser no formato DICOM ou Analyze. Hoje, o InVesalius dá suporte a arquivos de tomografia computadorizada (CT) e ressonância magnética (RM). Para o suporte ao formato DICOM [Pianykh 2007], é utilizada a biblioteca *GDCM* [Malaterre 2011], e a biblioteca *Nibabel* [NiBabel 2011] para arquivos Analyze. É importante mencionar que a biblioteca *GDCM* permitiu dar suporte a arquivos DICOM com alterações do padrão e também a arquivos compactados que vêm se tornando comuns devido ao seu menor tamanho.

As imagens de entrada são convertidas em uma matriz com três dimensões. Essa matriz é mantida numa estrutura *memmap* em arquivo em disco e mapeada em memória RAM por meio da chamada de sistema *mmap*, comum a vários sistemas operacionais. Assim, apenas os dados da matriz necessários são mantidos em memória. Quando outros dados forem necessários, eles são trazidos do arquivo em disco para a memória RAM. Essa estratégia permite que arquivos de exames médicos com grande quantidade de fatias

ocupem menos memória do que se todas as fatias fossem carregadas em memória. A estrutura *memmap* é provida pela biblioteca *Numpy* [Kinser 2009]. *Numpy* é uma biblioteca numérica para Python que provê matrizes e vetores e operações matemáticas sobre estes. Poderosa e de fácil uso, ela permite o acesso aos dados das matrizes usando índices ao invés de chamadas explícitas a funções ou métodos.

Após a abertura das imagens, o usuário tem acesso à visualização do exame e a várias ferramentas. É possível visualizar o exame percorrendo suas fatias, nas orientações axial, sagital e coronal e também a sua reconstrução tridimensional. É importante salientar que o uso de uma matriz de três dimensões facilitou o acesso às orientações citadas, bastando para isso acessá-la de acordo com a orientação através de índices.

Para a construção da interface gráfica (figura 1) foi utilizada a biblioteca multiplataforma *wxPython* [Rappin and Dunn 2006]. Por utilizar a biblioteca de interface gráfica padrão do sistema operacional em que está sendo executado, o InVesalius mantém a aparência de uma aplicação nativa, sendo necessários apenas pequenos ajustes.

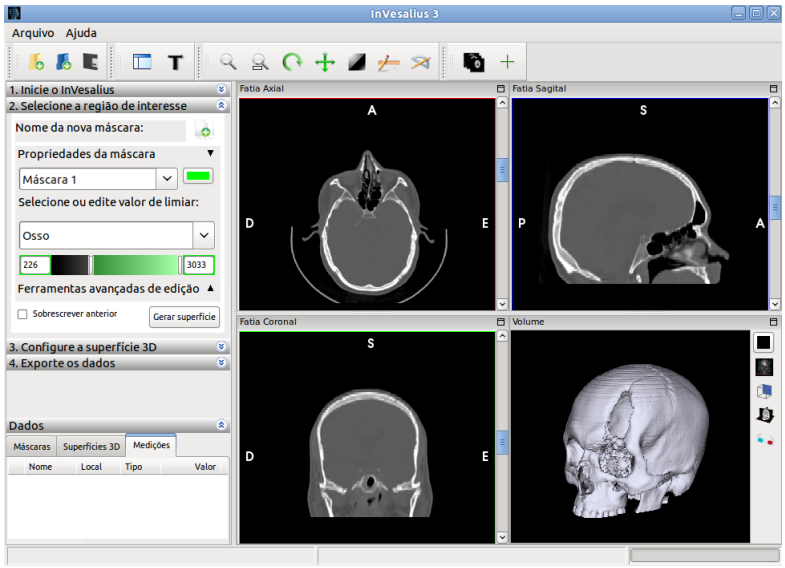


Figura 1. Software InVesalius

A biblioteca VTK (*Visualization ToolKit*) [Schroeder et al. 2004] foi utilizada para visualização, tanto 2D quanto 3D. Como os dados são armazenados em uma estrutura de matriz *memmap*, é necessário convertê-los para a estrutura VTK. Isso é feito apenas para as fatias do exame que estão sendo visualizadas no momento, ou seja, sob demanda. A biblioteca VTK também foi útil em outras funcionalidades que serão citadas no decorrer do texto.

4. Recursos

A seguir são apresentados os principais recursos do InVesalius na sua versão 3 e são citadas algumas de suas aplicações.

4.1. Segmentação de Imagens

Hoje, o InVesalius conta com duas técnicas de segmentação: *threshold* [González and Woods 2008] e manual. Na técnica de *threshold*, apenas os voxels

com valores entre um mínimo e um máximo são selecionados. A segmentação manual é uma adição à técnica por *threshold*. Com ela, é possível selecionar, interativamente, voxels de interesse do usuário não selecionados pela técnica de *threshold*, assim como eliminar da seleção voxels que não sejam de seu interesse. Adicionalmente, há ainda *presets* de *thresholds* para certas estruturas da anatomia humana cujos valores de voxels já são conhecidos, como por exemplo a estrutura óssea. O que é selecionado é demonstrado ao usuário por uma máscara colorida sobreposta às fatias do exame, como ilustra a figura 2. É possível criar várias máscaras, uma para cada segmentação. As máscaras também são armazenadas em uma matriz *memmap*.

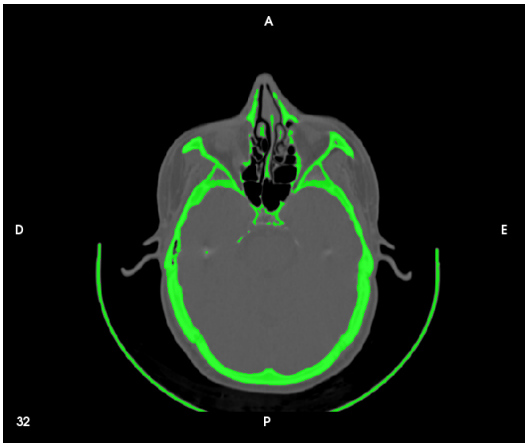


Figura 2. Imagem com *Threshold* para osso

4.2. Superfície 3D

Após segmentar a região de interesse, o usuário tem a opção de reconstrução 3D. Isso é feito através do algoritmo de *Marching Cubes* [Lorensen and Cline 1987], utilizando a implementação disponível na biblioteca VTK. Baseado nos voxels do exame segmentado, o algoritmo gera uma superfície poligonal, constituída geralmente por triângulos. Essa superfície pode ser impressa fisicamente por meio de técnicas de prototipagem rápida [Volpato 2007], para uso posterior em planejamento cirúrgico por médicos e odontologistas, ou, ainda, para o ensino, principalmente de anatomia.

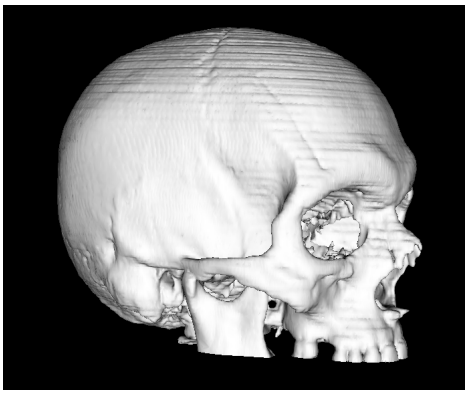


Figura 3. Superfície 3D do Osso

4.3. Visualização Volumétrica

Outro método de visualização 3D implementado no software é o *volume rendering*, cujas visualizações são mais realistas. No software foi utilizada a técnica conhecida como *Raycasting*. De modo simplificado, nesta técnica para cada pixel da tela é traçado um raio de luz (daí o nome *Raycasting*) em direção ao objeto. A cor do pixel será baseada na cor e transparência de cada *voxel* interceptado pelo raio de luz [Manssour and Freitas 2002]. Em TC os valores de cores e transparência são atribuídos aos níveis de cinza e estes valores dependem da necessidade do usuário que irá visualizar este volume e do estudo a ser realizado. Essa técnica foi implementada utilizando a biblioteca VTK. A figura 4 ilustra algumas projeções realizadas pelo software.

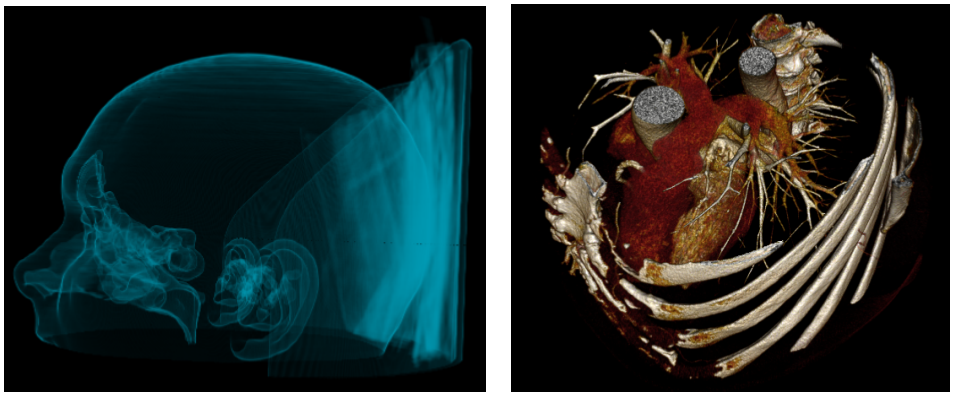


Figura 4. Visualização Volumétrica por *Raycasting*

4.4. Medidas

O InVesalius apresenta as ferramentas de medidas linear e angular, ambas disponíveis tanto nas fatias 2D quanto na reconstrução em 3D. A ferramenta de medida linear permite medir a distância entre dois pontos marcados pelo usuário. Já a ferramenta de medida angular mede o ângulo formado pelas duas semi-retas dos três pontos marcados pelo usuário. Essas ferramentas são importantes para, por exemplo, medir os avanços de um tratamento ou cirurgia, fazendo as medidas em exames antes e depois da cirurgia ou tratamento. São úteis também para medições de estruturas visíveis nos exames e para o planejamento cirúrgico. As representações como ilustra a figura 5, foram implementadas com a biblioteca VTK.

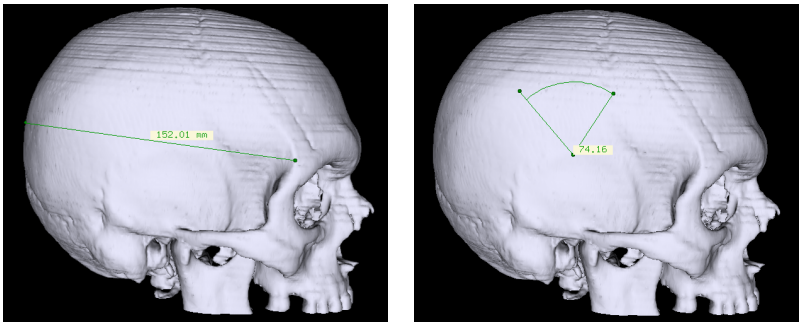


Figura 5. Medições linear e angular

5. Comunidade de usuários

O InVesalius, sua comunidade e todo o seu desenvolvimento estão centralizados no portal www.softwarepublico.gov.br. Atualmente, a comunidade conta com 4833 membros, espalhados em 71 países, sendo Brasil, Portugal, Estados Unidos, Espanha e Itália os que concentram o maior número de usuários.

De acordo com pesquisa realizada em fevereiro de 2011, respondida por 120 usuários, o perfil da comunidade encontra-se assim distribuído: 54% dos usuários são da área de saúde; 25%, de computação; 9%, das engenharias, e 12%, de outras áreas (veterinária, gestão hospitalar, paleontologia entre outras). A pesquisa mostrou também que 63% dos usuários estão satisfeitos e 12% muito satisfeitos com o uso do software.

6. Conclusão

Neste trabalho foi apresentado o InVesalius, software multiplataforma, multi-idioma e executável em computadores pessoais, uma alternativa livre aos softwares proprietários de visualização médica e reconstrução tridimensional. Desde a sua concepção, o InVesalius tem sido bem aceito pela comunidade de usuários, tanto no Brasil quanto no exterior. Além disso, o fato de parte significativa da comunidade não ser da área de saúde demonstra a versatilidade do software e sua aplicação em outras áreas do conhecimento, contribuindo também para o fomento no desenvolvimento de novas aplicações no Brasil. O desenvolvimento do InVesalius é constante. Para o futuro tem-se como metas o suporte a outros formatos de exames médicos, suporte a outras técnicas de segmentação, novas maneiras de interação com o software, e transmissão de imagens por rede.

Referências

González, R. and Woods, R. (2008). *Digital Image Processing*. Pearson/Prentice Hall.

Kinser, J. (2009). *Python for Bioinformatics*. Jones and Bartlett Publishers.

Lorensen, W. E. and Cline, H. E. (1987). Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21:163–169.

Malaterre, M. (2011). Grassroots dicom - http://sourceforge.net/projects/gdcm/.

Manssour, I. H. and Freitas, C. M. D. S. (2002). Visualização volumétrica. *Revista de Informática Teórica e Aplicada IX*, pages 97–126.

NiBabel (2011). Access a cacophony of neuro-imaging file formats - http://nipy.sourceforge.net/nibabel/.

Pianykh, O. S. (2007). *Digital Imaging and Communications in Medicine (DICOM)*. Springer Publising, 1th edition.

Rappin, N. and Dunn, R. (2006). *wxPython in action*. Manning Pubs Co Series. Manning.

Schroeder, W., Martin, K., Martin, K., and Lorensen, B. (2004). *The visualization toolkit*. Prentice Hall PTR.

Subversion (2011). Subversion. http://subversion.tigris.org/. accessed 29-Mar-2011.

Trac (2011). The trac project. http://trac.edgewall.org/. accessed 29-Mar-2011.

Volpato, N. (2007). *Prototipagem Rápida: Tecnologias e Aplicações*. Edgard Blucher.

Yoo, T. S. (2004). *Insight into Images*. A. K. Petters, 1th edition.

Identificação de Nódulos Pulmonares Utilizando a Construção de Mapas de Conexidade Fuzzy Partindo da Escolha de uma Semente Ideal

Tiago Emmanuel Praxedes Silva[1], Marcelo Costa Oliveira[1]

[1]Curso de Ciência da Computação – Universidade Federal de Alagoas (UFAL) – Campus Arapiraca
Caixa Postal 61 – 57300-970 – Arapiraca – AL – Brasil

{teps19, oliveiramc}@gmail.com

Abstract. *The main objective of this study was to compare quantitatively the accuracy of the segmentation through the algorithm for map construction of fuzzy connectedness based on the choice of an ideal seed compared to manual segmentation. We randomly selected 30 nodules of the project "Lung Image Database Consortium". The developed algorithm was effective in identifying lung nodules reaching 90.4% accuracy. Moreover, it surpassed the accuracy of the traditional segmentation algorithm for region growing at 9.1%. Therefore, this technique has the potential to be used as a tool for the aid to diagnosis of lung cancer.*

Resumo. *O objetivo principal deste trabalho foi comparar de maneira quantitativa a precisão da segmentação por meio do algoritmo de construção de mapas de conexidade fuzzy partindo da escolha de uma semente ideal em relação à segmentação manual. Foram selecionados de maneira aleatória 30 nódulos do projeto "Lung Image Database Consortium". O algoritmo desenvolvido se mostrou eficaz em identificar os nódulos pulmonares atingindo 90,4% de precisão. Além disso, superou a precisão do tradicional algoritmo de segmentação por crescimento de região em 9,1%. Logo, esta técnica possui potencial para ser utilizada como uma ferramenta de auxílio ao diagnóstico do câncer de pulmão.*

1. Introdução

O câncer é um crescimento celular anormal, incontrolável, que invade tecidos vizinhos e os destroem. Somente nos Estados Unidos em 2005, foram registrados mais de 175 mil casos de câncer do pulmão que levaram a morte mais de 160 mil pessoas. Este total é tão expressivo que supera o número de óbitos provocados pela soma dos 3 tipos de câncer de maior incidência (mama, colorretal e próstata). Segundo dados do Instituto Nacional do Câncer - INCA (www.inca.org.br), o número de novos casos do câncer de pulmão estimados para o Brasil no ano de 2008 foi de 17.810 entre homens e de 9.460 para mulheres. Entretanto, pacientes que são diagnosticados precocemente, logo nos primeiros meses da doença, possuem 60% a 90% de chance de sobrevivência [Bellotti e Carlo 2007]. Logo, a detecção e o tratamento precoce do câncer de pulmão são formas efetivas de garantir a vida dos pacientes. Todavia, a detecção e a classificação dos

nódulos em benigno ou maligno são tarefas desafiadoras aos especialistas, pois os nódulos são pequenos, apresentam baixo contraste e normalmente estão inclusos em estruturas anatômicas complexas. A segmentação do nódulo pulmonar é tipicamente realizada de forma manual, por um médico especialista. Porém, a extração manual do contorno do nódulo é tediosa, consome muito tempo e sofre de variações intra e inter-observador, necessitando assim de uma forma rápida e precisa para identificar nódulos pulmonares. Logo a implementação de um método semi-automático de segmentação favorece o diagnóstico precoce do câncer de pulmão aumentando a chance de sobrevivência do paciente.

Diante deste contexto, diversos métodos de auxílio computadorizado ao diagnóstico (CAD) foram propostos na literatura para assistir os especialistas em detectar o câncer de pulmão precocemente. A segmentação é uma das principais técnicas de processamento de imagem utilizadas no CAD, pois permite ao especialista separar o(s) nódulo(s) suspeito(s) da complexa estrutura anatômica pulmonar [Yoo 2004]. Além disso, permite analisar o volume do nódulo de maneira quantitativa e qualitativa por meio da reconstrução 3D que permite mensurar a gravidade e a evolução do nódulo mais facilmente. Uma técnica bastante utilizada na segmentação de imagens do pulmão é a de Crescimento de Região [Sluimer e Schilham 2006]. A literatura apresenta índices de precisão na segmentação de nódulos pulmonares isolados de até 68,5% [Sonka, Park e Hoffman 2008]. Todavia, um método ouro para a segmentação precisa dos diversos tipos de nódulos permanece um problema cujo qual a literatura não resolve. Dehmeshki *et al.* [Dehmeshki e Amin 2008] propuseram um algoritmo baseado no crescimento de região limitado por um mapa de conexidade fuzzy utlizando o critério de busca por uma semente ideal capaz de segmentar diversos tipos de nódulos com precisão média de 80%. Os tipos de nódulos segmentados foram: os anexos a estruturas vasculares, nódulos próximos ao diafragma e também os que apresentavam baixo e alto contraste. Contudo, os resultados foram obtidos adotando uma avaliação qualitativa dos resultados pelos especialistas, que em muitos casos pode ser tendenciosa.

Logo, o objetivo principal deste trabalho foi comparar de maneira quantitativa a precisão da segmentação por meio do algoritmo de construção de mapas de conexidade fuzzy partindo da escolha de uma semente ideal em relação à segmentação manual. Como objetivo secundário pode-se citar o teste da hipótese de que os algoritmos de crescimento de região produzem resultados mais precisos quando direcionados pela conexidade fuzzy na segmentação de nódulos pulmonares.

Este artigo é estruturado da seguinte forma: seção 1 (Introdução), a justificativa e o objetivo deste trabalho são descritos nesta seção; seção 2 (Materiais e Métodos), esta seção descreve os principais métodos (subseções 2.1 e 2.2) e materiais que foram utilizados ao longo do desenvolvimento deste trabalho; a seção 3 (Resultados e Discussão) mostra os resultados obtidos a partir dos testes realizados e a discussão acerca destes resultados em comparação com outros métodos de segmentação de imagens; seção 4 (Conclusão), além de enfatizar as realizações que este trabalho conseguiu alcançar também cita uma proposta para desenvolver e inserir uma nova funcionalidade ao algoritmo.

2. Materiais e Métodos

Os algoritmos computacionais foram desenvolvidos utilizando a linguagem de programação Java 1.6 no sistema operacional Linux Ubuntu 10.04. Este trabalho utilizou as imagens de uso público do projeto "*Lung Image Database Consortium*" (LIDC). O LIDC contém 90 exames com 829 nódulos malignos e benignos. Cada exame possui em média 200 imagens DICOM ("*Digital Imaging Communications in Medicine"*) de Tomografia Computadorizada (CT) de Tórax. Um arquivo XML descreve as posições cartesianas da segmentação manual realizada por um especialista dos nódulos e a classificação de cada nódulo pertencente ao exame [Mcnitt-Gray e Armato 2007].

Neste trabalho foram desenvolvidas as técnicas de segmentação utilizando apenas o algoritmo de crescimento de região e a segmentação por crescimento de região orientada por mapas de conexidade fuzzy. Foram selecionados de maneira aleatória 30 nódulos do LIDC. Para cada nódulo foi selecionado o seu corte mediano para aplicar os algoritmos. Os testes dos algoritmos foram aplicados dez vezes para cada nódulo buscando demonstrar consistência entre os resultados, pois como os algoritmos são semi-automáticos estes necessitam de ajustes no nível de segmentação para poder assim alcançar uma precisão satisfatória. Assim, neste trabalho foram avaliadas as precisões dos métodos de segmentação semi-automática tendo como base a segmentação manual. A precisão é obtida por meio do cálculo da área em pixels resultante da subtração de imagens entre os nódulos segmentados, via segmentação automática ou manual.

O trabalho foi baseado em três etapas de desenvolvimento: interface, leitura da imagem DICOM e a principal, que é a segmentação da imagem utilizando os algoritmos de Crescimento de Região e Conexidade Fuzzy. A interface oferece algumas opções, dentre elas estão o leitor e visualizador de imagens no padrão DICOM e as ferramentas básicas de processamento de imagens (zoom, translação e janelamento).

A partir da interface gráfica (Figura 1 - A) o especialista seleciona o arquivo de imagem DICOM de interesse, ao ser exibida a imagem o especialista seleciona a região de interesse e "planta" uma semente (pixel base) de onde o algoritmo de busca por uma semente ideal tem início, a seleção dessa região de interesse (ROI) é feita por meio do mouse. Também na interface existe um "*slider*" (Figura 1 - B) que controla o nível de segmentação da imagem de acordo com os critérios da Afinidade Fuzzy (subseção 2.2). Para facilitar a interação do especialista, os valores de afinidade foram normalizados entre 0 e 1. Desta forma, quanto mais próximo do valor 0 maior o número de ramificações no nódulo, caso contrário, quando se aproxima mais do valor 1 o número de ramificações diminui ocasionando uma segmentação mais contida.

Desta forma o especialista encontra mais recursos não só por meio dos algoritmos de crescimento de região e conexidade Fuzzy, mas também por meio do "*slider*" para um maior controle no nível de segmentação, conseguindo assim uma maior precisão e detalhamento sobre o nódulo a ser segmentado.

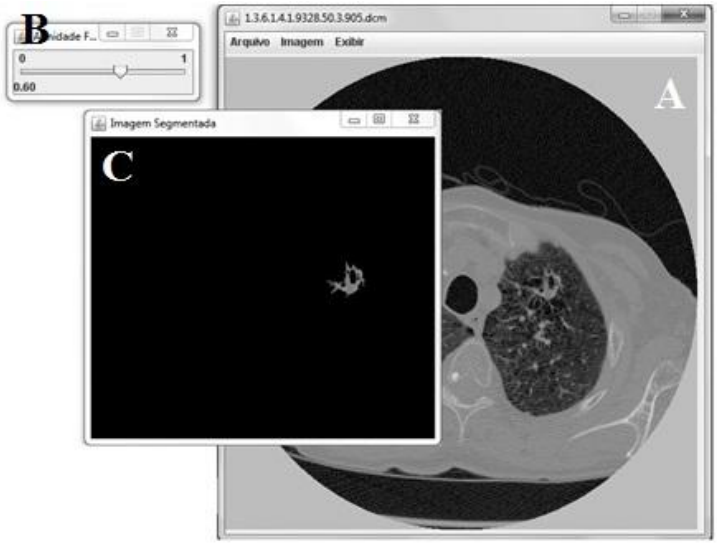


Figura 1. (A) Interface de execução do algoritmo de segmentação; (B) Slider de ajuste; (C) Resultado da Segmentação.

2.1. Busca por uma semente ideal

Devido à complexidade da anatomia e a dificuldade do especialista em definir a melhor "semente", foi desenvolvido um algoritmo de busca capaz de identificar uma "semente" ideal de acordo com a semente inicial que foi selecionada pelo usuário, esta semente ideal é um pixel que possui um nível de intensidade maior do que a dos seus vizinhos, este método foi implementado em forma de uma busca em profundidade em um grafo, efetuando a comparação dos quatro vizinhos para assim poder verificar se estes têm relação de conectividade. Por exemplo, um pixel *p* nas coordenadas (*x, y*) possui quatro vizinhos horizontais e verticais, cujas coordenadas são dadas por (*x+1*, *y*), (*x-1*, *y*), (*x*, *y+1*), (*x*, *y-1*).

Segundo [Gonzalez e Woods 2002], a conectividade entre pixels é um conceito importante usado no estabelecimento das bordas de objetos e componentes de regiões em uma imagem. Para estabelecer se dois pixels estão conectados, é preciso determinar se eles são de alguma forma adjacentes (digamos, se são *vizinhos-de-4*) e se seus níveis de cinza satisfazem um certo critério de similaridade (digamos, se eles são iguais). Por exemplo, em uma imagem binária com valores 0 e 1, dois pixels podem ser vizinhos-de-4, mas eles não são ditos conectados a menos que tenham o mesmo valor.

2.2. Conexidade Fuzzy

As funções conhecidas como fuzzy permitem quantificar o quão próximo dois pixels estão e quanto parecidos eles são em relação a propriedades baseadas nos brilhos de pixels presentes em vizinhanças dos pixels considerados. O valor de conexidade entre dois pixels *p* e *q* é dado pela maior força de conexidade entre todos os caminhos possíveis de *p* a *q* na imagem. A força de conexidade de um caminho é definida como sendo o valor do elo mais fraco neste caminho, ou seja, a menor afinidade entre pixels vizinhos que estejam neste caminho. A conexidade fuzzy permite a segmentação de objetos irregulares e complexos, pois é fortemente baseado na intensidade do pixel,

assim, pode utilizar o valor da conexidade para direcionar de forma mais acurada o algoritmo de crescimento de região [Zhou e Bai 2007].

A segmentação fuzzy é uma técnica que determina para cada elemento da imagem um grau de pertinência (entre zero e um) indicando a certeza dele pertencer (ou não) a um determinado objeto (que acredita-se estar contido na imagem) [Herman e Carvalho 2001]. Essas informações de pertinência possibilitam que o algoritmo segmente a imagem tomando uma decisão mais flexível acerca da classificação de cada elemento. A semente ideal dará início ao algoritmo de crescimento de região, que também usa a *vizinhança-de-4* para verificar e atribuir os pesos aos pixels da imagem construindo assim um mapa de conexidade entre os pixels da imagem com o auxílio de uma função, esta função pode ser entendida como uma função de afinidade fuzzy que irá determinar o quanto um pixel pertence à região de interesse numa escala de 0 a 1, onde o valor 1 corresponde a pertinência total (pixel semente ideal da segmentação) e o valor 0 como não pertencendo a região. Um pixel *p* é dito conectado ao pixel semente se este pixel *p* obtiver um nível de afinidade forte, ou seja, se o seu grau de pertinência (de 0 a 1) satisfizer as condições para pertencer a uma determinada região de interesse, este processo é conhecido como Conexidade Fuzzy.

No mapa de conexidade pode ser definido um limiar para assim poder obter como resultado a segmentação da imagem. Abaixo segue a descrição geral do algoritmo da conexidade fuzzy segundo [Sonka, Hlavac e Boyle 2008]:

1. Definir propriedades de adjacência e afinidade fuzzy;
2. Determinar os valores de afinidade para todos os pares de spels ("*spacial elements"* ou pixels) adjacentes;
3. Determinar a semente (pixel semente ideal) da segmentação;
4. Determinar todos os caminhos possíveis entre a semente e todos os outros elementos da imagem, dentro do domínio da imagem (sem formação de loops) considerando o relacionamento da adjacência fuzzy;
5. Para cada caminho, determinar a força deste caminho de acordo com a afinidade mínima dos próximos caminhos;
6. Para cada elemento da imagem, determinar a conexidade fuzzy entre este elemento e a semente conforme a força máxima de todos os possíveis caminhos e como resultado formar o mapa de conexidade desta imagem;
7. Limitar o mapa de conexidade com um limiar apropriado para segmentar a imagem dentro de um objeto contendo a semente e o "*background"*.

3. Resultados e Discussão

A segmentação fuzzy de imagens tem se mostrado eficaz nos casos em que a característica que distingue o objeto de interesse do "*background"* é alguma propriedade textural, ao invés do exato valor atribuído aos pixels, como por exemplo, imagens contendo sombras, iluminação não uniforme e/ou ruídos [Herman e Carvalho 2001]. A presença dessas interferências em uma imagem requer dos algoritmos de segmentação um esforço adicional durante o reconhecimento dos objetos presentes na mesma. Devido a estas características a conexidade fuzzy tem-se tornado uma promissora técnica de

auxílio ao diagnóstico do câncer de pulmão, visto que, soluciona as principais problemáticas intrínsecas do nódulo como o baixo contraste e a inclusão junto a estruturas.

Neste trabalho foi desenvolvida uma ferramenta de código livre para auxiliar especialistas na identificação de nódulos pulmonares. O banco de nódulos desenvolvido obedece à estrutura do arquivo XML criado pelo LIDC. Dessa forma, exames futuros inseridos no banco obedecerão às especificações do projeto LIDC. Isto permitirá a padronização da base de nódulos e, logo, a colaboração com os idealizadores do projeto.

O sistema desenvolvido permitiu ao especialista visualizar em detalhes o resultado da segmentação obtida por meio de uma interface gráfica (Figura 1 - C). Assim, caso a segmentação não tenha sido satisfatória ele pode retornar para a interface de segmentação e definir novos parâmetros à segmentação.

A segmentação semi-automática utilizando a Conexidade Fuzzy partindo da escolha de uma semente ideal obteve precisão média de 90,4% e superou em 9,1% a precisão do algoritmo de Crescimento de Região (Tabela 1). Na figura 2 é ilustrado o resultado da segmentação de três nódulos malignos utilizando a segmentação manual e os algoritmos semi-automáticos.

Tabela 1. Médias dos níveis de precisão dos algoritmos de segmentação.

	1º Grupo	2º Grupo	3º Grupo
Crescimento de Região	86,6%	79,9%	78,4%
Conexidade Fuzzy	98,4%	84,1%	88,8%

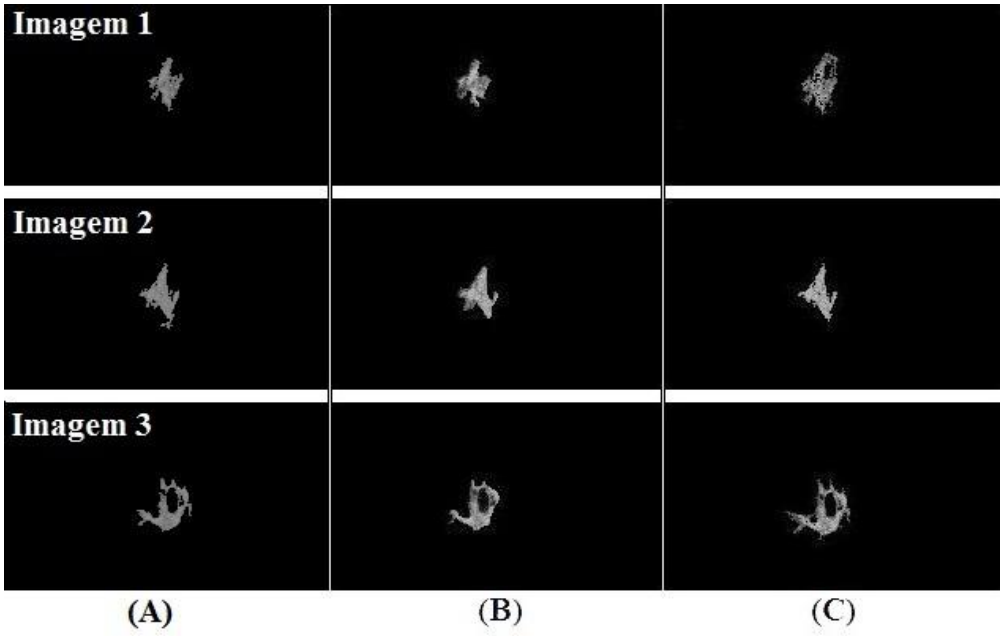


Figura 2. Imagens segmentadas pelos diferentes métodos. (A)Conexidade Fuzzy; (B) Manual; (C) Crescimento de Região.

A Tabela 1 apresenta diferentes níveis de precisão para cada um dos grupos de imagens, isto ocorre em virtude da variedade dos tipos de nódulos segmentados, nódulos que se encontram junto à parede pulmonar, anexos a estruturas vasculares ou ao

diafragma. A tarefa de segmentar o nódulo que se encontra junto à parede pulmonar é bem desafiadora, pois se torna consideravelmente difícil identificar a região do nódulo da região pertencente à parede pulmonar, isto é devido à semelhança quase que indistinguível dessas duas regiões, a segmentação neste caso é de complexidade considerada de nível alto. Já no caso em que o nódulo encontra-se junto a estruturas vasculares a segmentação é considerada de nível de complexidade mediano, pois as características que distinguem o nódulo das estruturas vasculares como intensidade do pixel e conexidade entre eles tornam-se mais evidentes. Portanto, como estes dois casos diferenciam em termos de complexidade da segmentação, os níveis de precisão entre os métodos de segmentação tendem também a ser diferentes, como já foi mencionada e explicada a conexidade fuzzy não faz uso somente do algoritmo de crescimento de região, mas também da criação de um mapa de conexidade fuzzy partindo da escolha de uma semente ideal o que a faz obter níveis de precisão superiores a segmentação que utiliza apenas o algoritmo de crescimento de região. Esta variedade nos tipos de nódulo foi de importância fundamental para a análise dos diferentes métodos de segmentação.

4. Conclusão

O algoritmo de construção de mapas de conexidade fuzzy partindo da escolha de uma semente ideal mostrou níveis bastante satisfatórios, isto para diferentes tipos de imagens. O algoritmo obteve resultados próximos ao da segmentação realizada de forma manual pelo especialista. Além disso, em todos os casos avaliados o algoritmo obteve precisão superior a técnica de segmentação que utiliza apenas o algoritmo de crescimento de região. A conexidade fuzzy foi capaz de isolar o nódulo suspeito, excluindo demais estruturas anatômicas que não pertenciam ao mesmo. A ferramenta desenvolvida neste trabalho possui potencial para ser utilizada como uma ferramenta de auxílio ao diagnóstico do câncer de pulmão. É de código livre, fácil uso e multiplataforma.

Como proposta para trabalhos futuros uma nova funcionalidade será desenvolvida e inserida no algoritmo, para realizar a segmentação tridimensional dos nódulos pulmonares.

Referências

Ayres, P. A. A., Cavalcante, R. C., Oliveira, M. C. (2010) “Auxílio à Classificação de Nódulos Pulmonares Usando Recuperação de Imagens Similares Baseada em Análise de Textura 3D e Registro de Imagem 3D”, XXX Congresso da Sociedade Brasileira de Computação - WIM - X Workshop de Informática Médica, Belo Horizonte/MG - Brasil.

G. T. Herman and B. M. Carvalho. (2001) “Multiseeded segmentation using fuzzy connectedness”, IEEE Transactions on Pattern Analysis and Machine Intelligence, p. 460–474.

INCA – Instituto Nacional do Câncer. (2010) “www.inca.org.br”.

J. Dehmeshi, H. Amin, M. Valdivieso, and X. J. Ye. (2008) "Segmentation of pulmonary nodules in thoracic CT scans: A region growing approach", IEEE Transactions on Medical Imaging, vol. 27, p. 467-480.

M. F. McNitt-Gray, S. G. Armato, C. R. Meyer, A. P. Reeves, G. McLennan, R. C. Pais, J. Freymann, M. S. Brown, R. M. Engelmann, P. H. Bland, G. E. Laderach, C. Piker, J. Guo, Z. Towfic, D. P. Y. Qing, D. F. Yankelevitz, D. R. Aberle, E. J. R. van Beek, H. MacMahon, E. A. Kazerooni, B. Y. Croft, and L. P. Clarke. (2008) "The Lung Image Database Consortium (LIDC) data collection process for nodule detection and annotation," Academic Radiology, vol. 14, pp. 1464-1474.

M. Sonka, W. Park, and E. A. Hoffman. (1996) “Rule-based detection of intrathoracic airway trees,” IEEE Trans. Med. Imag, vol. 15, no. 3, p. 314-326.

Milan Sonka, Vaclav Hlavac, and Roger Boyle. (2008) “Image Processing, Analysis, and Machine Vision”, Thomson, 3th edition.

R. Bellotti, F. De Carlo, G. Gargano, S. Tangaro, D. Cascio, E. Catanzariti, P. Cerello, S. C. Cheran, P. Delogu, I. De Mitri, C. Fulcheri, D. Grosso, A. Retico, S. Squarcia, E. Tommasi, and B. Golosio. (2007) "A CAD system for nodule detection in low-dose lung CTs based on region growing and a new active contour model," Medical Physics, vol. 34, p. 4901-4910.

R. Gonzales and R. Woods. (2002) “Digital Image Processing”, Upper Saddle River - NJ: Prentice-Hall, 2th edition.

Sluimer, I., A. Schilham. (2006) "Computer analysis of computed tomography scans of the lung: A survey." IEEE Transactions on Medical Imaging, p. 385-405.

T. Yoo. (2004) “Insight into images: principles and practice for segmentation, registration and image analysis”, vol. 1, AK Peters LTDA, Massachusetts, USA.

Y. X. Zhou and J. Bai. (2007) "Atlas-based fuzzy connectedness segmentation and intensity nonuniformity correction applied to brain MRI", IEEE Transactions on Biomedical Engineering, vol. 54, p. 122-129.

Um Método para Correção de Imagens de *Phantom* e Classificação de suas Estruturas de Interesse

Bruno Barufaldi[1], Amanda B. Cavalcanti[1], Leonardo V. Batista[1], Renata F. Góis[2], Homero Schiabel[2], José F. G. Carvalho[3]

[1]Departamento de Informática - Universidade Federal da Paraíba (UFPB)
Programa de Pós Graduação em Informática - João Pessoa, PB - Brasil

[2]Departamento de Engenharia Elétrica - Universidade de São Paulo (USP)
Escola de Engenharia de São Carlos - São Carlos, SP - Brasil

[3]Governo da Paraíba - Agência Estadual de Vigilância Sanitária da Paraíba
João Pessoa, PB – Brasil

{barufaldi, amanda, leonardo}@di.ufpb.br, {rfgoes, homero}@sc.usp.br

Abstract. *To execute the quality control of mammographic systems, the Brazilian Health Ministry demands the use of breast phantoms. Aiming to reduce the subjectivity present in the evaluation of phantom images through human visual inspection, a computerized system has been developed that uses a correction method in its digitised images, associated with the classification of its structures of interest by the visibility criterion. Comparing the results of the classification using the J48 algorithm of the WEKA package with and without image correction, this method presented a significant improvement in the effectiveness for determining structures of the phantom.*

Resumo. *Para realizar o controle de qualidade de sistemas mamográficos o Ministério da Saúde exige o uso de simuladores radiográficos (phantoms) de mama. Com o objetivo de reduzir a subjetividade na avaliação das imagens de phantom pela inspeção visual humana, está sendo desenvolvido um sistema computadorizado que utiliza um método de correção em imagens digitalizadas, associado à classificação de suas estruturas de interesse pelo critério de visibilidade. Ao comparar os resultados da classificação através do algoritmo J48 da ferramenta WEKA com e sem a correção das imagens, essa técnica apresentou uma melhora significativa na eficácia para determinadas estruturas do phantom.*

1. Introdução

O câncer de mama é a segunda neoplasia mais incidente na população mundial feminina. A cada ano, cerca de 22% dos casos novos de câncer em mulheres são de mama [INCA 2009]. De acordo com os dados da GLOBOCAN (*International Agency for Research on Cancer*), em 2008 foram registrados em todo o mundo 691 mil novos casos de câncer de mama em países em desenvolvimento e 692 mil em países desenvolvidos. No entanto, o número total de mortes pela doença em países em desenvolvimento foi de 268 mil, em contraste com 189 mil mortes nos países desenvolvidos [Tiezzi 2010].

A mamografia é o exame mais confiável e acessível para o rastreamento do câncer de mama. A sua eficácia e indicação são aceitas mundialmente e, na faixa etária entre 50 e 69 anos, a redução da mortalidade pela doença é da ordem de 30%. Já o benefício para mulheres entre 40 a 50 anos gira em torno de 15 a 18% [Tiezzi 2010].

O mecanismo de inspeção das mamografias é um procedimento custoso e predisposto a erros. Para a redução destas falhas, o Programa de Certificação da Qualidade em Mamografia do Colégio Brasileiro de Radiologia (CBR) exige que os serviços de mamografia efetuem o registro radiográfico de um simulador de mama, denominado *phantoms* [Cinti *et al.* 2004].

A digitalização de imagens radiográficas é uma etapa importante para processo de detecção das estruturas de interesse para o diagnóstico médico. O método mais simples para a aquisição de uma imagem no formato digital é através da utilização de um *scanner*. Para que os esquemas computacionais tenham sucesso, a digitalização deve seguir alguns critérios mínimos quanto à relação sinal ruído da imagem e do dispositivo digitalizador utilizado.

A utilização de sistemas computadorizados de auxílio ao diagnóstico torna-se necessária para manter a qualidade das atividades e serviços clínicos [Hogge *et al.* 1999]. O objetivo principal da produção desses sistemas está em tentar eliminar a subjetividade na análise especialista, devida às limitações próprias do ser humano, diminuindo substancialmente a carga de trabalho e saturação dos radiologistas.

Atualmente existem diversas aplicações na área de Inteligência Artificial utilizadas na Medicina. Entre as mais conhecidas, estão os sistemas de apoio ao diagnóstico médico. Para o computador elaborar uma hipótese diagnóstica, é necessário que o especialista informe todos os dados relevantes observados. Com isso, o sistema é capaz de processar essas informações de maneira eficiente, auxiliando a elaboração do diagnóstico.

O objetivo deste trabalho é desenvolver um subsistema do SIstema de Auxílio ao Diagnóstico médico (SIAD), denominado QualiMamo, voltado para automatização da maior parte das etapas do processo de controle de qualidade de mamógrafos atualmente adotado pela Agência de Vigilância Sanitária do Estado da Paraíba (AGEVISA-PB), visando ser uma ferramenta de auxílio ao controle de qualidade. Esse subsistema busca reduzir a subjetividade na avaliação das imagens do *phantom* MAMA-CDM [Medeiros *et al.* 2002] pela inspeção visual humana, utilizando um método de correção em imagens digitalizadas, associado à classificação de suas estruturas de interesse pelo critério de visibilidade.

2. Fundamentação Teórica

Essa seção apresenta os conceitos considerados relevantes à completa compreensão deste trabalho.

2.1 Filmes Radiográficos

O principal objetivo da realização de qualquer tipo de tomada radiográfica está na fiel reprodução dos objetos de interesse para o diagnóstico. Para a representação radiográfica é utilizado um filme sensível à luz juntamente a uma tela florescente (écran), capaz de converter cada fóton de raios X em um feixe de fótons de luz visível que é

absorvido pelo filme. Para descrever a resposta de um sistema écran-filme à exposição radiográfica utiliza-se um gráfico chamado curva característica, que relaciona a exposição radiográfica com a densidade ótica produzida no filme. A resposta de um filme ou de uma combinação écran-filme à exposição de raios X é dada pela curva característica, que usa uma escala logarítmica para representar a exposição.

Por intermédio da curva característica pode-se determinar o contraste do filme, através do grau de inclinação da parte linear da curva, ou seja, da faixa de densidades úteis. Uma inclinação mais acentuada indica alto contraste.

2.2 Processo de Digitalização

Os aspectos de digitalização de filmes radiográficos devem ser observados para garantir a eficiência do processo do diagnóstico. Os níveis de cinza vão ser processados a fim de informarem o tamanho e localização das estruturas de interesse da imagem. Se a digitalização não fornecer um tamanho suficientemente pequeno de pixel, algumas estruturas pequenas podem ser desprezadas e até mesmo desaparecer [Chan *et al*. 1994; 1997].

Muitos equipamentos radiográficos digitais têm sido desenvolvidos para a realização do diagnóstico, mas a aceitação clínica tem sido lenta, devido ao seu alto custo. Tendo em vista as vantagens de se adquirir a imagem radiográfica em sua forma digital, a opção imediata é adquiri-la por meio da digitalização utilizando um *scanner* que contenha um sistema adaptador de filmes, onde se registra a luz transmitida e não a refletida.

No entanto, sabe-se que o processo de digitalização da imagem radiográfica gera ruídos na imagem que correspondem a distorções visuais que não contribuem com informação, muitas vezes dificultando a visualização das estruturas de interesse das imagens [González-López 2007].

Sendo assim, é necessário realizar um acompanhamento do desempenho do digitalizador utilizado, verificando a confiabilidade da resposta dada por este sistema bem como a qualidade de suas imagens.

2.3 Simulador Radiográfico MAMA-CDM

Nos centros de radiodiagnósticos são utilizados rotineiramente simuladores radiográficos (ou *phantoms*) que contêm estruturas que representam os tecidos da mama. O principal objetivo dos *phantoms* é simular a interação do feixe de raios X com o paciente, para avaliar a atenuação da radiação e qualidade da imagem no filme mamográfico.

O *phantom* MAMA-CDM simula uma mama comprimida entre 4cm e 5cm e possui detalhes que produzem imagens radiográficas semelhantes às estruturas presentes na mama (microcalcificações, discos, fibras e massas tumorais).

São avaliados nas imagens dos simuladores a definição (resolução espacial), os detalhes de alto contraste (microcalcificações), os limiares de baixo contraste (discos), os detalhes lineares de baixo contraste (tecido fibroso), as massas tumorais e a densidade ótica de fundo. A Figura 1 apresenta uma foto do *phantom* MAMA-CDM e sua imagem radiográfica.

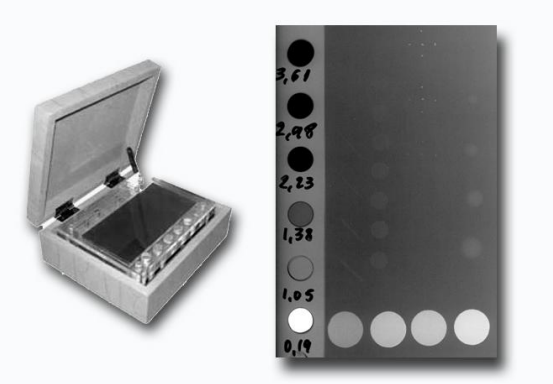


Figura 1. Simulador radiográfico MAMA-CDM e sua imagem radiográfica.

2.4 WEKA

O pacote WEKA (*Waikato Environment for Knowledge Analysis*) é formando por um conjunto de implementações de algoritmos utilizados em Mineração de Dados. Esse pacote inclui uma variedade de métodos de aprendizagem para o conjunto de dados a ser avaliado, denominados classificadores.

As árvores de decisão têm diversas vantagens quando comparadas a outras técnicas. A principal está em sua simplicidade e facilidade na interpretação dos resultados. O J48 do WEKA é a implementação otimizada do algoritmo de construção de árvores de decisão C4.5, que foi a última versão pública do algoritmo [Witten e Frank 2005]. Esse algoritmo constrói um modelo de árvore de decisão baseado em um conjunto de dados de treinamento e usa esse modelo para classificar as instâncias de um conjunto de testes.

3. Materiais e Métodos

Uma das abordagens de inspeção visual do filme radiográfico consiste em determinar a visibilidade das estruturas de interesse presentes na imagem do *phantom* MAMA-CDM.

Para a obtenção das imagens digitalizadas do *phantom*, utilizou-se o scanner Microtek, modelo *ScanMaker i800*, e foram digitalizadas em níveis de cinza, com 16 bits de resolução de contraste e 1200 x 1200 dpi de resolução espacial. Para a produção da curva característica do filme foi utilizado um filme radiográfico da marca Agfa, relevado por uma processadora de marca *Microtek-MX-2*.

A eficácia do sistema foi aferida utilizando 100 imagens corrigidas (conforme descrito na seção 3.2) do simulador radiográfico, com um total de 1900 estruturas analisadas entre discos de baixo contraste, fibras e massas tumorais. O comportamento do software é visualizado através das curvas ROC (*Receiver Operating Characteristic*), em que são verificadas a sensibilidade e a especificidade do sistema, à medida que se reduz o contraste das estruturas.

3.1 Curva Característica

Para a produção da curva característica do filme foram necessários um sensitômetro e um densitômetro (Mod. *X- Rite 334*) calibrados para medir as densidades óticas através de uma varredura sobre a região do filme sensibilizado com 21 passos de exposição. Com a utilização desses equipamentos, foi possível determinar a relação entre

as densidades óticas e a exposição do filme. Assim, as densidades óticas correspondentes a cada exposição diferente possibilitaram o levantamento da curva característica.

Uma vez definida a curva característica do filme, foi necessário conhecer a curva característica do digitalizador das imagens dos *phantoms* (*Microtek ScanMaker i800)*. Para isso, digitalizou-se dez vezes uma mesma imagem do filme neste mesmo digitalizador em diferentes posições, o que permitiu identificar as variações de intensidade existentes na escala produzidas pelo sensitômetro, correspondentes aos 21 passos de exposição do filme.

A construção da curva característica do digitalizador depende da relação entre o nível de cinza dos pontos de exposição e as medidas das densidades óticas. Essa relação associa a média dos valores dos pixels de cada região de interesse com o valor de densidade que tenha o mesmo passo de exposição. Desta forma, foi possível analisar a variação dos valores de pixels de uma região para diferentes digitalizações e, assim, verificar se há variação significativa nos pixels das imagens de um filme digitalizado em diferentes posições do digitalizador. A Figura 2 apresenta um exemplo de uma curva característica de um filme.

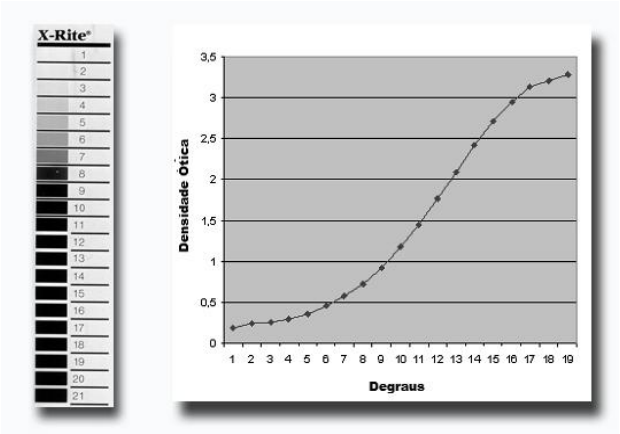


Figura 2. Exemplo de gráfico da curva característica de um filme radiográfico.

3.2 Correção das Imagens

Conhecida a curva característica do *scanner Microtek*, foi possível realizar a correção das imagens digitalizadas. O procedimento tem como função calcular, através de interpolação matemática, o polinômio que melhor representa os pontos amostrados referentes às densidades óticas obtidas para cada digitalizador. Esse polinômio tem como objetivo corrigir os níveis de cinza da imagem de modo a torná-los similares aos das imagens tomadas como referências.

Após determinado o polinômio, é necessário aplicá-lo às imagens do simulador radiográfico que foram digitalizadas. Para isso, foram realizadas operações pontuais na região que contém as estruturas de interesse da imagem. A partir desse pré-processamento, foi possível compensar o desvio do digitalizador no que se refere à intensidade da imagem e, posteriormente, localizar e classificar as estruturas da imagem resultante.

3.3 Classificação com WEKA

Para realizar a etapa de classificação, é necessário detectar as estruturas de interesse das imagens dos *phantoms* utilizados. As estruturas são detectadas através de uma técnica baseada em casamento por correlação [Gonzales e Woods R.E. 2002] apresentada em trabalhos anteriores pelos autores deste trabalho [Barufaldi *et al*. 2009]. Com o sistema desenvolvido é possível localizar as estruturas de interesse com precisão.

Para a construção dos modelos de classificação que serão gerados pelo algoritmo J48, é necessário pré-selecionar alguns atributos da imagem para realização da etapa de treinamento do sistema. É importante destacar que nem todos esses atributos serão empregados para a classificação das estruturas, já que somente algumas das características da imagem são consideradas relevantes pelo algoritmo.

Os atributos utilizados na etapa de treinamento do sistema foram determinados através do seletor de atributos da ferramenta WEKA. Esses atributos podem ter natureza global, em relação à imagem, ou local, referentes às estruturas de interesse detectadas pelo sistema.

A Tabela 1 define os atributos globais utilizados para realização da etapa de treinamento do sistema. Esses atributos serão extraídos de cada região de interesse da imagem, onde *p(i,j)* é o nível de cinza do pixel na posição *(i,j)* e *w* e *h* são, respectivamente, a largura e altura da imagem.

Tabela 1. Atributos globais extraídos das imagens dos *phantoms*.

Atributo	Média dos pixels da imagem	Média do módulo da diferença entre os pixels da imagem e a média da imagem	Variância da imagem	Desvio Padrão da imagem	Moda dos pixels da imagem
Equação	$\mu = \sum_{i=1}^{w}\sum_{j=1}^{h}\frac{p(i,j)}{(w*h)}$	$x = \sum_{i=1}^{w}\sum_{j=1}^{h}\frac{\lvert p(i,j)-\mu\rvert}{(w*h)}$	$v = \sum_{i=1}^{w}\sum_{j=1}^{h}\frac{(p(i,j)-\mu)^2}{(w*h)}$	$\sigma = \sqrt{v}$	Nível de cinza mais frequente em *p(i,j)*

A Tabela 2 apresenta os atributos extraídos de cada estrutura também utilizados na etapa de treinamento, onde $p_e(i,j)$ e $p_b(i,j)$ são os níveis de cinza dos pixels da região interna (estrutura) e externa (plano de fundo) do filtro de tamanho $w * h$ na posição *(i,j)* da imagem.

Tabela 2. Atributos extraídos das imagens dos *phantoms* após a detecção das estruturas de interesse.

Atributo	Média dos pixels da estrutura	Média dos pixels do plano de fundo	Diferença das médias dos níveis de cinza dos pixels da estrutura e plano de fundo	Razão de Weber [Gonzales e Woods 2002]
Equação	$\mu_e = \sum_{i=1}^{w}\sum_{j=1}^{h}\frac{p_e(i,j)}{(w*h)}$	$\mu_b = \sum_{i=1}^{w}\sum_{j=1}^{h}\frac{p_b(i,j)}{w*h}$	$\Delta\mu = \mu_e - \mu_b$	$W = \frac{\Delta\mu}{\mu_e}$

A etapa de treinamento consiste em produzir, através do algoritmo J48 do pacote WEKA, um modelo de árvore de decisão para cada estrutura de interesse, que permanecerá estático durante o processamento das imagens pelo sistema na etapa de classificação. Para tanto, utilizou-se 50 imagens laudadas por especialistas do Programa de Controle de Qualidade de Mamógrafos da AGEVISA-PB.

Um total de 300 estruturas foi extraído dessas imagens para treinamento, contendo discos de baixo contraste, fibras e massas tumorais. Essas estruturas foram classificadas por especialistas de acordo com sua visibilidade, ou seja, se são visíveis ou

não. É importante destacar que os laudos das imagens foram feitos por dois ou mais técnicos especializados, de maneira a diminuir a subjetividade própria da inspeção visual e aumentar a consistência do sistema.

Nessa etapa são criados os arquivos do formato padrão ARFF do WEKA, com todos os dados de entrada citados anteriormente. É possível selecionar os atributos relevantes para o treinamento através da própria ferramenta, na etapa de pré-processamento.

Na etapa seguinte, de classificação, foram implementadas árvores de decisão a partir dos modelos produzidos na etapa anterior. Foram utilizadas 100 imagens de simulador radiográfico laudadas para classificação, com um total de 1900 estruturas. É importante destacar que as imagens processadas na etapa de classificação são distintas das de treinamento.

O comportamento do software é visualizado através de curvas ROC para cada estrutura de interesse, em que são verificadas a sensibilidade e a especificidade do sistema. Para a construção das curvas ROC o valor do atributo $\Delta\mu$ foi alterado dentro de um certo intervalo, uma vez que este foi o atributo de maior relevância nas etapas de treinamento de todas as estruturas de interesse. A escolha desse atributo se deu por este ter sido o de maior relevância nas etapas de treinamento de todas as estruturas de interesse.

4. Resultados

A Figura 3 apresenta os gráficos da curva característica do filme e do *scanner Microtek ScanMaker i800*, juntamente com seu ajuste polinomial filme/*scanner*.

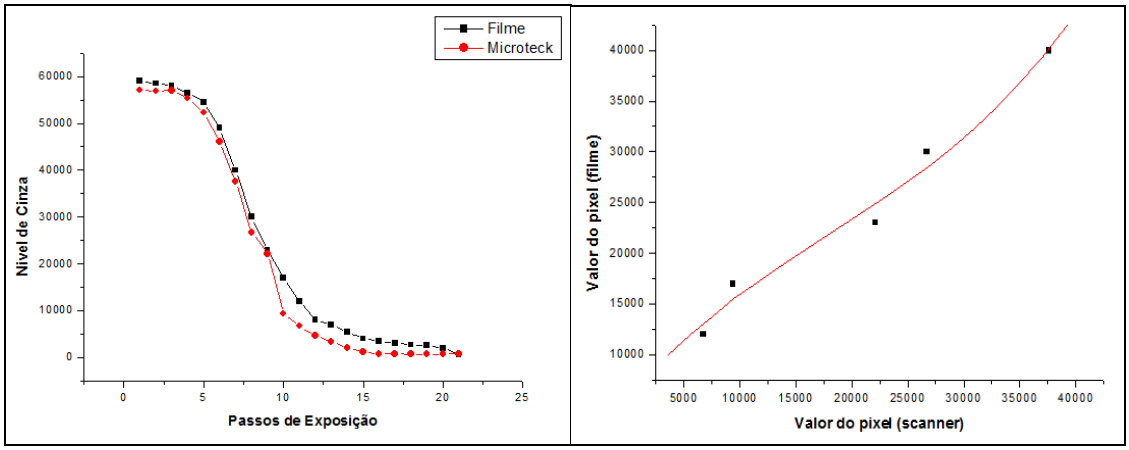


Figura 3. Curva característica filme/*scanner* e ajuste polinomial de grau 3 filme/*scanner*, $f(x) = 5{,}76E^{-10} * x^3 - 3{,}13E^{-5} * x^2 + 1{,}28 * x + 5689$.

Na comparação das imagens da Figura 4, nota-se que quando aplicado o polinômio de correção, as imagens digitalizadas apresentaram mais fidelidade às densidades óticas reais do filme.

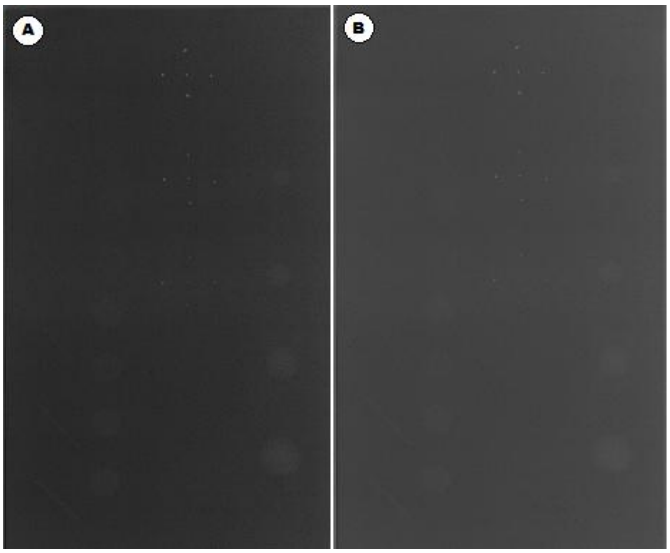


Figura 4. (A) Imagem do simulador radiográfico original digitalizada no *scanner Microteck ScanMaker i800* e (B) resultado da aplicação do polinômio de correção na mesma imagem.

Para representação da eficácia do sistema, são apresentadas as tabelas de contingência (ou matrizes de confusão), que indicam as taxas de acerto da classificação. A Tabela 3 mostra as matrizes de confusão das massas, discos e fibras, classificadas de acordo com o modelo gerado pelo algoritmo J48 com e sem a correção.

Tabela 3. Tabela de contingência das estruturas após a etapa de classificação.

		Classificação das fibras sem correção		**Classificação das massas sem correção**		**Classificação dos discos sem correção**	
		Visível	Não Visível	Visível	Não Visível	Visível	Não Visível
Classificado pelo Especialista	Visível	**0,963**	**0,037**	0,915	0,085	0,922	0,078
	Não Visível	**0,593**	**0,407**	0,04	0,96	0,11	0,89
		Classificação das fibras com correção		**Classificação das massas com correção**		**Classificação dos discos com correção**	
		Visível	Não Visível	Visível	Não Visível	Visível	Não Visível
Classificado pelo Especialista	Visível	**0,954**	**0,046**	0,915	0,085	0,922	0,078
	Não Visível	**0,407**	**0,593**	0,04	0,96	0,11	0,89

A Tabela 4 apresenta as taxas de preditividade positiva e negativa, sensibilidade, especificidade, eficiência e acurácia do sistema, para a classificação de cada estrutura de interesse.

Tabela 4. Medidas de eficácia do software para classificação das estruturas utilizando o algoritmo J48.

	Classificação das estruturas sem correção			Classificação das estruturas com correção		
	Massas	Discos	Fibras	Massas	Discos	Fibras
Acurácia	0,924	0,915	**0,833**	0,926	0,915	**0,870**
Sensibilidade	0,989	0,968	**0,844**	0,990	0,968	**0,887**
Especificidade	0,738	0,758	**0,760**	0,734	0,758	**0,790**
Eficiência	0,864	0,863	**0,802**	0,866	0,863	**0,839**
Preditividade Positiva	0,915	0,922	**0,961**	0,912	0,922	**0,952**
Preditividade Negativa	0,960	0,890	**0,410**	0,980	0,890	**0,597**

O comportamento do sistema pode ser verificado através das curvas ROC, conforme apresentado na Figura 5.

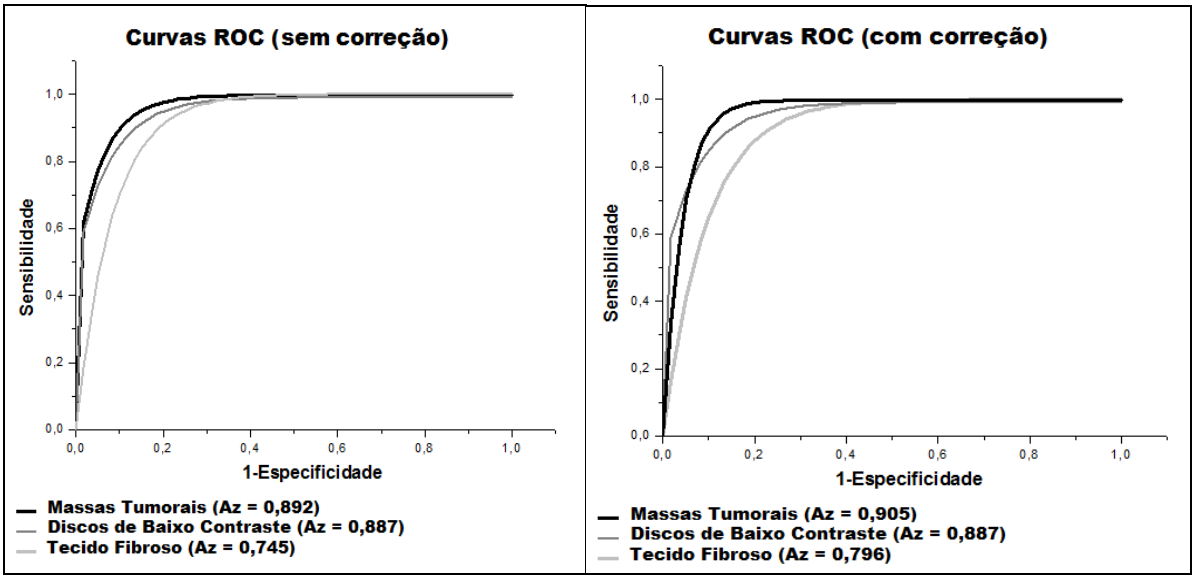


Figura 5. Curvas ROC do algoritmo J48 sem correção (esquerda) e com correção (direita) para classificação do tecido fibroso, discos de baixo contraste e massas tumorais.

5. Discussões e Conclusão

O processo de correção da digitalização das imagens do simulador radiográfico apresentou uma melhora na classificação das estruturas que compõem o tecido fibroso. Apesar de existir uma redução na taxa de sensibilidade dessas estruturas, houve um aumento em sua especificidade, o que indica uma melhor taxa de eficiência do método de correção. Acredita-se que por os discos de baixo contraste e massas tumorais apresentarem melhor definição com relação ao tecido fibroso, não houve uma melhora significativa no desempenho da técnica para a classificação.

Além disso, foi possível analisar a variação dos valores de pixels em diferentes posições do digitalizador. O método de correção realizado pelo o ajuste da curva característica do scanner em relação à curva do filme sensibilizado garante a uniformidade da digitalização em qualquer região do *scanner* avaliado (*Microteck ScanMaker i800*).

As estruturas que simulam as massas tumorais são as que apresentam um maior contraste em relação às outras estruturas analisadas. Por esse motivo, a classificação a respeito de sua visibilidade se torna mais fácil quando comparada com a das estruturas que representam os discos ou fibras.

Um dos problemas enfrentados é a presença de ruído que se assemelha a algumas estruturas. O tamanho reduzido das microcalcificações favorece a confusão entre estrutura e ruído, especialmente nos conjuntos menos evidentes. Estão sendo desenvolvidas técnicas para o tratamento de ruído dessas imagens para uma melhor detecção e classificação dessas estruturas.

Com os resultados apresentados, o método de correção das imagens do simulador radiográfico será inserido ao sistema Qualimamo com finalidade de melhorar a classificação das estruturas do tecido fibroso e, futuramente, microcalcificações. Após o término do trabalho, essa ferramenta de auxílio ao diagnóstico médico pode-se tornar viável para automação de parte dos procedimentos de controle de qualidade dos mamógrafos.

6. Agradecimentos

Ao CNPq, à CAPES e à SESU/MEC, pelo apoio na forma de bolsas de pesquisa e custeio.

Referências

Barufaldi B. et al. (2009) “Sistema para o Controle de Qualidade de Mamógrafos e Geração Automática de Laudos Através da Análise de Imagens de Fantoma”, V Workshop de Visão Computacional, São Paulo.

Cinti, M. N. et al. (2004) “Custom Breast Phantom for an Accurate Tumor SNR Analysis”, IEEE Transactions on Nuclear Science, vol. 51(1) 198-204.

Gonzales R.C. and Woods R.E. (2002) “Digital Image Processing”, 2nd ed. Prentice Hall.

González-Lopez, A. (2007) “Useful optical density range in film dosimetry: Limitations due to noise and saturation” Phys. Med. Biol. 52, p.321-327.

Hogge J. et al. (1999) “Quality Assurance in Mammography: Artifact Analysis”, RadioGraphics, vol. 19 503-522.

INCA – Instituto Nacional do Câncer (2009) “Estimativa 2010: incidência de câncer no Brasil / Instituto Nacional de Câncer”, ISBN 978-85-7318-161-6 (98).

Medeiros R. B. , Alves F.F.R., Ruberti E. M and Ferreira D.F.P (2002) “Influência das condições de processamento no desempenho de dois sistemas tela/filme utilizados na mamografia”, Revista Brasileira de Engenharia Biomédica, vol. 18 (2) 57-63.

Tiezzi D.G. (2010) “Câncer de mama: um futuro desafio para o sistema de saúde nos países em desenvolvimento”` Rev. Bras. Ginecol. Obstet. 32(6).

Witten I. H. and Frank, E. (2005) “Data Mining – Practical Machine Learning Tools and Techniques”. San Francisco, Elsevier, 2nd Edition.

Utilização de um Novo Modelo para Representação de Sinais, como Extrator de Características, na Recuperação de Imagens Médicas por Conteúdo

João L. Fernandes, Cristina N. Vasconcelos, Giovanni C. Carelli

Departamento de Ciência da Computação – Universidade Federal Fluminense (UFF)
Rua Passo da Pátria, 156 – 24210-240 – Niterói – RJ – Brasil

{ccmjlf,crisnv}@ic.uff.br, giovannicarelli@hotmail.com

Abstract. *The recovery of similar items from medical images databases can promote the acquisition of important knowledge to aid the diagnosis. The recovery of images by similarity involves the extraction of representative features of the images, which is not a trivial task, since there is usually a large number of factors that must be considered to simulate the human perception. Such techniques are known in literature as CBIR (Content-based Image Retrieval). In this paper, we propose studying a new model for signal representation, as features extractor, aiming at developing metrics of similarity between medical images.*

Resumo. *A recuperação de itens semelhantes, em bancos de dados de imagens médicas, pode ser um auxílio eficiente à elaboração de diagnósticos. Recuperar imagens por similaridade envolve a extração de características representativas dos exemplares, o que não é uma tarefa trivial, já que, normalmente, há um grande número de fatores que devem ser considerados, na tentativa de simular a percepção humana. Tais técnicas são conhecidas na literatura como CBIR (Content-based Image Retrieval). Neste trabalho, propomos o estudo de um novo modelo para representação de sinais, como extrator de características, visando ao desenvolvimento de métricas de semelhança entre imagens médicas.*

1. Introdução

É inegável a importância das imagens médicas como potenciais fontes de informação, onde diversos profissionais podem adquirir conhecimentos que, de forma geral, contribuam para o avanço das pesquisas em Saúde. Entretanto, considerando que tais fontes exigem, na maioria das vezes, complexos tratamentos para extração e análise dos dados relevantes a uma determinada aplicação, profissionais da área da Saúde ainda não obtêm todos os benefícios derivados desse valioso repositório [Silva, 2009].

Uma área de crescente interesse, envolvendo imagens médicas, é aquela relacionada à recuperação, em um banco de imagens, daquelas semelhantes a um exemplar de consulta. Recuperar uma imagem por similaridade envolve a extração de características representativas dos exemplares, o que não é uma tarefa trivial, já que, normalmente, há um grande número de fatores que devem ser levados em consideração, na tentativa de simular a percepção humana. As técnicas relacionadas a essa área, são

conhecidas como CBIR (*Content-based Image Retrieval*) e têm sido alvo de muitas pesquisas. Na literatura disponível, encontramos vários trabalhos que utilizam abordagens baseadas na análise multirresolução de *wavelets* para a produção de sistemas CBIR [Silva, 2009 e Castañon, 2003].

A produção de sistemas utilizando-se técnicas de CBIR inclui, de forma geral, o estabelecimento de vetores de características, que são descritores numéricos das imagens, incorporando, portanto, seus detalhes relevantes. Tais vetores são armazenados no banco de dados, de forma a facilitar a recuperação dos itens correlatos a um exemplar de consulta. Neste trabalho, propomos o estudo de um novo modelo para representação de sinais, como extrator de características, para o estabelecimento de vetores de semelhança entre imagens médicas, com vistas à sua utilização em sistemas CBIR.

2. Modelo proposto

Recentemente, no trabalho em [Fernandes, 2006], foi proposto o estudo de um novo modelo para a representação de sinais: partindo da Transformada de Fourier, desenvolveu-se uma representação conjunta espaço-frequência, baseada em funções de Gabor. Em [Torreão et al., 2009] demonstrou-se que tal modelo apresenta vantagens na representação de sinais a partir de poucas componentes de frequência, e possui características que parecem torná-lo adequado à modelagem de processos neurofisiológicos observados no córtex visual dos mamíferos. A seguir, uma breve descrição sobre a fundamentação teórica do processo.

Seja $I(x,y)$ um sinal bidimensional de quadrado integrável sobre um domínio ilimitado, de tal modo que a sua transformada de Fourier exista. A seguinte identidade se verifica:

$$I(x,y) = \iint_{-\infty}^{\infty} d\omega_x d\omega_y \left\{ \exp[i\omega_x(x-a) + i\omega_y(y-b)] \exp[-\frac{(x-a)^2 + (y-b)^2}{2\sigma^2}] * \right. \\ \left. * \exp[i(\omega_x x + \omega_y y)] \right\} \quad (1)$$

onde a, b e σ são funções das frequências ω_x e ω_y, dadas por

$$\sigma(\omega_x, \omega_y) = \frac{1}{(2\pi)^{3/2}} \sqrt{\left| \tilde{I}(\omega_x, \omega_y) \right|} \quad (2)$$

e

$$\omega_x a(\omega_x, \omega_y) + \omega_y b(\omega_x, \omega_y) = -\varphi_{\tilde{I}}(\omega_x, \omega_y) \quad (3)$$

com $\tilde{I}(\omega_x, \omega_y)$ denotando a transformada de Fourier de $I(x,y)$ – cuja amplitude e fase são indicadas por $|\tilde{I}|$ e $\varphi_{\tilde{I}}$, respectivamente – e com o asterisco na Eq. (1) denotando a operação de convolução de uma função de Gabor (esquerda) por uma função exponencial complexa (direita).

Utilizando as definições acima, a identidade em (1) pode ser comprovada tomando-se a Transformada de Fourier de ambos os lados da equação, ou então, efetuando-se a convolução indicada sob a dupla integral. Em quaisquer dos casos, o resultado obtido para o lado direito de (1) será exatamente a definição de $I(x,y)$, pela Transformada de Fourier Inversa. A representação em (1), portanto, apenas reescreve a

fórmula tradicional da Transformada Inversa de Fourier, expressando o sinal $I(x,y)$ ainda em termos de uma expansão em funções de base exponenciais complexas. Obteve-se, portanto, a partir da transformada de Fourier tradicional, uma representação conjunta espaço-frequência, sob a forma de funções de Gabor.

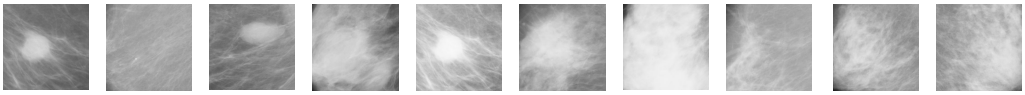
Como descrito em [Torreão et al., 2009], essa representação permite uma caracterização completa do sinal. Neste trabalho, propomos utilizar as próprias Funções de Gabor de Representação para fornecer subsídios para caracterização de uma imagem e elaboração de vetores de semelhança.

3. Metodologia

Para composição dos vetores de características, dividimos a imagem de teste em janelas de tamanho 5x5 *pixels* e calculamos o valor da função de representação:

$$g_a(x, y) = \exp[i\omega_x(x-a) + i\omega_y(y-b)] \exp[-\frac{(x-a)^2 + (y-b)^2}{2\sigma^2}] \quad (4)$$

para cada *pixel* (x,y) da janela. Os elementos da função em (4) são exatamente os mesmos descritos em (1), (2) e (3). Os valores deduzidos, em módulo, são consolidados em um somatório por janela e por frequência. O vetor final, portanto, é a média entre os valores das janelas, ao longo da imagem, por par de frequência (ω_x,ω_y) considerado. Cada vetor de característica deduzido possui 16 valores, correspondentes às 16 frequências mais baixas. Para cálculo do índice de semelhança (D) entre imagens, estamos utilizando, entre os vetores, a distância de Minkowsky [Castañon, 2003], com r=3.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figura 1: (a) Exemplar de consulta, (b) a (j) Imagens utilizadas para pesquisa por similaridade, em relação à imagem em (a).

Tabela 1: Índices de similaridade *D,* calculados para as imagens da Figura 1.

Imagem de Pesquisa	*D*	Imagem de Pesquisa	*D*	Imagem de Pesquisa	*D*
(b)	4.995277	(e)	0.365419	(h)	20.685948
(c)	2.079339	(f)	3.271543	(i)	3.205847
(d)	18.664704	(g)	8.205047	(j)	3.358907

4. Experimentos

Nos testes que realizamos, o limiar para detecção de uma imagem similar a um exemplar de consulta foi avaliado, empiricamente, em $D = 2.5$. Assim, ao compararmos duas imagens, elas foram consideradas similares quando o cálculo da distância D, entre os respectivos vetores de características, resultou um valor abaixo desse limite.

A título de exemplo, vamos mostrar o comportamento do algoritmo para um pequeno subconjunto das imagens de teste. As imagens na Figura 1 mostram recortes de tecido mamário, obtidos através de imagens oriundas de exames mamográficos, disponibilizadas em [Suckling, 2010]. Consideramos a imagem em (a) como um exemplar de consulta; conforme disposto em [Suckling, 2010], ela apresenta uma região com massa circunscrita e bem definida. A Tabela 1 mostra valores para o índice *D*, comparando-se a imagem em (a) com aquelas de (b) a (j). Como podemos perceber através das linhas sombreadas na tabela, as imagens em (c) e (e) foram aquelas detectadas como similares ao exemplar em (a) e tal semelhança pode ser comprovada por simples inspeção visual. Deve-se destacar que a imagem em (e) é o mesmo exemplar em (a) utilizado para a consulta, mas em condições bem diferentes de brilho e contraste – mesmo assim, o algoritmo foi robusto o suficiente para detectar a semelhança.

5. Conclusão

Os resultados iniciais obtidos demonstram que o nosso processo possui excelentes condições de aplicação. Ainda há muitos testes a serem realizados, estudando-se o comportamento dos parâmetros envolvidos no processo, bem como experimentando outras metodologias de aplicação. Já está em curso uma mudança na estrutura do algoritmo, na qual consideraremos uma "pirâmide" de imagens de resoluções crescentes e a consequente obtenção de um conjunto de vetores de características, um para cada nível de imagem – esperamos aumentar a precisão do método a partir dessa mudança.

Na próxima fase do trabalho, testaremos nossa aplicação em um conjunto diferente de imagens (e não somente provenientes de exames mamográficos), propiciando gráficos de precisão e revocação [Castañon, 2003]; além disso, vamos propor o estudo da Transformada Sintonizada de Gabor [Torreão e Victer, 2010], como extratora de características das imagens. Da mesma forma, esperamos obter bons resultados com essa abordagem inovadora.

Referências

Castañon, C. A. B. (2003) "Recuperação de imagens por conteúdo através de análise multirresolução por Wavelets", Dissertação de Mestrado, ICMC-USP, São Carlos.

Fernandes, J. L. (2006) "Shape from shading: Novas abordagens a partir de movimento e de foco", Tese de Doutorado, Universidade Federal Fluminense, Niterói.

Silva, L. A. (2009) "Categorização de Imagens Médicas baseada em Transformada Wavelet e Mapas Auto-Organizáveis", Tese de Doutorado, USP, São Paulo.

Suckling, J. (Organiser) (2010) *site* "The mini-MIAS database of mammograms" [http://peipa.essex.ac.uk/info/ mias.html], Último acesso: Agosto/2010.

Torreão, J. R .A. e Victer, S. M. C. (2010) "A Transformada de Gabor Sintonizada", Relatório técnico apresentado ao Instituto de Matemática Pura e Aplicada (IMPA), Rio de Janeiro.

Torreão, J. R. A., Fernandes, J. L. and Victer, S. M. C. C. (2009) "A Model for Neuronal Signal Representation by Stimulus-Dependent Receptive Fields" In: 19th International Conference on Artificial Neural Networks (ICANN 2009), Limassol.

Identificação biométrica de recém-nascidos através da superfície palmar

Rubisley P. Lemes[1], **Olga R.P. Bellon**[1], **Luciano Silva**[1], **Monica N.L. Cat**[1]

[1]IMAGO Research Group – Universidade Federal do Paraná (UFPR)

{rubisley,olga,luciano,monica_lima}@ufpr.br

Abstract. *We present results on newborn identification through automatic matching of high-resolution images of palmar surfaces. The use of friction ridge patterns for newborns is challenging due to both the small size of newborn's papillary ridges and their fragility. In this context we present our main contributions: a protocol for image acquisition, and algorithms for image processing and recognition. Experiments were performed on images collected from 250 newborns at a university hospital. For good quality palmar images, the results show that the proposed approach is promising.*

Resumo. *Apresentamos resultados em identificação automática de recém-nascidos através de imagens de alta-resolução da superfície palmar. Devido as cristas papilares em recém-nascidos serem mais finas, o problema da identificação é um grande desafio, agravado pela fragilidade dessas cristas. Neste contexto apresentamos nossas principais contribuições envolvendo: um protocolo de coleta de imagens, algoritmos de processamento e de identificação de imagens. Experimentos foram realizados em imagens coletadas de 250 bebês da maternidade de nosso Hospital das Clínicas (HC). Analisando imagens de boa qualidade, os resultados mostram que o método proposto é promissor.*

1. Introdução

Identificação de recém-nascidos é crucial para evitar troca em hospitais, tráfico e adoções ilegais. Dados estatísticos mostram que dentre os 3 milhões de nascimentos ocorridos anualmente no Brasil, existe uma troca de bebê a cada 6000 nascimentos[1]. A identificação é um problema desafiador devido tanto à reduzida espessura das cristas palmares quanto à sua fragilidade, tornando-as suscetíveis à deformação. Práticas manuais em todo o mundo utilizam pulseiras e/ou impressões plantares (coletadas através de tinta e papel) que não apresentam uma solução efetiva para o problema [Weingaertner et al. 2008].

Apesar do grande avanço das tecnologias biométricas [Bolle et al. 2003], até o momento nenhum sistema foi desenvolvido para identificação neonatal, pois as biometrias usuais são de difícil aplicação em bebês. Outra limitação é que tais características (e.g., face) mudam durante o desenvolvimento infantil dificultando uma identificação futura. Portanto, este trabalho apresenta uma nova abordagem para o problema desafiador da identificação de recém-nascidos. Nosso trabalho se baseia em imagens de alta-resolução da superfície palmar e tem como principais contribuições o desenvolvimento de técnicas apropriadas para aquisição, processamento e reconhecimento destas imagens.

[1]Ministério da Saúde do Brasil - http://portal.saude.gov.br/portal/

2. Aquisição das Imagens

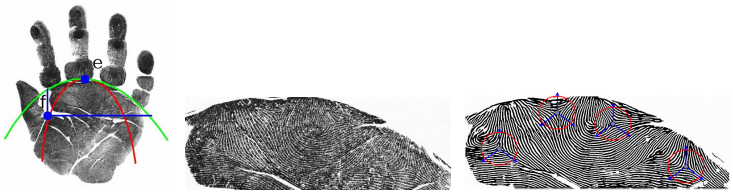
Utilizamos um sensor comercialmente disponível (*CrossMatch* LSCAN 1000P - 1000 dpi). Coletamos 1.221 impressões palmares e 1.221 impressões plantares de 250 bebês da maternidade do HC-UFPR. As imagens foram coletadas entre 1 e 48 horas após o nascimento e o número de coletas por bebê varia entre 3 e 15. Para automaticamente aferir a qualidade das imagens, utilizamos o método proposto por [Wu et al. 2006] para classificá-las em cinco níveis de qualidade: (1) boa, (2) normal, (3) seca, (4) úmida e (5) borrada. Somente 5% das impressões palmares e somente 1,5% das impressões plantares possuem qualidade suficiente (nível 1) para executar o reconhecimento automático.

Considerando um sistema capaz de assegurar a identificação dos bebês (imagens nível 1), apresentamos uma de nossas principais contribuições: desenvolvimento de um protocolo eficaz de coleta de imagens com boa qualidade. Determinamos que a melhor abordagem envolve: (I) usar álcool para remover a substância oleosa da palma da mão; (II) segundo estágio de limpeza utilizando água e sabão; (III) secar utilizando papel toalha; (IV) rápido tempo de exposição e aplicação de pressão média entre a mão e o sensor.

3. Processamento das Imagens

O processamento das impressões palmares possui os seguintes estágios: (1) correção da orientação da imagem, (2) remoção dos dedos, (3) segmentação da Região de Interesse (RI), (4) Short Time Fourier Transform (STFT) [Chikkerur et al. 2007], (5) estimação do Campo Direcional (CD), (6) estimação da frequência, (7) detecção de cristas recuperáveis, (8) Filtro de Gabor (FG) e (9) detecção dos trirrádios.

No estágio 1, estimamos um ângulo de rotação para retificar a orientação da imagem, similar a [Silva et al. 2006]. No estágio 2, removemos os dedos como feito por [Silva et al. 2006], utilizando uma parábola que menos intercepta a palma da mão (Fig. 1a - parábola verde). Em seguida, toda a imagem abaixo e a esquerda do ponto f é eliminada, extraindo a RI abaixo dos dedos (Fig. 1a,b).



(a) (b) (c)

Figure 1. Processamento: (a) remoção dos dedos, (b) RI e (c) FG e trirrádios.

Uma operação de reconstrução da imagem baseada em STFT [Chikkerur et al. 2007] é executada no estágio 4. Os estágios 5, 6, 7 e 8 são baseados em [Hong et al. 1998]. Nos estágios 5 e 6, a imagem é dividida em um conjunto de blocos não sobrepostos de tamanho 16×16 pixels [Hong et al. 1998] para se estimar localmente a orientação e frequência de cada bloco. No estágio 7, as regiões da imagem são classificada como: recuperáveis e não recuperáveis para evitar a extração de falsas informações em áreas muito ruidosas/borradas. O FG no estágio 8 utiliza as informações mais robustas de orientação e frequência obtidas no STFT em conjunto com aquelas obtidas nos estágios 5 e 6 para melhorar a visibilidade da estrutura de cristas na imagem

reconstruída (Fig. 1c). Finalmente, no estágio 9, o algoritmo conhecido como Índice de Poincare [Bolle et al. 2003] é utilizado para extrair os trirrádios (Fig. 1c).

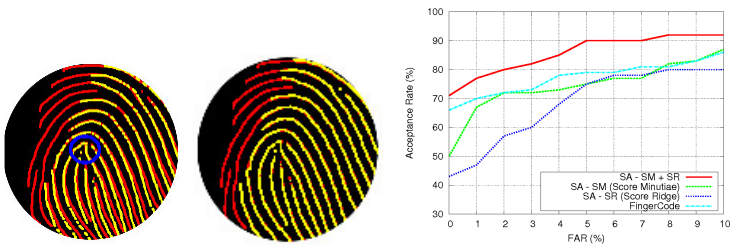
4. Algoritmos de Identificação

Propomos uma abordagem híbrida baseada em registro de imagem utilizando Simulated Annealing (SA) [Kirkpatrick et al. 1983] e campos de textura orientada (FingerCode - FC) [Jain et al. 2000]. Para criar o molde de autenticação, um algoritmo de esqueletização é executado na imagem reconstruída para extrair as cristas e minúcias [Hong et al. 1998]. Os moldes incluem $(n + C_{(n,2)})$ sub-moldes (pontos de referência), onde n é o número de trirrádios detectados e $C_{(n,2)}$ é o número de possíveis combinações de n trirrádios tomados 2 a 2 (Regiões Médias - RM, média das coordenadas entre dois trirrádios).

Cada sub-molde inclui todas as informações de orientação, coordenadas e distância radial dos pontos de cristas, minúcias e ponto de referência dentro de uma região com raio de 200 pixels em torno do ponto de referência. Complementando, 8 componentes orientados (0^o, 22.5^o, 45^o, 67.5^o, 90^o, 112.5^o, 135^o e 157.5^o) são extraídos utilizando-se um banco direcional de Filtros Gabor na região setorizada em torno do ponto de referência, gerando um vetor de características (VC) com 640 características [Jain et al. 2000].

Tanto na abordagem FC quanto SA, a palma da mão é dividida em quatro regiões de igual tamanho. Teoricamente, existe um trirrádio abaixo de cada dedo [Weingaertner et al. 2008]. O cálculo de similaridade é executado somente entre sub-moldes do mesmo tipo (trirrádios ou RM) e pertencentes a mesma região, resultando em $(n + C_{(n,2)})$ possíveis valores de teste, sendo o menor deles selecionado no caso do FC e o maior no caso do SA.

Partindo de um pré-alinhamento obtido através do registro dos pontos de referência (Fig. 2a), o SA analisa soluções vizinhas buscando o melhor alinhamento, que é computado como sendo a taxa de pontos e minúcias sobrepostas com as seguintes propriedades: devem estar dentro de uma região circular com raio $r_P = 1$ para pontos e $r_M = 10$ para minúcias; devem possuir direções equivalentes ($\theta_1 - \theta_2 < \theta = 10^o$); devem possuir distâncias radiais equivalentes ($rad_1 - rad_2 < rad_dist = 10$). Estes limiares foram selecionados empiricamente. Fig. 2 exemplifica a abordagem proposta.



(a) (b) (c)

Figure 2. SA: (a) pré-alinhamento, (b) alinhamento final e (c) resultados.

Após o registro, a similaridade entre dois sub-moldes é definida através da combinação do número de pontos sobrepostos S_R e o número de minúcias sobrepostas S_M. Estes escores (S_R e S_M) são combinados em um escore final $S_{R+M} = \alpha_1 S_M + \alpha_2 S_R$, onde α_1 e $\alpha_2 \in [0,0; 1,0]$. Fig. 2e mostra os resultados do SA combinando-se S_M e S_R. Maiores detalhes técnicos foram omitidos devido a limitação do número de páginas.

5. Resultados Experimentais

Devido a problemas de ilegibilidade das imagens, usamos apenas impressões palmares de 20 bebês. Estas impressões totalizam 60 imagens com informações de cristas/minúcias suficientes para executar o reconhecimento e foram classificadas como boas pelo algoritmo de classificação. A Tab. 1 mostra os resultados de verificação (FAR) e identificação (Rank) [Bolle et al. 2003] combinando-se trirrádios (DP - delta points) e RM.

Table 1. Resultados do reconhecimento em impressões palmares.

	FC (DP + RM)	FC DP	FC RM	SA (DP + RM)	SA DP	SA RM
0% FAR	66%	55%	49%	71%	58%	62%
Rank1	84%	72%	67%	95%	93%	85%
Rank5	94%	83%	86%	98%	98%	97%

O SA apresentou os melhores resultados em ambos os experimentos, mas é computacionalmente mais caro do que o FC. Assim, o FC procura pelas cinco amostras mais parecidas (Rank5) com a amostra de entrada (filtragem). Então, o SA atesta se uma das amostras refere-se ao par da amostra de entrada. Considerando Rank5 para o FC e 0% FAR para o SA, obtivemos (81%) de taxa correta de reconhecimento combinado.

6. Conclusão

Resultados preliminares em impressões datiloscópicas com alta-resolução de 20 recém-nascidos mostram que o processo de identificação automática é viável, embora difícil, pois dois requisitos devem ser seguidos: (1) as imagens devem possuir resolução apropriada (1500 dpi ou superior), e (2) um protocolo de coleta de imagens apropriado deve ser seguido. Entretanto, não há equipamento comercialmente disponível capaz de atender ao primeiro requisito. Como trabalho futuro, começaremos uma nova fase de coleta de imagens de mais de 100 bebês em uma das maiores maternidades de nossa cidade. O processo de coleta fará uso do melhor protocolo de aquisição desenvolvido nesta pesquisa.

References

Bolle, R. M., Connell, J. H., Pankanti, S., Ratha, N. K., and Senior, A. W. (2003). *Guide to Biometrics*. Springer.

Chikkerur, S., Cartwright, A. N., and V., G. (2007). Fingerprint enhancement using STFT analysis. *Pattern Recognition*, 40(1):198–211.

Hong, L., Wan, Y., and Jain, A. K. (1998). Fingerprint image enhancement: Algorithm and performance evaluation. *IEEE TPAMI*, 20(8):777–789.

Jain, A. K., Prabhakar, S., Hong, L., and Pankanti, S. (2000). Filterbank-based fingerprint matching. *IEEE TIP*, 9(5):846–859.

Kirkpatrick, S., Gelatt, C., and Vecchi, M. (1983). Optimization by simulated annealing. *Science*, 220:671–680.

Silva, L., Bellon, O. R. P., Lemes, R. P., and et al. (2006). An image processing tool to support gestational age determination. In *Proc. CBMS*, pages 867–874.

Weingaertner, D., Bellon, O., Silva, L., and Cat, M. (2008). Newborn's biometric identification: can it be done? In *Proc. VISAPP*, pages 200–205.

Wu, C., Tulyakov, S., and Govindaraju, V. (2006). Image quality measures for fingerprint image enhancement. *Lecture Notes in Computer Science*, 4105:215–222.

Segmentation and Classification of Breast Cancer Using Independent Component Analysis, Texture Features and Neural Networks

Lucio Flávio A. Campos[1,2], Emanuel C. M. Lemos[1], Luis C. O. Silva[2], Daniel D. Costa[2], Allan Kardec Barros[2]

[1]Instituto de Engenharia e Ciências da Computação – Universidade Estadual do Maranhão (UEMA)
Campus Paulo VI – São Luís – MA – Brazil

[2]Laboratório de Processamento da Informação Biológica– Universidade Federal do Maranhão (UFMA)

lucioflavio@engcomp.uema.br,emanuel.cml@gmail.com,
luisoliveirasilva@hotmail.com, danieldc@gmail.com, akbarros@ieee.org

Abstract. *We propose a method for segmentation and classification of breast cancer in digital mammograms using Independent Component Analysis (ICA), Texture Features and Multilayer Perceptron (MLP) Neural Networks. The method was tested for a mammogram set from MIAS database, resulting in 90.15% success rate, with 92% of specificity and 88.3% of sensitivity.*

1. Introduction

Breast cancer is the major cause of death by cancer in the female population [INCa, 2010]. It is know that the best prevention method is early diagnosis, which lessens the mortality and enhances the treatment. Therefore, a great effort has been made to improve the early diagnosis techniques. Among them, the most used is the mammogram, for it is low cost and easy access. However, mammogram has a high error value for medical diagnosis, ranging from 10 to 25%, resulting in a great number of false-positives diagnostics, which causes unneeded biopsies, or false-negatives, which delays the cancer diagnosis. The analysis of digital mammography is a complex cognitive task that includes various aspects of medical expertise and conclusive clinical findings [Newsread, Baute, 1992], [Meyer et al, 1989]. The visual task of clinical evaluation and diagnosis, based on digital mammography, consists of a number of different factors in multiple scales and levels of decomposition [Sickles, 1989], [Bocchi et al, 1997]. The fine-scale organization of the informational content on the mammogram is a key factor in the detection of breast cancer, as it represents the nature, structure and the quality of biological tissues, as they are projected on the mammogram [Homer, 1987], [Homer,1985]. Similar textural features are also present in rare clinical cases, where direct inference on probably benignancy or malignancy is much more complex.

These fine-scale structural details are realized as visual patterns in the image and they are often referred to as ‘‘texture’’ of the corresponding region. When the digital mammogram is obtained with adequate quality and resolution, these textural patterns can be identified, analyzed and classified with an aided of image processing and

computational vision algorithms, combined with artificial intelligence for features extraction.Those algorithms are able to decrease the error and make the mammograms more reliable [Bick, 1996].

The CAD (computer-aided diagnosis) systems can aid radiologists by providing a second opinion and may be used in the first stage of examination. For this to occur, it is important to develop many techniques to detect and recognize suspicious lesions and also to analyze and discriminate them.

Segmentation is an important task to identify regions suspicious of cancer in digital mammograms. After identified, each region must be classified as benign, malignant cancer or normal tissue.
The principal objective of image classification is to assign all pixels in the image to particular classes or themes, based on some features, for example. This type of classification is called pattern recognition.

Some methods of segmentation and classification of breast cancer in digital mammography have been reported.

[Campos et al, 2007] used independent component analysis (ICA) and neural network multilayer perceptron to classify mamogramas in 3 class: normal, benign and malignant, with 98,7% of successful. [Braz et al, 2007] classified the regions of interest of screening mammogram using spatial statistics, with performance of 98.24% to discriminate Mass from Non-Mass elements. [Dominguez, Nandi, 2007] use enhanced multilevel thresholding segmentation and region selection based on rank mammogram segmentation. According the authors, this method had a better performance, with 80% of sensitivity. [Campos et al, 2008] use independent component analysis, feature extraction and K - means cluster to segment digital mammography. The proposed method obtained 86.6% of success.

In this paper we propose to use the independent component analysis (ICA) to generate a data dependent filter bank for mammograms segmentation, and after generate a basis functions through the coefficients (features) extracted using ICA. Then, those features are used as input parameters to a Neural Network do the classification.

We divide this work as follows. Into section 2 we show the techniques for segmentation and classification the mammograms. In section 3 we present the results and discuss about the application of the techniques under study. Finally, section 4 presents some concluding remarks.

2. Methods

The block diagram of the proposed method is shown in Figure 1.

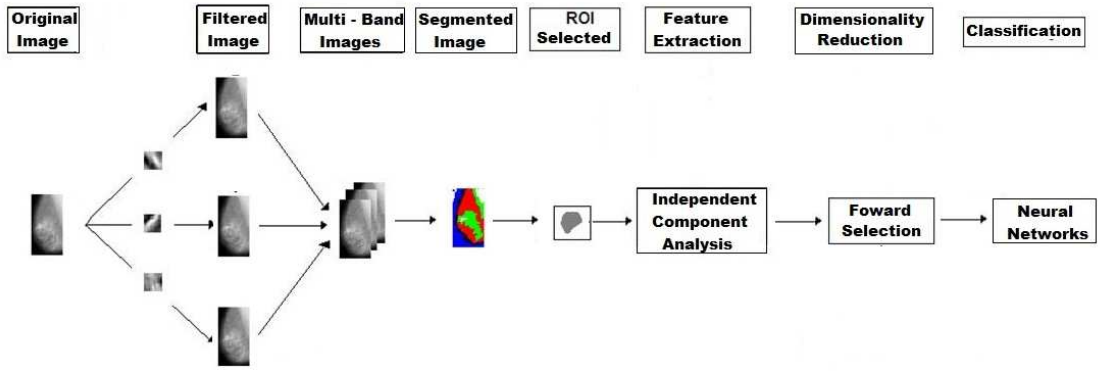


Figure 1: Block Diagram

The method was divided in two steps. Step one basically consists on filtering an original image, using ICA filter bank, feature extraction using non–linear operators, and segmentation using K-means clustering. The step two consists of the extraction of features in the ROI using ICA, reduction of insignificants features using the *forward-selection* technica and the classification of the tissue through neural networks.

2.1. Independent Component Analysis

Let us assume that an image x may be expressed as a linear combination of basis images $a_1, a_2, \cdots, a_n$ pondered by mutually statistically independent coefficients $s_1, s_2, \cdots, s_n$ [Hyvärinen, 1997], such that

$$x = a_1 s_1 + a_2 s_2 + \cdots + a_n s_n \tag{1}$$

In equation 1, only the variable x is known, and from that we estimate the coefficients a_i and the independent components s_i.

In the sequel, we represent an image as a column vector

$$x = \left[x_1, x_2, \cdots x_m \right]^T \tag{2}$$

by re-shaping the image matrix row-by-row into a single column. In this way we avoid using two-dimensional matrices in the ICA filtering, and an image may thus be represented using the standard ICA methodology. Mathematically, we express the model as,

$$x = \sum_{i=1}^{N} a_i s_i = As \tag{3}$$

Where x is the image data, the basis functions a_i, $i = 1, 2, \cdots, N$, are the columns of the $(M \times N)$ matrix A, and $s = \left[s_1, s_2, \cdots s_N \right]^T$ the independent components.

The image model described here is illustrated in Fig. 2.



Fig. 2: The ICA model.

2.2. FastICA Algorithm

The data matrix $\boldsymbol{X}$ is considered to be a linear combination of non-Gaussian (independent) components i.e., $\boldsymbol{X} = \boldsymbol{A}.\ \boldsymbol{S}$ where columns of $\boldsymbol{S}$ contain the independent components and $\boldsymbol{A}$ is a linear mixing matrix. In short ICA attempts to "un-mix" the data by estimating an un-mixing matrix $\boldsymbol{W}$, where $\boldsymbol{X}.\boldsymbol{W} = \boldsymbol{S}$.

Under this generative model the ICA, the measured in $\boldsymbol{X}$ will tend to be more Gaussian than the source components $\boldsymbol{S}$. Thus, in order to extract the independent components we search for an un-mixing matrix $\boldsymbol{W}$ that maximizes the non-gaussianity of the sources. In FastICA, non-gaussianity is measured using approximations to negentropy (J) which are more robust than kurtosis based measures and fast to compute [Marchini et al, 2004]. The approximation takes the form

$$J_{G(y)} = \left| E_y\{G(y)\} - E_v\{G(v)\} \right|^p \tag{4}$$

Where v is a standartized Gaussian randon variable, y is assumed to be normalized to unit variance, and the exponent p=1,2 typically (The notation J_g do not be confused with notation on for negentropy , J).

2.3. ICA Filter Bank

For generate the ICA filter bank, we use the fastICA algorithm applied to Regions on Interests (ROIs), containing textures of normal and abnormal tissues. We reduced the dimension of the ROIs at the Principal Components Analysis (PCA) step.

The generation of training data includes the following computations:

1. Select the ROI of an ensemble;
2. Select how many patches to take per image;
3. Select the given number of (m×m) sized patches from random locations;
4. Subtract from each patch its mean value;
5. Represent samples as columns, and store in a matrix consisting of training vectors.

The computations described above result in an ensemble of training vectors which are presented to the FastICA algorithm [Marchini, 2004], after additional preprocessing by PCA. After convergence of the algorithm, matrix W has been learned from the available training data. Matrix A is obtained by finding the inverse (or pseudo inverse) of W. As a last step, the columns A, are reshaped into size $(m \times m)$ to constitute the basis functions. These are now the impulse responses of our filter bank.

Each filter responds to specific frequency and orientation characteristics and specific texture properties. Thus, each filtered image will have high frequency in regions corresponding to textures which are tuned to the filter, and low energy corresponding to textures which are not tuned to that filter. If the banks consist of N filters, the result is N filtered images of the same size as the input image.

2.4. Features Extraction

Only filter outputs by default are not appropriate for identifying key texture features. The objective of the feature extraction using non-linear operators is to estimate the energy in a local region of the filter outputs [Jain, Farrokhnia, 2001], [Unser, Eden, 1990].

A number of feature extraction methods were proposed to extract useful information from the filter outputs, however, the most commonly used non-linearities are:

- Magnitude function; $F(y)=|Y|$
- Non-linear sigmoidal function; $F(y)=\tanh(Y)$
- Square function. $F(Y)=(Y)^2$

Where Y is a filtered image by ICA.

The feature image obtained F(Y) has the variance disparities converting of into mean values differences, an essential step if standard clustering are to be employed.

2.5. Clustering in the Feature Space

At the end of the feature extraction step we are left with a set of feature images extracted from the filter outputs. Pixels that belong to the same texture region have the same texture characteristics, and should be close to each other in the feature space. The final step in unsupervised texture segmentation is to cluster the pixels into a number of clusters representing the original texture regions. Labeling each cluster yields the segmented image.

Different approaches were taken for the clustering process [Jain, Farrokhnia, 2001]. In this paper we use the basic K-means clustering algorithm for simplicity.

K-means [Moore, 2001] is one of the most used unsupervised learning algorithms that solve the clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of **k** clusters fixed a priori.

The main idea is to define **k** centroids, one for each cluster. These centroids shoud be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other.

The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate **k** new centroids as barycenters of the clusters resulting from the previous step.

After we have these **k** new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the **k** centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function is:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 \tag{5}$$

where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre c_j, is an indicator of the distance of the *n* data points from their respective cluster

centers. When clustering is done, each pixel is labeled with its respective cluster, finally producing the segmented image.

2.6. Multilayer Perceptron Neural Networks

The Multilayer Perceptron (MLP), a feed-forward back-propagation network, is the most frequently used neural network technique in pattern recognition [Duda, Hart, 1973], [Bishop, 1999]. Briefly, MLPs are supervised learning classifiers that consist of an input layer, an output layer, and one or more hidden layers that extract useful information during learning and assign modifiable weighting coefficients to components of the input layers. In the first (forward) pass, weights assigned to the input units and the nodes in the hidden layers and between the nodes in the hidden layer and the output, determine the output. The output is compared with the target output. An error signal is then back propagated and the connection weights are adjusted correspondingly. During training, MLPs construct a multidimensional space, defined by the activation of the hidden nodes, so that the three classes (malignant, benign and normal tissue) are as separable as possible. The separating surface adapts to the data.

2.7. Selection of Most Significant Features

Our main objective to identify the effectiveness of a feature or a combination of features when applied to a neural network. Thus, the choice of features to be extracted is important.

Forward selection is a method to find the “best” combination of features (variables) by starting with a single feature, and increasing the number of used features, step by step [Funukaga, 1990]. In this approach, one adds features to the model one at a time. At each step, each feature that is not already in the model is tested for inclusion in the model. The most significant of these feature is added to the model, so long as P-value is below some pre-selected level.

2.8. Evaluation of Classification Methods

Sensitivity and specificity are the most widely used statistics to describe a diagnostic test. Sensitivity is the proportion of true positives that are correctly identified by the test and is defined by $S = TP/(TP+FN)$. Specificity is the proportion of true negatives that are correctly identified by the test and is defined by $TN/(TN+FP)$. Where ***FN*** is false-negative, ***FP*** is false-positive, ***TN*** is true negative and ***TP*** is true positive diagnosis.

3. Experimental Results and Discussion

Here are describe the results obtained using the method proposed in the previous section.

3.1. Mammogram Database

The database used in this work is the *Mammographic Institute Society Analysis* (MIAS) [Sucking et al, 1994]. The mammograms have a size of 1024x1024 pixels, and resolution of 200 micron. This database is composed of 332 mammograms of right and left breast, from 161 patients, where 53 were diagnosed as being malignant, 69 benign and 206 normal. The abnormalities are classified by the kind of found abnormality

(calcification, circumscribed masses, architectural distortions asymmetries, and other ill-defined masses).

For the segmentation**,** to generate the ICA filter bank, we selected 150 ROI from this database: 25 benign, 25 malignant and 100 normal tissues of the mammograms. The ROIs was found through of ***xy*** images coordinates of centre of abnormality contained in file list of MIAS database. To the normal tissue, only pectoral muscle wasn't considered as a possible ROI, but tissue and fat tissue were.

3.2. ICA Application, in Segmentation of mammograms

For generate the matrix *x* of equation 3, were randomly selected patches of size 12 x 12 of the 95 ROIs, and each patches generate de vector of size 1 x 144. It was generate 30000 training vectors that were used as input to fastICA algorithm.

In order to reduce the dimensionality of the filters, we made a preprocessing using principal component analysis. ICA was then performed successively on the 8 most significant filters.

Each row of matrix A, of size 1 x 144, was reshaped in windows, of size 12 x 12. Each window was used as a filter, localized in spatial frequency, and orientation.

3.3. Extracting Features

For feature extraction of filtered images by ICA, we used the magnitude function, combined with a square function:

$$F(Y) = (|\, Y \,|)^2 \tag{6}$$

In order to carry out the tests, we selected 60 mammograms: 35 malignant and 25 benign diagnoses. These mammograms that do not used to generate the ICA filter bank.

If the 60 mammograms, the algorithm detected 186 regions suspicious. Based in MIAS Database, can affirm that 186 regions founded, 60 regions have cancer, and 126 haven't. Figure 3 shows a comparison between the original mammograms and the segmented mammograms using K-means clustering, with 8 filters.

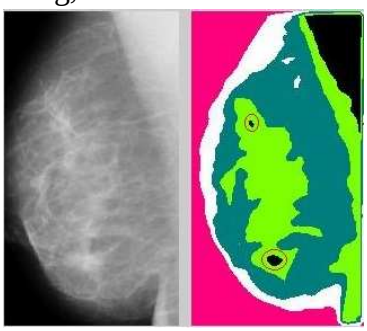


Fig.3: Comparison between: Original mammograms and mammogram segmented. The circumscribed regions indicate suspicious regions.

3.3. Classification of Regions Suspicious

The 186 suspicious regions, the ROI were manually selected, containing a lesion, in the case of abnormal mammograms. The ROI's was found using the informations through of *xy* images coordinates of centre of abnormality, contained in file list of MIAS

database, and the segmentation response. In the case of regions where not founded cancer, was extracted regions based only in segmentation response.

If the tissues had different sizes, it was rescaled each ROI. Therefore, they were resized to 24x24 pixels. Figure 4 exemplifies the ROI selection of a mammogram.

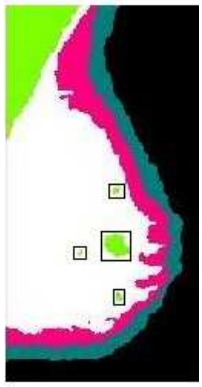


Fig. 4: Chosen tissue of mammogram

3.4. ICA Application for Classification of Mammograms

X of Equation 2 was represented using the chosen ROIs.

The images with ROI were rescaled and transformed into a one-dimensional vector

$$P = P_x \, x \, \boldsymbol{P_y} \quad \textbf{(6)}$$

Where ***Px*** is a rows and **Py** is a columns of ***P*** and ***P*** has dimension 1x 576.

Each sample represents one row of the mixture matrix. The matrix **X** is represented by the samples into the dimension of **P**, that is, 1x576. Thus, each row of the matrix **A** corresponds to a ROI, and each column corresponds to an attributed weight to a base image, i.e., an input parameter to the neural network [Christoyanni, 2002].

Using the FastICA algorithm and the matrix ***X***, we obtain the basis function matrix **A,** which contains the features of each sample.

3.5. Multilayer Perceptron Neural Networks

Using the *forward-selection* algorithm, 21 basis functions were selected as being the most significant features. The chosen features (a_i) are the input to the MLP Network. The best result was obtained using one input layer (21 neurons), one hidden layer (2 neurons) and one output layer (2 Neurons).

In order to realize the tests, we divided a sample into 186 ROI: 93 for training and 93 for tests.

4. Results

Table I shows the performance of application of the ICA technique and MLP network to each regions suspicious.

Table I: Classification of cancer and not cancer in regions suspicious. The lines and columns show the database and algorithm classification, respectively.

	Cancer	Not Cancer
Cancer	**53**	**10**
Not Cancer	**7**	**116**
Success (%)	**88.3**	**92**

Based on Table I, the method obtained a success rate of 90.15%, for discriminating normal (not cancer) and abnormal (cancer) tissues. The specificity was 92 %, and sensitivity was 88.3%. The method obtained 53 true positives, 116 true negatives. 10 false positives and 7 false negatives.

5. Conclusions

The presented results demonstrated that Independent Component Analysis, texture features and MLP Neural Networks were good techniques to Segment and classify Digital Mammograms, discriminating cancer and not cancer tissue.

Furthermore, the best performance was found using 8 ICA filters segmented those mammograms, and after MLP Neural Networks to classification, with a success rate of 90.15%. It can decrease the number of unneeded biopsies and late cancer diagnosis.

Nevertheless, there is the need to perform tests with a larger database and more complex cases in order to obtain a more precise behavior pattern.

References

Braz , G. Jr., E. C. Silva, A. C. Paiva and A. C. Silva, “Breast Tissues Classification Based on the Application of Geostatistical Features and Wavelet Transform”, In: International Special Topic Conference on Information Technology Applications in Biomedicine, ITAB 2007, 6th, 227-230, 2007.

Bick U., M. Giger, R. Schmidt, R. Nishikawa, D. Wolverton and K. Doi. “Computer-aided breast cancer detection in screening mammography”, In: Digital Mammogr'9Chicago, Il (1996), pp. 97–103.

Bishop, C.M.*:* “Neural Networks for Pattern Recognition”. Oxford University Press, New York (1999)

Bocchi L, G. Coppini, R. De Dominicis and G. Valli, *Tissue characterization from X-ray images*, *Med Eng Phys* 1997;19(4); pp. 336–342.

Campos, L. F. A. ; Barros, A. K. ; Silva, A. C. “Independent Component Analysis and Neural Networks Applied for Classification of Malignant, Benign and Normal Tissue in Digital Mammography”, In: Special Issue - Methods of Information in Medicine, v. 46, p. 212-215, 2007.

Campos, L. F. A. ;Costa, D. D.; Barros, A. K. “Segmentation on Breast Cancer Using Texture Features and Independent Component Analysis”, In: Bioinspired Cognitive Systems, BICS 2008.

Christoyianni I., Koutras A., Kokkinakis G., “Computer aided diagnosis of breast cancer in digitized mammograms”, In: Comp. Med. Imag. & Graph., 26:309-319, 2002.

Domínguez, A. Rojas. Nandi,A. K. “Detection of masses in mammograms using enhanced multilevel thresholding segmentation and region selection based on rank”, In: Proceedings of the fifth conference on Proceeding of the Fifth IASTED International Conference: biomedical engineering. 2007

Duda, R.O., Hart, P.E.: “Pattern Classification and Scene Analysis”, In: Wiley-Interscience Publication, New York (1973)

Fukunaga, K.: “Introduction to Statistical Pattern Recognition”. 2nd ed. London: AcademicPress. 1990.

Homer M. J, “Imaging features and management of characteristically benign and probably benign lesions”, In: RadiolClin N Am 1987; 25 (5); pp. 939–951.

Homer M. J, “Breast imaging: Pitfalls, controversies and some practical thoughts”, In: Radiol Clin N Am **23** (1985) (3), pp. 459–472.

Hyvärinen A. and E. Oja. “A fast fixed-point algorithm for independent component analysis”, In: Neural Computation, 9(7):1483-1492, 1997.

INCa, Internet site address: http://www.inca.gov.br accessed in 04/12/2010.

Jain, A.K, F. Farrokhnia, “Unsupervised texture segmentation using Gabor filters”, In: Pattern Recognition 24 (12) (1991) 1167–1186. 2001

Marchini J. L, Heaton C, Ripley B D. “FastICA algorithms to perform ICA and Projection Pursuit”. (2004) Available at http://www.stats.ox.ac.uk/~marchini/software.html

Meyer J. E. , E. Amin, K.K. Lindfors, J.C. Lipman, P.C. Stomper and D. *Genest*, “Medulary carcinoma of the breast: Mammographic and US appearance”, In: Radiology, 1989; 79-82

Moore A.: “*K-means and Hierarchical Clustering -Tutorial Slides”*, 2001. Internet site address: http://www2.cs.cmu.edu/~awm/tutorials/kmeans.html

Newstead G. M, P.B. Baute and H.K. Toth, “Invasive lobular and ductal carcinoma: Mammographic findings and stage at diagnosis”, In: Radiology 1992; 184; pp. 623–627.

Sickles E. A.: “Breast masses: mammographic evaluation”, In: *Radiology* 1989; 173; pp. 297–303.

Suckling, J et al (1994): “The Mammographic Image Analysis Society Digital Mammogram Database”, In: Exerpta Medica. International Congress Series 1069 pp375-378.

Unser M, M. Eden, “Nonlinear operators for improving texture segmentation based on features extracted by spatial filtering”, In: IEEE Trans. Syst. Man Cybern. 20 (1990) 804–815.

Classificação de Arritmias Utilizando Sinais de Variabilidade da Freqüência Cardíaca e o Algoritmo *Prediction by Partial Matching*

Thiago Fernandes Lins de Medeiros[1], Amanda Barreto Cavalcanti[2], Berg Élisson Sampaio Cavalcante[2], Erick Vagner Cabral de Lima Borges[2], Igor Lucena Peixoto Andrezza2, Leonardo Vidal Batista[2,3]

[1] Centro de Informatica, Universidade Federal de Pernambuco (UFPE) – Recife – PE - Brasil

[2] Departamento de Informática, Universidade Federal da Paraíba (UFPB) – João Pessoa – PB – Brasil

[3] Programa de Pós-Graduação em Informática, Universidade Federal da Paraíba (UFPB), Brasil

{bergelisson01,erickvagnerr,igorlpa90,thiagofernandeslm}@gmail.com, {amanda, leonardo}@di.ufpb.br

Abstract. *This paper describes a method of heart arrhythmia classification based on the signal of the heart frequency variability (HFV) and the compression algorithm Prediction by Partial Matching The extraction of the HFV signal is performed by analyzing the electrocardiogram to detect the R peak from the QRS complex of the heartbeats. Then it is possible to generate the signal. The classification of the heart arrhythmias is done in two steps. In the learning stage the PPM algorithm builds statistic models for the extracted tachogram. In the classification stage, the tachograms are compressed by the models obtained and attributed to the class whose models obtain the best ratio of compression.*

Resumo*. Este artigo descreve um método de classificação de arritmias baseado no Sinal de Variabilidade da Freqüência Cardíaca (VFC) e no algoritmo de compressão Prediction by Partial Matching(PPM). A extração do sinal VFC é realizada através da análise do eletrocardiograma para detectar o pico R no complexo QRS dos batimentos cardíacos. Então é possível gerar o sinal. A classificação das arritmias cardíacas é feito em dois passos. Na etapa de treinamento o algoritmo PPM constrói modelos estatísticos para os tacogramas extraídos. Na etapa de classificação, os tacogramas são comprimidos pelos modelos obtidos e atribuídos às classes que obtêm a melhor razão de compreensão.*

1. Introdução

Os métodos de diagnóstico de doenças cardíacas podem ser invasivos ou não-invasivos. O eletrocardiograma (ECG), por ser um método eficiente, não-invasivo e de baixo risco, é um sinal extremamente útil na análise e diagnóstico das doenças cardíacas

[Goldberger 1977]. É também muito utilizado para monitorar pacientes em estado grave ou que necessitam de cuidados especiais. O ECG permite o diagnóstico de arritmias, que são distúrbios do batimento rítmico regular do coração. As arritmias podem ocorrer em corações saudáveis e ter conseqüências mínimas, mas também podem indicar anomalias e levar a graves doenças do coração como derrames e paradas cardíacas [Goldberger 1977]. A detecção automática de arritmias em registros de ECG tem sido considerada uma linha de investigação importante. A análise da variabilidade da freqüência cardíaca (VFC), extraída dos ECGs, é atualmente uma das formas mais precisas para diagnosticar doenças do coração e avaliar a condição de saúde de um individuo [Eur. Heart J 1996]. A VFC refere-se às alterações, batimento a batimento, da freqüência cardíaca, e representa o resultado de influências diversas dos centros simpáticos e parassimpáticos.

A análise da VFC é em geral efetuada com base em medidas estatísticas. Tsipouras et al [Tsipouras e Fotiadis 1997] propuseram um método baseado redes neurais, capaz de classificar um segmento apenas como normal ou arritmia, sem identificar o tipo da arritmia. Acharya et al. [Tsipouras, Fotiadis e Sideris 2005], com um classificador Fuzzy, e Tsipouras et al. [Asl, Setarehdan e Mohebbi 2008], com um método baseado em conhecimento, classificaram arritmias nas cinco seguintes categorias: Fibrilação Atrial(AF), Ritmo Sinusal Normal (NSR), Bloqueio de Segundo Grau do Coração (BII), Bradicardia Sinusal (SBR) e Fibrilação Ventricular (VF). Mohammadzadeh et al. [Honorio, Batista e Duarte 2009] apresentaram um método usando o algoritmo de classificação Support Vector Machine em conjunto com análise discriminante generalizada.

Modelos estatísticos de compressão de dados têm sido utilizados em diversos problemas de reconhecimento de padrões [Carvalho 2008]. Este trabalho apresenta um novo método de classificação de arritmias cardíacas baseado na VFC e no algoritmo de compressão de dados Prediction by Partial Matching.

2. Fudamentação Teórica

O eletrocardiograma (ECG) é um sinal bioelétrico captado na superfície do corpo por meio de eletrodos, resultante da atividade elétrica que se espalha pelo coração durante o ciclo cardíaco [Madeiro 2007]. O ECG é um sinal extremamente útil na análise e diagnóstico das doenças cardíacas e é caracterizado por seis picos e vales identificados. Estas ondas proporcionam informações importantes sobre o funcionamento do coração cuja análise é amplamente utilizada para diagnóstico de muitas doenças cardíacas [Bell, Cleary e Witten 1984].

A variabilidade da freqüência cardíaca (VFC) é caracterizada pelas oscilações dos intervalos entre batimentos cardíacos consecutivos (intervalos R-R em eletrocardiogramas). O sinal composto pela sucessão temporal das durações de intervalos R-R consecutivos é denominado tacograma. O tacograma é extraído a partir de um registro de eletrocardiograma onde todos os intervalos de tempo entre picos R adjacentes são determinados [Eur. Heart J 1996].

Uma fonte de informação discreta e estacionária que gera mensagens de um alfabeto finito escolhe símbolos sucessivos do alfabeto de acordo com uma distribuição de probabilidades que depende dos símbolos selecionados anteriormente. Uma mensagem genérica será modelada como um processo estatístico estacionário pertencente ao alfabeto.

O conceito de entropia como uma medida de informação é crucial para a Teoria da Informação e a compressão de dados provê uma perspectiva concreta a ele. Define-se a taxa de compressão de um sistema de codificação como a média do número de bits por símbolo que o sistema usa para codificar a saída de uma fonte.

O algoritmo PPM [Acharya, Bhat, Iyengar, Rao e Dua 2003] é considerado um dos mais eficazes algoritmos de compressão de dados sem perdas, de propósito genérico, embora seja bastante exigente computacionalmente.

O algoritmo PPM-C é baseado na modelagem e previsão de contexto. O modelo é atualizado à medida que os símbolos vão sendo codificados e, baseando-se em toda a seqüência de símbolos já codificados, as distribuições de probabilidade para o próximo símbolo são calculadas.

Uma abordagem seria primeiramente extrair o sinal da VFC do sinal de ECG, registrando seus sucessivos picos R (intervalos RR), para depois processá-lo. Este é um método mais robusto já que a detecção dos intervalos RR é menos afetada pelo ruído. Com esta abordagem, Tsipouras and Fotiadis [Tsipouras e Fotiadis 1997] propuseram um algoritmo baseado na análise do tempo e da freqüência do sinal da VFC utilizando um conjunto de redes neurais. Esse método é capaz de classificar um segmento como “normal” ou “arritmia”, sem identificar o tipo da arritmia.

Acharya et al. [Salomom 2004], utilizando um classificador Fuzzy, e Tsipouras et al. [Asl, Setarehdan e Mohebbi 2008], utilizando um método baseado em conhecimento, classificaram arritmias nas quatro seguintes categorias: Fibrilação Atrial (AF), Ritmo Sinusal Normal (NSR), Bloqueio de Segundo Grau do Coração (BII) e Bradicardia Sinusal (SBR).

3. Métodos

Para fontes estacionárias, o algoritmo PPM aprende um modelo gradualmente melhor durante a codificação. Vários resultados experimentais mostram a superioridade do desempenho de compressão do PPM comparada a outros compressores, resultado da sua capacidade de construir um modelo preciso para a fonte logo no início do processo de compressão. Em outras palavras, o PPM gera um modelo eficiente da mensagem a ser comprimida mais rapidamente que a maioria dos outros algoritmos modernos de compressão de dados. Por este motivo o PPM será utilizado no método de classificação proposto.

Devido ao bom compromisso entre precisão do modelo e requisitos computacionais, a variante PPM-C foi implementada [Salomom 2004] e adaptado para realizar adequadamente as etapas exigidas para o treinamento e para a classificação dos sinais de ECG.

Os sinais da VFC utilizados foram gerados a partir dos sinais de ECG do MIT-BIH Arrhythmia Database [Mark e Moody 1997]. Foram extraídos segmentos relativos a cinco arritmias: Fibrilação Atrial (AF), Ritmo Sinusal Normal (NSR), Bloqueio de Segundo Grau do Coração (BII), Bradicardia Sinusal (SBR) e Fibrilação Ventricular (VF). Cada segmento extraído teve seu ritmo caracterizado de acordo com um arquivo de anotações de especialistas. Um total de 130 segmentos de ECG, com 32 intervalos RR cada, foi selecionado do banco.

A Figura 1 demonstra as etapas do método proposto.

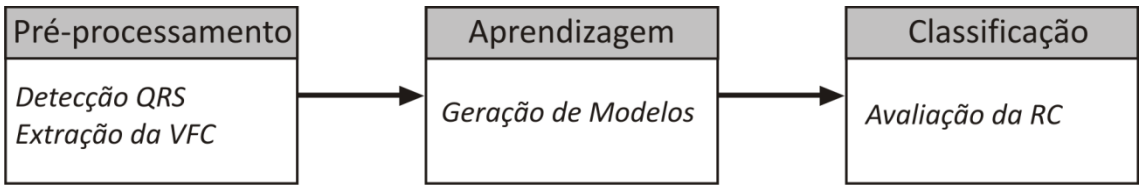


Figura 1: Etapas do processo de classificação do método proposto

Os sinais da VFC dos sinais de eletrocardiograma dos bancos de dados precisam ser extraídos. Para tanto, utilizando o algoritmo de Hamilton e Tompkins [Pan e Tompkins 1985, Hamilton e Tompkins 1986] são detectados os complexos QRS e identificados os picos R de cada complexo e seus respectivos valores de tempo. O sinal da VFC então é construído a partir das diferenças de tempo entre os sucessivos picos R (intervalos RR). Cada tacograma gerado é divido em segmentos contendo 32 intervalos RR e a classe de arritmia é identificada de acordo com os arquivos de anotações dos bancos de dados. Foram selecionadas 26 amostras (segmentos) de tacograma de cada classe.

Na fase de treinamento, o número de classes é definido (N = 5) e cada conjunto de treinamento Ti é formado por 17 das 26 amostras de tacogramas de sua respectiva classe Ci, i = 1, 2, ... N. O algoritmo PPM comprime seqüencialmente os tacogramas em Ti, e a tabela de probabilidades condicionais resultantes Mi é mantida como um modelo para os tacogramas da classe Ci, i = 1, 2,… N.

A razão de compressão (RC) é usada para aferir a qualidade dos modelos construídos durante o treinamento, e, portanto, para classificar amostras de classes desconhecidas. A RC é dada pela equação 1.

$$RC = n / m \quad (1)$$

em que n representa a quantidade de bits do tacograma comprimido, e m a quantidade de bits do tacograma original.

A classificação é realizada da seguinte forma. Um tacograma x de uma classe desconhecida é codificada pelo algoritmo PPM com modelo estático Mi e a razão de compressão correspondente RCi é gravada, i = 1, 2, … N. A amostra x é atribuída à classe Ci que obtiver a menor razão de compressão. A idéia é que, se x é uma amostra da classe Ci, Mi contem o modelo que melhor descreve sua estrutura, produzindo assim, a menor razão de compressão.

A validação cruzada é uma importante técnica para estimar o desempenho de um modelo preditivo, que tenta prever qual a melhor probabilidade de um resultado a partir de um conjunto inicial de amostras. A técnica foi usada para o classificador proposto e consiste em avaliar como os resultados de uma análise estatística irão se comportar para um conjunto de dados independentes.

Uma rodada de validação cruzada consiste em separar amostras de dados em subconjuntos complementares, realizar a análise de um único subconjunto (chamado de conjunto de treinamento) , nesse caso por intermédio do ppm, e finalmente obter o valor resultante dessa rodada.

A fim de reduzir a variância do resultado final, que é a média aritmética dos valores resultantes parciais, várias rodadas de validação cruzada são realizadas usando partições diferentes.

Para os testes de classificação apresentados nesse artigo foram aplicadas duas rodadas de validação cruzada. Previamente, testes foram realizados para determinar a quantidade de elementos por partição que gera melhores resultados e obteve-se um modelo com eficiência comparável a outros que utilizavam mais amostras.

Na primeira rodada foram selecionadas 17 amostras de tacogramas de cada classe para o conjunto de treinamento. Nove amostras de tacogramas não presentes no treinamento foram utilizadas para a etapa de classificação. Na segunda rodada, outras 17 amostras de cada classe foram selecionadas para o treinamento e nove para a classificação. Conforme recomendado na literatura, não há interseção entre os conjuntos de treinamento e classificação. Após a obtenção dos resultados de cada etapa, o resultado final é calculado.

4. Resultados e Discussões

A avaliação do desempenho do classificador proposto foi feita utilizando três medidas típicas para esse propósito: a sensibilidade, a especificidade, a acurácia, a preditividade positiva e a preditividade negativa apresentadas nas equações (2), (3), (4), (5) e (6).

$$\text{Sensibilidade}\,(\%) = \frac{VP}{(VP + FN)} \tag{2}$$

$$\text{Especificidade}\,(\%) = \frac{VN}{(VN + FP)} \tag{3}$$

$$\text{Acurácia}\,(\%) = \frac{(VP + VN)}{(VP + VN + FP + FN)} \tag{4}$$

$$\text{Preditividade Positiva}\,(\%) = \frac{VP}{(VP + FP)} \tag{5}$$

$$\text{Preditividade Negativa}\,(\%) = \frac{VN}{(VN + FN)} \tag{6}$$

em que VP, VN, FP e FN representam o número de verdadeiro-positivos, verdadeiro-negativos, falso-positivos e falso-negativos, respectivamente. Se, por exemplo, um segmento da VFC com a arritmia AF é classificado como um AF tem-se um VP; se um segmento não-AF é classificado como não-AF tem-se um VN; um segmento não-AF classificado como AF irá resultar em um FP; e um segmento AF classificado como não-AF produzirá um resultado FN. Pode-se definir a sensibilidade como a freqüência com a qual o classificador indica a presença de determinada classe quando ela está presente de fato, e a especificidade como a freqüência com a qual o classificador indica a ausência de determinada classe quando ela está ausente. A acurácia é a proporção de predições corretas em relação a todas as predições. Preditividade positiva é a proporção de amostras classificadas como verdadeiro-positivos em relação a todas as predições positivas. Preditividade negativa é a proporção de amostras classificadas como verdadeiro-negativos em relação a todas as predições negativas.

Foram realizados testes com valores variados para o tamanho do contexto, k, utilizado no PPM-C, mas os melhores resultados foram obtidos com o contexto igual a dois (k = 2). O melhor caso (k = 2) produziu os valores mostrados na Tabela 1.

Tabela 1. Sensibilidade (S), Especificidade (E) ,Acurácia (A), Preditividade Positiva(PP) e Preditividade Negativa(PN) para cada classe com k = 2.

Classes	S %	E %	A %	PP%	PN%
NSR	94,45%	100,00%	98,81%	50,00%	100,00%
BII	94,45%	100%	98,81%	100,00%	50,0%
AF	100%	97,06%	97,68%	50,00%	100,00%
VF	88,89%	97,22%	95,56%	100,00%	50,00%
SBR	100,00%	100,00%	100,00%	100,00%	100,00%
Média	95,56%	98,86%	98,17%	80,00%	80,00%

A Tabela 2 apresenta um comparativo dos resultados de diferentes métodos de classificação de arritmias cardíacas e o método proposto.

As classes usadas pelos autores na tabela 2 foram em [Tsipouras e Fotiadis 1997] – Normal ou Arritimia; em [Tsipouras, Fotiadis e Sideris 2005] - NSR, PVC, VF e BII; em [Asl, Setarehdan e Mohebbi 2008] – NSR, PVC, AF, VF, SBR e BII; no Método Proposto – NSR, BII, AF, SBR e VF.

Tabela 2. Comparação dos resultados obtidos com o método proposto neste trabalho com outros métodos extraídos da literatura.

Autor	Sensibilidade (%)	Especificidade (%)	Acurácia (%)
Tsipouras et al. [3]	89,95	92,91	-
Tsipouras et al. [4]	-	-	94,26
Asl Mohammadzadeh et al. [5]	95,77	99,40	99,16
Método Proposto	95,56	98,86	98,17

5. Conclusão

Este trabalho apresentou um novo classificador de arritmias cardíacas baseado no sinal da VFC, com resultados comparáveis aos melhores classificadores modernos. O método proposto apresentou uma média de 95,56% de sensibilidade, 98,86% de especificidade e 98,17% de acurácia, valores superiores aos apresentados em [Tsipouras e Fotiadis 1997] e [Tsipouras, Fotiadis e Sideris 2005], mas não superiores aos observados em [Asl, Setarehdan e Mohebbi 2008]. Como se trata de uma abordagem inovadora, os resultados podem ser melhorados em trabalhos futuros. Testes preliminares indicam que o aumento dos conjuntos de treinamento tem impacto positivo, levando a acurácia para 99,71%, a sensibilidade para 97,09% e a especificidade para 99,85%. Contudo, testes mais sistemáticos devem ser efetuados para consolidação destes últimos avanços.

Agradecimentos

Agradecemos às nossas instituição de fomento Ministério da Educação (MEC) e à Secretaria de Educação Superior (SESU), além do Departamento de Informática da Universidade Federal da Paraíba.

Referências

Acharya R.U., Bhat BS, Iyengar SS, Rao A, Dua S. (2003) "Classification of heart rate data using artificial neural network and fuzzy equivalence relation". Pattern Recogn 2003;36(1): p. 61-68.

Asl B.M., Setarehdan SK, and Mohebbi M. (2008). "Support vector machine-based arrhythmia classification using reduced features of heart rate variability signal", Artif. Intell. Med. 44, 1 (Sep. 2008), p. 51-64.

Bell T. , Cleary J. , Witten I. (1984), "Data compression using adaptive coding and partial string matching", IEEE Transactions on Communications, v. 32, n. 4, p. 396-402.

Shannon C. E. (1948) "A mathematical theory of communication",' Bell System Technical Journal,vol. 27, p. 379-423 and 623-656, July and October.

Carvalho L.C. (2008) "Instrumentação Médico-Hospitalar", 1ª ed. Barueri - São Paulo: Manole. v. 1.

Hamilton P.S., Tompkins W.J. (1986) "Quantitative investigation of QRS detection rules using the MIT/BIH arrhythmia database". IEEE Trans Biomed Eng 1986;33(12), p.1157-1165.

Honorio, T. C. S. ; Batista, L. V. ; Duarte, R. C. M. (2009) ―Texture Classification Using Prediction by Partial Matching Models‖. Workshop de Visão Computacional, São Paulo, p. 21-26.

Goldberger, L. and E. (1977), Clinical Electrocardiography, Saint Louis: The Mosby Company, 1st edition.

Madeiro J.P.V. et al (2007), "A new approach to QRS segmentation based on wavelet bases and adaptive threshold technique", Medical Engineering and Physics, v. 29, p. 26-37.

Mark R.G., Moody G.B. (1997). “MIT-BIH Arrhythmia Database” [Online]. Disponível em: http://ecg.mit.edu/dbinfo.html [acessado: 23.02.2009].

Pan J., Tompkins W.J. (1985) “A real time QRS detection algorithm” IEEE Trans Biomed Eng 1985;32(3): p. 230-236.

Salomom D. (2004) “Data Compression – The Complete Reference”. 3ª ed. Springer.

Task force of The European Society of Cardiology and The North American Society of Pacing and Electrophysiology (1996), “Heart Rate Variability: Standards of measurement, physiological interpretation and clinical use”, Eur. Heart J, vol. 17, p. 354-381.

Tsipouras M.G., Fotiadis D.I. (2004) “Automatic arrhythmia detection based on time and time—frequency analysis of heart rate variability” Comp Meth Prog Biomed 2004;74(2): p. 95-108. Comput 1997;35(6): p. 626-632.

Tsipouras M.G., Fotiadis D.I., Sideris D. (2005) “An arrhythmia classification system based on the RR-interval signal” Artif Intell Med 2005;33(3): p. 237-250.

FREDS: Framework para Redução da Descontinuidade Semântica em Imagens Médicas

Hugo Cesar Pessotti[1], Luiz Otavio Murta Junior[1], Edson Garcia Soares[2], Alessandra Alaniz Macedo[1]

[1] Departamento de Computação e Matemática, FFCLRP-USP, Brasil

[2]Departamento de Patologia, FMRP-USP, Brasil

{pessotti, murta}@usp.br, egsoares@fmrp.usp.br, ale.alaniz@usp.br

Abstract. *This paper describes the development of a framework focused on the definition of conceptual mappings between the content of medical images and textual information contained in medical records, applied in a scenario of computer-aided diagnosis in thyroid cancer. It is also intended to contribute towards the reduction of the semantic gap between computerized medical image retrieval and the human interpretation of its content.*

Resumo. *Este trabalho relata o desenvolvimento de um framework voltado para a definição de mapeamentos conceituais entre o conteúdo de imagens médicas e informações textuais contidas em laudos, aplicado no cenário de sistemas de apoio ao diagnóstico em câncer de tireoide. Pretende-se também contribuir com a redução da descontinuidade semântica existente entre a recuperação computadorizada de imagens médicas e a interpretação humana de seu conteúdo.*

1. Introdução

Os nódulos tireoidianos estão entre as doenças mais comuns envolvendo a tireóide. Nódulos palpáveis são encontrados em 4 a 7% da população adulta (com maior incidência entre mulheres e idosos), aumentando para 19 a 67% em exames de ultra-som, devido a sua maior precisão [Maia et al. 2007]. Embora comuns, apenas 5% dos nódulos são malignos, justificando a necessidade de um diagnóstico mais acurado para início do tratamento mais adequado ao paciente com câncer. Um sistema de apoio ao diagnóstico capaz de diferenciar entre os tipos de nódulos poderia ajudar o patologista a concluir diagnósticos com o fornecimento de informação extra, podendo evitar procedimentos desnecessários em pacientes com nódulos benignos e aumentar a sobrevida dos pacientes no caso de câncer. A sobrevivência estimada em 10 anos é de 95% para o câncer papilífero e 75% para o folicular em situações onde o diagnóstico é dado de forma precoce [Coeli et al. 2005].

O uso de sistemas computacionais pode auxiliar o estabelecimento do diagnóstico pela análise de imagens, uma vez que fornecem informações complementares de modo a diminuir o tempo e esforço necessário para analisar um exame. Tradicionalmente, esses sistemas baseiam-se principalmente em técnicas de Processamento de Imagens e Recuperação de Imagens Baseada em Conteúdo, que se concentram nos atributos das imagens [Doi 2007]. Segundo pesquisadores, a extração de atributos pode não oferecer ajuda substancial durante o processo de recuperação, pois o profissional de saúde muitas vezes não está familiarizado com os atributos escolhidos, especialmente os de baixo nível como, por exemplo, energia e entropia de uma imagem [Mueen et al. 2008]. O uso de

informações em nível semântico pode melhorar a precisão da recuperação de imagens, tornando possível ao usuário especificar sua intenção de busca e também aprimorar a interpretação dos resultados.

Neste artigo, propõe-se o *framework* FREDS (Framework para Redução da Descontinuidade Semântica em Imagens Médicas) para a definição de mapeamentos conceituais entre o conteúdo de imagens microscópicas e informações textuais presentes em exames. Este mapeamento será aplicado no contexto de sistemas de apoio ao diagnóstico, com o objetivo de extrair informações complementares de exames que descrevam componentes celulares similares à aqueles identificados em uma imagem microscópica sendo avaliada pelo patologista. Pretende-se também contribuir com a redução da descontinuidade semântica existente entre a recuperação computadorizada de imagens médicas e a interpretação humana de seu conteúdo, uma vez que o mapeamento semântico pode resultar na geração de conhecimento.

A seguir, na Seção 2, será apresentado um breve estudo de conceitos relacionados ao trabalho, na Seção 3 será apresentado o *framework* proposto para manipulação de imagens e laudos médicos, na Seção 4, a metodologia de experimentação proposta, na Seção 5, os resultados obtidos e, por fim, na Seção 6, as conclusões deste trabalho.

2. Fundamentos Teóricos

Neste seção será apresentado um breve estudo dos fundamentos teóricos relacionados ao trabalho, para uma melhor compreensão das técnicas abordadas pertencentes às áreas de Recuperação de Informação e Processamento de Linguagem Natural.

2.1. Recuperação de Imagens Baseada em Conteúdo

O ambiente médico é caracterizado por um grande volume de informações textuais ou imagens produzidas ao longo das práticas de tratamento e acompanhamento dos pacientes. Pesquisadores de Recuperação de Imagens Baseada em Conteúdo investigam maneiras de otimizar a recuperação e a análise de imagens. Normalmente, esses mecanismos baseiam-se na extração de assinaturas das imagens, ou vetor de características, a partir de um ou mais descritores. Descritores são funções que descrevem algum aspecto relacionado ao conteúdo de uma imagem, que pode ser relacionado à sua cor, forma ou textura. As buscas por imagens são realizadas utilizando os vetores de características e não as imagens propriamente ditas, sendo necessário um critério para definir a similaridade entre duas assinaturas e, por consequência, entre as imagens que representam.

2.2. Rotulação de Imagens

Rotulação de imagens é o processo de atribuição de valores numéricos para regiões conexas de uma imagem. Este processo consiste na identificação de regiões conexas, por exemplo, através da segmentação de componentes de uma imagem. Todos os *pixels* pertencentes a uma mesma região recebem um mesmo valor numérico, chamado de rótulo, que é atribuído sequencialmente conforme as regiões vão sendo identificadas.

A rotulação é necessária para identificar de forma única os diferentes componentes de uma imagem microscópica e pode ser conduzida de duas maneiras diferentes: (a) rotulação automatizada, onde são utilizados algoritmos de processamento de imagens para segmentar e identificar os rótulos e (b) rotulação auxiliada pelo usuário, onde é apresentado ao especialista uma imagem microscópica e este é responsável por identificar os achados patológicos e posteriormente rotulá-los.

2.2.1. Rotulação Automatizada

A rotulação automatizada é realizada através da aplicação de técnicas de Processamento de Imagens e Recuperação de Imagens Baseada em Conteúdo. Inicialmente deve ser criada uma base de exemplos de achados patológicos, estruturada com o auxílio de um especialista, contendo exemplos de regiões de interesse para a definição de diagnóstico em câncer de tireoide.

Uma vez estruturada a base de exemplos de achados patológicos, a rotulação automatizada pode ser conduzida através da segmentação de uma imagem microscópica. Cada componente segmentado é submetido à extração de atributos e posteriormente é realizada sua classificação por similaridade, onde é atribuído um rótulo de acordo com sua similaridade com os achados patológicos armazenados na base de exemplos. Este processo é apresentado no Algoritmo 1a.

```
função rotularImagem(I)
  nroRótulo <- 0
  rótulos <- vetor[]
  componentes <- segmentarImagem(I)

  para cada componente da imagem I
    nroRótulo <- nroRótulo + 1
    áreaSegm <- matriz[]

    para cada pixel (x,y) do componente
      rótulosImagem[x,y] <- nroRótulo
      áreaSegm[x,y] <- I[x,y]
    fim para

    vetorAtributos <- extraçãoAtrib(áreaSegm)
    classe <- classificSimilaridade(vetorAtributos)

    rótulos[nroRótulo] <- obterRótulo(classe)
  fim para
fim
```

a) Rotulação automatizada

```
função rotularImagemAuxilioUsuario(I)
  nroRótulo <- 0
  rótulos <- vetor[]
  componentes <- segmentarImagem(I)

  para cada componente da imagem I
    nroRótulo <- nroRótulo + 1
    áreaSegm <- matriz[]

    para cada pixel (x,y) do componente
      rótulosImagem[x,y] <- nroRótulo
    fim para

    destacarRegiãoImagem(áreaSegm)
    classe <- classificUsuario(áreaSegm)

    rótulos[nroRótulo] <- obterRótulo(classe)
  fim para
fim
```

b) Rotulação Auxiliada pelo Usuário

Pseudo-Algoritmo 1. Rotulação de imagens (abstrações criadas pelos autores)

2.2.2. Rotulação Auxiliada pelo Usuário

Na rotulação auxiliada pelo usuário, o patologista é responsável pela identificação das estruturas e achados patológicos presentes na imagem microscópica. Inicialmente a imagem é segmentada e cada componente extraído é apresentado ao patologista, que deve classificá-lo e atribuir manualmente o rótulo, de acordo com o Algoritmo 1b.

A rotulação auxiliada é útil nos casos onde o patologista consegue identificar os padrões em uma imagem mas apresenta dificuldade em construir o seu diagnóstico. Esta abordagem também pode servir como uma forma de *feedback* para a rotulação automática, permitindo ao patologista corrigir eventuais erros durante a classificação e adicionar rótulos que não foram identificados de forma automatizada, podendo levar a um aumento na taxa de precisão da rotulação.

2.3. Modelo Vetorial de Recuperação de Informação

O Modelo Vetorial [Salton et al. 1975] é um modelo utilizado em Recuperação de Informação para representação de informação textual. Neste modelo, os termos de índice

de um documento ou de uma expressão de busca são representados através de pesos numéricos, ao contrário do Modelo Booleano que apenas registra a presença ou ausência dos termos. A escolha de pesos numéricos torna possível a recuperação parcial de documentos, permitindo o cálculo do grau de similaridade entre documentos e uma expressão de busca.

No modelo vetorial, o documento é representado por um vetor onde cada dimensão corresponde ao peso de um termo em relação ao documento. Caso o termo não ocorra no documento, seu peso atribuído é igual a zero. Existem diversas maneiras de calcular estes pesos, sendo a mais comum a medida TF-IDF (*Term Frequency-Inverse Document Frequency*). Dado uma coleção de D documentos e um termo t_k, a medida TF-IDF é calculada segundo a equação a seguir:

$$w_{ki} = \frac{n_{ki}}{N_i} log\left(\frac{|D|}{|d : t_k \in d|}\right) \tag{1}$$

onde n_{ki} é o número de vezes que o termo t_k aparece no documento d_i, N_i o número total de palavras neste documento, $|d : t_k \in d|$ o número de documentos que apresentam o termo t_k e $|D|$ o número total de documentos na coleção.

2.4. Similaridade do Cosseno

A similaridade entre documentos desestruturados geralmente é medida através do grau de similaridade de seus conteúdos, partindo do princípio que documentos que compartilham termos em comum têm uma maior probabilidade de serem semelhantes.

Considerando-se o espaço vetorial gerado pelo Modelo Vetorial, onde o perfil de cada documento é definido como sendo o vetor de pesos dos termos que o compõe, a similaridade entre dois documentos pode ser medida através do cosseno do ângulo formado por seus vetores, dada a seguinte equação:

$$cos\left(d_i, d_j\right) = \frac{\sum\limits_{k} w_{ki} w_{kj}}{\sqrt{\sum\limits_{k} w_{ki}^2}\sqrt{\sum\limits_{k} w_{kj}^2}} \tag{2}$$

onde k é o número de termos presentes na coleção de documentos e w_{ki} e w_{kj} representam a importância do termo t_k para os documentos d_i e d_j, respectivamente, utilizando a medida TF-IDF.

O resultado é o cosseno do ângulo formado pelos dois vetores de pesos. Devido ao fato dos pesos calculados segundo a medida TF-IDF serem sempre positivos, o cosseno resultante estará entre 0 e 1. Um cosseno próximo a zero indica que há poucos termos comuns entre os documentos, ao passo que quanto mais próximo de 1, maior será a similaridade.

2.5. Normalização Linguística

Uma das tarefas mais comuns realizadas antes do processamento de informação textual é sua normalização. A normalização linguística tem como objetivo facilitar o reconhecimento das variações que ocorrem durante a escrita de textos em linguagem natural. Essas variações podem ser de origem morfológica ou semântica.

A variação morfológica ocorre quando várias palavras apresentam um radical em comum, seja pela variação de gênero, número e grau ou até mesmo em suas diferentes formas verbais. *Stemming*, ou radicalização, é uma técnica utilizada para extrair o radical de uma palavra através da remoção de sufixos, permitindo que palavras de mesmo significado sejam comparadas independente de sua variação de gênero, número, grau e flexão. Em [Viera and Virgil 2007] são apresentados e comparados os principais algoritmos de radicalização para a língua portuguesa, destacando o algoritmo de Orengo [Orengo 2004] e a adaptação do algoritmo de Porter [Porter 1980].

A variação semântica é observada principalmente nos casos de sinonímia, onde palavras de formas diferentes podem representar o mesmo conceito ou ter um significado semelhante. Sua normalização é feita pela identificação de sinônimos em comum, criando agrupamentos de palavras de significados similares. Isto pode ser feito, por exemplo, através da busca de sinônimos em tesauros. Um tesauro pode ser visto como uma lista de palavras e seus sinônimos, dado um domínio específico.

2.6. Redução da Ambiguidade

Um dos maiores desafios na área de Processamento de Linguagem Natural é identificar e resolver problemas de ambiguidade presente nos textos. A ambiguidade surge quando uma expressão apresenta mais de uma interpretação possível, podendo ser principalmente de origem léxica ou estrutural [Ullmann 1964].

Observamos a presença da ambiguidade léxica principalmente nos casos de polissemia, onde uma mesma palavra pode ter significados diferentes dado o contexto em que se apresentam. Por exemplo, o termo "célula" pode significar "pequena cavidade", "unidade microscópica fundamental da matéria viva" ou até mesmo "agrupamento de pessoas para fins políticos". Uma abordagem semântica pode ser empregada para determinar o significado de uma palavra dentro do domínio, através do uso de ontologias de domínio.

A ambiguidade estrutural está relacionada às diferentes formas que as palavras podem ser organizadas para gerar uma interpretação, como no seguinte exemplo: "câncer de tireoide ausente". Nesta construção, existem duas interpretações possíveis: (1) Não há câncer de tireoide ou (2) o tipo de câncer é "tireoide ausente". Embora neste caso a primeira opção seja a mais provável de ser verdadeira, não há uma regra para determinar qual é correta, dado que a interpretação é sensível ao contexto. Uma forma de resolver este tipo de ambiguidade é atribuir pesos para cada combinação de palavras, que podem ser calculados segundo a frequência de cada um na coleção de documentos. Neste cenário, os termos são agrupados através do uso de *n-grams*. Considerando o exemplo anterior, as representações *tri-gram* (onde n=3) seriam: "câncer de tireoide" e "de tireoide ausente". Desta forma, se o *tri-gram* "câncer de tireoide" ocorrer com mais frequência que "de tireoide ausente", esta estrutura será a preferida em favor da outra.

3. *Framework* FREDS

O *framework* FREDS (*Framework* para Redução da Descontinuidade Semântica em Imagens Médicas) foi construído com o objetivo de suportar aplicações que necessitem de acesso à componentes de Processamento de Linguagem Natural, Recuperação de Informação e Recuperação de Imagens Baseada em Conteúdo para investigar o mapeamento do conteúdo de imagens médicas e informações textuais em exames de tireoide. Esta proposta é uma continuação do *framework* de [Pessotti et al. 2008].

Para determinar as necessidades e compreender as funcionalidades do sistema proposto, foram conduzidas atividades de análise de requisitos e desenvolvimento de um diagrama de classes utilizando a linguagem UML. Os principais requisitos funcionais do FREDS são: (i) permitir que o patologista submeta imagens microscópicas obtidas através de microscópios com câmeras digitais; (ii) rotular (identificar e nomear) regiões de interesse em uma imagem médica; (iii) retornar laudos cujas descrições sejam similares a expressões de busca obtidas a partir dos rótulos de uma imagem; (iv) obter e agrupar os diagnósticos de um conjunto de laudos e (v) obter os termos mais frequentes de um conjunto de diagnósticos.

O *framework* tem cinco interfaces de software: Segmentador, Extrator, Classificador, Recuperador de Informação e Agrupador de Exames. Essas interfaces foram definidas com o objetivo de prover um meio de acesso comum às classes do sistema, permitindo que novos componentes sejam implementados futuramente sem a necessidade de readequar os componentes que os utilizam. Este trabalho explora a característica de modularidade do *framework* inicial para modificar seus componentes de modo a poder aplicá-lo no cenário proposto. A Figura 1 apresenta o Diagrama de Classes elaborado, baseado no diagrama proposto inicialmente em [Pessotti et al. 2008]. Os componentes sombreados no diagrama foram reutilizados do trabalho original, enquanto os demais foram desenvolvidos especificamente para este trabalho.

4. Experimentação

Nesta seção serão abordadas a estruturação da base de imagens e laudos e, em seguida, a metodologia desenvolvida para a condução dos experimentos com a base criada.

4.1. Construção da base de imagens e laudos

A estruturação da base de imagens e laudos foi realizada em quatro etapas:

1. Levantamento de Exames: Inicialmente, foram selecionadas lâminas de exames de tireoide realizados no Serviço de Patologia do HCFMRP-USP, dos quais se obtiveram as imagens digitalizadas e o respectivo laudo patológico.
2. Escolha de exames relevantes: Com a ajuda do Prof. Dr. Edson Garcia Soares foram escolhidas as imagens mais relevantes de cada exame, dado o diagnóstico em questão.
3. Identificação dos padrões: Após a escolha das imagens mais relevantes, foi solicitado ao patologista que identificasse os padrões presentes em cada imagem. Estes padrões são utilizados no treinamento dos algoritmos de rotulação automatizada.
4. Criação da base: A partir das regiões identificadas na etapa anterior foi construída a base de imagens e laudos, armazenando-se os limites da região marcada pelo patologista, os valores de pixel em seu interior e o laudo associado ao exame.

Cada laudo é composto por dois documentos: descrição e diagnóstico, ambos redigidos pelo patologista. A descrição contém uma visão geral da lâmina em termos de achados patológicos e organização celular. O diagnóstico contém a conclusão do laudo, tomada a partir dos achados da lâmina.

4.2. Rotulação e Seleção de Laudos

A fim de aplicar o *framework* FREDS no contexto de sistemas de apoio ao diagnóstico, foi definida uma metodologia para explorar a base de imagens e laudos. A rotulação foi

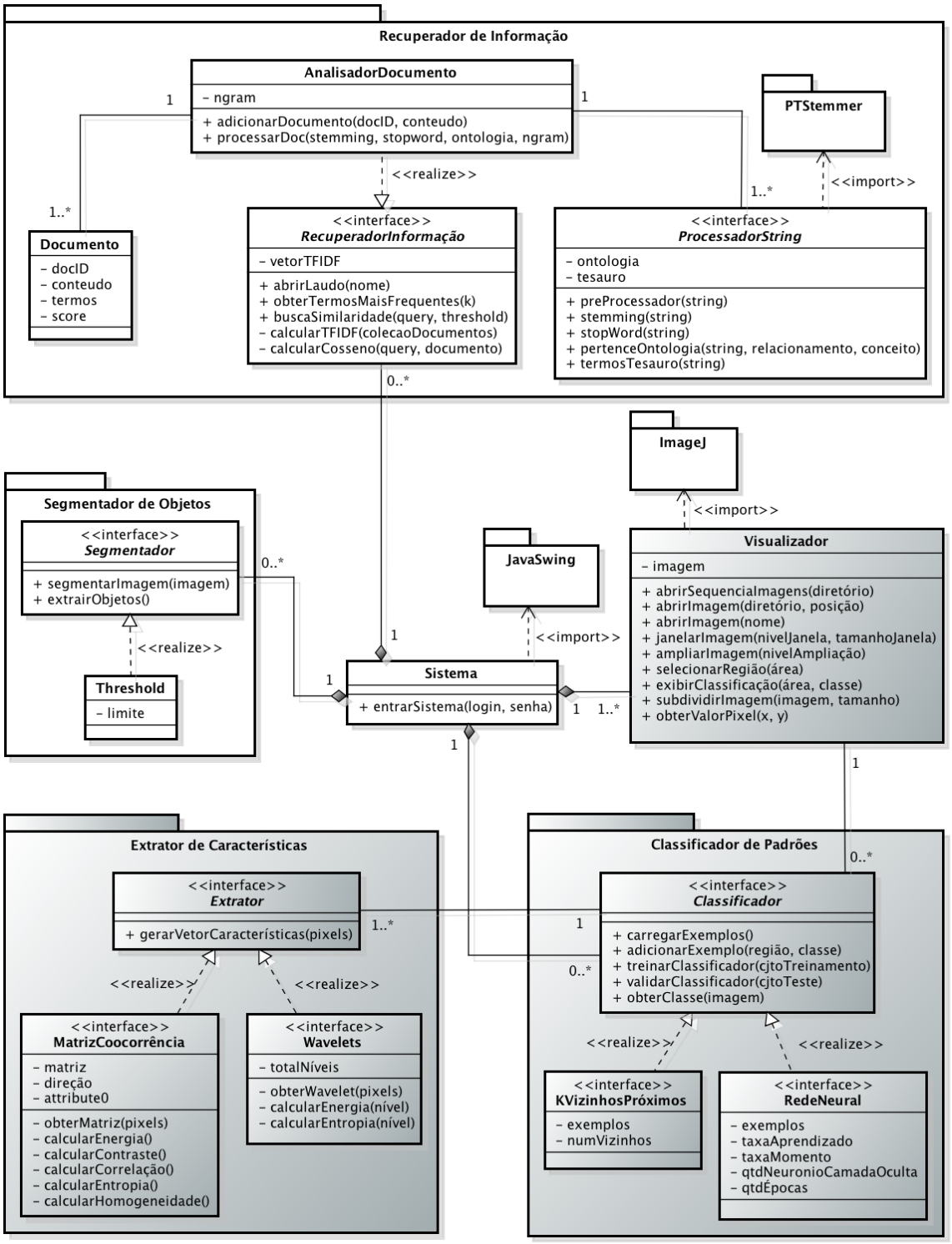


Figura 1. Classes de Software do FREDS. Em cinza, os componentes propostos originalmente em [Pessotti et al. 2008] e em branco os desenvolvidos neste trabalho.

conduzida de forma automatizada, utilizando os algoritmos de Matriz de Coocorrência [Haralick et al. 1973] e Wavelets [Daubechies 1992] para a extração de atributos e a classificação foi realizada através de Redes Neurais Artificiais [Haykin 2001] com taxa de aprendizado de 0.1, 0.05 de momento e 1000 épocas de treinamento. Uma visão geral da metodologia é apresentada na Figura 2.

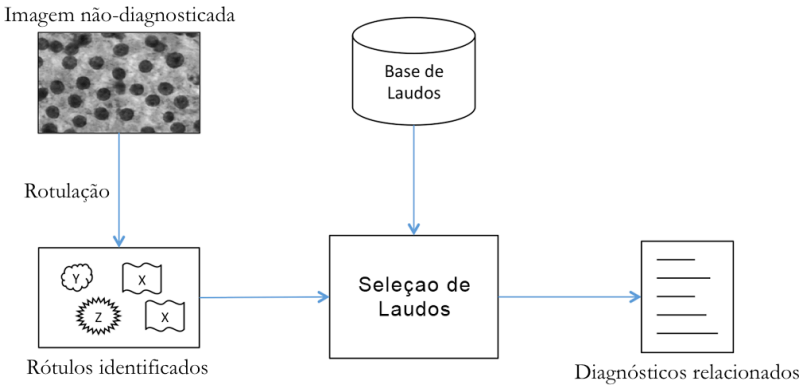


Figura 2. Metodologia da experimentação

Após identificados os rótulos de uma imagem, estes são utilizados na formação de expressões de busca (*queries*) para a etapa de seleção de laudos patológicos. Em seguida, os rótulos identificados passam por uma expansão de termos por meio de um tesauro. Para verificar se os termos expandidos continuam dentro do contexto de câncer de tireoide, é utilizada uma ontologia de domínio definida junto ao especialista. No estágio atual, ela contém termos associados a achados patológicos e diferentes tipos de câncer de tireoide. Conforme a necessidade em trabalhos futuros, essa ontologia poderá ser expandida. Embora existam ontologias de carcinomas que contenham relacionamentos do tipo "achado patológico" [Golbeck et al. 2003], não foi encontrada nenhuma especificamente dentro do domínio de tireoide para a língua portuguesa.

A formação das *queries* é dada pela combinação dos termos expandidos e conectivos lógicos AND e OR. Estas *queries* foram utilizadas para a seleção de laudos patológicos, realizada através do cálculo de similaridade do cosseno entre a descrição presente nos laudos e cada combinação formada.

O resultado final é apresentado sob duas formas diferentes: (1) listagem dos termos mais relevantes encontrados nos laudos, utilizando o peso de cada termo como medida de *score* e (2) agrupamento dos diagnósticos presentes nos laudos, utilizando a similaridade cosseno como medida de *score*. Acredita-se que a listagem de termos relevantes pode ser útil para fornecer auxílio ao patologista durante a escrita do seu diagnóstico final, enquanto o agrupamento de diagnósticos é mais indicado caso o patologista queira comparar seu diagnóstico com diagnósticos similares presentes na base de laudos.

5. Resultados

Os Quadros 1a, 1b e 2 apresentam resultados iniciais obtidos por meio da experimentação com a base de laudos, atualmente composta por 25 laudos patológicos, totalizando 723 palavras ligadas à descrição dos casos e 310 palavras presentes nos diagnósticos. Considerou-se como entrada os rótulos "pseudo-inclusão" e "fendas", que foram obtidos através da rotulação automatizada. Esses rótulos representam achados patológicos que são frequentemente associados à câncer papilífero.

“papilífero” = 0.1079 “folicular” = 0.0596 “microcarcinoma” = 0.0568	“compatível microcarcinoma papilífero” = 0.1152 “lesão padrão folicular” = 0.1152 “papilífero recomenda-se confirmação” = 0.0578
a) n-gram = 1	b) n-gram = 3

Quadro 1. Conjuntos de termos mais relevantes e score obtido

O Quadro 1a contém os três termos de maior *score* encontrados em 5 exames retornados pelas expressões de busca formadas pelos rótulos. Já o Quadro 1b apresenta a mesma consulta, mas considerando o parâmetro n-gram = 3 nos algoritmos de Recuperação de Informação. O uso de n-gram retornou um conjunto de termos mais significativo segundo o especialista, pois as palavras retornadas sem o uso de n-gram não apresentam significado quando aparecem isoladas. Os termos “papilífero” e “folicular” não demonstram o grau de desenvolvimento do carcinoma, apenas definem o seu tipo. Ao aparecerem associados a outros termos, como “lesão”, “microcarcinoma” ou “carcinoma”, os conjuntos passam a ter um significado maior para o patologista.

O baixo *score* dos termos é explicado pelo fato da base atual de exemplos conter uma baixa variedade de diagnósticos, pois obteve-se acesso apenas aos laudos dos exames selecionados para captura de imagens. Consequentemente, os termos aparecem com frequência maior na coleção de documentos e seu *score* é reduzido pela medida TF-IDF, de acordo com a equação (1).

“Quadro morfológico compatível com carcinoma papilífero. Requer confirmação histopatológica” = 0.3309 “Quadro morfológico sugestivo de lesão de padrão microfolicular a esclarecer (hiperplásica x neoplásica)” = 0.2514 “Quadro compatível com microcarcinoma papilífero. Recomenda-se confirmação histopatológica com exame per-operatório de congelação” = 0.2065

Quadro 2. Exemplo contendo três diagnósticos selecionados e score obtido

O Quadro 2 mostra três diagnósticos que obtiveram o maior *score* durante a seleção automatizada dos laudos a partir dos rótulos de entrada. Segundo o especialista, os documentos retornados estão de acordo com o esperado para os rótulos de imagem considerados (“pseudo-inclusão” e “fendas”), pois representam termos associados à câncer papilífero.

6. Conclusão

Uma forma de construir um mapeamento entre imagens médicas e diagnósticos é a utilização dos rótulos da imagem. Através da rotulação automatizada foi possível identificar os achados patológicos presentes em uma imagem não-diagnosticada, a partir dos quais obteve-se termos relacionados a diagnósticos de câncer pela busca em uma base de laudos. Desta forma, constituiu-se uma ponte entre o conteúdo da imagem e a descrição do laudo, permitindo a localização de diagnósticos que são similares à imagem em questão.

Embora os resultados apresentados estejam de acordo com o esperado, pretende-se ampliar a base de laudos para aumentar a variedade de exemplos de casos. Espera-se que o aumento do número de exemplos ocasione a inclusão de casos de difícil interpretação, que por sua vez são importantes na etapa de treinamento dos algoritmos. Desta forma, os algoritmos poderão reconhecer estes tipos de casos com um maior grau de confiança.

Como trabalho futuro, propõe-se que sejam verificadas oportunidades de aplicação no contexto de aprendizado eletrônico, sistemas de vigilância e, em especial, auditoria de exames. Dado um exame avaliado pelo patologista, é possível verificar a validade de seu diagnóstico através da comparação do diagnóstico final com os diagnósticos sugeridos dentro do nível de confiança fornecido pelo sistema.

7. Agradecimentos

Os autores agradecem a FAPESP pelo apoio financeiro (08/08098-1).

Referências

[Coeli et al. 2005] Coeli, C., Brito, A., Barbosa, F., et al. (2005). Incidência e mortalidade por câncer de tireóide no Brasil. *Arq Bras Endocrinol Metab*, 49:503–509.

[Daubechies 1992] Daubechies, I. (1992). *Ten lectures on wavelets*. Society for Industrial and Applied Mathematics.

[Doi 2007] Doi, K. (2007). Computer-aided diagnosis in medical imaging: Historical review, current status and future potential. *Comput Med Imaging Graph*, 31:198–211.

[Golbeck et al. 2003] Golbeck, J., Fragoso, G., Hartel, F., et al. (2003). The National Cancer Institute's thesaurus and ontology. *Journal of web semantics*, 1(1):75–80.

[Haralick et al. 1973] Haralick, R. M., Shanmugam, K., and Dinstein, I. (1973). Textural features for image classification. *IEEE Trans. Syst. Man Cybern*, 3(6):610–621.

[Haykin 2001] Haykin, S. (2001). *Redes Neurais Princípios e Práticas*. Bookman, Porto Alegre.

[Maia et al. 2007] Maia, A., Ward, L., Carvalho, G., et al. (2007). Consenso brasileiro em nódulos de tireoide e câncer diferenciado de tireóide. *Arq Bras Endocr Metab*, 51(5).

[Mueen et al. 2008] Mueen, A., Zainuddin, R., and Baba, M. S. (2008). Automatic multilevel medical image annotation and retrieval. *J. Digital Imaging*, 21(3):290–295.

[Orengo 2004] Orengo, V. (2004). *Assessing relevance using automatically translated documents for cross-language information retrieval*. PhD thesis, School of Computing Science, Middlesex University, London.

[Pessotti et al. 2008] Pessotti, H., de Azevedo-Marques, P. M., and Macedo, A. A. (2008). Framework para classificação automática de tomografias computadorizadas de alta resolução para auxílio ao diagnóstico de lesões intersticiais de pulmão. In *Anais do XXI Congresso Brasileiro de Engenharia Biomédica (CBEB 2008)*, Salvador.

[Porter 1980] Porter, M. (1980). An algorithm for suffix stripping. *Program*, 14(3):130–137.

[Salton et al. 1975] Salton, G., Wong, A., and Yang, C. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

[Ullmann 1964] Ullmann, S. (1964). *Semântica: uma introdução à ciência do significado*. Fundação Calouste Gulbenkian, Lisboa, 3 edition.

[Viera and Virgil 2007] Viera, A. and Virgil, J. (2007). Uma revisão dos algoritmos de radicalização em língua portuguesa. *Information Research*, 12(3):12–3.

Análise de Imagens de Retinografia Digital com ImageJ Customizado

Daniel Maraschin[1], Marcelo Maestri[2], Nadjia Feller[1], Luciana Nedel[1], Bruce Duncan[3]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91501-970 – Porto Alegre – RS – Brasil

[2]Serviço de Oftalmologia – Hospital de Clínicas de Porto Alegre (HCPA)
Rua Ramiro Barcelos 2350 – 90035-903 – Porto Alegre – RS – Brasil

[3]Departamento de Medicina Social – Universidade Federal do Rio Grande do Sul (UFRGS)
Rua Ramiro Barcelos 2600/414 – 90035-903 – Porto Alegre – RS – Brasil

{dmmgutterres,njfeller,nedel}@inf.ufrgs.br, mmaestri@terra.com.br, bbduncan@ufrgs.br

Abstract. *This paper present ImageJ for Fundus of the Eye analysis, a modification on ImageJ, an open-source multiplataform tool, to ease reading and evaluation of fundus of the eye images while searching for clues for cardiovascular problems or those related to diabetes. Functionalities added to this tool were developed according to the evaluation protocol suggested by the Department of Ophthalmology and Visual Sciences of Wisconsin University, along with added communication to a DICOM image server. In this paper we present the added functionalities, aspects of their implementation and preliminary results of its use in evaluation of participant exams for a longitudinal study.*

Resumo. *Este artigo apresenta o ImageJ para Retinografia, uma modificação do ImageJ, ferramenta multiplataforma de código aberto, para facilitar a leitura e avaliação de imagens de fundo de olho em busca de indícios de problemas cardiovasculares ou relacionados à diabetes. As funcionalidades adicionadas a essa ferramenta foram desenvolvidas conforme protocolo de leitura proposto pelo Departamento de Oftalmologia e Ciências Visuais da Universidade de Wisconsin, além da comunicação com um servidor de imagens no padrão DICOM. Nesse artigo apresentamos as funcionalidades agregadas à ferramenta, aspectos de sua implementação e resultados preliminares do uso na avaliação de participantes de um estudo longitudinal.*

1. Introdução

Retinografia é a tomada de fotos de fundo de olho (retina), seguida da avaliação dessas imagens para a constatação de anormalidades que ponham em risco a acuidade visual e a saúde do paciente. Esse procedimento é muito importante principalmente em indivíduos com doenças cardiovasculares e diabetes. Em pacientes hipertensos, a identificação de

anormalidades da circulação da retina auxilia na predição de complicações em outros órgãos como coração e cérebro. Nos pacientes com diabetes, auxilia na identificação da retinopatia diabética, lesão da retina causada por diabetes que pode levar à cegueira, além de contribuir para predição de outras complicações gerais dessa doença. A retinopatia diabética, uma vez identificada, pode ser tratada através de procedimentos oftalmológicos específicos, como *laser* ou cirurgia de retina, para reduzir a chance de perda visual.

Um dos problemas atuais no Sistema Único de Saúde (SUS) é a dificuldade de acesso da população ao exame de retinografia. São realizados entre 10000 e 11000 exames desse tipo, por mês, em todo Brasil, segundo dados do Ministério da Saúde [DATASUS, 2010]. Uma possível solução para esse problema seria aumentar a disponibilização da retinografia digital na rede pública credenciada. Nesse ponto encontra-seobstáculos importantes: o fato das ferramentas existentes para retinografias digitais serem proprietárias, licenciadas por um custo elevado por unidade, ou estarem, em alguns casos, em processo de tornarem-se obsoletas; ou ainda, em formatos de arquivo proprietário e sem conformidade com padrões internacionais, o que impede a troca de informações de imagens entre diferentes equipamentos. Além disso, as ferramentas proprietárias mais utilizadas usualmente exigem treinamento complexo dos profissionais especializados da área médica oftalmológica, com maior custo de recursos humanos. Uma vez que a compensação financeira oferecida pelo SUS é baixa, o estímulo para aumentar a vinculação desses profissionais à rede pública é reduzido.

Assim, o desenvolvimento de uma ferramenta de imagens para retinografia digital de baixo custo operacional, que possibilite a troca de imagens de diversos padrões entre diferentes equipamentos, de manuseio mais simples, possibilitando a identificação de anormalidades da retina de maneira eficiente até por profissionais não especializados, treinados e acompanhados sob supervisão médica oftalmológica, poderia reduzir seu custo global e estimular serviços médicos oftalmológicos a vincularem-se a rede pública, permitindo a ampliação da realização da retinografia pelo SUS.

O projeto ELSA [http://elsa.org.br] consiste em avaliar e acompanhar a saúde de 15.000 funcionários de instituições federais de ensino do Brasil ao longo de vinte anos. Para esse fim é realizada uma série de exames e questionários para avaliar os padrões sócio-econômicos e de saúde dos participantes. Um desses exames é a retinografia, que pode ajudar a prever doenças cardiovasculares e sinais de retinopatia diabética ao avaliar imagens de fundo do olho dos participantes.

No Centro de Dados do projeto ELSA, localizado na UFRGS, foi proposta uma ferramenta para auxiliar no exame da retinografia digital. Ela foi projetada para oferecer compatibilidade com o padrão DICOM NEMA PS3 [ISO 12052] permitindo o trabalho em conjunto com grandes repositórios de exames médicos. Para isso, foi escolhido o visualizador de imagens, com suporte a imagens DICOM e de código aberto, ImageJ [Rasband 2011] que permite modificações para agregar as funcionalidades necessárias. Essa ferramenta se propõe a ser uma solução simples e eficiente, além de estar atualizada com necessidades relacionadas à leitura de imagens de fundo de olho em busca de retinopatia diabética, aumentando a eficiência do trabalho envolvido na leitura e identificação de anormalidades da retina, tanto por parte profissionais médicos oftalmologistas como profissionais não especialistas treinados e acompanhados sob supervisão médica.

Esse artigo está dividido da seguinte forma: na sessão 2 é apresentado o projeto de pesquisa onde a ferramenta foi proposta; na sessão 3 são apresentados os requisitos que a ferramenta deve possuir para a leitura e avaliação dos exames de retinografia; na sessão 4 são descritos os passos que levaram à modificação do ImageJ e a maneira que foram conduzidos os treinamentos dos usuários; na sessão 5 são expostas as impressões e resultados da utilização da ferramenta por usuários; e por fim na sessão 6 estão as conclusões e considerações finais desse artigo.

2. O Projeto ELSA

O Projeto ELSA (Estudo Longitudinal de Saúde do Adulto) é uma pesquisa de âmbito nacional, composta por seis instituições públicas de ensino superior, e tem o propósito de investigar a incidência e os fatores de risco para doenças crônicas, em particular, as cardiovasculares e o diabetes. Em cada centro integrante do estudo, os sujeitos da pesquisa – 15 mil funcionários de instituições públicas de ensino superior, entre 35 e 74 anos de idade – fazem exames e entrevistas nas quais são avaliados aspectos como condições de vida, diferenças sociais, relação com o trabalho, gênero e especificidades da dieta da população brasileira. Estes indivíduos serão acompanhados e terão seus dados coletados durante mais de vinte anos, e esses serão armazenados e gerenciados através de um sistema *online*.

A produção de dados dentro do projeto se dá através de entrevista face-a-face e da realização de diversos tipos de exames. Os participantes também recebem monitoramento para averiguar o desdobramento de desfechos de saúde (morte, internação, afastamento do trabalho por motivo de doença). Além disso, espécimes biológicos são armazenados para investigações futuras. Após a realização dos exames, é gerado um laudo que é enviado para o participante. Esse laudo contém o resultado do exame, assim como orientações para o participante e seu médico, caso seja detectada alguma anomalia ou problema.

Participam do estudo seis instituições de ensino superior e pesquisa de seis estados das regiões nordeste, sudeste e sul do país: FIOCRUZ, UFBA, UFES, UFMG, UFRGS e USP. Como estudo multicêntrico, envolve diversas entidades em seis diferentes cidades brasileiras, onde cada uma delas possui um Centro de Investigação que coleta as informações do participante e as envia ao Centro de Dados. O Estudo também nucleia seis Centros de Leitura de Exames Especializados, Biotecas e o Laboratório Central, além do já mencionado Centro de Dados.

No Centro de Dados ELSA-RS, foi identificada uma deficiência em ferramentas para leitura e avaliação de exames de retinografia que comunicassem com um repositório DICOM, centralizando a aquisição desses exames entre seus centros. Por esse motivo foi proposta a ferramenta apresentada a seguir.

3. Ferramentas de Auxilio à Leitura

Algumas ferramentas podem ser utilizadas para aumentar a precisão do laudo da retinografia. Manipulação do contraste e brilho e magnificação da imagem ajudam a distinguir entre pontos

ou focos benignos de alteração de pigmentação da retina ou hemorragias, isquemias e edemas. O aumento do contraste também auxilia a visualização de imagens que ficaram muito escuras devido à absorção de luz pelo pigmento, a pupila pouco dilatada ou ainda pela utilização de um *flash* com intensidade baixa. A magnificação também pode ser utilizada para perceber estreitamento dos vasos. A seguir apresentamos com mais detalhes as funcionalidades que foram adicionadas ao ImageJ ou tiveram sua acessibilidade melhorada.

3.1. Grade Macular

Uma grade macular centralizada sobre a mácula da retina permite a divisão da imagem em diversos setores compreendidos entre as arcadas vasculares temporais superior e inferior (área entre os vasos principais que nascem no disco óptico e aproximam-se da têmpora), o que possibilita a classificação das lesões de acordo com os quadrantes nos quais elas se encontram. Após ser aplicada sobre a mácula, a grade é rotacionada para compensar torção do globo ocular. Com isso, conforme vemos na figura 1, ela divide a imagem com três círculos concêntricos e diversos setores: A região da mácula, dentro do círculo central; Uma região circular interna dividida em quatro quadrantes ao redor da mácula, entre o círculo interno e o círculo central; Uma região intermediária dividida em quatro quadrantes entre o círculo externo e o círculo central; E uma região externa dividida em oito setores fora do círculo exterior.

O objetivo da grade macular é auxiliar a classificação topográfica das lesões na área acima em relação à posição da mácula, pois essa é a principal região para a acuidade visual máxima além de determinar o risco de prejuízo visual da seguinte forma: Lesões mais distantes do centro têm menor chance de afetar a visão; Lesões mais próximas ao centro (mácula) trazem maior perigo à visão, influenciando a decisão médica de tratamento ou observação. A grade também auxilia na classificação do risco por extensão da lesão: Lesão que ocupa apenas uma área de círculo ou até duas áreas de anel; Lesão em apenas um quadrante (menor risco); Lesão presente nos quatro quadrantes (maior chance de afetar a visão). Se ficarem no limite, serão registradas como presente nas duas zonas, mas também há o registro de tamanho (por isso a necessidade de círculos de tamanho pré-determinado na ferramenta de medição de lesões, para possibilitar comparação de tamanhos). Sabe-se do perigo maior ou menor por outros trabalhos prévios, que já quantificaram o risco pelo tamanho e localização. A ferramenta é proposta apenas para auxiliar a identificação dessas lesões de maior ou menor risco [Wang, 2003].

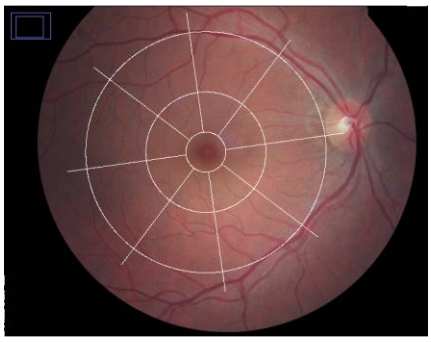


Figura 1. Exemplo de grade macular

3.2. Grade de Disco Óptico

A grade do disco óptico, apresentada na figura 2, é composta de três círculos concêntricos. É posicionada sobre o disco óptico e utilizada para definir, orientar e delimitar zonas válidas para medida do calibre vascular, detecção de anormalidades vasculares como cruzamentos arteriovenosos, espasmos arteriolares focais (estreitamento súbito dos vasos), além de facilitar a divisão em quadrantes para o registro de localização dessas anormalidades. O índice artério-venular é um dado biológico que se associa com risco de morbidade ou mortalidade cardiovascular. Ou seja, um paciente com razão alterada (baixa) tem uma chance maior de sofrer um evento cardiovascular (infarto, AVC, etc.) e morte, do que um paciente com a razão normal.

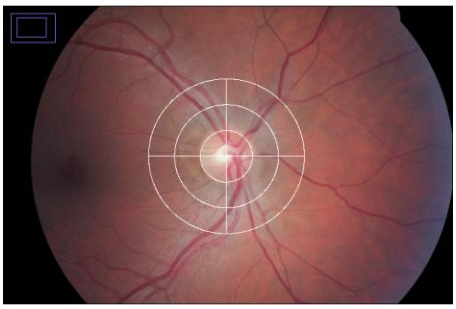


Figura 2. Exemplo de grade de disco óptico

3.3. Filtros

Um filtro de tons de cinza da cor verde permite acentuar o contraste entre estruturas vermelhas e a retina. Isso salienta o desenho e trajeto dos vasos e pontos hemorrágicos, eliminando reflexos indesejáveis na avaliação dessas estruturas, principalmente em fundos de olho com pigmentação muito avermelhada (como por exemplo, em caso de olhos claros). Isso permite identificar mais facilmente constrições focais de vasos, lesões hemorrágicas associadas à retinopatia diabética, hipertensiva ou outras alterações vasculares ou retinianas. Além disso, esse filtro facilita a detecção de glaucoma, pois acentua o contraste entre a retina sadia e a porção alterada por perda de fibras nervosas, que ocorre a partir do disco óptico e segue de forma arqueada em direção lateral superior ou inferior, nos casos mais graves.

A ferramenta de filtro de inversão de cores processa a imagem que está sendo avaliada, e calcula o negativo colorido, ressaltando artefatos de câmera (sujeira ou danos à lente), e lesões que podem apresentar-se obscurecidas por outras estruturas na imagem.

3.4. Grade de Lesões

A ferramenta de Grade de Lesões permite dimensionar e aplicar círculos à imagem, para dessa forma caracterizar periculosidade de lesões e edemas de acordo com seu tamanho. Essa ferramenta também é usada para medir a cavidade do disco óptico, para indicar presença ou formação de glaucoma. Os círculos são aplicado clicando no centro da área de interesse, com isso, um menu será aberto pedindo o diâmetro em micra que pode ser digitado ou selecionado de uma lista pré-definida [Klein, 1991].

4. ImageJ para Retinografia

Nessa sessão são apresentadas as abordagens utilizadas para a resolução de cada problema de implementação da ferramenta e as decisões tomadas durante seu projeto. Em seguida, a mesma é apresentada, com o passo-a-passo utilizado para o treinamento de usuários.

4.1. Implementação

Algumas decisões que foram tomadas no início e durante o projeto refletiram na ferramenta resultante. Além do ImageJ foram estudadas outras ferramentas de visualização de imagens como Osirix [http://www.osirix-viewer.com] e MedicalStudio [Trevisan, 2007]. Ao início do projeto, foi identificada a necessidade de padronização de imagens, pois a ferramenta poderá ser utilizada em larga escala em imagens obtidas dos diversos centros de realização de exames (no escopo do Projeto ELSA), e a utilização de diversos tipos de imagens, em formatos diferentes dificultaria a intercomunicação entre esses centros. Para isso foi selecionada uma ferramenta base que fornecesse suporte ao padrão DICOM [ISO 12052]. Outro fator considerado decisivo foi o conhecimento e experiência prévios com a linguagem Java, além do fato de esta linguagem facilitar a portabilidade da ferramenta para diversos sistemas operacionais, sem a necessidade de modificação de código ou bibliotecas. O menu modificado do ImageJ para Retinografia pode ser visto na figura 3, com D para grade de Disco Óptico, M para Grade Macular, L para Grade de Lesões, F para o menu de filtros, C para calibragem da proporção micra/pixel e R para restaurar a imagem original.

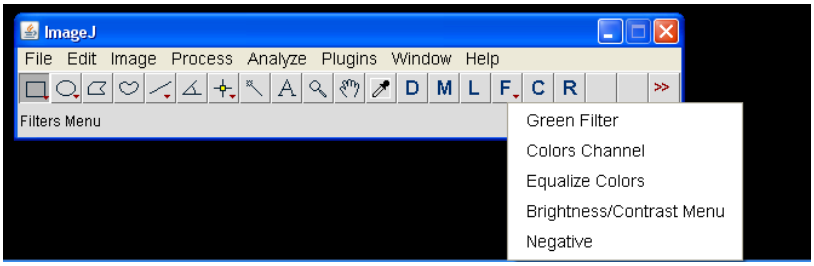


Figura 3. Menu modificado para o ImageJ para Retinografia

4.1.1. *Plugins*

Após a leitura da documentação do ImageJ, fica clara a sugestão de implementação de novas funcionalidades em forma de *plugins* que possam ser instalados na ferramenta sem a necessidade de modificação no seu núcleo. Essa abordagem foi utilizada nas fases iniciais do projeto sendo a ferramenta de filtro de tons de cinza da cor verde, o componente de comunicação com o servidor de imagens e a ferramenta de manipulação de canais de cores construídas dessa forma. Porém, esse método logo se provou muito limitado, e aos poucos as dependências com modificações do núcleo se tornaram visivelmente necessárias para a criação das demais funcionalidades.

4.1.2. Macro *Language*

Após perceber-se a dificuldade para criar as novas funcionalidades a partir de *plugins*, partiu-se para um segundo ponto sugerido pela documentação, que alterava apenas a interface do programa original, sem modificar o seu núcleo. Essa solução provou-se

extremamente complexa, pois cada modificação exigia uma grande alteração em diversos outros itens devido ao grande acoplamento da interface. Isso impactou negativamente, tornando maior o tempo para resposta do programa e aumentando o esforço para redesenhar qualquer item da interface. Em especial, destaca-se a dificuldade de criar objetos desenhados em tela, pois esses precisam implementar um objeto ROI. Esses objetos são descritos ao longo de várias classes de interface, com muita interdependência, tornando impraticável a criação e teste de qualquer uma das funcionalidades propostas. Por fim, ao analisar mais profundamente o código do ImageJ, foi possível perceber que a pseudolinguagem utilizada para desenhar e executar itens na barra de ferramentas poderia ser utilizada para implementar as ferramentas desejadas. A pseudolinguagem, chamada de "macro *language*" é interpretada pelo programa que gera então manipulações de imagens. Essa linguagem é descrita na documentação oficial como uma linguagem para criar chamadas às funções da ferramenta ou *plugins*. Porém, o que a documentação falha em citar, é que essa linguagem possui um grande poder de expressão para funções de desenho, e permite a adição de mais expressões através de modificações no *kernel* do ImageJ. Com isso em mente, foram desenvolvidas as novas funcionalidades como modificações da interface do programa. Dessa maneira, o acoplamento foi reduzido a ponto de serem necessárias apenas três adições ao *kernel* do ImageJ. O resultado da utilização desse método foi uma ferramenta de baixo acoplamento e facilmente modificável.

A descrição da pseudolinguagem pode ser encontrada no documento "functions.html" dentro do pacote "macros" do código fonte do ImageJ, e sua implementação no pacote "ij.macro" na classe "Functions.java". Essa classe implementa as interfaces "measurements" com definições de constantes para identificar medidas e "macroConstants", com a definição de constantes que identificam cada funcionalidade. Em "Functions.java" cada identificador de uma funcionalidade está atrelado a um método específico de acordo com sua função, sendo elas "doFunction", "getFunctionValue", "getStringFunction" e "getArrayFunction", representando funções que, respectivamente, executam ações, retornam valores numéricos, retornam objetos do tipo *string*, e retornam *arrays*. Nessa classe foram inseridas duas funções, "rotateX" e "rotateY", que retornam valores inteiros e recebem como parâmetros o ângulo de inclinação e módulo do raio em coordenadas polares. Essas funções retornam a projeção desse raio em coordenadas cartesianas no eixo correspondente a cada função, resultante da aplicação de seno para "rotateY" e cosseno para "rotateX". Essas extensões da pseudolinguagem são utilizadas para efetuar a rotação das divisões da grade macular. O diagrama de classe associado à "Functions.java" pode ser visto na figura 4. A barra de ferramentas foi criada em um arquivo texto, que é lido pelo ImageJ e executado de acordo com as funções declaradas em "Functions.java". Esse arquivo foi colocado na pasta "Macros" do diretório de instalação do ImageJ.

4.2. Passos do treinamento

A seguir é apresentado o passo-a-passo utilizado para o treinamento dos usuários, neste caso, oftalmologistas ou estudantes de medicina.

Os usuários foram instruídos a escolher a imagem que se deseja trabalhar, acessando o menu de importação de imagens a partir de arquivos locais (disco ou mídia removível) e a partir de

um servidor de imagens médicas no padrão DICOM previamente configurado, onde era digitado o número identificador do participante. Após pesquisa de exames de retinografia desse participante, as imagens relacionadas ao exame foram resgatadas para a memória da estação de trabalho. Em seguida, os usuários foram instruídos a selecionar a Grade Macular para aplicar na imagem de retina centrada na mácula, e após clicar com o mouse no centro da mácula visualizou-se a caixa de entrada de dados para a digitação da inclinação da grade em graus. A inclinação padrão é zero, porém dependendo da imagem utilizada é alterada para compensar a torção do globo ocular, presente na imagem. A grade deve tangenciar o disco óptico.

Os usuários em seguida utilizam a ferramenta de *zoom* para buscar anomalias ou indícios de problemas presentes na imagem, como lesões ou alterações na pigmentação da retina. No passo seguinte os usuários foram apresentados ao filtro de equalização de cor, para facilitar a busca de alterações na pigmentação da retina e ignorar possíveis artefatos de câmera que poderiam ser confundidos com lesões. A imagem original foi então restaurada (removendo os filtros e grades aplicados). Após, era indicada a utilização do filtro inversão de cores, que facilita a visualização de acúmulo de lipídios. A imagem foi novamente restaurada, e aplicava-se o filtro de cores, que permite a remoção de espectros de uma ou mais cores (vermelho, verde e azul), e o menu de contraste e brilho, que permite aumentar e diminuir estes valores para compensar fotos saturadas em vermelho ou com baixa iluminação (problemas causados por uso demasiado ou reduzido de *flash* da câmera do retinógrafo). Nesse ponto a imagem era novamente restaurada e após magnificação de área que apresentou problemas, apresentava-se a grade de lesão como ferramenta para medição de diâmetro de pontos de interesse (lesões, drusas, acúmulo de lipídios, derrames, etc). A ferramenta permite a entrada manual de diâmetro em micra ou a seleção de valores pré-determinados. Essas medidas são anotadas em formulário específico que é encaminhado ao oftalmologista para avaliação mais detalhada de risco à visão. O próximo filtro utilizado era o de tons de cinza do canal de cor verde, que retorna uma imagem muito nítida para visualização da espessura dos vasos presentes na retina, que permite ao usuário avaliar cruzamentos de vasos e artérias na retina. Esses tipos de alterações podem indicar problemas de hipertensão, que pode vir a causar derrames na retina, prejudicando a visão, além de servir como alerta para possíveis complicações cardiovasculares gerais. O mesmo procedimento foi então realizado para a imagem centrada no disco óptico, porém substituindo a utilização da grade macular pela grade de disco óptico, clicando sobre o centro do disco, para classificação de lesões e sinais de alerta próximos. Tornou-se a utilizar a Grade de Lesões, dessa vez para medição da escavação presente no disco óptico da imagem original. O disco óptico é composto por diversos feixes de nervos que funcionam analogamente à fibra ótica, transmitindo ao cérebro informações decorrentes de sinais luminosos captados na retina. Ao longo do tempo, as fibras nervosas atrofiam, causando uma impressão de escavação acentuada no disco óptico, característica do glaucoma. Sabendo que o diâmetro do disco óptico possui aproximadamente 1800 micra, aplica-se a grade de Lesão sobre a escavação do disco óptico com 1200 micra de diâmetro. Se a escavação ultrapassar a grade, o glaucoma estará assinalado, e o paciente é então encaminhado para tratamento adequado.

5. Avaliação da Ferramenta por Usuários

Após o treinamento, foram fornecidos aos usuários conjuntos de quatro fotos de retinografia, e foi solicitado que cada um utilizasse as ferramentas disponíveis para detecção de lesões e anormalidades como pode ser visto na figura 4. Esse procedimento foi gravado em vídeo, e ao mesmo tempo foi gerado um *log* pelo programa. O teste de utilização foi realizado por estudantes de graduação em medicina, estudantes de pós-graduação e um professor, todos com especialização em oftalmologia, sendo o professor, especialista em exame de retinografia. Os participantes são oriundos da faculdade de medicina da UFRGS, sendo as avaliações dos estudantes comparadas posteriormente à avaliação realizada pelo professor, para confirmar os problemas detectados e assinalar os não detectados com o auxílio da ferramenta.

Cada sessão de teste demorou em torno de 18 minutos, para a avaliação das quatro imagens, e ao final, cada usuário preencheu um questionário, onde fizeram uma avaliação pessoal sobre seus conhecimentos e uso da ferramenta em comparação com experiências prévias. Todas as imagens utilizadas foram obtidas utilizando um retinógrafo Canon RC1 – Não-Midriático (sem uso de colírio para a dilatação de pupila). Os resultados foram posteriormente comparados aos resultados obtidos utilizando a ferramenta ImageSpectrum Review da Canon Medical Systems.

Todos os usuários demonstraram satisfação e maior conforto na utilização da ferramenta, passando a utilizá-la com confiança após um breve treinamento que durou em torno de 10 minutos. Os resultados obtidos com o uso da ferramenta foram muito próximos aos do especialista, com diversas estruturas, artefatos e lesões encontradas em comum e com as mesmas ferramentas. Os resultados discordantes surgiram por dúvidas em relação à classificação de problemas ou foram atribuídos a pouca experiência de alguns usuários na avaliação retinografias, que pode ser aprimorada com maior treinamento. Em comparação com o ImageSpectrum Review, o especialista avaliado durante esse trabalho obteve um tempo médio de 20 minutos para cada conjunto de quatro fotos, e os estudantes obtiveram um tempo médio de 26 minutos. Já com a utilização da ferramenta ImageJ para Retinografia foi possível reduzir o tempo do especialista para 13 minutos (um ganho de 35%), e em relação aos estudantes foi obtida uma média de tempo de 16 minutos (ganho de 20% em relação ao especialista utilizando a ferramenta comercial).

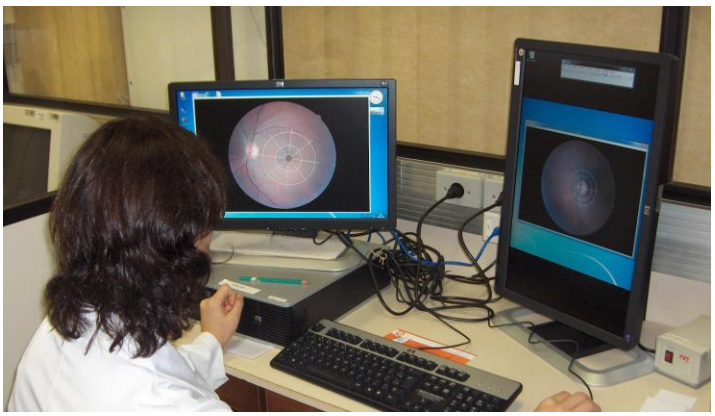


Figura 4. Usuário avaliando um exame de retinografia.

6. Conclusões

As modificações à ferramenta ImageJ foram desenvolvidas no contexto do Centro de Dados Elsa-RS, na UFRGS, e implantada em dois computadores de um laboratório do Centro de Leitura de Retinografia do projeto Elsa, localizado no Hospital de Clínicas de Porto Alegre, e está em uso para a avaliação de exames de retinografia desde o dia quinze de junho de 2010. Posteriormente a ferramenta foi instalada em mais três computadores no Centro de Pesquisas Clínicas anexo ao hospital.

Foi atingido o objetivo principal desse trabalho, que era a redução do tempo para a avaliação de exames de retinografia através de uma ferramenta de baixo custo, uso simples e ágil, que exigiu treinamento de curta duração, mesmo de usuários não especialistas, e manteve a eficiência na identificação de anormalidades da retina. Foram levantadas questões de interação que poderiam tornar o trabalho ainda mais ágil sem perda de qualidade nas avaliações, como por exemplo, a possibilidade de modificar a rotação da Grade Macular após a fixação da mesma, e a possibilidade de mesclar o menu de brilho e contraste com o menu de cores. Essas questões estão sendo estudadas, assim como propostas de novos métodos de interação com a ferramenta, para serem agregados em trabalhos futuros.

7. Referências

Wang, J. J.; Foran, S.; Smith, W.; Mitchell, P. Risk of Age-Related Macular Degeneration in Eyes With Macular Drusen or Hyperpigmentation. **Archives of Ophtalmology – American Medical Association**, Chicago, Illinois, USA. Arch Ophtalmol. 2003;121:658-663, 2003. Disponível em: http://archopht.ama-assn.org/cgi/reprint/121/5/658.pdf. Acesso em: jun. 2010.

DATASUS. Produção Ambulatorial do SUS – Brasil – Ministério da Saúde/DATASUS – Departamento de Informática do SUS. Brasília, Distrito Federal, Brasil, 2010. Disponível em: http://tabnet.datasus.gov.br/cgi/deftohtm.exe?sia/cnv/qgbr.def. Acesso em: jun. 2010.

imageSpectrum Review – Canon Medical Systems, Lake Success, New York, USA, 2009. Disponível em: http://www.usa.canon.com/html/industrial_medeq/ispectrum.html. Acesso em mai. 2009.

Trevisan, D. G.; Nicolas, V.; Macq, V.; Nedel, L. P. – MedicalStudio: a medical component-based framework. VII Workshop de Informática Médica (WIM, 2007), p. 86-95.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. **ISO 12052-2006**: Health informatics – Digital imaging and communication in medicine (DICOM). [S.l.:s.n].

Klein R., Davis M.D., Magli Y.L. *et al.*, The Wisconsin age-related maculopathy grading system. Ophthalmology 98 (1991), pp.1128-1134.

Rasband, W.S., ImageJ, U.S. National Institute of Health, Bethesda, Maryland, USA, http://imagej.nih.gov/ij/, 1997-2011.

Detecção automática de Pontos Cefalométricos a partir da análise do contorno anatômico.

Mauricio Cunha Escarpinati[1], Michele Fúlvia Angelo[2], Lucas Lima Batista[2], Luiz Bernardo Souza e Souza[2], Maiane Barbosa dos Santos Souza[2], Fernando de Menezes Oliveira[2]

[1]Faculdade de Computação – Universidade Federal de Uberlândia (UFU)
Uberlândia – MG – Brasil

[2]Departamento de Tecnologia – Universidade Estadual de Feira de Santana (UEFS)
Feira de Santana, BA - Brasil

mauricio@facom.ufu.br, m.fulvia@terra.com.br

Abstract. *This paper describes the introductory phase of a study that aims to develop mathematical models and computational algorithms for, from the analysis of anatomical contours, perform the automatic detection of Cephalometric Landmarks. This search is justified by the attempt to eliminate subjectivity in the process of the cephalometric analysis. As preliminary results were modeled algorithms to perform detection of eleven different Cephalometric Landmarks, nine of them were between good and satisfactory results and only two of them have problems in the detection process.*

Resumo. *Este artigo descreve a etapa introdutória de um trabalho que visa o desenvolvimento de modelos matemáticos e algoritmos computacionais para, a partir da análise de contornos anatômicos, realizar a detecção automática de Pontos Cefalométricos. Tal busca se justifica pela tentativa de eliminar a subjetividade no processo de realização dos exames de traçados e análises cefalométricas. Como resultados preliminares foram modelados algoritmos para realizar a detecção de onze diferentes Pontos Cefalometricos, sendo que nove deles apresentaram resultados entre bons e satisfatórios e somente dois deles apresentam problemas no processo de detecção.*

1. Introdução

A Cefalometria Radiográfica é a parte da ciência odontológica que se dedica à mensuração de grandezas lineares e angulares, em radiografias da cabeça, a qual se constitui num dos elementos essenciais ao diagnóstico ortodôntico e à avaliação de padrões de crescimento facial [Pereira 1998]. Em 1931, Broadbent [Broadbent 1931] estabeleceu o uso da cefalometria como meio de diagnóstico moderno e desde então, este método tem sido utilizado como parte dos registros, para auxiliar o diagnóstico e o planejamento dos tratamentos ortodônticos. A abundância de informações radiográficas passou a ser organizada em análises cefalométricas. Esta tem sido realizadas por meio de traçados sobre as radiografias de crânio laterais, porém, a determinação dos pontos cefalométricos tem se mostrado não reprodutível quando comparados por diversos observadores. Essa não reprodutibilidade, segundo Houston [Houston 1982], é

conseqüência da falta de precisão na determinação dos pontos cefalométricos, bem como dos erros de medida por instrumentos manuais (régua, compasso, lápis). Tais erros podem mudar significativamente o diagnóstico e o planejamento do tratamento ortodôntico.

Inserido neste contexto, o presente trabalho tem por objetivo desenvolver métodos computacionais capazes de detectar de forma automática os pontos cefalométricos necessários à realização do exame de traçado cefalométrico. Para tanto, serão realizadas análises computacionais na representação discreta dos contornos anatômicos das estruturas morfológicas envolvidas no processo.

Neste ensaio serão apresentados resultados preliminares da localização de onze pontos cefalométricos a partir de análises matemáticas dos contornos anatômicos, ajustados manualmente com o auxílio de uma ferramenta computacional desenvolvida pelo grupo e apresentada em trabalhos anteriores [Bastos et al 2009].

2. Materiais e Métodos

As regras estabelecidas pela literatura [Ravelli, 2007] para determinação da posição dos pontos cefalométricos a partir dos contornos anatômicos se baseiam, em sua maioria, em um conjunto de regras lógicas e matemáticas sobre a conformação geométrica dos contornos anatômicos. Desta forma, foram desenvolvidas formulações matemáticas para reproduzir tais regras e determinar, como descrito na seção 2.1, a posição de onze Pontos Cefalométricos.

A formulação matemática desenvolvida permitiu a implementação de algoritmos que a partir da análise de representações vetoriais dos contornos anatômicos pudessem determinar as coordenadas de cada um dos Pontos Cefalométricos modelados que, então, puderam ser comparados com a marcação manual realizada por especialistas.

2.1 – Modelagem matemática da posição dos Pontos Cefalométricos

A seguir serão apresentados os onze Pontos Cefalométricos analisados, bem como a definição cefalométrica de suas localizações de acordo com [Ravelli, 2007] e a formulação matemática desenvolvida. Para uma melhor compreensão faz-se necessário algumas definições, nas formulações matemáticas desenvolvidas para a determinação das coordenadas (x_p, y_p) de cada um dos Pontos Cefalométricos aqui tratados será assumido que:

- $\boldsymbol{T}$ é o conjunto de todas as $\boldsymbol{n_s}$ coordenadas cartesianas na forma $\boldsymbol{P_i}(x_i, y_i)$, com $1 < i < n$, que compõem a representação vetorial da Estrutura Anatômica investigada em casa caso;
- Desta forma temos que para o estudo de cada ponto cefalométrico, assumiremos um $\boldsymbol{T}$ diferente formado pelo conjunto de coordenadas da representação vetorial da estrutura anatômica em análise.

Os pontos estudados até o momento foram: Ponto Sela (S), Ponto Orbitário (Or), Ponto Espinha Nasal Posterior (ENP), Ponto Espinha Nasal Anterior (ENA), Ponto Porio Mecânico (Po), Ponta do Nariz (PN), Ponto Fossa Pterigo Maxilar (Ptm), Ponto Mentoniano ou Mentual (Me), Ponto (D), Ponto Násio (N) e o Ponto Pterigóide (Pt)

[Ravelli 2007]. As formulações matemáticas para localização das coordenadas de cada um dos pontos descrito podem ser visualizadas na Tabela 1.

Tabela 1 – Formulação matemática da localização dos Pontos Cefalométricos

PONTOS	Formulação Matemática
S	$P_S = (x_p, y_p) \rightarrow x_p = \frac{1\sum_1^n x_i}{n},\ y_p = \frac{1\sum_1^n y_i}{n}$
Or	$P_{Or} = (x_p, y_p) \rightarrow (x_p, y_p) = (\forall (x_i, y_i) \in T \mid y_i = max(y_1, y_2, \dots, y_n))$
ENP	$P_{ENP} = (x_p, y_p) \rightarrow (x_p, y_p) = (\forall (x_i, y_i) \in T \mid x_i = min(x_1, x_2, \dots, x_n))$
ENA	$P_{ENA} = (x_p, y_p) \rightarrow (x_p, y_p) = (\forall (x_i, y_i) \in T \mid x_i = max(x_1, x_2, \dots, x_n))$
Po	$P_{Po} = (x_p, y_p) \rightarrow x_p = \frac{1\sum_1^n x_i}{n},\ y_p = min(y_1, y_2, \dots, y_n)$
PN	$P_{PN} = (x_p, y_p) \rightarrow (x_p, y_p) = (\forall (x_i, y_i) \in T \mid x_i = max(x_1, x_2, \dots, x_n))$
Ptm	$P_{Ptm} = (x_p, y_p) \rightarrow x_p = \frac{1\sum_1^n x_i}{n},\ y_p = \frac{1\sum_1^n y_i}{n}$
Me	$P_{Me} = (x_p, y_p) \rightarrow (x_p, y_p) = (\forall (x_i, y_i) \in T \mid y_i = max(y_1, y_2, \dots, y_n))$
D	$P_D = (x_p, y_p) \rightarrow x_p = \frac{1\sum_1^n x_i}{n},\ y_p = \frac{1\sum_1^n y_i}{n}$
N	$P_N = (x_p, y_p) \rightarrow (x_p, y_p) = (\forall (x_i, y_i) \in T \mid y_i = min(y_1, y_2, \dots, y_n))$
Pt	$P_{Pt} = (x_p, y_p) \rightarrow (x_p, y_p) = (\forall (x_i, y_i) \in T \mid y_i = min(y_1, y_2, \dots, y_n))$

3. Resultados e Conclusões

Para a análise dos resultados obtidos com os algoritmos implementados, foram analisados 10 casos reais. Primeiramente as imagens digitalizadas foram submetidas à ferramenta de traçado digital dos contornos anatômicos desenvolvida pelo grupo e apresentada em trabalhos anteriores [Bastos et al 2009]. O ajuste da representação vetorial dos contornos anatômicos foi realizado por especialistas da área de Radiologia Odontológica, em seguida, os mesmos especialistas, com o auxílio da ferramenta, demarcaram manualmente a posição dos pontos cefalométricos. Por último os contornos anatômicos foram analisados pelos algoritmos desenvolvidos e os resultados obtidos pelo sistema (posição dos onze Pontos Cefalométricos descritos na Tabela 1) foram confrontados com os resultados das análises dos especialistas.

Como resultado da análise, foi mensurada a distância em *mm* entre a posição de cada Ponto Cefalométrico indicada pelos especialistas e a posição dos mesmos pontos calculada pelo algoritmo. Na Tabela 2 é apresentado o erro obtido entre a marcação manual e a automática de cada um dos onze pontos nos 10 casos analisados. Neste trabalho tomou-se como erro a não reprodutibilidade da marcação manual por parte da ferramenta. Segundo El-Feghi [EL-Feghi 2008], o erro médio considerado pelos ortodontistas na reprodutibilidade da marcação dos pontos cefalométricos, possui a seguinte classificação: bom, para erro menor ou igual a 2mm e aceitável o erro entre 2mm e 4mm.

Embora, nos dez casos analisados os resultados obtidos na marcação dos pontos Ptm e Me estejam fora do aceitável, a reprodutibilidade na marcação dos demais Pontos Cefalométricos por parte do algoritmo se encontra dentro das categorias “bom” e

"aceitável", o que demonstra que o trabalho, ainda em desenvolvimento, apresenta resultados promissores. Investigações preliminares apontam que embora o algoritmo tenha errado na marcação da posição dos Pontos ***Ptm*** e ***Me***, este erro segue um padrão uniforme para todos os casos, o que pode indicar uma solução trivial para o problema.

Tabela 2 – Resultados obtidos com a ferramenta de marcação automática dos Pontos Cefalométricos

Erro em mm da marcação automática dos Pontos Cefalométricos tendo como parâmetro a marcação manual realizadas por especialistas												
Pontos	Casos Analisados										Média	D. Padrão
	Caso 1	Caso 2	Caso 3	Caso 4	Caso 5	Caso 6	Caso 7	Caso 8	Caso 9	Caso 10		
S	1,3	1,4	2,6	1,4	1,1	1,4	0,6	0,4	1,0	3,2	1,43	0,84
Or	0,7	0,3	1,1	0,0	0,3	0,3	1,9	0,3	0,0	3,2	0,81	1,01
N	0,0	0,4	0,4	0,6	0,3	0,6	1,0	0,4	0,7	1,4	0,60	0,39
ENP	0,3	0,3	0,0	0,3	0,7	0,6	0,4	0,3	0,3	1,4	0,47	0,38
ENA	0,6	0,3	0,0	0,3	0,7	0,3	0,3	0,3	0,3	0,0	0,32	0,22
Po	0,9	0,3	0,3	0,4	0,3	0,5	0,4	0,3	0,9	1,4	0,58	0,37
PN	2,1	2,2	0,4	3,1	3,1	2,6	0,9	0,9	0,3	1,4	1,72	1,06
Ptm	4,2	7,6	5,6	6,9	8,0	9,4	3,8	5,7	7,5	24,2	8,27	5,86
Pt	3,1	4,4	0,4	4,2	4,1	5,3	2,0	3,3	1,4	11,0	3,92	2,91
Me	1,3	1,6	7,6	0,9	1,6	2,0	4,1	3,1	2,5	16,5	4,12	4,76
D	8,6	7,1	8,2	9,4	6,9	7,6	7,4	7,4	8,4	24,1	9,50	5,18

Diante do apresentado, acredita-se que a modelagem e os algoritmos propostos poderão contribuir de maneira significativa para o processo de realização dos exames de traçados e análise cefalométrica. Como futuras tarefas, serão modelados mais pontos cefalométricos e estudadas soluções para o posicionamento automático das representações vetoriais dos contornos anatômicos, auxiliando de uma forma ainda mais eficaz o dia-a-dia do especialista na realização de tal exame e na redução da subjetividade do processo.

Referencias

Bastos, I.L.O, Batista, L.L., Marques, R.S., Labo, R., ÂNGELO, M. F., ESCARPINATI, M. C. Esquema CAD para realização de traçados e análises cefalométricas In: IX Workshop de Informática Médica, 2009, Bento Gonçalves – RS.

Broadbent B H. A new x-ray technique and its aplication to orthodontics. Angle Ortodont, 1931; 1(2): 45-66.

EL-Feghi I, Galhood M, Sid-Ahmed M, Ahmadi M. Automated 2-D Cephalometric Analysis of X-ray by Image Registration Approach based on Least Square Approximator. 30th Annual International IEEE EMBS Conference 2008; 3949-3952.

Houston WJB. A comparision of the reability of measurement of cephalometric radiographs by tracings and direct digitization. Swed Dent J. 1982; 15: 99-103.

Pereira CB, Mundostock CA, Berthold TB. Introdução à cefalometria radiográfica. 3 ed. São Paulo: Pancast; 1998.

RAVELLI, D. B. et al. Ortodontia análises cefalomátricas mais usuais ao seu alcance. 1. ed. São Paulo: Rima, 2007. ISBN 978-85-7656-105-7.

Estudo do Parâmetro Tamanho de *Motif* para a Classificação de Séries Temporais de ECG

André Gustavo Maletzke[1], Huei Diana Lee[1,2], Willian Zalewski[1], Jefferson Tales Oliva[1], Renato Bobsin Machado[1,3], Cláudio Saddy Rodrigues Coy[3], João José Fagundes[3], Feng Chung Wu[1,2,3]

[1]Centro de Engenharias e Ciências Exatas – Universidade Estadual do Oeste do Paraná
Laboratório de Bioinformática – LABI
Caixa Postal 39, 85856-970 – Foz do Iguaçu, PR, Brasil

[2]Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo
Laboratório de Inteligência Computacional – LABIC
Caixa Postal 668, 13560-970 – São Carlos, SP, Brasil

[3]Faculdade de Ciências Médicas – Universidade Estadual de Campinas
Pós-Graduação em Ciências da Cirurgia
CEP 13083-887 – Campinas, SP, Brasil

{andregustavom,hueidianalee}@gmail.com

Abstract. *Currently, there is a growing interest in different areas for the analysis of data that present temporal dependency. In the medical area, information of this kind is daily recorded, however, only a small portion is analyzed due to lack of methods and tools. In this sense, the identification of morphological patterns (motifs) in time series is an important tool for analyzing such data. In this work, we present an initial study about the influence of the parameter size of the motif considering the classification task, applied to medical temporal databases, such as the electrocardiogram tests. The results show a significant influence of the motif´s size in the classification process.*

Resumo. *Atualmente, existe um crescente interesse em distintas áreas pela análise de dados temporais. Na área médica são registradas diariamente informações com essa característica, no entanto, somente uma pequena parcela é analisada devido à carência de métodos e ferramentas. Nesse sentido, a identificação de padrões morfológicos (motifs) em séries temporais constitui uma ferramenta importante para a análise desses dados. Neste trabalho, é apresentado um estudo inicial sobre a influência do parâmetro tamanho de motif, por meio da tarefa de classificação, quando aplicado em bases de dados temporais da área médica, como exames de Eletrocardiograma. Observou-se influência significativa desse parâmetro em relação aos resultados de classificação.*

1. Introdução

O desenvolvimento de tecnologias computacionais ao longo dos anos tem aumentado de maneira acelerada a capacidade de adquirir e armazenar informações [Hilbert and López 2011]. No entanto, o poder de análise de informação não tem seguido o mesmo ritmo de crescimento. Esse fato pode ser observado em distintas áreas

como na medicina, na qual, diariamente, inúmeros equipamentos médicos capturam informações sobre variáveis físicas, químicas e biológicas que descrevem o quadro clínico de pacientes. Porém, somente uma pequena parcela dessas informações é armazenada para posterior análise de um determinado momento ou ao longo do tempo, seja de modo manual ou por meio de ferramentas computacionais.

Um exemplo são os exames de Eletrocardiograma – ECG –, os quais consistem na monitoração ao longo do tempo da variação dos potenciais elétricos gerados pela atividade cardíaca e são de fundamental importância para o diagnóstico de um grande número de cardiopatias e outros distúrbios [Thanapatay et al. 2010].

De acordo com a Organização Mundial da Saúde[1] as doenças cardiovasculares são a principal causa de morte no mundo, representando cerca 29% dos casos. Segundo o Ministério da Saúde, as doenças cardiovasculares também constituem uma das principais causas de morte no Brasil, totalizando 32% dos óbtidos bem definidos, sendo que esses índices atingem estratos populacionais mais jovens de maneira mais acentuada quando comparados a outros países como Estados Unidos, Japão e Europa Ocidental. Apenas em 2005, 22% dos gastos com internação hospitalar (exceto partos) estão ligados a doenças cardiovasculares seguido por doenças respiratórias crônicas (15%) e neoplasias (11%) [Ministério da Saúde 2009], estatísticas que ressaltam a necessidade de políticas públicas mais eficientes para a prevenção e o diagnóstico dessa enfermidade.

Nesse contexto, métodos computacionais podem prover suporte à análise de dados obtidos ao longo do tempo como exames de ECG podendo auxiliar na identificação de padrões importantes de modo a contribuir na realização de diagnósticos precoces e com maior precisão. Todavia, grande parte dos métodos computacionais disponíveis aos especialistas são destinados à análise de dados não temporais. Desse modo, é necessário que técnicas específicas sejam desenvolvidas e aplicadas a dados com essa característica.

Dentre as técnicas existentes, a identificação de padrões morfológicos, denominados de *motifs*, em Séries Temporais – ST – tem sido utilizada em distintas áreas por meio do apoio a tarefas como descrição de dados, classificação, predição, entre outras [Mitchell 1997, Monard and Baranauskas 2003]. Assim sendo, a identificação de *motifs* em exames com coleta de dados ao longo do tempo constitui uma ferramenta importante podendo contribuir significativamente no estudo desses dados. No entanto, a definição do parâmetro tamanho de *motif* é um processo complexo devido ao grande número de possibilidades que podem ser consideradas para cada problema e está diretamente relacionada ao fracasso ou ao sucesso da análise que se pretende realizar. Portanto, o estudo e o entendimento desse parâmetro é uma tarefa de interesse e necessária para a utilização de *motifs* em problemas reais.

A identificação de *motifs* é uma tarefa custosa computacionalmente, podendo se tornar inviável em situações nas quais as ST possuem número elevado de observações, pois o método tradicional apresenta complexidade quadrática em relação ao número de observações da ST [Maletzke and Batista 2010]. Neste trabalho, é utilizado um método probabilístico baseado em [Chiu et al. 2003] e adaptado em [Maletzke 2009] para a construção de tabelas atributo-valor para posterior aplicação de métodos de extração de conhecimento. Embora apresente um comportamento probabilístico, em [Maletzke et al. 2008]

[1] `http://www.who.int`

é apresentada uma avaliação experimental ilustrando a eficiência do método.

Este trabalho faz parte do projeto de Análise Inteligente de Dados desenvolvido mediante uma parceria entre o Laboratório de Bioinformática – LABI – da Universidade Estadual do Oeste do Paraná – UNIOESTE/Foz do Iguaçu, o Serviço de Coloproctologia da Faculdade de Ciências Médicas da Universidade Estadual de Campinas – UNICAMP/Campinas, o Laboratório de Inteligência Computacional – LABIC – do Instituto de Ciências Matemáticas e de Computação – ICMC – da Universidade de São Paulo – USP/São Carlos e o Grupo Interdisciplinar em Mineração de Dados e Aplicações – GIMDA – da Universidade Federal do ABC – UFABC, e tem como objetivo realizar um estudo inicial referente à influência do parâmetro tamanho de *motif* aplicado a base de dados temporais de exames médicos, com um estudo de caso de ECG.

O restante deste trabalho está organizado da seguinte maneira: nas Seções 2 e 3 são apresentados os trabalhos relacionados e o material e método, respectivamente. Na Seção 4 são apresentados e discutidos os resultados e na Seção 5 são apresentadas as conclusões e os trabalhos futuros.

2. Trabalhos Relacionados

O desenvolvimento de métodos para auxiliar na análise de dados oriundos de exames de ECG tem sido tema de estudo de inúmeros trabalhos, sendo que várias abordagens foram propostas envolvendo problemas como predição, descrição/sumarização, classificação e visualização de dados de ECG. Em relação ao problema de classificação grande parte dos trabalhos destina-se à extração de características relevantes para posterior indução de classificadores baseados, principalmente, em redes neuronais artificiais [Neagoe et al. 2003, Thanapatay et al. 2010, Mar et al. 2011].

Em [Yu and Chen 2007] os exames de ECG são representados em função dos coeficientes obtidos a partir da transformada discreta de *wavelet* e utilizados como dados de entrada para construção de uma rede neuronal artificial. Os resultados são promissores, no entanto, a inteligibilidade dos modelos construídos é bastante complexa e pouco intuitiva, prejudicando a utilização do conhecimento por parte dos especialistas.

Em [Jovic and Bogunovic 2010] é realizada a extração de características baseadas em medidas estatísticas e em medidas como dimensão de correlação e entropia para indução de distintos classificadores dentre os quais redes neuronais artificiais, árvores de decisão e máquinas de vetor e suporte. A indução de árvores de decisão apresentou resultados significativos e promissores, permitindo que o conhecimento gerado fosse analisado por meio das regras geradas. Embora os resultados tenham sido promissores as características utilizadas apresentam baixa inteligibilidade.

A abordagem de *ensemble* para combinar vários classificadores neuronais em um sistema eficiente para a classificação de exames de ECG é proposta em [Osowski et al. 2011], sendo que a combinação dos classificadores é realizada por meio da utilização de algoritmos genéticos. Os resultados da aplicação dessa abordagem foram superiores quando comparados à abordagem sem *ensemble*.

De modo geral, a maioria dos trabalhos combinam extração de características e diferentes algoritmos de aprendizado de máquina para aumentar as taxas de classificação de problemas envolvendo dados de ECG. Nesse sentido, muitos trabalhos tem alcançados

resultados importantes e relevantes para a área, no entanto, pouco esforço tem sido destinado à construção de classificadores inteligíveis permitindo e apoiando a especialistas no estudo e entendimento de distúrbios cardiovasculares.

3. Material e Método

Nesta seção são descritos o método utilizado para a identificação de padrões morfológicos, o conjunto de dados e o método aplicado para avaliar o parâmetro tamanho de *motif*.

3.1. Descrição do Conjunto de Dados

O Eletrocardiograma é um exame que tem por objetivo descrever ao longo do tempo os fenômenos elétricos relacionados à atividade cardíaca por meio de eletrodos pré-posicionados no corpo. Esse exame quando realizado de modo completo é constituído por doze eletrodos, no entanto, aparelhos que realizam a análise de maneira automática, frequentemente utilizam somente uma parte desses eletrodos [Olszewski 2001]. O conjunto de dados selecionado foi obtido do repositório de dados da UCR *Time Series Classification/Clustering*[2] e está composto por 200 exames de ECG de diferentes pacientes. Na Tabela 1 é apresentada uma breve descrição a respeito do conjunto de dados, na qual as colunas 2 e 3 representam o quantidade de exemplos, *i.e.*, a número de exames, e o tamanho de cada exemplo, respectivamente. As colunas 4 e 5 referem-se ao número e distribuição de classes e as colunas 6 e 7 ao erro da classe majoritária e a indicação da existência ou não de valores desconhecidos nos dados.

Tabela 1. Características do conjunto de dados de ECG

Conjunto de dados	# Ex.	#Observações da ST	Classes	% Classe	Erro majoritário	Valores desconhecidos
ECG	200	96	1 2	66,5% 33,5%	33,5%	Nenhum

Os exames de ECG que constituem o conjunto de dados referem-se a pacientes que apresentaram sinal clínico de taquicardia supra-ventricular (classe 2) e não supra-ventricular (classe 1), previamente analisados por especialistas do domínio. No gráfico da Figura 1(a) é apresentado um exame com sinal clínico de taquicardia não supra-ventricular e no gráfico da Figura 1(b)com sinal clínico de taquicardia supra-ventricular.

3.2. Método para a Identificação de Padrões Morfológicos (*motifs*)

Como mencionado, a identificação de *motifs* é uma tarefa custosa computacionalmente podendo apresentar, no pior caso, uma complexidade de $O(m^2)$, em que m é o tamanho da ST. Desse modo, buscando-se reduzir a complexidade computacional foi utilizado um método baseado na abordagem desenvolvida em [Chiu et al. 2003] e adaptada em [Maletzke 2009] para aplicação em ST envolvendo problemas de classificação. O método utilizado pode ser divido em três passos, apresentados a seguir.

3.2.1. Passo 1 – Construção da Matriz de Subsequências

O processo de construção da Matriz de Subsequências – MS – consiste em extrair todas as subsequências de tamanho n da ST, por meio do conceito de janela deslizante. Esses conceitos são apresentados na Definição 1 e Definição 2, respectivamente.

[2] `http://www.cs.ucr.edu/~eamonn/time_series_data/`

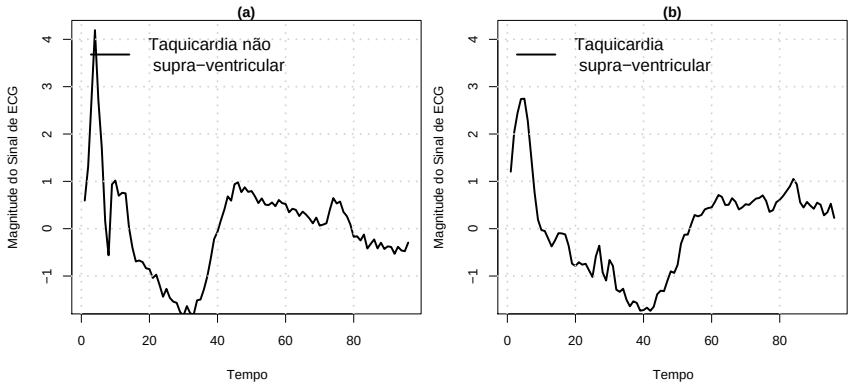


Figura 1. Exemplos de exames de ECG extraídos do conjunto de dados

Definição 1 *(Subsequência) [Chiu et al. 2003] Dada a ST Z de tamanho m, uma subsequência C de Z é uma amostra contínua de Z de tamanho n, sendo $n << m$. Portanto, $C = (z_p, \ldots, z_{p+n-1})$ para $1 \leq p \leq m - n + 1$.*

Definição 2 *(Janela Deslizante) [Maletzke 2009] consiste em extrair todas as subsequências de tamanho n de uma ST Z de tamanho m, ou seja, como resultado obtém-se as subsequências $(z_1, \ldots, z_n), (z_2, \ldots, z_{n+1}), \ldots, (z_i, \ldots, z_{n+i-1})$, para $1 \leq i \leq m-n+1$.*

Desse modo, uma vez construída a MS é aplicado o algoritmo *Random Projections* [Buhler and Tompa 2002] com o objetivo de identificar subsequências similares. Porém, esse algoritmo foi originalmente proposto para sequências de DNA e de proteínas e requer como entrada uma sequência de símbolos. Para tanto, no método utilizado, cada subsequência da MS é transformada para uma subsequência simbólica (cada ponto da subsequência correspondendo a um valor do alfabeto), de acordo com a Definição 3, por meio do método *Symbolic Aggregate aproXimation* – SAX – [Lin et al. 2002].

Definição 3 *(Série Temporal Simbólica) Uma ST $\hat{Z}$ de tamanho m' é uma coleção ordenada de valores $\hat{Z} = (\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_{m'})$ com $\hat{z}_t \in \Sigma$, onde Σ é um alfabeto finito.*

Uma variação que pode ser realizada, é utilizar o SAX, em uma etapa preliminar, para a redução de dimensionalidade por meio da definição de uma janela de redução, a qual pode ser ajustada em relação ao tamanho da sequência. Após, a sequência é discretizada utilizando um determinado alfabeto. No exemplo da Figura 2 (a) são extraídas subsequências de tamanho 16 da ST Z, com tamanho $m = 1000$, que após realizada a redução de dimensionalidade por meio de uma janela de redução de 25% e discretizadas mediante o alfabeto (a,b,c) formam subsequências simbólicas de tamanho $n_{sax} = 4$.

3.2.2. Passo 2 – Construção da Matriz de Colisão

A Matriz de Colisão – MC – é utilizada como ferramenta para apontar possíveis *motifs* existentes na ST. Essa matriz possui número de linhas e colunas iguais ao número de linhas da MS e, inicialmente, a MC é nula. Essa matriz é preenchida por meio de um processo iterativo, sendo que a cada iteração toda a MS é percorrida, de modo que a localização de cada subsequência seja inserida em uma estrutura *hash* – Figura 2 (c). Para isso, é utilizada uma máscara, gerada aleatoriamente, que indica quais colunas da MS devem ser utilizadas como parâmetro para a função de *hash*. Na Figura 2 (b) a máscara

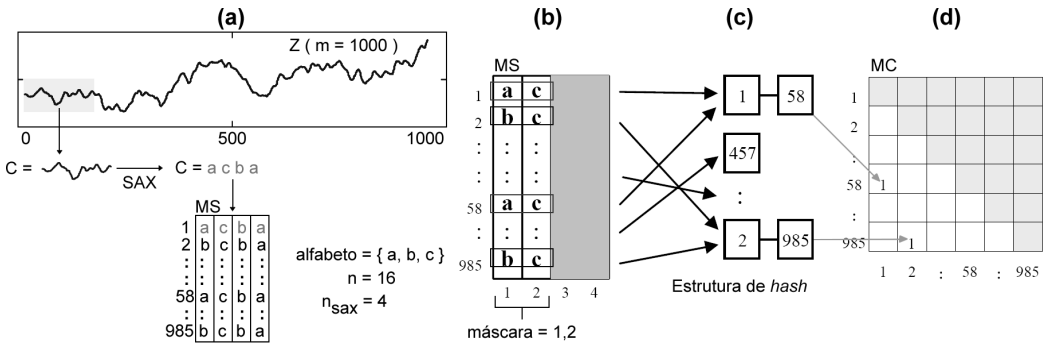


Figura 2. Processo de identificação de *motifs* (modificada de [Chiu et al. 2003])

é igual a $(1{,}2)$, portanto as subsequências das linhas $(1{,}58)$ e $(2, 985)$ da MS irão colidir, pois ambas possuem os mesmos valores nas colunas 1 e 2. Ao final de cada iteração é atualizada a MC por meio do incremento de um contador nas posições correspondentes às subsequências que colidiram – Figura 2 (d). O processo é repetido um número determinado de vezes, com o objetivo de tornar as máscaras mais representativas.

3.2.3. Passo 3 – Análise da Matriz de Colisão

Um valor alto em uma posição da MC é um indício da existência de um *motif*. Para que esse indício apontado pela MC seja comprovado verifica-se na MC a localização das subsequências que obtiveram maior número de colisões. Após, é calculada a distância entre essas subsequências utilizando uma medida de similaridade, mais especificamente neste trabalho a distância Euclidiana. Caso duas subsequências estejam dentro de uma distância r, essas podem ser consideradas como *motifs*. Neste método, o limiar r refere-se a uma percentagem que corresponde ao erro médio aceito para a diferença existente entre duas observações de duas subsequências distintas.

Para que seja determinado um valor em percentagem condizente com o valor de r é necessário que as subsequências sejam normalizadas para o intervalo 0 e 1 para posterior determinação da distância. Esse procedimento permite que seja desconsiderada a informação de amplitude dos dados em relação aos valores que compõem as sequências comparadas, apresentando resultados eficientes na comparação da morfologia. Desse modo, uma vez comprovada a similaridade morfológica é necessário verificar se as subsequências apresentam amplitudes de valores similares. Para isso, são resgatados os valores, sem normalização, das subsequências e determinada a área de cada subsequência, de modo que subsequências com áreas que diferem em no máximo r serão consideradas similares. O cálculo da área é realizado por meio da regra dos trapézios [Barroso et al. 1987].

Outras subsequências também podem estar dentro de r, porém podem não ter sido selecionadas devido à característica probabilística do método e, portanto, precisam ser adicionadas à condição de *motif*. Neste trabalho, é realizada uma busca sequencial a partir da subsequência definida como *motif* por toda a ST.

3.3. Avaliação Experimental

Para avaliar o parâmetro tamanho de *motif* envolvendo dados de exames de ECG foram extraídos *motifs* de distintos tamanhos mediante o método apresentado na Seção 3.2. A

definição do tamanho de *motifs* a serem extraídos foi realizada considerando uma percentagem em relação ao número total de observações de cada exame, o qual é equivalente a 96 observações. Desse modo, para realizar o processo de identificação de *motifs* foi utilizada a seguinte configuração experimental:

- **Tamanho de *motif*:** foram extraídos *motifs* nos intervalos de 5% a 50%, com incrementos de 5%, em relação ao número total de observações dos exames;
- **Limiar de similaridade:** foi utilizado um limiar r de 5%, isto é, as subsequências selecionadas como *motifs* diferem em no máximo 5%, considerando conjuntamente a combinação entre a distância Euclidiana e a área de cada subsequência;
- **Janela de redução de dimensionalidade:** não foi aplicado nenhum método de redução de dimensionalidade;
- **Tamanho da máscara:** utilizou-se uma máscara de tamanho dois corroborando com estudos realizados em [Chiu et al. 2003] com distintos conjuntos de dados;
- **Tamanho do alfabeto:** o alfabeto utilizado é composto por seis símbolos;
- **Número de iterações:** as iterações realizadas corresponderam a 50% de todas as possíveis combinações de máscaras.

Para cada variação do parâmetro tamanho de *motif* foi construída uma tabela atributo-valor, na qual cada *motif* de um mesmo tamanho, porém de morfologias distintas, constitui um atributo cujo valor é dado pela frequência desse *motif* em cada exame. Portanto, considerando o intervalo selecionado para se realizar a variação desse parâmetro foram obtidas 10 tabelas atributo-valor.

Para determinar a capacidade de descrição das tabelas atributo-valor obtidas da variação do parâmetro tamanho de *motif* foram realizadas a indução de classificadores, um para cada tabela atributo-valor, e mensurado o erro de classificação de cada classificador. Para a construção dos classificadores foi utilizado o algoritmo *J*48, por meio da ferramenta WEKA[3], para indução de árvores de decisão.

Para reduzir a possibilidade de que os resultados tenham sido gerados ao acaso foi utilizado o método de amostragem denominado de validação cruzada 5×2. Esse método consiste em particionar cinco vezes o conjunto de dados, considerando sementes diferentes, gerando um conjunto de teste e outro de treinamento a cada vez, de modo a preservarem as características do conjunto de dados original.

A análise dos resultados foi realizada por meio do teste estatístico ANOVA com pós-teste Tukey para comparações múltiplas [Motulsky 1995].

4. Resultados e Discussão

Na Tabela 2 são apresentadas as taxas médias de erro e os respectivos desvios padrão dos classificadores gerados pelo algoritmo *J*48 para cada variação do tamanho de *motif*.

Na análise dos resultados não foi observada diferença estatisticamente significativa entre os erros de classificação obtidos para cada variação de tamanho de *motif* (*p-valor* $> 0{,}05$), exceto para o tamanho de *motif* 5% que apresentou diferença estatisticamente significativa em relação aos demais tamanhos (*p-valor* $< 0{,}001$). Observou-se que o tamanho do *motif* influencia no desempenho do modelo e que, embora não tenha sido

[3] `http://www.cs.waikato.ac.nz/ml/weka/`

possível detectar diferença estatísticamente significativa entre os tamanhos 10% a 50%, houve uma têndência de maiores erros para os tamanhos 45% e 50%. Esse fato evidencia a importância de se investigar a existência de comportamentos locais nos dados.

Tabela 2. Taxas de erro médio e desvios padrão para cada tamanho de *motif*

	Tamanhos de *Motif* em Percentagem									
	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%
Erro Médio	9,61	25,08	27,51	25,34	24,82	23,09	25,09	25,70	28,00	28,31
DP	3,66	5,79	5,36	6,35	5,42	3,06	3,79	2,24	4,12	2,34

Muitas das análises em exames de ECG são realizadas de maneira global por meio, por exemplo, de características que buscam descrever cada exame. A análise de pequenos comportamentos pode auxiliar na identificação de anomalias ainda em estágios iniciais. Portanto, neste estudo os resultados apontaram uma maior relevância, considerando a classificação de exames de ECG, para padrões morfológicos de menor tamanho em relação ao número de observações dos exames. Nesse sentido, acredita-se que os resultados ainda podem ser melhorados por meio da busca de *motifs* de outros tamanhos, porém restritos a tamanhos menores corroborando com esses resultados iniciais apresentados.

Em relação ao método utilizado para a identificação de *motifs*, este apresenta melhor eficiência computacional quando comparado ao método denominado de força bruta [Lin et al. 2002], no entanto possui uma quantidade maior de parâmetros. Dentre esses parâmetros, o tamanho de *motif*, a medida de similaridade e o limiar de similaridade são considerados de maior importância, pois estão diretamente relacionados à aceitação de subsequências como sendo similares ou não. Para realizar a avaliação do parâmetro tamanho *motif* variou-se o tamanho desse parâmetro de maneira crescente buscando identificar padrões locais nos dados e não uma morfologia global que os descreva.

Já para o parâmetro limiar de similaridade optou-se por abrandar o nível de erro de 0% para 5% considerando que com um limiar de 0% não foi possível identificar padrões morfológicos nos dados que pudessem dar apoio à tarefa de classificação. Esse parâmetro é um dos mais complexos de ser determinado, pois está diretamente relacionado com a geração de falsos positivos e falsos negativos, sendo necessário um estudo mais detalhado a respeito em trabalhos futuros. Juntamente com esse parâmetro a medida de similaridade é outro parâmetro a ser ajustado, embora estudos realizados na literatura apontem a distância Euclidiana como sendo em média a que apresenta resultados melhores [Keogh and Kasetty 2002], existem outras métricas ou combinações de métricas que podem ser utilizadas para a comparação de subsequências.

Os demais parâmetros são utilizados para reduzir o espaço de busca e não possuem influência na aceitação de subsequências como *motifs*, *i.e.*, não estão diretamente relacionados à geração de falsos positivos. Desse modo, o número de iterações foi fixado em 50% do total de iterações possíveis, considerando que em estudos anteriores foi verificada a eficiência do método com valores menores [Maletzke 2009]. O tamanho do alfabeto foi determinado considerando estudos anteriores [Lin et al. 2002].

Em relação à amostragem de dados, foi escolhida a validação cruzada 5×2, pois o conjunto de dados possui poucos exemplos e apresenta classes desbalanceadas, podendo o resultado ser comprometido se utilizada a validação cruzada com dez partições.

É importante ressaltar que os resultados apresentados podem variar de acordo com

o algoritmo de indução de classificadores. Neste trabalho, foi realizada a indução de árvores de decisão considerando que o conhecimento gerado por algoritmos dessa classe apresentam maior inteligibilidade em relação a algoritmos de outros paradigmas, permitindo com que o raciocínio utilizado na classificação seja facilmente explicado em função dos *motifs* encontrados nos dados do exame [Mitchell 1997, Monard and Baranauskas 2003].

5. Conclusão e Trabalhos Futuros

De acordo com os resultados apresentados foi possível observar que a aplicação de *motifs* para a classificação de dados de ECG é promissora apresentando resultados interessantes em termos de taxa de erro e no nível de inteligibilidade dos modelos gerados. Neste trabalho, observou-se que *motifs* de menores tamanhos possuem maior capacidade de descrição quando utilizados em problemas de classificação de dados de ECG. Além disso, a definição do tamanho de *motif* não é uma tarefa trivial sendo necessário realizar análises mais completas para auxiliar na definição desse parâmetro.

Desse modo, espera-se que o estudo de padrões morfológicos locais em exames de ECG possa, futuramente, auxiliar a médicos e especialistas, bem como a centros de atendimento hospitalar, que não contam com especialistas da área, na identificação anormalidades com maior precisão e antecedência.

Como trabalhos futuros pretende-se explorar métodos para a identificação automática de *motifs*, bem como a avaliação de outros parâmetros relacionados ao processo de identificação de padrões morfológicos. Outro aspecto a ser estudado refere-se à construção de tabelas atributo-valor contendo *motifs* de tamanhos distintos de modo que os modelos construídos possam relacionar desde comportamentos locais até comportamentos mais gerais e também estudar o potencial existente na identificação de *motifs* quando aplicado a outros métodos de classificação. Ainda, um outro trabalho futuro será a aplicação dessa abordagem a dados médicos de outras especialidades, como dados provindos de exames de manometria anorretal.

Agradecimentos

A Fundação Araucária e a Universidade Estadual do Oeste do Paraná pelo auxílio na realização deste trabalho por meio do programa de bolsas de iniciação científica.

Referências

Barroso, L. C., de Araújo, M. M., Filho, F. F., de Carvalho, M. L. B., and Maia, M. L. (1987). *Cálculo Numérico*. Harbra.

Buhler, J. and Tompa, M. (2002). Finding motifs using random projections. *Journal of Computational Biology*, 9(2):225–242.

Chiu, B., Keogh, E., and Lonardi, S. (2003). Probabilistic discovery of time series motifs. In *Proceedings of the 9th International Conference on Knowledge Discovery and Data Mining*, pages 493–498, New York, USA.

Hilbert, M. and López, P. (2011). The world's technological capacity to store, communicate, and compute information. *Sciene Magazine*.

Jovic, A. and Bogunovic, N. (2010). Electrocardiogram analysis using a combination of statistical, geometric, and nonlinear heart rate variability features. *Artificial Intelligence in Medicine*, In Press, Corrected Proof.

Keogh, E. and Kasetty, S. (2002). On the need for time series data mining benchmarks: a survey and empirical demonstration. In *Proceedings of the 8th International Conference on Knowledge Discovery and Data Mining*, pages 102–110, New York, USA.

Lin, J., Keogh, E., Lonardi, S., and Patel, P. (2002). Finding motifs in time series. In *Proceedings of the Second Workshop on Temporal Data Mining at the Eighth International Conference on Knowledge Discovery and Data Mining*, pages 53–68, Canada.

Maletzke, A. G. (2009). Uma metodologia para extração de conhecimento em séries temporais por meio da identificação de motifs e da extração de características. Dissertação de mestrado, Universidade de São Paulo, São Carlos, São Paulo, Brasil.

Maletzke, A. G. and Batista, G. E. (2010). Mineração de dados temporais mediante a identificação de motifs e a extração de características. In *Anais do VII Best MSc Dissertation/PhD Thesis Contest - Joint Conference 2010*, volume 1, pages 1–12, São Bernardo do Campo - SP.

Maletzke, A. G., Batista, G. E., and Lee, H. D. (2008). Uma avaliação sobre a identificação de motifs em séries temporais. In *Anais do Congresso da Academia Trinacional de Ciências*, volume 1, pages 1–10, Foz do Iguaçu, Brasil.

Mar, T., Zaunseder, S., Cortes, J. P. M., Soria, M. L., and Poll, R. (2011). Optimization of ecg classification by means of feature selection. *Biomedical Engineering*, (99):1.

Ministério da Saúde (2009). Elsa brasil: maior estudo epidemiológico da américa latina. *Revista Saúde Pública*, 43(1).

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, Boston, USA.

Monard, M. C. and Baranauskas, J. A. (2003). *Sistemas Inteligentes: fundamentos e aplicações*, chapter Conceitos sobre Aprendizado de Máquina, pages 89–114. Editora Manole, Barueri, Brasil.

Motulsky, H. (1995). *Intuitive Biostatistics*. Oxford University Press, New York, USA.

Neagoe, V.-E., Iatan, I.-F., and Grunwald, S. (2003). A neuro-fuzzy approach to classification of ecg signals for ischemic heart disease diagnosis. In *Proceedings of the Annual Symposium Proceedings Archive at the Americam Medical Informatics Association*, pages 494–498.

Olszewski, R. T. (2001). *Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA.

Osowski, S., Siwek, K., and Siroic, R. (2011). Neural system for heartbeats recognition using genetically integrated ensemble of classifiers. *Computers in Biology and Medicine*, 41(3):173 – 180.

Thanapatay, D., Suwansaroj, C., and Thanawattano, C. (2010). Ecg beat classification method for ecg printout with principle components analysis and support vector machines. In *Proceedings of The International Conference on Electronics and Information Engineering*, pages 72–75.

Yu, S.-N. and Chen, Y.-H. (2007). Electrocardiogram beat classification based on wavelet transformation and probabilistic neural network. *Pattern Recognition Letters*, 28:1142–1150.

Uma Metodologia para Estruturação de Laudos Médicos usando Ontologias*

Oscar Picchi Netto[1,2], Alessandra Alaniz Macedo[1], Paulo Mazzoncini de Azevedo Marques[2], José Augusto Baranauskas[1]

[1]Departamento de Computação e Matemática — DCM
Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto — FFCLRP
Universidade de São Paulo — USP
Av. Bandeirantes, 3900 - Ribeirão Preto, SP - 14040-901 - Brasil

[2]Faculdade de Medicina de Ribeirão Preto — FMRP
Universidade de São Paulo — USP
Av. Bandeirantes, 3900 - Ribeirão Preto, SP - 14040-901 - Brasil

oscarpn@usp.br, ale.alaniz@usp.br, pmarques@fmrp.usp.br, augusto@usp.br

Abstract. *In computer systems aimed at generating and managing reports in radiology dominates the acquisition and storage of information in an open form of textual annotation. A major challenge faced in the development of these systems lies in the inadequacy of environments characterized by standardized interfaces that severely restrict the freedom of physicians in filling their reports, in contrast to the ineffectiveness of open text environments, which restrict a further analysis by computers. This paper aims to propose a methodology for structuring radiological reports from the Hospital das Clínicas da Faculdade de Medicina at Ribeirão Preto aiming in the future to build semi-automatically a report structure that allows knowledge extraction in order to facilitate the activities involved in teaching and researching in this hospital school. The methodology was evaluated on three bases each containing 5000 reports and the results compared with a standard ontology.*

Resumo. *Nos sistemas informatizados voltados à geração de laudos em radiologia predomina a aquisição e o armazenamento da informação na forma de anotação textual aberta. Um grande desafio enfrentado no desenvolvimento destes sistemas reside na inadequabilidade de ambientes caracterizados por interfaces de preenchimento padronizados e pré-definidos, que restringem fortemente a liberdade do médico na geração de seus relatos; contrastando com a ineficácia de outros ambientes que trabalham com textos abertos, restringindo fortemente as possibilidades de análise futura da informação. Este trabalho tem por objetivo propor uma metodologia de estruturação para a base de laudos radiológicos do Hospital das Clínicas da Faculdade de Medicina de Ribeirão Preto visando, em trabalhos futuros, construir de forma (semi-)automática um laudo estruturado que permita a extração de conhecimento para facilitar as atividades ligadas ao ensino e pesquisa deste hospital escola. A metodologia proposta foi avaliada em três bases contendo 5000 laudos cada uma e os resultados comparados com uma ontologia padrão.*

*Projeto de pesquisa realizado com apoio financeiro FAPESP e CNPq/FAPEAM — INCT Adapta.

1. Introdução

Com o avanço da tecnologia nos últimos anos ficou muito fácil armazenar grandes quantidades de dados. Na área médica, esse crescimento é notável dada toda variedade de informações de pacientes que se encontram em formulários médicos, processos laboratoriais e laudos médicos armazenados atualmente sob a forma digital. Especificamente, no Hospital das Clínicas da Faculdade de Medicina de Ribeirão Preto (HCFMRP) todos os dias centenas de exames radiológicos são realizados. Em muitos destes exames são geradas milhares de imagens, as quais são então utilizadas pelos especialistas médicos para gerar um laudo para cada exame no formato de texto livre. Tal laudo, junto com as imagens e informações complementares sobre o paciente (idade, sexo, exames realizados, etc), permite o diagnóstico médico dos pacientes. Portanto, laudos de radiologia contêm uma grande quantidade de informação que caracteriza a condição médica do paciente. Todavia, uma grande porcentagem desta informação não é estruturada, assumindo a forma de texto livre o que dificulta processos computacionais de busca, ordenação e análise.

Estudos na literatura mostram os benefícios potenciais de informações médicas estruturadas envolvendo práticas médicas, pesquisa e ensino. Na prática clínica, relatórios estruturados podem auxiliar na organização e melhoria da apresentação de registros médicos [Shortliffe and Hubbard 1989, Aberle et al. 1996]. Na pesquisa e ensino, relatórios estruturados podem melhorar significativamente a precisão e relevância em tarefas de recuperação de informação. Apenas dados estruturados viabilizam técnicas de modelagem de base de dados causal, espacial, temporal e evolucionária que estão em desenvolvimento nas áreas de informática médica e ciência da computação. Por exemplo, no estudo de [Honorato et al. 2009] é apresentada uma metodologia que pode ser aplicada automaticamente ou semi-automaticamente com a ajuda de especialistas do domínio e tem por objetivo realizar o mapeamento de documentos não estruturados para uma tabela atributo-valor. Os resultados encontrados fornecem um indicativo que a metodologia pode auxiliar na redução do tempo de atuação dos especialistas na análise de grandes quantidades de documentos não estruturados. Já no estudo de [Taira et al. 2001] é desenvolvida uma abordagem para um processador de linguagem natural que automaticamente estrutura informações médicas importantes que estão contidas em um documento de radiologia em formato de texto livre. Este sistema não necessita de nenhuma ajuda de especialistas no domínio, utilizando extensivamente métodos estatísticos e de Aprendizado de Máquina.

Neste contexto, quando as informações encontram-se em forma estruturada, como em uma base de dados, é possível realizar a busca por modelos para suporte à tomada de decisão médica por meio de algoritmos de Aprendizado de Máquina [Mitchell 1997]. Uma vez que os textos dos laudos não estão estruturados é necessário um pré-processamento dos textos. O pré-processamento deve ser adequado, de forma a permitir que os modelos extraídos representem adequadamente o conhecimento embutido nos textos. Em geral, o pré-processamento é uma etapa de um processo mais amplo, denominado Mineração de Textos e consiste em três etapas elementares [Feldman and Sanger 2006]: (i) pré-processamento, (ii) mineração e (iii) pós-processamento. Na etapa de pré-processamento ocorre a preparação dos dados (para o processo de mineração de textos) na qual o texto (formato não estruturado) é transformado em alguma representação estruturada. Existem diversas formas de representação sendo a mais utilizada o formato de uma tabela atributo-valor: em geral, as palavras existentes nos textos são transformadas em atributos

com sua frequência associada. Esta tabela é então utilizada na etapa de mineração, na qual, em geral, são utilizados algoritmos de Aprendizado de Máquina. No Aprendizado de Máquina utiliza-se inferência indutiva a partir de um conjunto de exemplos), induzindo um modelo sobre o conceito intrínseco nos dados. Na etapa de pós-processamento ocorre a interpretação dos resultados, que consiste na avaliação dos modelos extraídos. Assim como o armazenamento de grandes volumes de dados é de pouca valia, a menos que existam métodos computacionais adequados para analisá-los, também é pouco produtivo extrair modelos simbólicos que não causem surpresa ou representem conhecimento novo ou mesmo que sejam altamente redundantes para o especialista do domínio. É nesta etapa que ocorre a remoção de padrões irrelevantes ou redundantes e tradução de padrões úteis em termos inteligíveis pelos especialistas. Também é nessa etapa que ocorre o uso do conhecimento extraído, que consiste na incorporação do conhecimento ao domínio, seja tomando ações baseadas no conhecimento novo ou simplesmente documentando e relatando para as partes interessadas o conhecimento obtido, bem como remoção de conflitos potenciais com conhecimento previamente tido como correto.

Neste artigo é proposta uma metodologia de extração de termos em laudos médicos de forma estruturada. A proposta é que em trabalhos futuros e complementares seja possível construir de maneira (semi-)automática um laudo estruturado de forma a facilitar a extração de conhecimento, além de facilitar atividades ligadas ao ensino e pesquisa, uma vez que o estudo é em um hospital escola. A metodologia proposta foi avaliada em três bases laudos e os resultados comparados com uma ontologia padrão.

O restante deste trabalho está organizado da seguinte maneira: na Seção 2 é feita uma introdução sobre laudos médicos. Na Seção 3 é apresentada a metodologia proposta para a estruturação de laudos. Experimentos e resultados obtidos com uma base de laudos são apresentados na Seção 4 e, finalmente, na Seção 5 são apresentadas as conclusões.

2. Laudo Médico

O termo *laudo médico* é comumente utilizado para a interpretação de exame complementar ou resultado de perícia médica elaborado por médico. Na área médica, laudos são comumente descritos na forma textual, normalmente compostos por sentenças em que o médico descreve em linguagem natural as observações a respeito da saúde do paciente. Normalmente, o médico responsável por fazer o exame escreve suas suspeitas de acordo com o exame realizado e as imagens obtidas deste exame. Outro médico utilizando as mesmas imagens valida este laudo, confirmando assim o mesmo. O laudo então é levado ao médico que requisitou o exame e este, por meio do laudo do paciente e das informações previamente obtidas com o paciente, fornece o diagnóstico.

O HCFMRP possui um sistema de laudo digitalizado na realização de exames radiológicos, armazenando seus resultados na tabela Exame Radiológico do seu banco de dados. Nessa tabela existem diversos campos estruturados tais como código do paciente, número do exame, código da região anatômica, data da realização do laudo bem como campos no formato de texto livre tais como descrição e conclusão, dentre outros. São nestes campos expressos em linguagem natural que a mineração de textos mostra-se adequada para extrair ou estruturar o conhecimento, como o exemplo exibido na Figura 1. Para tanto, é necessário que os textos sejam representados em um formato estruturado que possa ser processado por algoritmos de extração de conhecimento. Uma forma diferente

Descrição: 'Controle pós-operatório de segmentectomia evidencia: Reduçao das dimensoes do lobo direito do fígado, com hipertrofia compensatória do lobo esquerdo. Baço pâncreas e rins com forma, contornos, dimensões e intensidade de sinal normais. Dilataçao de vias biliares intra-hepática e ductos hepáticos direito e esquerdo. O ducto hepático comum não é identificado e do ducto biliar comum (colédoco), visualiza-se apenas o terço distal que apresenta calibre normal. Na região do segmento IV junto à cúpula frênica, observa-se ductos biliares com dilatações saculares , com pequena coleção subcapsular com hipersinal em T2 (bilioma?). Vesícula biliar nao visualizada. Alças intestinais e gordura peritoneal e retroperitoneal sem alteração'

Conclusão: 'Dilatação de vias biliares intra hepáticas (Obstrução ao nível do hepático comum?), com formação de pequena coleção (bilioma) junto à cúpula frênica.'

Figura 1. Exemplo dos campos de *descrição* e *conclusão* de um laudo radiológico

daquela adotada neste estudo consistiria no mapeamento manual desses laudos em bases de dados estruturadas, o que apresenta-se como um método lento e com um determinado grau de subjetividade, uma vez que pode ser influenciado por fatores específicos do ser humano que realiza esta tarefa.

3. Metodologia Proposta

Existem várias maneiras de transformar textos em dados estruturados. Uma delas é transformá-los em uma representação atributo-valor utilizando a abordagem *bag of words*, na qual a frequência das palavras (termos), independente de seu contexto ou significado, são contadas. A ocorrência de palavras em sequência, em geral, pode conter mais informação do que palavras ocorrendo isoladamente. Portanto, ao criar atributos pela junção de duas ou mais palavras consecutivas, é esperado obter atributos com um maior poder de predição, compondo assim o n-grama, onde n representa o número de palavras que foram unidas para a geração de um atributo. Por exemplo, considerar as palavras *lobo* e *direito* individualmente pode agregar pouco conhecimento, pois *lobo* pode referir-se a um *animal* ou *região anatômica* e *direito* pode ser *uma faculdade de praticar algo* ou *oposto de esquerdo*, dentre outros significados. Todavia, o termo composto *lobo direito* pode agregar muito mais informação se o texto se refere a uma região anatômica do corpo humano.

A partir da contagem dos n-gramas é possível gerar uma tabela cujas entradas contêm informações relacionadas à frequência de cada n-grama. Neste estudo foi utilizada a ferramenta PreText [Soares et al. 2008] que permite a geração de n-gramas para qualquer valor de n. A partir dos conjuntos n-gramas é possível aplicar os demais passos do Algoritmo 1 para geração de uma ontologia.

Uma ontologia é especificada por uma coleção de termos (ou conceitos) e seus relacionamentos definindo uma ordem parcial, em geral, da forma *tipo-subtipo* [Gruber 1993, Sowa 2000]. Formalmente, dada uma linguagem lógica L, onde $L_p \subset L$ é o conjunto de símbolos predicados de L e $L_f \subset L$ é o conjunto de fórmulas bem formadas de L, uma ontologia é uma tupla (V, A) na qual o vocabulário $V \subset L_p$ e os axiomas $A \subset L_f$ [Heflin et al. 1999]. Taxonomias que especificam relacionamentos hierárquicos entre termos de um domínio específico do conhecimento humano — representados na forma de árvores ou grafos direcionados acíclicos — estão entre as ontologias mais utilizadas [Zhang et al. 2002]. Nesse caso, cada conceito tem a ele uma ontologia na forma de uma hierarquia (árvore), onde cada aresta da árvore representa um relacionamento entre entre dois nós. Os relacionamentos podem ser do tipo *é-um*, *é-parte-de* ou *é-um-processo-de*, dentre outros.

Algoritmo 1 Algoritmo para criação da ontologia a partir de laudos

Require: Documentos, uma coleção de documentos contendo laudos radiológicos
Região, uma região anatômica de interesse
N, número máximo de N-gramas a serem gerados (default 10)
Ensure: Ontologia, uma ontologia criada a partir dos Documentos
1: Docs_Selecionados ← Selecione(Documentos, Região)
2: LC ← ∅
3: $NDocs$ ← Número_de_Documentos(Docs_Selecionados)
4: (1-grama, 2-grama, ..., N-grama) ← PreText(Docs_Selecionados)
5: **for** i ← 1 **to** Número_de_Palavras(1-gram) **do**
6: **if** Frequência(palavra$_i$) $> \theta \times NDocs$ **then**
7: LC ← LC ∪ {palavra$_i$}
8: **end if**
9: **end for**
10: Ontologia ← LC
11: **for** $n \leftarrow 2$ **to** N **do**
12: **for** $i \leftarrow 1$ **to** Número_de_ Palavras(LC) **do**
13: **if** $palavra_i \subset n$-grama **then**
14: Associe $palavra_i$ ao nó correspondente na Ontologia
15: **end if**
16: **end for**
17: **end for**
18: **return** Ontologia

O Algoritmo 1 utiliza um conjunto de documentos e a região associada a esses documentos. Todas as palavras contidas no conjunto 1-grama são candidatas a estarem na LC (lista de candidatas). O algoritmo faz uso do parâmetro θ, onde $0 \leq \theta \leq 1$, explicado a seguir. Se $\theta = 0$ então todas as palavras, independente da sua frequência são adicionadas à LC; caso $\theta > 0$ as palavras do conjunto 1-grama com frequência maior que $\theta \times NDocs$, onde $NDocs$ é o número de documentos selecionados, serão adicionadas à LC (linhas 3 à 9). Esta estratégia foi adotada pois, em geral, n-gramas que apresentam uma maior frequência podem ser mais representativos para o aprendizado. Após isso, para todos os conjuntos n-gramas ($n = 2, 3, \ldots, N$) e para todas as palavras contidas na LC, o algoritmo verifica se a palavra$_i$ da LC está contida na lista de palavras do conjunto n-grama; em caso afirmativo associa a palavra$_i$ ao nó correspondente na Ontologia (linhas 10 à 17).

Por exemplo, suponha uma base de dados radiológicos hipotética contendo apenas exames de joelho. Primeiramente, os documentos são processados pelo PreText, gerando assim os conjuntos n-gramas. Neste exemplo, assuma que $N = 3$, ou seja, que serão gerados os conjuntos 1-grama, 2-gramas e 3-gramas. Agora suponha que no conjunto 1-grama aparece a palavra *joelho* e que a frequência dela seja maior do que $\theta \times NDocs$. Sendo assim *joelho* entra na LC. Assuma então que no conjunto 2-gramas aparecem as seguintes palavras *joelho_ligamento* e *joelho_menisco* e que no conjunto 3-gramas existam *joelho_ligamento_lateral*, *joelho_ligamento_posterior* e *joelho_ligamento_cruzado*. Então *joelho_ligamento* e *joelho_menisco* são colocados no sub-ramo cuja raiz é *joelho*. Analogamente, *joelho_ligamento_lateral*, *joelho_ligamento_posterior* e *joelho_ligamento_cruzado* são colocados no sub-ramo cuja raiz é *joelho_ligamento*. Com essas suposições é mostrada a ontologia gerada pelo Algoritmo 1 na Figura 2.

4. Experimentos, Resultados & Discussão

Para avaliar a metodologia proposta neste estudo foram utilizados laudos radiológicos do base de dados do HCFMRP, atualizada até a data de 05/02/2009. Os dados consistem em exames realizados pelo Centro de Ciências das Imagens e Física Médica e inseridos no

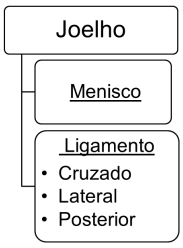


Figura 2. Exemplo de ontologia criada sobre laudos hipotéticos de joelho

sistema desde 1999 até fevereiro de 2009. A pesquisa foi aprovada pelo Comitê de Ética em Pesquisa do HCFMRP, sob processo número 10791/2007. Essa base possui 1.134.071 laudos, porém nem todos possuem todos os campos mencionados na Seção 2. Portanto foi necessário pré-processar esta base de dados para utilizar somente laudos completos; esta redução da base de dados foi de 570.641 laudos, restando assim 563.430 laudos completos para serem analisados. A base de laudos completos encontra-se sem nenhum pré-processamento adicional, sendo composta por todos os dados da tabela Exame Radiológico do banco de dados relacional do HCFMRP. Como é possível notar, no Algoritmo 1 é possível selecionar uma região anatômica específica, como joelho, pulmão, mama ou até mesmo região não tão específica como tórax e abdômen.

Portanto, com base no Algoritmo 1 foram criadas 3 ontologias, utilizando 3 bases de 5000 laudos cada, selecionados de forma aleatória dentre os 563.430 laudos completos: uma base mais específica (joelho), uma base intermediária (tórax) e uma base geral (utilizando todos os tipos de laudo). Em cada uma dessas bases foram realizados experimentos para a geração das ontologias, utilizando-se $\theta = 0,045$, valor definido de forma heurística. É importante salientar que a ferramenta PreText utilizada trabalha com 'radicais' (*stems*) das palavras. Assim sendo, nesta fase inicial da metodologia proposta, as ontologias aqui mostradas contém apenas os radicais encontrados pelo PreText.

De forma a avaliar a metodologia proposta seria possível (i) utilizar o auxílio de especialistas médicos no domínio da aplicação e/ou (ii) utilizar uma ontologia pré-estabelecida e confiável e procurar por partes que sejam comuns àquelas encontradas pela metodologia aqui proposta, dentre outras formas [Brank et al. 2007]. Neste estudo efetuou-se uma avaliação intermediária entre (i) e (ii), comparando-se manualmente os resultados obtidos com uma ontologia internacionalmente conhecida e validada, a saber RadLex[1], procurando por ramos que fossem comuns a ambas ontologias. Há, entretanto, o obstáculo para se fazer comparações pelo fato que a ontologia RadLex está escrita na língua inglesa, enquanto que os laudos utilizados neste estudo estão na língua portuguesa, o que impede a comparação automática de ontologias, por exemplo, pelo algoritmo proposto por [Brank et al. 2007]. Todavia, é importante salientar que é de igualmente importante a validação por especialistas, algo que poderá ser efetuado em trabalhos futuros.

Um resumo sobre os experimentos realizados pode ser encontrado na Tabela 1. Nesta tabela são mostrados os parâmetros utilizados bem como o número de palavras e de *stems* de cada base e o número de palavras no primeiro LC(1) e no segundo LC(2) níveis da ontologia. Um fato que merece atenção é que diante do grande número de palavras e

[1] http://radlex.org/viewer

radicais (*stems*) a metodologia proposta gerou poucos candidatos para o primeiros níveis da ontologia, o que é interessante do ponto de vista de estruturação do conhecimento. A seguir, são descritos os resultados em cada uma das 3 bases de laudos selecionados para estudo. As ontologias criadas foram visualizadas utilizando o software Protégé[2].

Tabela 1. Resumo sobre os experimentos

Base	θ	N	Documentos	Palavras	Stems	LC(1)	LC(2)
Joelho	$0,045$	10	5000	172.145	5.880	1	20
Tórax	$0,045$	10	5000	154.413	10.665	1	10
Geral	$0,045$	10	5000	312.008	17.514	2	29

4.1. Base Específica - Joelho

Neste experimento a base de laudos selecionada contém apenas laudos referentes a região anatômica do joelho. Após a aplicação do Algoritmo 1, obtém-se a ontologia mostrada à esquerda da Figura 3. Neste experimento utilizando uma base bem específica foi possível encontrar dois sub-ramos cuja a representação pode ser encontrada no RadLex de forma quase total, ou seja, quase todas as folhas selecionadas aparecem no RadLex. Os sub-ramos de *ligament* (radical de *ligamento*) e *metafis* (radical de *metáfise*) gerados na ontologia foram encontrados no RadLex. Nas Figuras 4(a) e 5(a) são mostrados os resultados encontrados pela metodologia aqui proposta; nas Figuras 4(b) e 5(b) são mostradas partes da ontologia que foram encontradas no RadLex; os nomes sublinhados aparecem na ontologia gerada pelo Algoritmo 1.

4.2. Base Intermediaria - Tórax

Neste experimento utilizou-se uma base de laudos intermediária, ou seja, selecionando uma região anatômica e, portanto, vários tipos de exames e, consequentemente, vários tipos de laudos, aumentando um pouco o escopo em relação ao experimento anterior. Observa-se que, apesar de ser uma base mais geral, o número de palavras diminiu; isso ocorreu pois alguns laudos de joelho são extremamente detalhados, enquanto que, ao que tudo indica, alguns laudos da região torácica não parecem apresentar um detalhamento tão profundo. Entretanto, isso requer a análise por especialistas no domínio para sua efetiva comprovação. Outro fato que pode ser observado é que o número de *stems* aumentou, mostrando que são necessários mais radicais de palavras para descrever uma região anatômica maior. Já na LC ocorreu a diminuição de 45 (joelho) para 24 (tórax). A ontologia gerada para esta base é mostrada no centro da Figura 3.

Para esta base de laudos não houve sub-ramos que possam ser encontrados no RadLex. Isso pode ter ocorrido por 2 motivos: o primeiro é que a base utilizada pode não representar uma amostra significativa da base original; o segundo motivo pode ser devido ao fato de que o algoritmo proposto não conseguiu trabalhar com uma LC tão reduzida a ponto de gerar algo que poderia ser encontrado no RadLex. Apesar disso, um dos sub-ramos que pode ser destacado por apresentar alguns resultados possivelmente relevantes. O radical *process* possui quatro filhos, todos esses filhos de *process* podem ser reconhecidos como processos que ocorrem no corpo humano como, por exemplo, o filho *inflamatori* que representa o *processo inflamatório*. Depois de uma busca no RadLex sobre a palavra inflamatório, observou-se que este não possui nenhuma menção a este

[2]http://protege.stanford.edu/



Figura 3. Ontologia criada sobre laudos de joelho (esquerda), laudos de tórax (centro) e laudos gerais (direita)

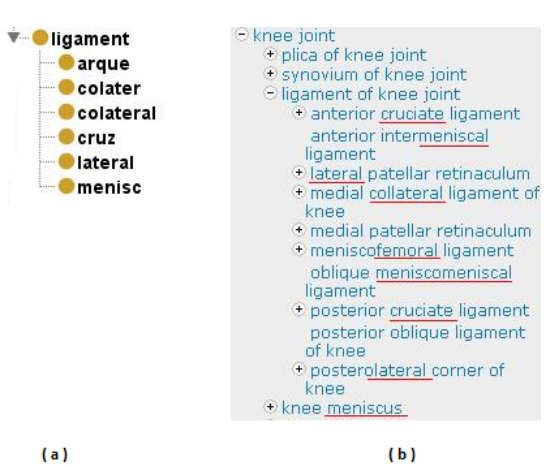


Figura 4. Subramo da ontologia Ligament (a) e sua contrapartida no RadLex (b)

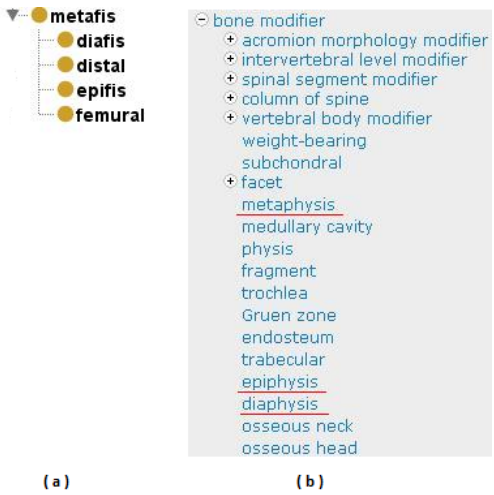


Figura 5. Subramo da ontologia Metafis (a) e sua contrapartida no RadLex (b)

processo. Nesta base, não foram encontrados resultados que possam ser achados em contrapartida no RadLex.

4.3. Base Geral

Neste experimento não houve nenhum tipo de restrição quanto a região anatômica. Neste caso, a base possui 312.008 palavras e 17.514 *stems*. É possível perceber que estes números são maiores que os das bases utilizadas anteriormente. A LC gerada para esta base possui 61 palavras. A ontologia gerada é mostrada à direita da Figura 3. Apesar de possuir uma LC maior que as das outras bases, a ontologia gerou um resultado que pode ser interessante: o sub-ramo *estenos*, radical da palavra *estenose*. Apesar de o termo *estenose* poder ser encontrado no RadLex, os filhos gerados pela ontologia não são encontrados no mesmo. Na Figura 6(a) e 6(b) são mostrados os dois sub-ramos, o da ontologia gerada pelo Algoritmo 1 e a ontologia do RadLex, respectivamente. Apesar de ser uma base mais geral que a base de tórax, esta base apresentou um dos sub-ramos presentes no RadLex, portanto, é possível observar que apesar de utilizar uma base geral o algoritmo é capaz de encontrar resultados que podem ser relevantes.

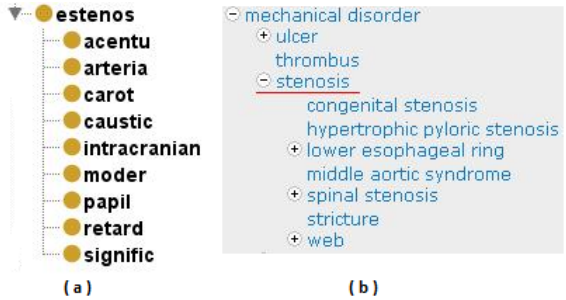


Figura 6. Subramo da ontologia *estenose*(a) e sua contrapartida no RadLex(b)

5. Conclusão

Nos sistemas informatizados voltados à geração de laudos em radiologia, predomina-se a aquisição e o armazenamento da informação na forma de anotação textual aberta. Um grande desafio enfrentado no desenvolvimento destes sistemas reside na inadequabilidade de ambientes caracterizados por interfaces de preenchimento padronizados e pré-definidos, que restringem fortemente a liberdade do médico na geração de seus relatos; contrastando com a ineficácia de outros ambientes que trabalham com textos abertos, restringindo fortemente as possibilidades de análise futura da informação.

Neste trabalho foi proposta uma metodologia que gera uma lista de candidatos a serem os primeiros sub-ramos de uma ontologia utilizando-se das palavras geradas e a frequência das palavras como métrica de corte. Foram geradas 3 ontologias, cada uma avaliando o comportamento da metodologia proposta em bases específica, intermediária e geral. Contatou-se que para uma base de 5000 laudos, os resultados foram mais interessantes para a base mais específica. Trabalhos adicionais podem utilizar mais níveis na ontologia, outros valores para a métrica de corte, além de utilizar um número de laudos maior. As ontologias também poderão ser mostradas com as palavras originais encontradas nos documentos ou mesmo agrupá-las em um nó da ontologia ao invés de somente

os radicais encontrados. Por último, seria importante para o país a adoção de ontologias padronizadas em língua portuguesa, o que permitiria avaliar mais objetivamente o conhecimento extraído por técnicas automáticas como a aqui proposta.

Agradecimentos

Este projeto de pesquisa foi desenvolvido na Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto (USP), Faculdade de Medicina de Ribeirão Preto (USP) e Hospital das Clínicas da Faculdade de Medicina de Ribeirão Preto com apoio financeiro FAPESP, CNPq, FAPEAM e INCT Adapta.

References

Aberle, D. R., Dionisio, J. D., McNitt-Gray, M. F., Taira, R. K., Cárdenas, A. F., Goldin, J. G., Brown, K., Figlin, R., and Chu, W. W. (1996). Integrated multimedia timeline of medical images and data for thoracic oncology patients. *Radio-Graphics*, 16:669–681.

Brank, J., Grobelnik, M., and Mladenić, D. (2007). *Automatic Evaluation of Ontologies*, chapter 11. In [Kao and Poteet 2007].

Feldman, R. and Sanger, J. (2006). *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data.* Cambridge University Press.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220.

Heflin, J., Hendler, J., and Luke, S. (1999). Coping with changing ontologies in a distributed environment. In *Proceedings of AAAI-99 Workshop on Ontology Management*, pages 74–79. AAAI Press.

Honorato, D. F., Monard, M. C., Lee, H. D., Neto, A. P., and Chung, W. F. (2009). Avaliação de uma metodologia de mapaemento de laudos médicos para uma representação estruturada: estudo de caso com laudos de endoscopia digestiva alta. *WIM - IX Workshop de Informática Médica.*

Kao, A. and Poteet, S. R., editors (2007). *Natural Language Processing and Text Mining.* Springer.

Mitchell, T. M. (1997). *Machine Learning.* McGraw–Hill.

Shortliffe, E. H. and Hubbard, S. M. (1989). *Information systems in oncology. In: De Vita VT, Hellman S, Rosenberg S, eds. Cancer: principles and practice of oncology.*

Soares, M. V., Prati, R. C., and Monard, M. C. (2008). Pretext ii: Descrição da reestruturacão da ferramenta de pré-processamento de textos. Technical Report 333, ICMC-USP, São Carlos - SP.

Sowa, J. (2000). *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* Brooks Cole Publishing Co., Pacific Grove, CA.

Taira, R. K., Soderland, S. G., and Jakobovits, R. M. (2001). Automatic structuring of radiology free-text reports. *RadioGraphics*, 21:237–245.

Zhang, J., Silvescu, A., and Honavar, V. G. (2002). Ontology-driven induction of decision trees at multiple levels of abstraction. In *Proceedings of the 5th International Symposium on Abstraction, Reformulation and Approximation*, pages 316–323, London, UK. Springer-Verlag.

Modelagem dos Conceitos Clínicos e Psicológicos de Demências para o Diagnóstico Auxiliado por Computador

Michele M. V. Ferreira[1], Timothy Wayne Cook[2], Luciana T. Cavalini[3]

[1]Especialização em Psicologia Hospitalar – Hospital Universitário Antonio Pedro
Rua Marquês de Paraná, 303 – 24.033-900 – Niterói – RJ – Brazil

[2]Instituto Nacional de Ciência e Tecnologia – Medicina Assistida por Computação Científica
Laboratório Associado "Multilevel Healthcare Information Modelling" – Brazil

[3]Instituto de Saúde da Comunidade
Universidade Federal Fluminense – Niterói, RJ – Brazil

mimaripsi@yahoo.com.br, timothywayne.cook@gmail.com, lutricav@vm.uff.br

Resumo. *A informação em saúde apresenta importantes complexidades espacial, temporal e de domínio, o que traz desafios à criação de sistemas de diagnóstico auxiliado por computador. Este artigo apresenta a estratégia de modelagem do conhecimento clínico e psicológico para um sistema de diagnóstico sindrômico de demências auxiliado por computador, baseado em modelagem multinível de sistemas de saúde. Para a modelagem do conhecimento, foram identificados os conceitos envolvidos na síndrome demencial, e foram elaborados arquétipos, de acordo com as especificações openEHR. Espera-se, com o desenvolvimento deste projeto, um melhor apoio às ações de detecção precoce de demências nos serviços de atenção básica.*

Abstract. *Health information presents significant spatial-temporal and domain complexities, which brings challenges to the implementation of computer-aided diagnostic systems. This paper presents the modeling strategy of the clinical and psychological knowledge for a computer-aided syndromic diagnosis of dementia, based on multilevel modeling of healthcare systems. To model the knowledge, the concepts related to the dementia syndrome were identified, and the respective archetypes were elaborated, according to the openEHR specifications. It is expected, with the development of this project, a better support for the activities of early detection of dementia in primary care settings.*

1. Introdução

1.1. Demência na Atenção Primária

A demência é uma das principais causas de incapacidade para os idosos. De uma perspectiva de Carga Global de Doença, contribui com 11,2% de todos os anos vividos com incapacidade [Mathers et al. 2003]. De uma perspectiva da atenção primária, no entanto, a demência pode não ter grande impacto sobre a carga de trabalho de um profissional de saúde da atenção básica, que atualmente poderá diagnosticar um ou dois

pacientes novos a cada ano (em uma área de tamanho populacional médio) e terá 12 a 15 pacientes com demência em um total de 2000 [Iliffe et al. 2009]. Esta carga de trabalho muda de acordo com a estrutura etária da população.

Apesar disso, a detecção precoce de demência na atenção primária é difícil de ser obtida entre diferentes sistemas de saúde [Olafsdottir et al. 2001], sendo que o diagnóstico ocorre entre 18 a 30 meses, e em casos extremos, em até quatro anos [Bamford et al. 2007]. Em primeiro lugar, a detecção pode ser adiada por motivos imputáveis ao paciente, à família do paciente ou ao cuidador. Por exemplo, algumas das características iniciais da demência podem ser pensadas como “normais” no processo de envelhecimento [Pollitt 1996]. Além disso, o estigma que pode ser associado à condição pode impedir o paciente ou a família de procurar ajuda médica, por causa do embaraço, vergonha ou incerteza [Iliffe et al. 2000].

Assim, é possível que o diagnóstico de demência na atenção primária possa ser melhorado através do desenvolvimento de sistemas informatizados de apoio a decisão [Johnston et al., 1996].

1.2. Registros Eletrônicos em Saúde e Sistemas de Apoio à Decisão na Atenção Primária

O desenvolvimento de sistemas de informação em saúde pode apoiar diversas ações em atenção primária [Wang et al. 2003]. Entretanto, sistemas de informação em saúde com base em modelos de dados tradicionais não são interoperáveis e têm elevados custos de manutenção. De fato, o desenvolvimento de aplicações saúde é um desafio complexo, especialmente devido ao elevado número de conceitos em evolução constante, o que torna o alcance de um consenso bastante difícil [Blobel 2002].

Algumas soluções para esses problemas têm sido propostas ao longo das últimas duas décadas, comumente envolvendo a separação entre o modelo de domínio e a persistência de dados [Wollersheim et al. 2009]. Esta abordagem de modelagem multinível propõe a definição de pelo menos dois níveis: o Modelo de Referência, que define os tipos genéricos de dados e de estruturas de dados e um Modelo de Domínio, definido pelas restrições ao Modelo de Referência. Sistemas de informação em saúde baseados na modelagem multinível são mais facilmente interoperáveis e podem ser implementados em qualquer hardware. [Garde et al. 2007].

Assim, o objetivo deste trabalho é definir a estratégia para modelar os conceitos psicológicos e clínicos de demência, como uma base de conhecimento para o desenvolvimento de sistemas de diagnóstico assistido por computador para uso em atenção primária.

2. Estratégia de Modelagem

Mediante revisão sistemática da literatura, foram encontrados 3.307 artigos científicos que abordavam critérios diagnósticos de demência, dos quais 581 foram selecionados para a análise, por apresentarem a validação de um critério diagnóstico. Para o presente estudo, foi modelado o instrumento de diagnóstico de demência leve conhecido como *Free and Cued Selective Reminding Test* (FCSRT) [Grober et al. 2010].

A modelagem do conhecimento utilizada foi baseada nos princípios da modelagem multinível de sistemas de informação em saúde. Esta abordagem é superior à modelagem tradicional de dados em um nível, que tem sido utilizada no

desenvolvimento de sistemas de informação em saúde nos últimos 45 anos, em termos de interoperabilidade do sistema, coerência semântica e persistência de longo prazo da informação [Garde et al. 2007]. Nesta abordagem, as classes de Modelo de Referência são persistentes e, portanto, tendem a ser estáveis ao longo do tempo. No Modelo de Domínio, as Definições de Restrições ao Modelo de Referência (chamadas de "arquétipos" pela Fundação openEHR [Madsen et al. 2010] fornecem a interpretação semântica dos objetos armazenados através do Modelo de Referência.

A modelagem do conhecimento necessário para a produção dos arquétipos foi desenvolvida usando o editor de arquétipos Ocean® Archetype Editor®, que permite a geração de arquivos no formato Archetype Definition Language (ADL) correspondentes à implementação do Modelo de Referência das especificações openEHR.

O FCSRT foi modelado como uma EVALUATION, com a estrutura de uma ITEM_TREE, na qual cada um dos três formulários do teste (A, B e C) foram definidos como CLUSTER, cada CLUSTER contendo 16 ELEMENTS, um para cada cartão apresentado ao paciente. O tipo de dados de cada ELEMENT foi definido como DV_COUNT, com valor mínimo inclusive igual a zero (0) e valor mínimo inclusive igual a três (3). Um quarto CLUSTER foi modelado, com 2 estruturas do tipo ELEMENT, uma para registrar o resultado do teste (com tipo de dados DV_COUNT) e outra para o registro da fórmula de cálculo do resultado (com tipo de dados DV_TEXT).

3. Conclusões

Este estudo demonstrou a viabilidade de modelagem do conhecimento de conceitos clínicos e psicológicos utilizados na prática dos serviços de saúde, mediante a utilização da modelagem multinível, como proposta nas especificações *open*EHR. O grande desafio que se coloca no futuro é o desenvolvimento de aplicativos baseados nesta metodologia, para uso na rotina dos serviços de saúde. Entretanto, para que isto seja efetivado, é necessário o desenvolvimento de uma massa crítica dentro da própria especialidade de informática em saúde, a partir de projetos de pesquisa e inovação, e de esforços educacionais de considerável porte.

Este projeto foi inteiramente baseado no uso de componentes baseados em software livre que já se provaram eficazes e universalmente adotados como padrões verdadeiros ou de facto. O mesmo se aplica às especificações que orientam a modelagem do conhecimento. Assim, não há restrições quanto ao licenciamento dos componentes derivados deste projeto. Isto também pode ser considerado um avanço potencial significativo, dado que é possível promover uma liberação do campo da informática em saúde das tradicionais políticas de *lock-in* disseminadas na indústria do software médico proprietário [Reynolds e Wyatt, 2011].

Referências

Bamford, C., Eccles, M., Steen, N. and Robinson, L. (2007). Can primary care record review facilitate earlier diagnosis of dementia? In *Family Practice*, pages 108–116. Oxford University Press.

Blobel, B. (2002). Comparing concepts for electronic health record architectures. In *Studies in Health Technology and Informatics*, pages 209–214. IOS Press.

Garde, S., Hovenga, E., Buck, J. and Knaup, P. (2007). Expressing clinical data sets with openEHR archetypes: a solid basis for ubiquitous computing. In *International*

Journal of Medical Informatics, pages S334–S341. Elsevier.

Garde. S., Knaup, P., Hovenga, E. and Heard, S. (2007). Towards semantic interoperability for electronic health records. In *Methods of Information in Medicine*, pages 332–343. Schattauer Publishers.

Grober, E., Sanders, A. E., Hall, C. and Lipton, R. B. (2010). Free and cued selective reminding identifies very mild dementia in primary care. In *Alzheimer Disease & Associated Disorders*, pages 284–290. Lippincott Williams & Wilkins.

Iliffe, S., Manthorpe, J. and Eden, A. (2003). Sooner or later? Issues in the early diagnosis of dementia in general practice: a qualitative study. In *Family Practice*, pages 376–381. Oxford University Press.

Iliffe, S., Robinson, L., Brayne C, Goodman C, Rait G and Manthorpe J. (2009). Primary care and dementia: 1. diagnosis, screening and disclosure. In *International Journal of Geriatric Psychiatry*, pages 895–901. John Wiley & Sons.

Iliffe, S., Walters, K. and Rait, G. (2000). Shortcomings in the diagnosis and management of dementia in primary care: towards an educational strategy. In *Aging & Mental Health*, pages 286–291. Taylor & Francis.

Madsen, M., Leslie, H., Hovenga, E. J. and Heard, S. (2010). Sustainable clinical knowledge management: an archetype development life cycle. In *Studies in Health Technology and Informatics*, pages 115–132. IOS Press.

Mathers, C. D., Bernard, C., Iburg, K. M., Inoue, M., Fat, D. M., Shibuya, K., Stein, C., Tomijima, N. and Xu, H. (2003). Global burden of disease in 2002: data sources, methods and results. Global Programme on Evidence for Health Policy Discussion Paper No. 54. World Health Organization, 1st edition.

Olafsdottir, M., Foldevi, M. and Marcusson, J. (2001). Dementia in primary care: why the low detection rate? In *Scandinavian Journal of Primary Health Care*, pages 194–198. Taylor & Francis.

Pollitt, P.A. (1996). Dementia in old age: an anthropological perspective. In *Psychological Medicine*, pages 1061–1074. Cambridge University Press.

Reynolds, C. J. and Wyatt, J. C. (2011). Open source, open standards, and health care information systems. In *The Journal of Medical Internet Research*, e24.

Wang, S. J., Middleton, B., Prosser, L. A., Bardon, C. G., Spurr, C. D., Carchidi, P. J., Kittler, A. F., Goldszer, R. C., Fairchild, D. G., Sussman, A. J., Kuperman, G. J. and Bates, D. W. (2003). A cost-benefit analysis of electronic medical records in primary care. In *American Journal of Medicine*, pages 397–403. Elsevier.

Wollersheim, D., Sari, A. and Rahayu, W. (2009). Archetype-based electronic health records: a literature review and evaluation of their applicability to health data interoperability and access. In *Health Information Management Journal*, pages 7–17. Health Information Management Association of Australia.

Avaliação do Algoritmo de *Stacking* em Dados Biomédicos*

Maria Izabela R. Caffé[1,2] , Pedro Santoro Perez[1] & José Augusto Baranauskas[1]

[1]Departamento de Computação e Matemática — DCM
Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto — FFCLRP
Universidade de São Paulo — USP
Av. Bandeirantes, 3900 - Ribeirão Preto, SP - 14040-901 - Brasil

[2]Faculdade de Medicina de Ribeirão Preto — FMRP
Universidade de São Paulo — USP
Av. Bandeirantes, 3900 - Ribeirão Preto, SP - 14040-901 - Brasil

mariairc@aluno.ffclrp.usp.br, pedrosperez@usp.br, augusto@usp.br

Abstract. *Stacking is a well studied ensemble technique, but some of its aspects still need to be explored, e.g., there are no recommendations on which and how many algorithms should be used at level-0 or even which algorithm should be used to compose the level-1 meta-classifier. The literature indicates the meta-algorithm at level-1 should be simple, and Naive Bayes has been typically used in studies. This study analyzed stacking in medical datasets, using three different paradigms of machine learning algorithms to compose the meta-classifier. The experiments indicate simple meta-algorithms do not provide good results, and therefore, the meta-classifier must have a degree of complexity for it to achieve a good performance.*

Resumo. *O stacking é uma técnica de combinação de classificadores bem estudada, mas ainda com muitos aspectos a explorar, e.g., não existem recomendações sobre quais e quantos algoritmos devem ser utilizados no nível-0 nem qual algoritmo deve compor o meta-classificador do nível-1. A literatura indica que o meta-algoritmo do nível-1 deve ser simples, sendo Naive Bayes geralmente utilizado nos estudos. Neste estudo analisou-se stacking em conjuntos de dados biomédicos, utilizando três paradigmas de aprendizado de máquina para o meta-classificador. Os experimentos mostram que meta-algoritmos simples não apresentam bons resultados, indicando que devem ter certo grau de complexidade para obterem bom desempenho.*

1. Introdução

Combinação de classificadores (*ensembles*) é um método de aprendizado de máquina no qual vários modelos (hipóteses) são combinados para gerar um único classificador final. Em geral, a utilização de *ensembles* tem a tendência de diminuir a taxa de erro, tornando o classificador final mais preciso, já que utiliza as predições de todos os algoritmos de

*Projeto de pesquisa realizado com apoio financeiro do CNPq/FAPEAM — INCT ADAPTA.

aprendizado envolvidos para gerar uma hipótese final para o mesmo conjunto de dados [Dzeroski & Zenko 2004].

Stacking [Wolpert 1992] é um dos métodos de combinação de classificadores heterogêneos. O método proposto é uma estrutura em duas camadas: no nível-0 vários algoritmos de aprendizado recebem o conjunto de treinamento, gerando os classificadores de nível-0; a camada seguinte (nível-1) tem como entrada as predições da camada anterior (nível-0), na qual um meta-algoritmo de nível-1 as combina para fornecer o meta-classificador final h^*, conforme é mostrado na Figura 1. Mais precisamente, suponha L diferentes algoritmos de aprendizado A_1, A_2, …, A_L e um conjunto de n exemplos $\{(x_1,y_2), (x_2,y_2), \ldots, (x_n,y_n)\}$, onde fica implícito o fato que cada elemento x_i é um vetor. Cada algoritmo de nível-0 é aplicado ao conjunto de treinamento, induzindo os classificadores $\{h_1, h_2, \ldots, h_L\}$ deste nível. Cada classificador do nível-0 é então utilizado para rotular os exemplos. Isso implica que, para cada exemplo x_i, uma tupla é formada, composta pela classe predita por cada um dos classificadores de nível-0 juntamente com a classe verdadeira daquele exemplo, isto é, $(h_1(x_i), h_2(x_i), \ldots, h_L(x_i), y_i)$. Essas tuplas constituem o conjunto de treinamento de nível-1, cujos atributos são as classes preditas por cada um dos L classificadores. Com isso, é possível aplicar o meta-algoritmo de aprendizado aos exemplos de nível-1 para aprender o meta-classificador h^*.

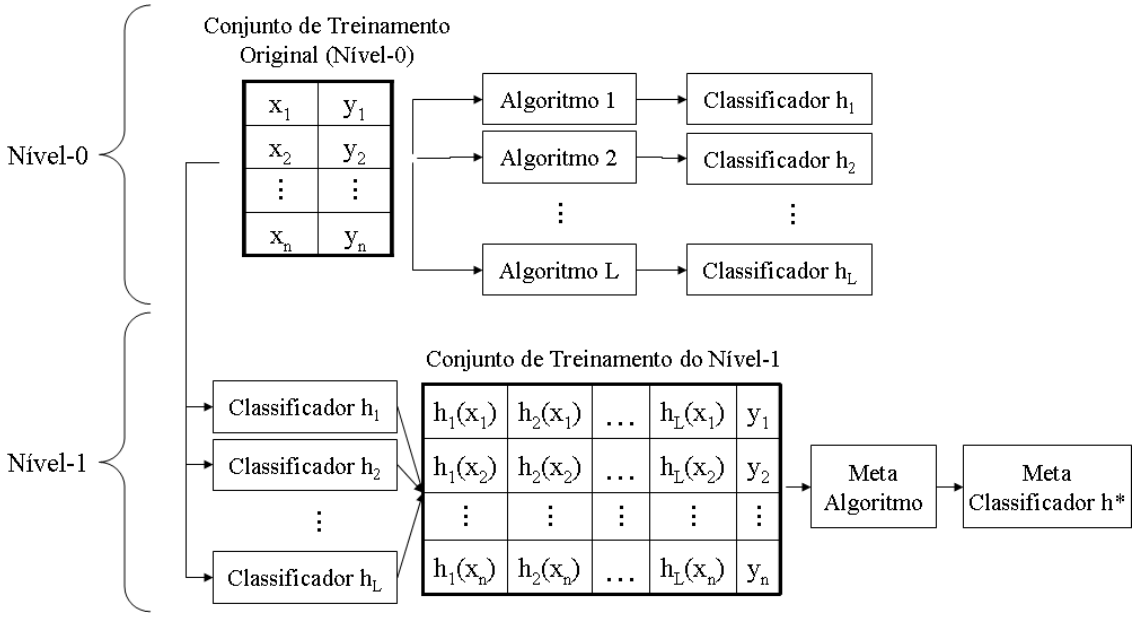


Figura 1: *Stacking*

Uma das explicações para o uso de *stacking* é que ao passar pelas camadas, o meta-classificador aprende os erros dos classificadores anteriores. [Wolpert 1992] relata que muito aspectos sobre *stacking* ainda são desconhecidos (*black art*) no sentido que não há recomendações sobre quais e quantos algoritmos devem ser utilizados no nível-0 nem qual o algoritmo que deve compor o meta-classificador do nível-1. Em geral, os trabalhos na literatura utilizam um número variável de algoritmos no nível-0 e para gerar o meta-classificador o algoritmo Naive Bayes.

Normalmente, dados biomédicos estão associados a uma grande quantidade de classes, quantidade de exemplos muito variada, excesso de atributos e exemplos com dados faltantes e redundantes, o que tem motivado a aplicação de algoritmos de aprendizado de máquina nesse domínio [Pereira & Schmitz 2010, Pollettini *et al.* 2009]. Neste sentido, o estudo de [Tanwani *et al.* 2009] buscou traçar um guia para utilização de *ensembles*, com ênfase em conjuntos de exemplos da área biomédica. Entretanto, os autores utilizaram apenas o algoritmo Naive Bayes como meta-classificador em sua análise do algoritmo de *stacking*. Com isso em mente, o objetivo deste estudo consiste em avaliar o método de *stacking* em conjuntos de dados biomédicos, utilizando diferentes classificadores tanto de nível-0 como de nível-1, estendendo assim, o trabalho de [Tanwani *et al.* 2009] referente a *stacking*.

O restante deste artigo encontra-se organizado da seguinte forma. Na Seção 2 estão descritos os algoritmos usados para induzir os classificadores e meta-classificadores e os conjuntos de exemplos utilizados, bem como os três experimentos que compõem este estudo. Na Seção 3 os resultados são expostos e discutidos; as conclusões encontram-se na Seção 4.

2. Metodologia Experimental

Como mencionado anteriormente, no estudo de [Tanwani *et al.* 2009] envolvendo *stacking* os autores utilizaram apenas o algoritmo Naive Bayes como meta-classificador. No estudo aqui desenvolvido, o uso de *stacking* é ampliado, comparando o desempenho do uso de três paradigmas diferentes no nível-1: estatístico (Naive Bayes), indução de regras simples (One Rule) e árvore de decisão de um único nível (Decision Stump). O objetivo aqui é tentar verificar se algoritmos de diferentes paradigmas de aprendizado resultam em uma melhor combinação no *stacking*.

Resumidamente, este estudo é composto por três experimentos envolvendo stacking realizados em dezoito conjuntos de exemplos biomédicos utilizando vários algoritmos, que são descritos nas próximas seções.

2.1. Conjuntos de Exemplos

Os conjuntos de exemplos utilizados neste trabalho referem-se ao domínio biomédico selecionados do *UCI Machine Learning Repository* [Frank & Asuncion 2010]. Na Tabela 1 são apresentados os conjuntos de dados, data de publicação, número de exemplos, o número de atributos, o número e a distribuição das classes e se há ou não dados faltantes. Neste trabalho não houve nenhum pré-processamento dos conjuntos de dados. A seguir é fornecida uma breve descrição biológica sobre os conjuntos de exemplos.

- **Breast Cancer**: a partir de atributos com informações clínicas e laboratoriais sobre pacientes, a tarefa é distinguir entre eventos recorrentes e não-recorrentes associados a câncer de mama (Instituto de Oncologia da Iugoslávia).
- **Haberman**: com atributos sobre cirurgias de câncer de mama, o objetivo é predizer se o paciente sobreviveu após a cirurgia durante o estudo.
- **Heart-statlog**: também baseado em aspectos clínicos e laboratoriais, o problema de classificação é prever se há ausência ou presença de doença cardíaca.

- **Hepatitis**: considerando dados clínicos e laboratoriais, a tarefa é predizer se um paciente com hepatite morreu ou sobreviveu durante o período do estudo.
- **Liver Disorders**: predição de uma determinada classe binária baseada em testes sanguíneos relacionados a problemas de fígado. A documentação sobre esta base de dados não deixa claro o que a classe significa. Sabe-se apenas que ela pode assumir 2 valores, cada um representado por um número simplesmente.
- **Pima Indians**: problema com atributos sobre dados clínicos e laboratoriais de mulheres descendentes dos indígenas Pima cuja tarefa de classificação é distinguir entre pacientes testadas positivamente para diabetes e pacientes testadas negativamente para a doença.
- **Promoters**: baseada em sequências de DNA de E. coli, a tarefa consiste em discriminar se uma dada sequência é promotora gênica.
- **Sick**: tarefa de prever se pacientes possuem doença relacionada a tireoide baseando-se em dados clínicos e laboratoriais.
- **Contraceptive Method**: baseada em atributos de natureza sócio-econômica de mulheres da Indonésia, a tarefa é predizer qual o método contraceptivo utilizado.
- **Lung Cancer**: os autores desta base de dados não fornecem informação sobre os atributos utilizados. A tarefa é diferenciar entre 3 tipos de câncer de pulmão.
- **Postoperative**: o problema é determinar para onde pacientes na área pós-operatória de recuperação devem ser encaminhados a seguir, sendo os atributos relacionados a medidas clínicas sobre os pacientes (temperatura corporal, pressão sanguínea).
- **Ann-thyroid**: a tarefa é distinguir entre 3 situações envolvendo hipotireoidismo. Os atributos trazem dados sobre medidas laboratoriais relacionadas a problemas na tireoide.
- **Lymphography**: baseados em achados clínicos e laboratoriais relacionados à linfografia, a tarefa é discriminar condições tumorais da linfa (Instituto de Oncologia da Iugoslavia).
- **Cleveland, Switzerland e Hungarian**: o problema aqui consiste em distinguir entre condições relacionadas a doenças do coração. Os atributos trazem informação sobre achados clínicos e laboratoriais.
- **Dermatology**: O problema é determinar o tipo de doença erimato-esquimatosa a partir de achados clínicos e histopatológicos. **Ecoli**: a tarefa é predizer os sítios de localização celular de proteínas a partir de medidas e pontuações relacionadas a sequências protéicas.

2.2. Algoritmos e Meta-Algoritmos

Nos experimentos realizados foram utilizados 9 algoritmos de aprendizado de máquina da biblioteca Weka [Witten & Frank 2005], com seus parâmetros default, exceto quando mencionado o contrário. Uma breve descrição sobre cada um deles é dada a seguir.

- Naive Bayes usa o método probabilístico para classificação [Rish 2001], assumindo independência entre os atributos.
- Bayes Network utiliza diversos algoritmos de busca e medidas de qualidade, fornecendo estruturas sobre os dados [Chickering *et al.* 2004].
- IBK (*Instanced Based Learner*) classifica segundo os K-vizinhos mais próximos [Aha *et al.* 1991]. Neste estudo foi utilizado K = 3.
- JRip induz regras, sendo uma reimplementação do algoritmo Ripper [Cohen 1995].

- One Rule é um classificador muito simples, que seleciona um atributo para compor a regra com menor taxa de erro [Witten & Frank 2005].
- J48 gera árvores de decisão, sendo uma reimplementação do algoritmo C4.5 [Quinlan 1993].
- Decision Stump constrói uma árvore de decisão de um único nível utilizando entropia [Iba & Langley 1992].
- MLP (*Multi Layer Perceptron*) é uma rede neural multicamada utilizando retropropagação para otimizar os pesos durante o treinamento [Haykin 1998].
- SMO (*Support Vector Machine*) algoritmo de otimização mínima sequencial de John Platt para treinar classificadores de vetores de suporte [Vapnik 1998].

Os experimentos envolvendo *stacking* são identificados como Meta 1, Meta 2 e Meta 3, conforme Tabela 2. Cada um dos 9 algoritmos também teve seu desempenho avaliado individualmente. O desempenho foi medido utilizando validação cruzada com 10 partições (10-*fold cross-validation*).

Tabela 1. Conjuntos de exemplos ordenados por número de classes

Conjunto	Ano	Exemplos	Atributos	Classes	Distribuição de Classes	Faltantes
Breast cancer	1988	286	9	2	(70.28, 29.72)	Sim
Haberman	1999	306	4	2	(73.53, 26.47)	Não
Heart statlog	N/D	270	13	2	(55.56, 44.44)	Não
Hepatitis	1988	155	20	2	(20.65, 79.35)	Sim
Liver disorders	1990	345	7	2	(42.03, 57.97)	Não
Pima indians	1990	768	9	2	(65.10, 34.90)	Não
Promoters	1990	106	58	2	(50.00, 50.00)	Não
Sick	1987	3772	30	2	(93.88, 6.12)	Sim
Contraceptive	1997	1473	9	3	(42.70, 22.61, 34.69)	Não
Lung cancer	1992	32	56	3	(28.13, 40.62, 31.25)	Sim
Postoperative	1993	90	9	3	(71.11, 2.22, 26.67)	Sim
Ann-thyroid	1992	7200	22	3	(2.31, 5.11, 95.58)	Não
Lymphography	1988	148	19	4	(1.35, 54.73, 41.22, 2.70)	Não
Cleveland	1988	303	13	5	(54.46, 45.54, 0.00, 0.00, 0.00)	Sim
Hungarian	1988	294	14	5	(63.95, 34.98, 0.00, 0.00, 0.00)	Sim
Switzerland	1988	123	14	5	(6.50, 39.02, 26.02, 24.39, 4.07)	Sim
Dermatology	1998	366	33	6	(30.60, 16.67, 19.57, 13.39, 14.20, 5.46)	Sim
Ecoli	1996	336	8	8	(42.56, 22.92, 15.48, 10.42, 5.95, 1.49, 0.60, 0.60)	Não

Tabela 2. Algoritmos e Meta Algoritmos em cada Experimento

Experimento	Meta Algoritmo	Algoritmos
Meta 1	Naive Bayes	Bayes Net, IBK, One Rule, JRip, J48, Decision Stump, MLP, SMO
Meta 2	One Rule	Bayes Net, Naive Bayes, IBK, JRip, J48, Decision Stump, MLP, SMO
Meta 3	Decision Stump	Bayes Net, Naive Bayes, IBK, One Rule, JRip, J48, MLP, SMO

3. Resultados e Discussão

Na Tabela 3 estão as medidas AUC (área sob a curva ROC) [Bradley 1997] obtidas para os experimentos Meta 1, Meta 2 e Meta 3 e todos os outros algoritmos utilizados neste trabalho, que obtiveram a seguinte colocação: Meta 1, Bayes Net, Naive Bayes, MLP, J48, IBK, SMO, Meta 3, Jrip, Meta 2, Decision Stump e One Rule. Nesta tabela é possível observar que o classificador obtido por Meta 1 teve um *ranking* melhor do que Meta 2 e Meta 3. Entretanto, os classificadores Meta 2 e Meta 3, em geral, tiveram desempenho pior do que os outros classificadores sozinhos, sendo estes Bayes Net, Naive Bayes, MLP, J48, IBK e SMO.

Para avaliar os resultados obtidos sob o ponto de vista estatístico, foi realizado teste de Friedman [Friedman 1940] com significância α=0.05 para determinar se há diferença significativa entre os valores AUC obtidos. Como a hipótese nula foi rejeitada (ou seja, existe diferença entre os classificadores), foi utilizado o teste *post hoc* descrito em [Benjamini & Hochberg 1995] (menos conservador que o teste de Nemenyi [Nemenyi 1963]) para buscar pares com diferença significativa em uma comparação todos-contra-todos; o teste confirmou que Meta 1 se saiu significativamente melhor do que todos os outros algoritmos bem como Meta 2 e Meta 3, exceto Naive Bayes, Bayes Net e MLP, perante o qual Meta 1 apenas se saiu melhor, mas não de forma significativa.

Os resultados do teste podem ser visualizados na Tabela 4, na qual $\vartriangle$ (▲) indica que o algoritmo da respectiva linha foi melhor (significativamente) do que o algoritmo da respectiva coluna e $\triangledown$ (▼) indica que o algoritmo da respectiva linha foi pior (significativamente) do que o algoritmo da respectiva coluna. Por simetria, o triângulo inferior dessa tabela possui resultados opostos ao triângulo superior e foi omitido por questões de clareza.

Como o desvio padrão pode ser visto como uma medida de estabilidade dos algoritmos a pequenas variações no conjunto de treinamento, o teste de Friedman também foi utilizado para avaliar diferenças significativas entre os desvios padrões de todos os algoritmos, incluindo *stacking*. O teste de Friedman encontrou diferença significativa entre os desvios padrão, entretanto o teste *post hoc* com correção por Benjamini & Hochberg não conseguiu detectar diferenças utilizando α=0.05. Neste sentido, os resultados obtidos permitem afirmar que, para os conjuntos de exemplos avaliados, a utilização de *stacking* não melhora nem piora a estabilidade dos algoritmos.

Tabela 3. Medida AUC(±Desvio Padrão) dos Experimentos

Conjunto	Naive Bayes	BayesNet	IBK	JRip	One Rule	Decision Stump	J48	MLP	SMO	Meta 1	Meta 2	Meta 3
Breast câncer	0.72±0.14	0.71±0.14	0.66±0.13	0.61±0.10	0.54±0.07	0.65±0.13	0.63±0.10	0.62±0.13	0.59±0.08	0.69±0.15	0.57±0.09	0.67±0.11
Haberman	0.67±0.11	0.69±0.09	0.63±0.11	0.62±0.10	0.59±0.08	0.64±0.08	0.58±0.08	0.66±0.13	0.51±0.02	0.72±0.05	0.51±0.07	0.55±0.04
Heart statlog	0.90±0.06	0.91±0.04	0.83±0.07	0.80±0.08	0.71±0.06	0.72±0.07	0.76±0.10	0.85±0.06	0.84±0.06	0.89±0.06	0.79±0.08	0.83±0.08
Hepatitis	0.86±0.11	0.89±0.08	0.79±0.15	0.60±0.15	0.65±0.13	0.67±0.13	0.70±0.20	0.82±0.15	0.75±0.13	0.83±0.12	0.73±0.16	0.73±0.16
Liver disorders	0.65±0.12	0.52±0.03	0.64±0.06	0.64±0.09	0.54±0.07	0.54±0.06	0.67±0.08	0.74±0.07	0.50±0.01	0.75±0.08	0.58±0.06	0.66±0.09
Pima indians	0.82±0.05	0.81±0.05	0.74±0.05	0.72±0.06	0.67±0.06	0.69±0.06	0.75±0.08	0.80±0.04	0.72±0.06	0.83±0.05	0.67±0.04	0.73±0.05
Promoters	0.96±0.11	0.96±0.11	0.92±0.08	0.81±0.13	0.70±0.11	0.71±0.12	0.83±0.14	0.98±0.08	0.93±0.09	0.97±0.08	0.58±0.11	0.90±0.09
Sick	0.93±0.05	0.96±0.02	0.88±0.05	0.94±0.05	0.89±0.04	0.94±0.03	0.95±0.04	0.95±0.03	0.50±0.00	0.99±0.01	0.93±0.03	0.94±0.03
Contraceptive	0.69±0.04	0.70±0.04	0.62±0.04	0.62±0.04	0.58±0.02	0.56±0.02	0.66±0.04	0.70±0.02	0.63±0.03	0.73±0.04	0.59±0.04	0.60±0.02
Lung cancer	0.71±0.32	0.71±0.29	0.68±0.31	0.55±0.15	0.54±0.15	0.55±0.11	0.68±0.23	0.56±0.22	0.63±0.20	0.80±0.19	0.70±0.21	0.57±0.15
Postoperative	0.39±0.22	0.40±0.21	0.31±0.13	0.50±0.00	0.48±0.03	0.46±0.13	0.49±0.02	0.41±0.19	0.47±0.04	0,57±0.18	0.46±0.11	0.43±0.12
Ann-thyroid	0.93±0.02	1.00±0.00	0.74±0.04	0.99±0.01	0.95±0.02	0.99±0.00	0.99±0.00	0.97±0.03	0.59±0.02	1.00±0.00	0.97±0.03	0.99±0.00
Lymph	0.91±0.07	0.91±0.07	0.89±0.10	0.78±0.14	0.77±0.11	0.78±0.11	0.79±0.14	0.91±0.07	0.87±0.08	0.87±0.11	0.80±0.06	0.81±0.07
Cleveland	0.90±0.06	0.91±0.04	0.85±0.07	0.84±0.07	0.72±0.07	0.71±0.07	0.80±0.09	0.89±0.06	0.84±0.08	0.91±0.05	0.79±0.08	0.81±0.07
Hungarian	0.90±0.06	0.91±0.07	0,87±0.06	0.76±0.10	0.75±0.11	0.77±0.11	0.77±0.15	0.89±0.04	0.80±0.11	0.89±0.07	0.83±0.06	0.80±0.08
Switzerland	0.53±0.12	0.54±0.04	0.49±0.09	0.56±0.07	0.53±0.10	0.47±0.06	0.55±0.09	0.54±0.12	0.57±0.09	0.52±0.13	0.50±0.07	0.51±0.04
Dermatology	1.00±0.00	1.00±0.00	0.99±0.01	0.95±0.02	0.67±0.03	0.79±0.02	0.97±0.02	1.00±0.00	0.98±0.01	1.00±0.00	0.82±0.03	0.78±0.01
Ecoli	0.97±0.02	0.97±0.02	0.94±0.03	0.91±0.04	0.75±0.04	0.78±0.02	0.92±0.05	0.96±0.02	0.95±0.02	0.93±0.04	0.81±0.04	0.83±0.02
Ranking Médio AUC	4.00	3.06	6.83	7.64	10.42	9.28	6.56	4.28	7.19	2.58	8.61	7.50
Ranking Médio Desvio Padrão	7.14	5.47	7.31	7.83	6.64	6.08	8.94	5.54	5.14	5.83	6.47	5.33

Tabela 4. Resultados do Teste de Friedman para medida AUC

Algoritmo	Naive Bayes	Bayes Net	IBK	JRip	One Rule	Decison Stump	J48	MLP	SMO	Meta 1	Meta 2	Meta 3
NaiveBayes		∇	▲	▲	▲	▲	∆	∆	▲	∇	▲	▲
Bayes Net			▲	▲	▲	▲	▲	∆	▲	∇	▲	▲
IBK				∆	▲	∆	∇	∇	∆	▼	∆	∆
Jrip					▲	∆	∇	▼	∇	▼	∆	∇
One Rule						∇	▼	▼	▼	▼	∇	▼
Decision Stump							▼	▼	∇	▼	∇	∇
J48								∇	∆	▼	∆	∆
MLP									▲	∇	▲	▲
SMO										▼	∆	∆
Meta 1											▲	▲
Meta 2												∇
Meta 3												

4. Conclusão

Neste estudo a avaliação de *stacking* foi realizada utilizando três paradigmas diferentes para geração do meta-classificador: estatístico, indução de regras simples e árvore de decisão de um único nível, estendendo assim, o trabalho de [Tanwani *et al.* 2009] no qual *stacking* utilizou somente um algoritmo estatístico como meta-classificador. Os resultados aqui obtidos permitem concluir que *stacking* utilizando como meta-classificador um algoritmo estatístico tem um desempenho significativamente superior do que indução de regras ou árvores simples.

Na literatura associada é recomendável a utilização de algoritmos simples para compor o meta-classificador [Dzeroski & Zenko 2002]. Entretanto, o presente estudo indica que o algoritmo utilizado para compor o meta-classificador deve ter um grau de sofisticação tal que o permita representar adequadamente os conceitos oriundos do nível-0, algo que regras contendo um único atributo ou árvores de um único nível claramente não o possuem. Com isso, é possível indicar que estudos futuros utilizem um algoritmo de complexidade controlada para compor o meta-classificador — por exemplo, uma árvore de decisão com níveis decrescentes de poda ou uma rede neural com quantidades crescentes de neurônios, sinapses ou ciclos de treinamento — de forma a identificar, se possível, o nível de complexidade exigido para compor um bom meta-classificador.

Referências

Aha, D. W., Kibler, D. & Albert, M. K. (1991) “Instance based learning algorithms” In Machine Learning, p. 37–66.

Benjamini, Y. & Hochberg, Y. (1995) “Controlling the false discovery rate: a practical and powerful approach to multiple testing”, In Journal of the Royal Statistical Society Series B, v. 57, p. 289–300.

Bradley, A. P. (1997) “The use of the area under the ROC curve in the evaluation of machine learning algorithms”. Pattern Recognition 30(7), 1145–1159.

Chickering, D. M., Heckerman, D. & Meek, C. (2005) “Learning of Bayesian Networks is NP – Hard” In Journal of Machine Learning Research, 5, p 1287–1330.

Cohen, W. W. (1995) “Fast effective rule induction” In Proceedings of Twelfth International Conference on Machine Learning, p. 115–123.

Dzeroski, S. & Zenko B. (2002) “Is combining Classifiers Better than Selecting the Best One?” In Proceedings of the 19th International Conference on Machine Learning, Morgan Kaufmann Publishers, San Francisco.

Frank, A. & Asuncion, A. (2010) “UCI Machine Learning Repository”, http://archive.ics.uci.edu/ml, School of Information and Computer Science, University of California at Irvine, Irvine CA.

Friedman, M. (1940) “A comparison of alternative tests of significance for the problem of m rankings”. The Annals of Mathematical Statistics 11(1), 86–92.

Haykin, S. (1998) Neural networks: a comprehensive foundation, 2nd edition, Pearson Education, London.

Iba, W. & Langley, P. (1992) “Induction of One – Level Decision Trees” In Proceedings of the Ninth International Conference on Machine Learning.

Nemenyi, P. B. (1963) Distribution-free multiple comparisons, PhD. Thesis, Princeton University.

Pereira, M. & Schmitz, A. (2010) "Inteligência Artificial e Geotecnologias Emergentes Aplicadas em Estudos Ecoepidemiológicos de Malária no Município de Bragança-Pará, Brasil, no Período de 2006 a 2008", In Proceedings do X Workshop de Informática Médica, Congresso da Sociedade Brasileira de Computação, p. 1630–1640, Belo Horizonte.

Pollettini, J. T., Tinos, R., Panico, S., Daneluzzi, J. C. & Macedo, A. A. (2009) "Vigilância em atenção básica à saúde a partir do uso de relevance feedback para classificação de pacientes em diferentes níveis de cuidado em saúde", In Proceedings do IX Workshop de Informática Médica, Congresso da Sociedade Brasileira de Computação, p. 1945–1954, Bento Gonçalves.

Quinlan, J. R. (1993) C4.5: programs for machine learning, Morgan Kaufmann, San Francisco.

Rish, I. (2001) “An empirical study of the naive Bayes classifier”, In IJCAI Workshop on Empirical Methods in Artificial Intelligence, p. 41–46.

Seewald, A. K. (2002) “How to make Stacking Better and Faster While Also Taking Care of an Unknown Weakness”, In Proceedings of the 19th International Conference on Machine Learning, p. 554–561, Morgan Kaufmann Publishers, Sydney.

Seewald, A. K. (2002) “Exploring the Parameter State Space of Stacking”, In Proceedings of the 2002 IEEE International Conference of Data Mining (ICDM'02), p. 685–688.

Tanwani, A. K., Afridi, J. Shafiq, M. Z. & Farroq, M. (2009) “Guidelines to Select Machine Learning Scheme for Classification of Biomedical Datasets”, C.Pizzuti, M.D. Ritchie, & M. Giacobini, LNCS 5483, Springer-Verlag Berlin Heidelberg 2009, p. 128–139.

Vapnik, V. N. (1998) Statistical learning theory, Wiley Interscience, USA.

Witten, I. H. & Frank, E. (2005) Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, 2. ed.: Morgan Kaufmann.

Wolpert, D. H. (1992) Stacked Generalization. In *Neural Networks,* p. 241–260.

HqRF: Novos Descritores de Forma para Auxiliar no Diagnóstico de Câncer de Mama

Walter Alexandre A. de Oliveira, Denise Guliato

[1]Faculdade de Computação – Universidade Federal de Uberlândia (UFU)
Av. João Naves de Ávila, 2121 – Uberlândia – MG – Brasil
INCT-MACC Medicina Assistida por Computação Científica

{walter,guliato}@facom.ufu.br

Abstract. *Malignant breast tumors and benign masses appear in breast cancer exams with different shape characteristics: the former usually have rough, spiculated, or microlobulated contours, whereas the latter commonly have smooth, round, oval, or macrolobulated contours. Visual features that characterize shape roughness can assist in distinguishing between tumor types. In this work we propose two new approachs to describe 2D and 3D shapes based on Hilbert curves. The descriptors are aplied to content-based images/volumes retrieval. The methods were evaluated using a set of breast contours and a 3D object synthetic database. As the experimental evaluations show, we achieved a mean average precision of 1.00 (to 2D shapes) and 0.99 (to 3D shapes).*

Resumo. *Tumores malignos e lesões benignas aparecem em exames radiológicos com diferentes características: os primeiros geralmente apresentam contornos grosseiros, espiculados ou microlobulados, enquanto os últimos comumente têm contornos suaves, arredondados, ovalados ou macrolobulados. Características visuais que descrevem a aspereza da forma podem auxiliar na distinção entre os tipos de tumor. Neste trabalho são propostos novos métodos para descrever formas 2D e 3D, baseados na curva de Hilbert. Os descritores são aplicados para a recuperação de imagens/volumes por conteúdo. O método foi avaliado usando uma base de dados de contornos de mama e uma base de dados sintética com objetos 3D. A avaliação dos experimentos mostram uma precisão média de 1.00 (para formas 2D) e 0.99 (para formas 3D).*

1. Introdução

O câncer de mama é o tipo mais comum de câncer entre mulheres no mundo todo, chegando a cerca de 16% [WHO 2010]. A Organização Mundial de Saúde (OMS) estima que 460 mil mulheres foram vítimas de câncer de mama durante o ano de 2008. Ainda de acordo com a OMS, cerca de 70% dos casos de câncer de mama ocorrem em países pobres ou em desenvolvimento. Em regiões como a América do Norte, a taxa de sobrevivência chega a 80%, graças, principalmente, à detecção precoce da lesão de mama [WHO 2010]. No entanto esta modalidade de exame apresenta limitações para detectar tumores malignos com achados radiológicos ocultos em mamas densas ou em mamas com implante, devido à sobreposição de tecidos causada pela projeção bidimensional da mama. Exaustivos esforços têm sido feitos nos últimos 15 anos para incluir imagens de ressonância magnética (MRI) e ultrasonografia como modalidades complementares para melhorar a interpretação de imagens de mama [Turnbull et al. 2010, Chen et al. 2006].

Independente da modalidade, lesões benignas e malignas podem ser diferenciadas pelas características da sua forma. É bem estabelecido que a maioria das lesões benignas tem contornos arredondados ou ovais, textura interna relativamente homogênea e margens bem definidas. As lesões malignas tipicamente apresentam contornos mal definidos, rugosos ou espiculados e textura interna heterogênea. Entretanto, casos atípicos podem ocorrer, o que torna o diagnóstico de câncer de mama uma tarefa difícil e sujeita a erros [Gupta et al. 2006, American College of Radiology, Reston, VA 1998].

Com o objetivo de auxiliar o radiologista na tomada de decisão, vários sistemas de apoio ao diagnóstico (CADx) de câncer de mama têm sido propostos no últimos anos. Uma das tarefas realizadas por um sistema CADx é analisar, objetivamente, as características do contorno das lesões presentes no exame radiológico para classificá-las como lesões benignas ou malignas. Diversos trabalhos têm sido desenvolvidos para caracterizar os contornos de lesões de mama de forma automática. Rangayyan et al. em [Rangayyan et al. 2000] propuseram métodos para obtenção de índices de espículos (SI) e de concavidade (F_{cc}), baseados em modelos poligonais para caraterização do contorno da lesão. Guliato el al. em [Guliato et al. 2008b] propuseram métodos para obtenção de índice de espículo (SI_{TA}), índice de concavidade (CX_{TA}) e índice de dimensão fractal (FD_{TA}) baseados na assinatura da função tangente (*Turning Angle Function*) de um dado contorno. Os métodos foram aplicados à classificação de lesões de mama, tendo alcançado altas taxas de precisão ($AUC = 0.93$). Caballero et al. em [R.G.-Caballero et al. 2007] utilizaram a análise de componentes independentes (*Independent Component Analysis* - ICA) para obter um conjunto de coeficientes usados na classificação de mamogramas com adição de ruído. A classificação foi realizada usando redes neurais e imagens da DDSM (*Digital Database for Screening Mammography*) [Heath et al. 1998]. Weiqiang et al. em [Weiqiang et al. 2008] combinaram diversos índices para representar as características do contorno de lesões de mama. Os índices utilizado no trabalho incluem: índice de espículo, índice de lóbulo, dimensão fractal e circularidade do contorno. Os autores relataram ter alcançado bons resultados na classificação de 93 imagens obtidas na base de dados do MIAS (*Mammographic Image Analysis Society*) [Suckling et al. 1994]. Em [Guliato et al. 2010] os autores apresentam um novo descritor de silhueta do contorno, utilizando a curva de Hilbert, denominado *2Hq Feature*. A silhueta de um contorno é definida como uma imagem em que o interior do contorno é preenchido pela cor branca, e o fundo, pela cor preta. O *2Hq Feature* foi utilizado na classificação de 111 contornos de lesão de mama [Alto et al. 2005]. A classificação das lesões como benignas ou malignas foi realizada usando uma rede neural. Os experimentos mostraram alta taxa de precisão ($AUC = 0.99$). Apesar dos resultados obtidos, o *2Hq Feature* proposto por [Guliato et al. 2010] apresenta alta dimensionalidade e foi desenvolvido para caracterizar silhuetas bidimensionais.

Neste artigo propomos uma modificação do descritor proposto por [Guliato et al. 2010], de tal forma a reduzir a sua dimensionalidade e será referido neste trabalho como *2HqRF*. Propomos também estender o *2HqRF* para descrever silhuetas tridimensionais, denominado *3HqRF*. Uma vez obtidas as características que descrevem as silhuetas 2D ou 3D das lesões de mama, estas podem ser utilizadas em tarefas de classificação de padrões ou de recuperação de imagens por conteúdo. Neste trabalho exploraremos o uso dos descritores *2HqRF* e *3HqRF* (*HqRF* para simplificar) para tarefas de recuperação de imagem, utilizando o método KNN (*K-Nearest Neigh-*

bors). Os experimentos para a avaliação dos resultados obtidos pelo descritor *2HqRF* foram realizados com a mesma base de dados utilizada em [Guliato et al. 2008b]. Para avaliar os resultados obtidos pelo descritor *3HqRF*, criamos uma base de dados sintética que simula lesões em silhuetas arredondas ou espiculadas.

O restante deste artigo é organizado da seguinte forma: na Seção 2 são apresentados os conceitos teóricos necessários ao entendimento do trabalho; na Seção 3, o descritor proposto *2HqRF* e sua extensão para o espaço 3D são apresentados; a Seção 4 descreve as bases de dados utilizadas; os experimentos e uma discussão dos resultados obtidos encontram-se na Seção 5; finalmente, a Seção 6 traz a conclusão do trabalho.

2. Conceitos Teóricos

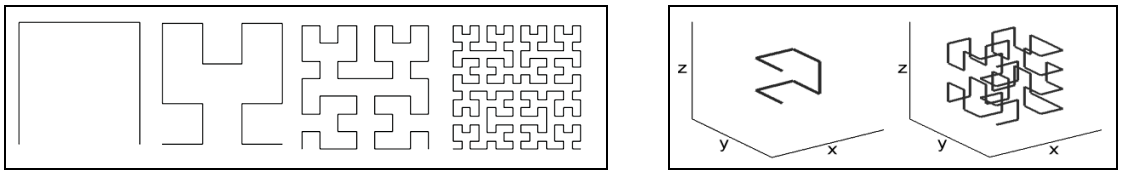
Nesta seção são resumidos os conceitos básicos necessários para o entendimento deste trabalho, apontando suas características de maior interesse.

2.1. Curva de Preenchimento de Espaço de Hilbert

Uma curva de preenchimento de espaço é um caminho contínuo que visita todos os pontos em uma grade n-dimensional e nunca se cruza [Prusinkiewicz and Lindenmayer 2004]. Ela pode ser aplicada a imagens ou volumes (neste caso a matriz de *pixels* ou o volume de *voxels* são as grades que a curva irá percorrer).

Uma característica deste tipo de curva é a sua capacidade de preservar a localização da informação, produzindo uma alta correlação dos dados obtidos. Em outras palavras, diz-se que características próximas na imagem tendem, tanto quanto possível, a permanecer próximas na representação unidimensional obtida pela curva. Dentre as diversas curvas de preenchimento de espaço, a de Hilbert é a mais popular, uma vez que é também a que apresenta uma maior correlação dos dados [Ebrahim et al. 2008].

A Figura 1a mostra as curvas de Hilbert no espaço 2D com níveis 1, 2, 3 e 4. Para obter a curva de nível $(n + 1)$, quatro cópias da curva de nível n são devidamente transladadas, rotacionadas e conectadas [Ebrahim et al. 2008]. A curva de Hilbert pode ser estendida para o espaço tridimensional. Sua construção, similar à curva no espaço 2D, utiliza oito cópias da curva de nível n para obter a curva de nível seguinte $(n + 1)$ [Prusinkiewicz and Lindenmayer 2004, Armstrong et al. 2009]. A Figura 1b ilustra as curvas de Hilbert no espaço tridimensional para $n = 1$ e $n = 2$.



(a) **(b)**

Figura 1. a) Curvas de preenchimento de espaço de Hilbert no espaço 2D (níveis 1 a 4). b) Curvas tridimensionais de Hilbert para os níveis 1 e 2.

2.2. Ganho Acumulativo Descontado

O ganho acumulativo descontado (do inglês *Discounted Cumulative Gain* - DCG) é uma medida de avaliação de desempenho que tem sido muito utilizada em aplicações

de recuperação de informação e em algoritmos de busca na *Web* [Croft et al. 2009]. A medida valoriza mais fortemente aqueles resultados corretos que ocorrem no início da lista de respostas em uma dada consulta (*ranking list*) [Manning et al. 2009], [Shilane et al. 2004].

Suponha-se que n candidatos sejam recuperados para uma consulta. O candidato na *i-ésima* posição da lista de respostas é associado a um valor rel_i, que indica a relevância deste candidato para a consulta. Os valores para o DCG e DCG normalizado para o intervalo [0,1] ($NDCG$) são definidos nas Equações 1 e 2, respectivamente [Croft et al. 2009]:

$$DCG_k = rel_1 + \sum_{i=2}^{k} \frac{rel_i}{log_2 i}, k \leq n \tag{1}$$

$$NDCG_k = \frac{DCG_k}{IDCG_k} \quad IDCG_k = 1 + \sum_{i=2}^{|C|} \frac{1}{log_2 i} \tag{2}$$

em que $rel_i \in \{0, 1\}$, k indica o número de candidatos na lista de respostas, IDCG corresponde ao valor ideal do DCG (obtido com uma lista de classificação perfeita) e $|C|$ é o número de elementos da classe do objeto de consulta.

2.3. O Descritor *2Hq Feature* Proposto em [Guliato et al. 2010]

Nesta Seção, descrevemos sucintamente o descritor *2Hq Feature* confomre proposto por [Guliato et al. 2010] para descrever silhuetas 2D usando a curva de Hilbert. A silhueta de um objeto é dividida em sub-regiões, de acordo com o nível de decomposição L definido *a priori*. Cada uma dessas sub-regiões é percorrida, independentemente, pela curva de Hilbert 2D com o objetivo de mapear a representação bidimensional para uma representação unidimensional, que será posteriomente subamostrada e normalizada para o intervalo [0,1]. Finalmente, o descritor *2Hq Feature* é obtido pela composição das representações normalizadas de cada sub-região.

Para um melhor entendimento, considere-se I uma silhueta e L o nível de decomposição da imagem. No nível L, a silhueta I é dividida em 4^L sub-regiões $r_i, i = 1, ..., 4^L$. A representção normalizada, denominada $2Hq_i$ é obtida para cada sub-região r_i. O processo para obter o $2Hq_i$ é ilustrado na Figura 2.3. O descritor *2Hq Feature* que descreve I é composto pelo conjunto de todos os Hq_i, ou seja, $2HqFeature = \{2Hq_1, ..., 2Hq_{4^L}\}$.

O processo para a obtenção do *2Hq Feature* é sumarizado a seguir:

1. *Redimensionamento*: nesta primeira etapa, as silhuetas são redimensionadas para as dimensões $2^N \times 2^N$, onde N é um inteiro positivo. Esta etapa evita a perda de informações, uma vez que a curva de Hilbert irá passar por todos os *pixels* de uma imagem com tamanho $2^N \times 2^N$.
2. *Decomposição*: cada imagem é dividida em quatro sub-regiões de tamanho igual. Cada sub-região pode ser novamente dividida em outras quatro sub-regiões e assim por diante, em diversos níveis.
3. *Aplicando a curva de Hilbert*: a curva de Hilbert é aplicada a cada sub-região. O resultado é uma representação unidimensional de cada sub-região, com a vantagem de manter a relação de localidade entre os *pixels* (uma característica da curva de Hilbert).

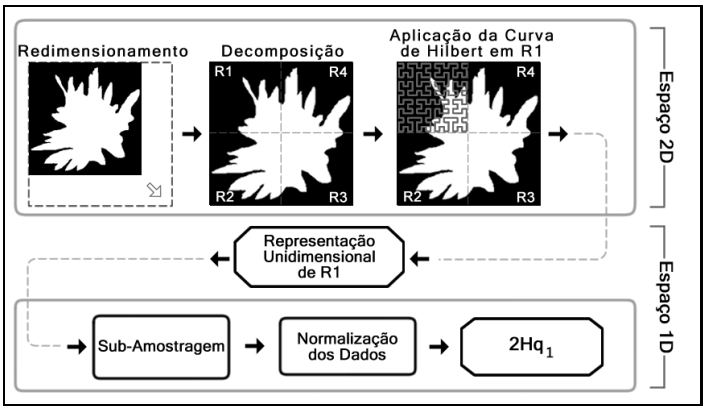


Figura 2. Fluxograma do processo empregado para obtenção do $2Hq_1$ de uma silhueta. Neste exemplo, é utilizada uma decomposição com $L = 1$.

4. *Subamostragem*: a representação obtida na etapa anterior é subamostrada, para um número predefinido q, pela aplicação sucessiva da transformada discreta de wavelet, considerando apenas o coeficiente de aproximação.
5. *Normalização*: os dados subamostrados para cada sub-região r_i são normalizados para o intervalo $[0, 1]$, produzindo o $2Hq_i$. É importante notar que, feita a decomposição, todas as sub-regiões são processadas de maneira independente, podendo ser processadas em paralelo.

3. Proposta dos descritores $HqRF$

3.1. Proposta para o Descritor de Silhuetas 2HqRF

O descritor *2Hq Feature* proposto por [Guliato et al. 2010] tipicamente utiliza como parâmetros: $q = 256$ e $L = 1$. Neste caso, o *2Hq Feature* é formado pela composição de quatro vetores de 256 elementos cada um. À medida que L aumenta, o número de vetores aumenta exponencialmelnte. Esta solução, tendo apresentado ótimos resultados em tarefas de classificação, seria praticável também na recuperação de imagens por conteúdo.

Neste trabalho propomos reduzir a dimensionalidade do *2Hq Feature* utilizando uma rede neural artificial (RNA). O número de elementos que compõem o novo descritor, referido neste trabalho como *2HqRF*, depende do número de classes da aplicação. O uso da RNA para este fim foi inspirado pelo trabalho de [Barcelos et al. 2008].

O mapeamento do descritor *2Hq Feature* para um vetor de característica de alto nível, aqui denominado por *2HqRF*, é feito por uma RNA *feed-forward*, multicamada com *back propagation*. A rede possui 4^L entradas, uma para cada $2Hq_i$ obtido, conforme Seção 2. A ilustração da RNA é apresentada na Figura 3.

3.2. Extensão do *2HqRF* para descrever silhuetas tridimensionais

O método para a obtenção do descritor $2HqRF$, descrito anteriormente, pode ser estendido para obter descritores de forma no espaço tridimensional. Para isto, é necessário que:

i) os objetos sejam definidos como volumes de *voxels*;

ii) as etapas 1, 2 e 3 descritas na Seção 2.3 sejam ajustadas para o espaço 3D e;

iii) a RNA seja ajustada para mapear os dados para obter o *3HqRF*.

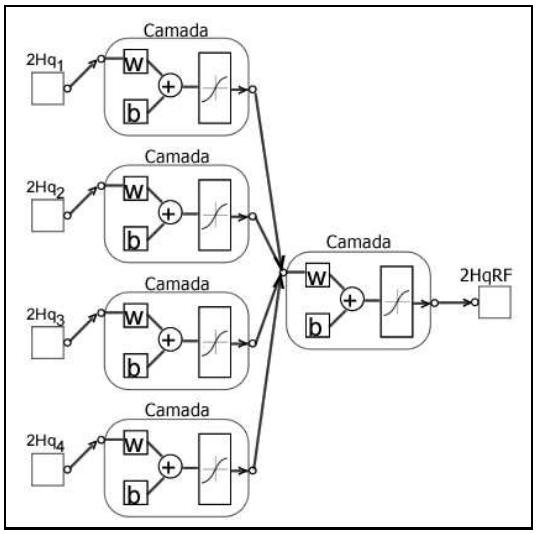


Figura 3. Arquitetura da RNA utilizada para obter o *2HqRF* (nesta ilustração é considerada a decomposição com $L = 1$).

4. Bases de Dados

Para a realização dos experimentos apresentados na Seção 5, duas bases de dados foram utilizadas: a primeira contendo contornos de lesões em duas dimensões, e a segunda contendo objetos tridimensionais que simulam lesões tridimensionais da mama. Esta seção descreve ambas as bases de dados.

4.1. Base de Dados de Contornos 2D

Os contornos presentes nesta base têm origem em três fontes: *Screen Test: The Alberta Program For The Early Detection Of Breast Cancer* [Alto et al. 2005]; *Mammographic Image Analisys Society* (MIAS, UK); e *Foothills Hospital* de Calgary [Rangayyan et al. 2000, Rangayyan et al. 1997]. O contorno de cada lesão foi manualmente traçado por um radiologista especializado em momografias. A base de dados combinada possui 111 contornos, incluindo casos típicos e atípicos de lesões benignas (65) e tumores malignos (46). A classificação como maligna ou benigna de cada contorno foi baseada em biópsia. Esta base de dados é aqui denominada BC-111. A Figura 4 exibe algumas silhuetas da base de dados BC-111.

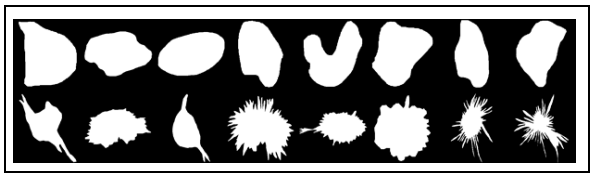


Figura 4. Exemplo de silhuetas de lesões de mama presentes na base BC-111.

4.2. Base de Dados de Volumes de *Voxels*

Com o objetivo de avaliar a potencialidade do descritor $3HqRF$ para caracterizar objetos tridimensionais arredondados ou espiculados, criamos a base de dados W3D-111[1]. A Figura 5 ilustra alguns objetos desta base.

[1]Disponível no endereço eletrônico `http://www.facom.ufu.br/~walter/hqrf`

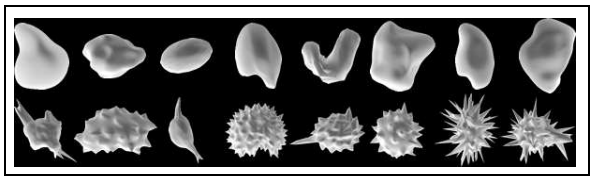


Figura 5. Exemplo de objetos da base de dados W3BC-111.

Os objetos são divididos em duas classes, espiculados (62 objetos) e arredondados (49 objetos), e são representados como volumes de *voxels* com dimensões $256 \times 256 \times 256$.

5. Experimentos e Discusão dos Resultados

Nesta seção, os descritores de forma $2HqRF$ e $3HqRF$ são avaliados por meio de experimentos de recuperação de formas. A RNA foi treinada utilizando a *10-fold cross-validation*. A similaridade entre dois descritores é medida por meio da distância Euclidiana. Os resultados são avaliados em termos da média das curvas de Precisão-Revocação (PR) e da média dos valores NDCG.

5.1. Experimentos Realizados no Espaço 2D

A fim de avaliar a qualidade dos resultados obtidos com o descritor de forma $2HqRF$, proposto neste artigo, é realizada uma comparação do método proposto com os fatores de forma SI_{TF} descrito em [Guliato et al. 2008a], CX_{TA} e FD_{TA} descritos em [Guliato et al. 2008b]. A base de dados utilizada neste experimento é a BC-111. Os parâmetros utilizados para a obtenção do $2HqRF$ são listados na Tabela 1. Na Figura 6 são exibidos os 20 primeiros resultados de consultas realizadas neste experimento (o primeiro objeto em cada linha é o objeto de consulta). A Figura 7a exibe as curvas PR média para o $2HqRF$ e para os métodos SI_{TF}, CX_{TA} e FD_{TA}. Os valores da média dos NDCG são apresentados na Figura 7b. Note que o método proposto obteve uma precisão média de 100% para todos os valores de revocação, superando os demais métodos utilizados neste experimento.

Tabela 1. Parâmetros utilizados na obtenção do descritor $2HqRF$.

Parâmetro	**Valor**
Nível de decomposição (L)	3
Família Wavelet	Haar
Num. de neurônios nas camadas escondidas	20
Tamanho do $2HqF_i$ (q)	256

5.2. Experimentos Realizados no Espaço 3D

A base de dados W3BC-111 é utilizada para os experimentos no espaço tridimensional. Os parâmetros utilizados nestes experimentos são apresentados na Tabela 2. A Figura 8 exibe os 20 primeiros objetos recuperados em consultas realizadas (note que o objeto de consulta é o primeiro em cada linha). A curva PR média obtida neste experimento é ilustrada na Figura 9. O valor de NDCG médio obtido no mesmo experimento é de 0,99.

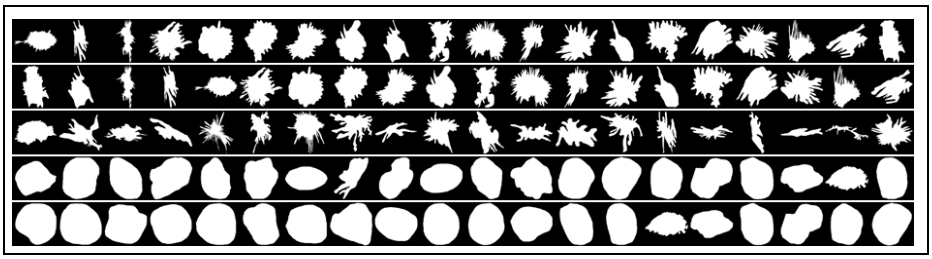
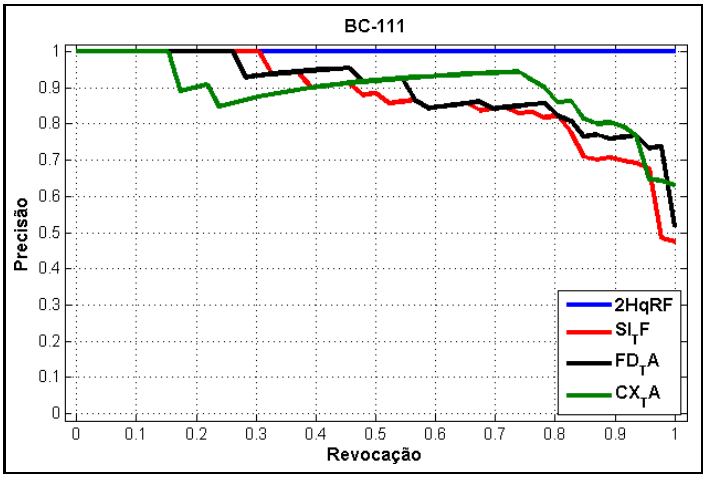


Figura 6. Consultas realizadas com a base de dados BC-111.



(a)

Descritor	NDCG
$2HqRF$	1.0000
SI_{TF}	0.8849
CX_{TA}	0.9656
FD_{TA}	0.9579

(b)

Figura 7. a) Curva precisão-revocação média obtida nos exeprimentos com a base BC-111. b) Valores de NDCG médio obtidos nos experimentos com a base de dados BC-111.

Tabela 2. Parâmetros utilizados na obtenção do descritor $3HqRF$.

Parâmetro	Valor
Nível de decomposição (L)	1
Família Wavelet	Haar
Num. de neurônios nas camadas escondidas	20
Tamanho do $2HqF_i$ (q)	256

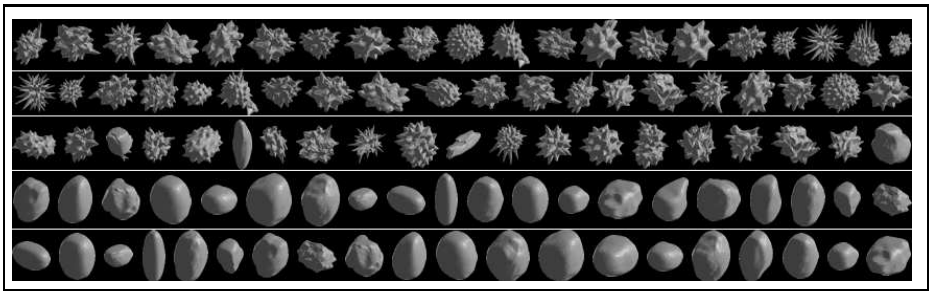


Figura 8. Consultas realizadas com a base de dados W3BC-111.

6. Conclusões

Neste artigo foram apresentados dois novos métodos para descrição de formas nos espaços 2D e 3D. Os descritores propostos foram aplicados à recuperação de tumores

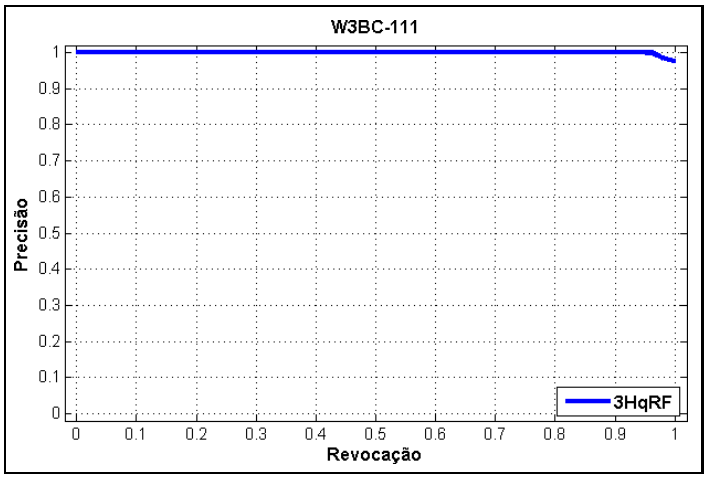


Figura 9. Curva precisão-revocação média obtida nos exeprimentos com a base W3BC-111.

de mama. A análise dos resultados mostrou uma superioridade do $2HqRF$, em termos dos valores médios de NDCG e de curvas PR, quando comparados a trabalhos relacionados. O $3HqRF$ mostrou potencialidade em discriminar formas arredondadas e espiculadas, que são características importantes para auxiliar o diagnóstico de câncer de mama em exames radiológios tridimensionais, como ressonância magnética ou tomografia computadorizada. Esforços têm sido feitos para se construir uma base com lesões de mama conseguidas a partir de exames de ressonânca magnética.

Agradecimentos

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) pelo apoio financeiro.

Referências

Alto, H., Rangayyan, R., and Desautels, J. (2005). Content-based retrieval and analysis of mammographic masses. *Journal of Electronic Imaging*.

American College of Radiology, Reston, VA (1998). Illustrated breast imaging reporting and data system BI-RADS^{TM}).

Armstrong, J., Ahmed, M., and Chau, S.-C. (2009). A rotation-invariant approach to 2d shape representation using the Hilbert curve. *Image Analysis and Recognition*, 5627:594–603.

Barcelos, C., Ribeiro, E., and Batista, M. (2008). Image characterization via multilayer neural networks. *International Conference on Tools with Artificial Intelligence*, pages 325–332.

Chen, W., Giger, M., and Bick, U. (2006). A fuzzy c-means (fcm)-based approach for computerized segmentation of breast lesions in dynamic contrast-enhanced mr images. *Acad. Radiol.*, 13(1):63–72.

Croft, W., Metzler, D., and Strohman, T. (2009). *Search Engines: Information Retrieval in Practice*. Addison Wesley.

Ebrahim, Y., Ahmed, M., Abdelsalam, W., and Chau, S.-C. (2008). Shape representation and description using Hilbert curve. *Pattern Recognition Letters*.

Guliato, D., de Oliveira, W., and Jr., C. T. (2010). A new feature descriptor derived from Hilbert space-filling curve to assist breast cancer classification. 1:303–308.

Guliato, D., Rangayyan, R., Carvalho, J., and Santiago, S. (2008a). Polygonal modeling of contours of breast tumors with the preservation of spicules. *IEEE Trans Biomed Eng*.

Guliato, D., Rangayyan, R., de Carvalho, J., and Santiago, S. (2008b). Feature extraction from the turning angle function for the classification of contours of breast tumors. *J. Digit Imaging*.

Gupta, S., P.F., C., and Markey, M. (2006). Breast cancer cadx based on BI_RADS^{TM} descriptors from two mammographic views. *Med. Phys.*, 33(6):1810–1817.

Heath, M., Bowyer, K., and Kopans, D. (1998). Current status of the digital database for screening mammography.

Manning, C., Raghavan, P., and Schütze, H. (2009). *An Introduction to Information Retrieval*. Cambridge University Press.

Prusinkiewicz, P. and Lindenmayer, A. (2004). *The Algorithmic Beauty of Plants*. Springer-Verlag.

Rangayyan, R., El-Faramawy, N., Desautels, J., and Alim, O. (1997). Measures of acutance and shape for classification of breast tumors. *IEEE Transactions on Medical Imaging*.

Rangayyan, R., Mudigonda, N., and Desautels, J. (2000). Boundary modelling and shape analysis methods for classification of mammographic masses. *Medical and Biological Engineering and Computing*.

R.G.-Caballero, C.J.G.-Orellana, H.M.G.-Velasco, and M.M.-Macías (2007). Independent component analysis applied to detection of early breast cancer signs. *Computational and Ambient Intelligence*.

Shilane, P., Min, P., Kazhdan, M., and Funkhouser, T. (2004). The princeton shape benchmark. *Shape Modeling International*.

Suckling, J., Parker, J., Dance, D., Astley, S., Hutt, I., Boggis, C., Ricketts, I., Stamatakis, E., Cerneaz, N., Kok, S., Taylor, P., Betal, D., and Savage, J. (1994). The mammographic image analysis society digital mammogram database.

Turnbull, L., Brown, S., Harvey, I., Olivier, C., Drew, P., Napp, V., Hanby, A., and Brown, J. (2010). Comparative effectiveness of mri in breast cancer (comice) trial: a randomised controlled trial. *The Lancet*, 375(9714):563–571.

Weiqiang, Z., Xiangmin, X., and Wei, H. (2008). Shape and boundary analysis for classification of breast masses. *International Symposium on Computational Intelligence and Design*.

WHO (2010). Breast cancer: prevention and control.

Desenvolvimento Dirigido a Modelos para Aplicações em Saúde

A.T.A. Gomes, A. Ziviani, V.U. Procópio, V.M. Moreira, B.S.P.M. Correa, I.M. Teixeira

Laboratório Nacional de Computação Científica (LNCC) – Petrópolis, RJ – Brasil
Instituto Nacional de C&T em Medicina Assistida por Computação Científica (INCT-MACC)

{atagomes,ziviani,vmacedo,brunos,iuri}@lncc.br

Abstract. *We propose a new model-driven method for healthcare information systems that targets a higher degree of maintainability and reuse by employing crosscutting clinical data models and architecture models. To support this method on the automatic synthesis of such systems, we are progressing work on the development of a software workbench that weaves together clinical data models described according to the openEHR specifications and architecture models specified in the Acme language.*

Resumo. *Neste artigo propomos um novo método de desenvolvimento dirigido a modelos para sistemas de informação em saúde que objetiva um alto grau de manutenibilidade e reuso empregando transversalidade na concepção de modelos de dados clínicos e modelos arquiteturais. Para dar suporte a esse método na geração automática desses sistemas, estamos desenvolvendo um ferramental de software capaz de acoplar modelos de dados clínicos descritos de acordo com as especificações openEHR e modelos arquiteturais especificados na linguagem Acme.*

1. Introdução

Recentemente, tem havido considerável atenção sobre o potencial de Desenvolvimento Dirigido a Modelos – DDM no suporte ao desenvolvimento de Sistemas de Informação em Saúde – SIS. De forma geral, as iniciativas existentes advogam o uso do padrão MDA (*Model Driven Architecture*) [Miller e Mukerji 2003]. Conforme descrito em [Raghupathi e Umar 2008], a abordagem MDA viabiliza o desenvolvimento de SIS: (i) com baixo custo e *time-to-market*; (ii) interoperáveis; (iii) independentes de plataforma; e (iv) melhor manuteníveis. Contudo, em nossa visão, há dois desafios principais a serem ainda tratados para que DDM alcance um nível maior de interoperabilidade, manutenibilidade e reuso no desenvolvimento de SIS:

Complexidade na modelagem dos dados clínicos. Nossa experiência prévia no desenvolvimento de SIS [Teixeira et al. 2009] mostra que mudanças na modelagem de dados clínicos são constantes devido à complexidade de transcrever o conhecimento do profissional médico para dentro do sistema. Essa complexidade pode se traduzir em problemas de manutenibilidade e interoperabilidade a médio e longo prazo. Como consequência, a modelagem dos dados clínicos acaba exigindo atenção demasiada da equipe de desenvolvimento, sendo que constitui apenas uma parcela do esforço de desenvolvimento total de um SIS.

Modelagem arquitetural ad-hoc. Técnicas de projeto arquitetural são especialmente úteis no domínio de SIS. Nesse sentido, a abordagem MDA é capaz de facilitar o reuso de uma boa arquitetura, mas não auxilia o projeto de novas boas arquiteturas. Tipicamente, sistemas dentro de uma mesma família de SIS (p.ex., emergência pré-hospitalar, vigilância epidemiológica, monitoramento intra-hospitalar de pacientes) empregam estilos arquiteturais semelhantes (p.ex., cliente-servidor, *publisher-subscriber*, *peer-to-peer*). Defendemos que esses estilos devem poder ser melhor explorados pelas equipes de desenvolvimento visando projetos arquiteturais que propiciem manutenibilidade e reuso de longo prazo.

Esses dois desafios têm nos motivado a desenvolver um método de DDM para SIS mais abrangente que os baseados somente na abordagem MDA. Nosso método tem como objetivos:

Transversalizar modelos arquiteturais e de dados clínicos, de modo que especialistas em saúde não se importem em como os dados são processados, armazenados e transportados, e que arquitetos de software não se importem em como os dados são representados em SIS;

Dualizar modelos de dados clínicos, para separar a modelagem computacional da modelagem médica, o que permite reduzir o esforço da equipe de desenvolvimento na manutenção de SIS;

Estilizar modelos arquiteturais, para capturar características comuns dentro de famílias de SIS como artefatos reusáveis de projeto.

Para dar suporte ao método proposto, temos desenvolvido um ferramental de software inovador, que é apresentado na Seção 2. O estado atual de desenvolvimento do ferramental e as perspectivas futuras de aplicação desse ferramental são apresentados na Seção 3.

2. Ferramental de DDM Proposto

O ferramental proposto é apresentado na Figura 1. Todas as ferramentas ilustradas estão sendo prototipadas como *plugins* para o ambiente Eclipse[1] e são detalhadas adiante.

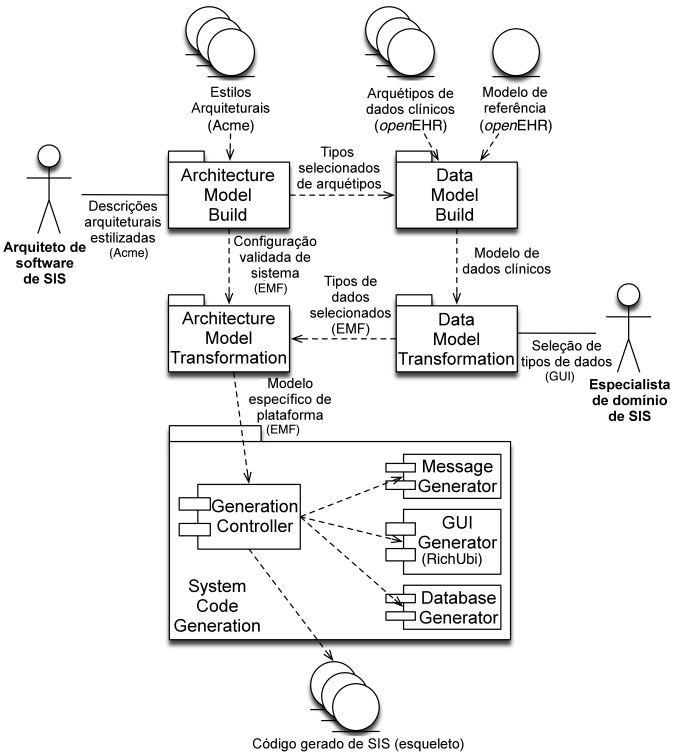


Figura 1. Ferramental de DDM proposto.

2.1. Tranversalização de Modelos

Para tranversalizar modelos arquiteturais e de dados clínicos, consideramos dois atores diferentes no método proposto: o *especialista de domínio* e o *arquiteto de software*. O primeiro

[1] http://www.eclipse.org

especifica modelos de dados clínicos de SIS (vide Subseção 2.2) e o segundo modelos arquiteturais de SIS (vide Subseção 2.3). Esses dois modelos são transformados, conforme descrito nas subseções seguintes, em formatos compatíveis com o metamodelo EMF (*Eclipse Modeling Framework*)[2] e passados à ferramenta Architeture Model Transformation. Essa ferramenta implementa um motor de composição e transformação de modelos baseado em EMF que gera um modelo de plataforma específica – PSM (*Platform Specific Model*) também baseado em EMF, a partir do qual esqueletos de SIS são gerados (vide Subseção 2.4).

2.2. Modelagem de Dados Clínicos

Para dualizar modelos de dados clínicos, adotamos no método proposto o conjunto de especificações *open*EHR [Beale 2002]. No *open*EHR, a tarefa da modelagem da informação é mantida separada daquela da modelagem de dados clínicos. Modeladores de informação definem um *modelo de referência*, sobre o qual comitês multi-disciplinares de analistas de sistemas e médicos produzem os chamados *arquétipos de dados clínicos*, que restringem os tipos de dados definidos no modelo de referência. Por exemplo, no *open*EHR "Pressão sanguínea" é um arquétipo com tipos de dados como "Sistólica" e "Diastólica", que definem restrições sobre os valores aceitos pelo tipo de dados mais geral "Quantidade" no modelo de referência.

No ferramental de software proposto, ambos o modelo de referência e os arquétipos de dados clínicos das especificações *open*EHR servem como entrada para a ferramenta Data Model Build. Essa ferramenta também recebe da ferramenta Architecture Model Build (vide Subseção 2.3) uma indicação sobre quais tipos de arquétipos de dados clinicos são de interesse – observação, instrução, avaliação ou ação – para a familia específica de SIS da qual o sistema a ser gerado faz parte. A partir dessas entradas, a ferramenta Data Model Build gera um modelo completo de dados clínicos. Esse modelo é repassado para a ferramenta Data Model Transformation, que: (i) permite que o especialista selecione os tipos de dados no modelo de dados clinicos completo que são de interesse para o SIS específico a ser gerado – por exemplo, o tipo de dado "Muscle Tone" é crucial para obstetras (devido ao método Apgar), mas não faz sentido para serviços de emergência para pacientes de infarto do miocárdio; e (ii) traduz os tipos de dados selecionados em um formato compatível com EMF.

2.3. Modelagem Arquitetural

Para estilizar modelos arquiteturais, adotamos no método proposto a linguagem de descrição arquitetural Acme [Garlan et al. 1997]. Acme oferece construções de alto nível – como componentes, conectores e configurações – que incorporam informações sobre a estrutura de um sistema abstraindo detalhes da implementação do mesmo. Acme também provê suporte para *estilos arquiteturais* [Garlan et al. 1994], os quais prescrevem tipos de construções e restrições na composição dos mesmos. Por exemplo, um estilo "Pipeline" pode incluir um tipo de componente "Filter" e um tipo de conector "Pipe" além de restrições que impedem composições circulares de tais construções em uma configuração de sistema. Uma configuração seguindo um estilo específico em Acme pode assim ser vista como uma instância de tal estilo.

No ferramental de software proposto, estilos arquiteturais descritos em Acme servem como entrada para a ferramenta Architecture Model Build. Tais estilos definem regras para a configuração dos sistemas pertencentes a diferentes famílias de SIS – até o momento definimos estilos Acme para sistemas de emergência pré-hospitalar e vigilância sindrômica. A

[2] `http://www.eclipse.org/modeling/emf/`

ferramenta Architecture Model Build: (i) valida uma configuração de sistema especificada por um arquiteto de software de acordo com as regras de estilo da família SIS pretendida; (ii) indica à ferramenta Data Model Build os tipos de arquétipos de dados clínicos de interesse para essa família – por exemplo, sistemas de emergência manipulam a maior parte dos dados de observação e instrução, enquanto que sistemas de vigilância estão normalmente interessados em dados de observação e avaliação; e (iii) traduz a configuração validada de sistema em um formato compatível com EMF.

2.4. Geração de Código

A ferramenta System Code Generation implementa um motor de tranformação baseado em EMF que gera esqueletos de código para SIS a partir de um PSM exportado pela ferramenta Architecture Model Transformation. A ferramenta System Code Generation é composta por quatro componentes. O componente Generation Controller analisa o PSM e invoca um dos três componentes geradores específicos dependendo da parte do PSM que está sendo analisada: (i) o componente Message Generator para a geração de entidades de protocolo e formatos de mensagens (p.ex. definições WSDL, *stanzas* XMPP) (ii) o componente GUI Generator, que é baseado no RichUbi, uma ferramenta baseada em EMF para geração de código para interfaces gráficas de usuário [Cirilo et al. 2010]; e (iii) o componente Database Generator para a geração de código especifico de banco de dados (p.ex., esquemas, mapeamentos O/R).

3. Perspectivas e Trabalhos Futuros

Até o momento, todas as ferramentas apresentadas, à exceção da Architecture Model Transformation, já foram prototipadas – a composição dos modelos arquiteturais e de dados clínicos ainda é feita manualmente. Temos testado o ferramental em condições experimentais usando EJB, AJAX, Webservices e Hibernate como plataforma alvo. Como trabalho futuro, pretendemos validar a proposta completa através da modelagem de SIS existentes e da comparação dos esqueletos de código gerados pelo ferramental proposto com tais sistemas.

Referências

Beale, T. (2002). Archetypes: Constraint-based domain models for future-proof information systems. *open*EHR standard document.

Cirilo, C. E., do Prado, A. F., de Souza, W. L., e Zaina, L. A. M. (2010). Model driven RichUbi: A model driven process for building rich interfaces of context-sensitive ubiquitous applications. Em *Proc. of the 28th SIGDOC*, pág. 207–214, New York, NY, USA. ACM.

Garlan, D., Allen, R., e Ockerbloom, J. (1994). Exploiting style in architectural design environments. *SIGSOFT Softw. Eng. Notes*, 19:175–188.

Garlan, D., Monroe, R. T., e Wile, D. (1997). ACME: An Architecture Description Interchange Language. Em *CASCON'97: Proceedings*, pág. 169–183, Toronto, Ontario.

Miller, J. e Mukerji, J. (2003). MDA guide version 1.0.1. Technical report, OMG.

Raghupathi, W. e Umar, A. (2008). Exploring a model-driven architecture (MDA) approach to health care information systems development. *Int J Med Inform*, 77:305–14.

Teixeira, I. M., Vicoso, R. P., Correa, B. S. M. P., Gomes, A. T. A., e Ziviani, A. (2009). Suporte remoto ao atendimento médico emergencial via dispositivos móveis. Em *Anais do IV Workshop de Informática Médica (WIM)*, Porto Alegre, RS, Brasil. SBC.

TMJsim - Simulador de Auxílio ao Diagnóstico, Planejamento Pré-cirúrgico e Acompanhamento de Tratamento Bucomaxilofacial.

Marta B. Villamil, Eduardo Garcia, Robson Machado da Rosa
Programa Interdisciplinar de Pós-Graduação Aplicada
Universidade do Vale do Rio dos Sinos
Av. Unisinos, 950 - B. Cristo Rei / CEP 93.022-000 - São Leopoldo (RS)
mbvillamil@unisinos.br,EduardoGarcia@terra.com.br,robsongdev@gmail.com

Abstract.

The use of jaw motion simulators in dentistry education, orthodontic adjustment of occlusions, or in the preoperative planning of craniofacial surgery can be extremely useful, improving diagnosis and postoperative treatment. This article presents the TMJsim, a simulator of mandible motion constructed from real data coming from Computed Tomography and Magnetic Resonance images. The virtual joints model which composes the simulator receives points captured from the lower incisal point motion. The contribution of each muscle in temporomandibular movement is approached from the implementation of Hill actuators model and the concept of curves of insertion.

Resumo.

O uso de simuladores de movimento da mandíbula na educação em odontologia, no ajuste ortodôntico de oclusões, ou no planejamento pré-operatório de cirurgias craniofaciais pode ser extremamente útil, melhorando o diagnóstico e o tratamento pós-operatório. Este artigo apresenta o TMJsim, um simulador de movimento da mandíbula construído a partir de dados reais advindos de imagens de Tomografia Computadorizada e Ressonância Magnética. Os parâmetros que alimentam o modelo de juntas virtuais, que compõe o simulador, são pontos capturados a partir do ponto incisal inferior mandibular. A contribuição de cada músculo temporomandibular no movimento é abordada a partir da implementação do modelo de atuadores de Hill e do conceito de curvas de inserção aliado a restrição de movimento pelos tendões.

1. Introdução

Os simuladores médicos tem auxiliado no ensino de procedimentos terapêuticos e de diagnóstico assim como na representação de conceitos médicos e no processo de decisão de profissionais de saúde. Os simuladores têm sido desenvolvidos para procedimentos básicos desde a extração de sangue até cirurgia laparoscópica e de trauma. Também são importantes no auxílio à prototipagem de novos utensílios da engenharia biomédica. São aplicados a pesquisa e desenvolvimento de ferramentas úteis a novas terapias, tratamentos e diagnóstico precoce em medicina. O maior mérito do uso de simuladores está na diminuição da desnecessária alta frequência com que os pacientes sofrem com as adversidades clínicas advindas de mãos de médicos inexperientes.

Na maior parte dos simuladores, os componentes visuais do procedimento são reproduzidos por técnicas de computação gráfica, enquanto os componentes baseados em toque são reproduzidos por dispositivos de *haptic feedback* combinados com simulações físicas em resposta às ações do usuário. Simulações médicas desta natureza muitas vezes usam dados reais de Tomografia Computadorizada e Ressonância Magnética do paciente para aumentar o realismo.

Na modelagem de figuras humanas, o maior desafio tem sido a representação de suas características com o maior grau de realismo possível. As principais investigações nesta área situam-se na modelagem do movimento humano, deformações de músculos e pele, *rendering* realista da pele, olhos e cabelo e na modelagem das emoções e do comportamento humano. Através destas investigações se chega a simulações para posterior análise de muitas situações envolvendo humanos inclusive as análises médicas. Estas simulações permitem que se derive informações dos modelos com o objetivo de predizer ou reproduzir comportamentos que seriam observados em situações reais.

Dentre os sistemas de articulações presentes no corpo humano, o sistema mastigatório propicia muitos aspectos para investigação, pois os eventos físicos que ocorrem neste sistema, durante a função e eventual disfunção, são de difícil conceituação. A simulação dos elementos estruturais e funcionais que trabalham em conjunto é especialmente útil quando relaciona causa e efeito. As contrações de vários músculos de diferentes formas e tamanhos guiados por duas juntas torna, ainda, a cinemática e física do sistema não totalmente conhecido (LEMOINE et al., 2005).

Este trabalho apresenta o TMJsim - Simulador de auxílio ao diagnóstico, planejamento pré-cirúrgico e acompanhamento de tratamento bucomaxilofacial. Ele consiste de um sistema, para utilização na odontologia e medicina, que permite um avanço no diagnóstico e no planejamento de tratamentos e procedimentos cirúrgicos, nos casos que envolvem a Articulação Temporomandibular, a ATM. Esta é uma articulação dupla bilateral e é através dela que a mandíbula, único osso móvel do crânio, conecta-se à base craniana.

Há um grande número de patologias relacionadas à ATM e isso se deve não só ao seu uso regular, mas também às grandes forças aplicadas pelos músculos e à grande variedade de movimentos que cada ATM é capaz de produzir. Dor de cabeça, dor na ATM durante a movimentação mandidular, sensação de travamento da mandíbula, estalos durante a mastigação, limitação de abertura da boca, dor em certas regiões da face e no ouvido e dificuldade na oclusão (fechamento) dos dentes, são alguns dos sintomas desta disfunção. Algumas das causas destas disfunções são os acidentes que envolvem a mandíbula, cabeça e pescoço ou doenças, como artrite.

Para diagnosticar as disfunções, uma das ferramentas essenciais são os Articuladores. Os Articuladores disponíveis atualmente no mercado, são instrumentos mecânicos que representam a articulação temporomandibular, a maxila e a mandíbula, na qual pode-se adaptar modelos que representam as duas arcadas. A utilização de um Articulador ocorre em casos de diagnóstico, pois ele busca simular os movimentos da ATM, a partir de ajustes feitos com base em certos dados do paciente. O Articulador auxilia na fase de tratamento pois permite analisar o impacto de determinados procedimentos.

A construção do TMJsim engloba pesquisas relacionadas as áreas de computação gráfica, realidade virtual e medicina e tende a tornar-se um novo paradigma como ferramenta pois ele simulará e reproduzirá os movimentos da ATM de forma realística, permitindo uma análise completa do caso em tratamento.

Ele é um software que vem a substituir e incrementar de maneira inovadora o trabalho que é realizado pelos articuladores mecânicos. Inicialmente o TMJsim reconstroe a ATM virtualmente, gerando um modelo 3d, a partir de exames como Tomografia Computadorizada e Ressonância Magnética. Uma vez obtida uma cópia virtual da ATM, o software simula os movimentos reais, com grande flexibilidade e facilidade na parametrização. Com ele será possível analisar em profundidade um determinado caso numa fase de diagnóstico ou antever o resultado de procedimentos,inclusive cirúrgicos.

2. Contexto

Recentemente, trabalhos como e de Stavness et al. (2008) estão sendo desenvolvidos já com o intuito de simular os efeitos de reconstruções mandibulares. Tratamentos de pacientes com câncer muitas vezes requerem a retirada de parte da estrutura óssea e muscular, alterando a anatomia facial e as funções dos músculos. Estas alterações usualmente requerem uma reconstrução estética e funcional da mandíbula. Um sistema de simulação pré-operatório pode ajudar os especialistas a planejar os procedimentos, potencializando a sua análise decorrente da experiência e intuição. O trabalho de Stavness et al. (2008) utiliza uma plataforma de simulação, inicialmente descrita por Stavness et al. (2006) e chamada de Artisynth, baseia-se na simulação de corpos rígidos, com algoritmos de detecção de colisão e atuadores ponto-a-ponto de tipos variáveis, como atuadores musculares dinâmicos não-lineares do tipo Hill (HILL, 1938). O controle da simulação é feito através de uma curva de nível de ativação versus linha de tempo, para cada músculo. Outra opção de controle neste caso é a otimização para controle automático, onde é utilizada modelagem reversa para geração dos padrões de ativação dos músculos, de forma que a mandíbula siga uma trajetória padronizada. O trabalho de Sifakis et al. (2005) foi utilizado como base para o desenvolvimento do controle automático.

3. Materiais e Métodos

O uso de simuladores de movimento da mandíbula na educação em odontologia, no ajuste ortodôntico de oclusões, ou no planejamento pré-operatório de cirurgias craniofaciais pode ser extremamente útil, melhorando o diagnóstico e o tratamento pós-operatório.

A relação entre as ações musculares e o movimento mandibular é provavelmente o aspecto menos compreendido da biomecânica mandibular, apesar de seu significado importante. A mandíbula é um objeto que se move de acordo com leis dinâmicas. Possui atributos como massa, centro de massa e propriedades inerciais que não são geralmente descritos.

A oclusão é definida como qualquer contato entre as superfícies de incisão e/ou mastigação dos dentes superiores e inferiores (ZWEMER, 1993). Contudo a oclusão tem sido mais bem definida como a relação dinâmica e morfológica e funcional entre todos os componentes do sistema mastigatório (McNEILL, 2000). Não existe ainda um

estudo matemático que analise a frequência e força das colisões e possibilite a modelagem deste complexo sistema.

Um modelo completo do sistema mastigatório engloba, ainda, algumas investigações no que concerne a forças musculares, deformação de tecidos, formas de tratar colisões entre dentes e alimentos e uso de dispositivos de retorno de força para ajuste oclusal que são descritos mais detalhadamente a seguir.

3.1 Reconstrução do Modelo 3D

Para reproduzir e visualizar o movimento mandibular, é necessário que a mandíbula seja um objeto separado do resto da cabeça. O modelo 3D foi planejado com três malhas de triângulos: uma para a mandíbula, outra para o maxilar superior e outra para o resto do crânio (Figura 1) (Villamil 2005). Para a obtenção destas malhas, primeiro houve um processo de segmentação manual dos ossos do crânio e dentes e depois uma técnica de extração de malhas foi aplicada. Estes procedimentos foram aplicados usando o programa (SLICER, 2005), um software livre e de código aberto para visualização, registro, segmentação e quantificação de imagens médicas.

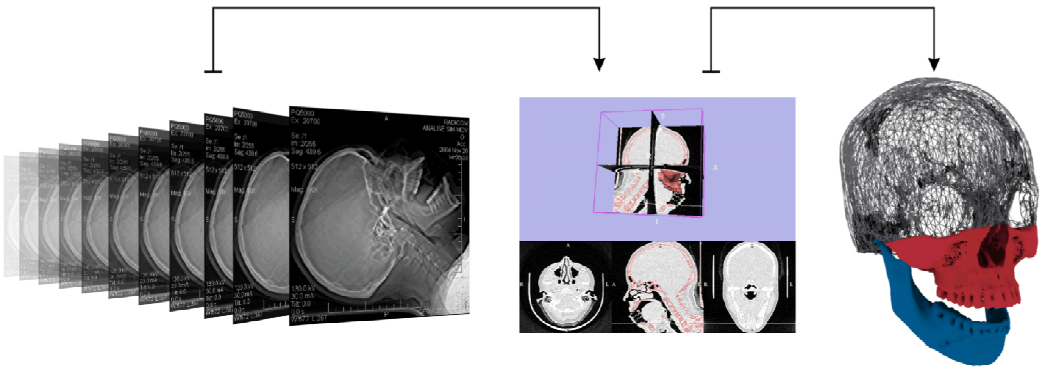


Figura 1. A partir das imagens de tomografia, a regiões dos ossos e dentes foram segmentadas para a obtenção das malhas de triângulos.

3.2 Posicionamento do de Centro de Rotação Inicial

O método conhecido como centro de rotação instantâneo (LEPERA, 1958), é frequentemente usado na análise mecânica de juntas para encontrar o centro de rotação quando o movimento é descrito apenas por rotações. O movimento da mandíbula é descrito como rotacional apenas nos primeiros milímetros do movimento de abertura da boca. Por isto, apenas o primeiro centro de rotação é calculado usando este método.

Após, a origem e direção dos próximos eixos de rotação são obtidos a partir do caminho descrito pelo ponto incisal e do tratamento da colisão entre a mandíbula e maxila.

Os centros de rotação iniciais (CRIs), para as ATMs esquerda e direita, são obtidos através dos cálculos que usam pontos anatômicos marcados em imagens de ressonância magnética. A área da mandíbula é segmentada em imagens da mandíbula em diferentes posições. Na primeira, com a boca totalmente fechada, e a outra, com a

boca parcialmente aberta com um centímetro entre os dentes incisivos superiores e inferiores (Figura 2).

Os três pontos anatômicos foram localizados no topo dos côndilos (*i*) e no centro entre os dentes incisivos inferiores(*j*). Para cada junta foi considerado o movimento destes dois pontos (o de cada côndilo e o incisal) (*i*t; *j*t) para o espaço de tempo t para t+1 (*i*t+1; *j*t+1). O centro de rotação para este incremento de movimento pode ser calculado erguendo bissetores perpendiculares entre as linhas (*i*t) a (*i*t+1) e (*j*t) a (*j*t+1). O centro de rotação está entre a interseção das duas bissetores (Figura 2).

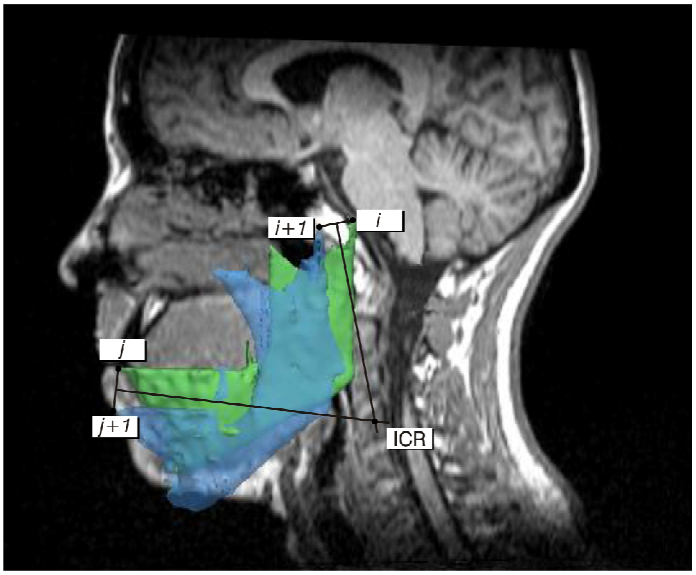


Figura 2. Representação do cálculo dos centros de rotação iniciais das ATMs ou origem da base vetorial que representam a ATM.

3.3 O Modelo de Articulação

O modelo da articulação temporomandibular (ATM) (Villamil et al. 2005, Villamil 2009, Olszewski et al. 2008) consiste em um par de juntas com três DOFs rotacionais e três DOFS translacionais cada uma, que movimentam um único osso, a mandíbula. As restrições do movimento se dão em função da colisão entre ossos e entre dentes, e pela própria topologia das juntas (o movimento gerado por uma junta interfere no posicionamento da junta adjacente). Cada junta tem o seu próprio sistema de referência, mas o seu comportamento em termos de rotação e translação é também dependente da junta adjacente. Os sistemas de referência da ATM são responsáveis pela posição e orientação do objeto gráfico que representa a mandíbula.

O modelo de movimento das ATMs é baseado numa sequência de posições e orientações de dois referenciais que representam as ATMs esquerda e direita. Esta sequência foi inferida a partir do caminho incisal capturado de um indivíduo com ATMs normais. O modelo pode ser considerado procedural, pois as transformações necessárias para realizar o movimento são obtidas a partir do uso de cinemática inversa, do tratamento de colisões e das restrições impostas pela topologia de juntas

interdependentes que define a relação de movimento das ATMs esquerda e direita (Figura 3).

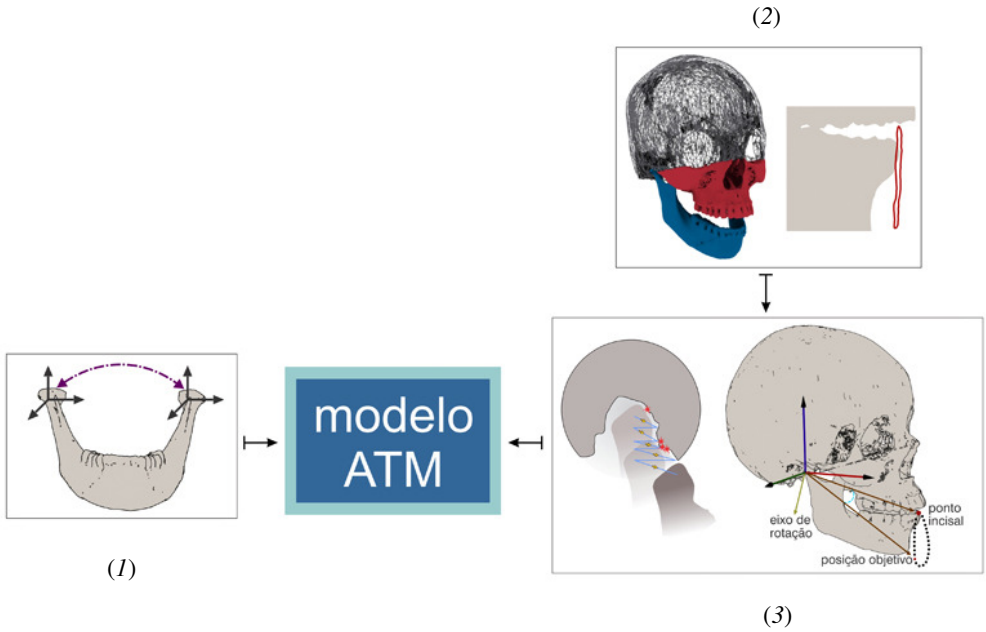


Figura 3: Modelo de movimento das ATMs, baseado numa topologia de juntas interdependentes (*1*): é necessária a aplicação da técnica de cinemática inversa no caminho incisal capturado(*2*) e tratamento de colisão entre as malhas dos ossos(*3*) (Villamil 2009).

3.4 Modelo de Inter-relação de Forças musculares

Uma articulação é o mecanismo que a natureza oferece a um sistema para capacitá-lo ao movimento com um mínimo de desgaste. A articulação atua como uma parte integral de um sistema de alavanca movido pela ação dos músculos. A determinação de como as forças dos músculos masseter, temporal e outros pequenos que atuam no controle fino de movimento da ATM ainda não está bem estabelecida. A complementação do modelo baseado em cinemática com um modelo baseado em Física que simplifique a contribuição dos músculos sobre a mandíbula através de linhas de ação levará a uma completa compreensão do movimento do sistema mastigatório.

O modelo de músculos envolve a utilização de um modelo tridimensional dos ossos e a modelagem de músculos utilizando atuadores de Hill (Figura 4). Foram modelados ligamentos no que concerne suas restrições no movimento mandibular. Além disso, foi utilizada uma nova metodologia na simulação dos músculos baseada no conceito de curvas de inserção com diversas linhas de ação para cada grupo muscular. Foi desenvolvido um simulador para visualização da manipulação do modelo em tempo real, além de permitir a ativação de músculos individualmente em tempo de simulação. (Garcia, 2011)

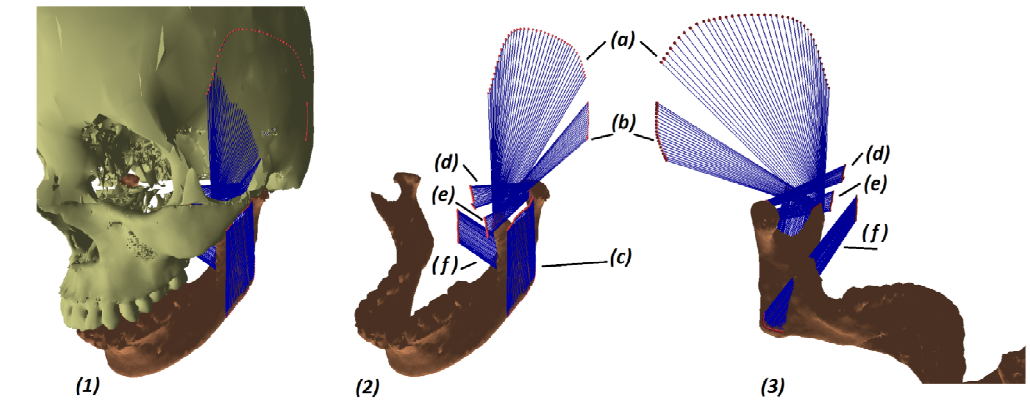


Figura 4. Músculos temporal posterior (a) temporal anterior (b), masseter (c), pterigóideo lateral superior (d) e inferior (e) e pterigóideo medial(f), vistos no modelo completo (1) e no modelo da mandíbula em vista externa (2) e interna (3).

5. Discussão e Comentários Finais

Na odontologia, o articulador tradicional, como já mencionado, é tido como uma ferramenta essecial no diagnóstico e planejamento de tratamentos e cirurgias. Os atuais articuladores são utilizados no estágio de diagnóstico, quando é necessária uma análise oclusal pré-ortodôntica, pré-protética ou para completar o diagnóstico de uma disfunção temporomandibular. No estágio terapêutico, os articuladores são utilizados quando é necessária a realização dos tratamentos protéticos, com uma necessária alternância entre o consultório médico e laboratório de prótese. Atualmente o mercado disponibiliza no mercado, somente os articuladores mecânicos. Os dados reais do paciente são extraídos de moldes de gesso feitos a partir de suas arcadas dentárias. Eles são colocados nesta ferramenta para exame e diagnóstico, ou para a construção de aparelhos dentários. Além do molde de gesso é necessário que o profissional extraia medidas craniomandibulares e da relação entre dentes e juntas do paciente para um correto ajuste do articulador. Todo este processo é bastante trabalhoso e toma bastante tempo do profissional. Mas o problema deste tipo de dispositivo é que o movimento por ele simulado não é totalmente correto, visto que o maxilar que movimenta é o superior, e não a mandíbula como acontece no movimento de mastigação real. Além disto, o movimento simulado de mastigação é sempre uniforme. Não é possível simular vários padrões de mastigação que variam depessoa para pessoa e os padrões que dependem do tipo de alimento (alimentos duros, moles, viscosos etc.). Outro ponto negativo do articulador mecânico é o de que não há visualização dos padrões de mastigação e curvas de movimento descritas pela mandíbula de vários ângulos, além de não haver a possibilidade de armazenar dados relativos a eles. Com isto o profissional de medicina ou odontologia não pode enviar estes dados para outro colega para que este emita uma outra opinião sobre o diagnóstico, além de não permitir o confronto e a análise dos dados antigos e novos ao longo do tratamento.

Portanto, conforme já mencionado em relação aos diferenciais do TMJsim, a concorrência existe somente em relação a uma parte do produto proposto e ainda assim, com recursos limitados e abordagem ultrapassada. O projeto do TMJsim ganhou o prêmio Santander Ciência e Inovação da Região Sul do Brasil e tem seu pedido de patente registrado junto ao INPI sob o protocolo número PI1002258-9.

6. Referências

Hill, A. The heat of shortening and the dynamic constants in muscle.Proc.Roy.Soc.London, v. 126, p. 136_195, 1938

Lemoine, J. J.; Xia, J. J.; Gateno, J.; Liebschner, M. A. K. Radiographic Analysis for Jaw Motion Normalization. Journal of Oral Maxillofacial Surgery, [S.l.], v.63, p.961–967, 2005.

Lepera, F. Determination of the hinge axis clutches on condyle position. Journal of Prosthetic Dentistry, [S.l.], v.8, p.260, 1958.McNeill C. Ciência e Prática da Oclusão. 1 ed, Ed. Quintessence, São Paulo, 2000

Olszewski, R.; Villamil, M. B.; Trevisan, D. G.; Nedel, L. P.; Freitas, C. M.; Reychler H.; Macq, B. Towards an integrated system for planning and assisting maxillofacial orthognathic surgery. Computer methods and programs in biomedicine, [S.l.], v.91, n.1, p.13–21, 2008.

Sifakis, E.; Neverov, I.; Fedriw, R. Automatic determination of facial muscle activations from sparse motion capture marker data. ACM Trans. Graph. (Proc. SIGGRAPH, n. 24, p. 417_425, 2005.

SLICER software (2005). http://www.slicer.org/

Stavness, I. et al. An integrated, dynamic jaw and laryngeal model constructed from CT data. In: Proc ISBMS06 in Springer LNCS 4072. [S.l.]: Springer, 2006. p.169_177.

Stavness, I. et al. Towards predicting biomechanical consequences of jaw reconstruction. Conf Proc IEEE Eng Med Biol Soc, v. 2008, p. 4567_4570, 2008.

Villamil M. B., Nedel L. P., Freitas C. S., Machado R.S, Silveira R. L., Silva A. M. M. Simulação do Movimento da Mandíbula e Comportamento da Articulação Temporomandibular**.** In V Workshop de Informática Médica (WIM´2005), Porto Alegre, Brazil, June 2005.

Villamil, M. B. Modelagem e Simulação da Articulação Tempromandibular. Tese de Doutorado, Universidade Federal do Rio Grande do Sul, 2009.

Garcia, E. Modelagem e Simulação dos Músculos Tempormandibulares. Dissertação de Mestrado, 2011.

Zwemer, T. J. Boucher´s Clinical Dental Terminology, ed 3 St Louis: Mosby, 1993

Sistema de Apoio à Decisão Aplicado ao Diagnóstico da Doença de Alzheimer

Flávio L. Seixas[1], Bianca Zadrozny[2], Jerson Laks[3], Débora C. M. Saade[1], Aura Conci[1]

[1]Instituto de Computação – Universidade Federal Fluminense (UFF)
Rua Passo da Pátria, 156 – 24.210-240 – Niterói – RJ – Brasil

[2]IBM Research Brasil – Av. Pasteur, 138 – 22.296-903 – Rio de Janeiro – RJ - Brasil

[3]Instituto de Psiquiatria – Universidade Federal do Rio de Janeiro (UFRJ),
Av. Venceslau Brás, 71 – 22.290-140 – Rio de Janeiro – RJ – Brasil

{fseixas, debora, aconci}@ic.uff.br, biancaz@br.ibm.com,
jlaks@centroin.com.br

Abstract. *This paper describes a clinical decision support system applied to Alzheimer's disease diagnosis. Such system includes: a Bayesian knowledge base, an ontology defined to represent uncertainty inherent in clinical knowledge, an inference engine and a Bayesian learning algorithm. The Bayesian network structure was based on clinical Alzheimer criteria published by NINCDS-ADRDA and DSM-IV. The parameters were obtained using a Bayesian learning algorithm known as Expectation Maximization (EM). As a training database, we used clinical items from about 1500 cases available in the CERAD database. Results were evaluated using a sensitivity analysis.*

Resumo. *Este artigo descreve um sistema de suporte à decisão clínica, aplicado ao diagnóstico da Doença de Alzheimer. Os principais componentes presentes no sistema são: um modelo de conhecimento baseado em rede bayesiana, uma ontologia para representação do conhecimento clínico e incertezas, um motor de inferência e aprendizagem computacional. A estrutura da rede bayesiana foi baseada nos critérios clínicos publicados pelo NINCDS-ADRDA e DSM-IV. Os parâmetros da rede foram definidos aplicando o algoritmo de aprendizagem computacional conhecido como Expectation Maximization (EM). A base de treinamento foi baseada em dados clínicos de cerca de 1500 indivíduos obtidos do CERAD. Os resultados foram avaliados através de uma análise de sensibilidade.*

1. Introdução

A doença de Alzheimer é uma doença neurodegenerativa progressiva que leva à perda irreversível de neurônios e à demência. Conforme DSM-IV *(Dementia Diagnostic and Statistical Manual for Mental Disorders)*, demência é definida como a verificação de múltiplos déficits cognitivos, incluindo prejuízos da memória e ao menos um dos seguintes distúrbios cognitivos: afasia, apraxia, agnosia ou funcionamento executivo [APA, 1994]. A doença de Alzheimer é classificada como um subtipo da demência, representando a principal causa de demência entre idosos. Em um levantamento envolvendo 1563 indivíduos idosos em uma comunidade de São Paulo, 12,9% foram

diagnosticados com algum tipo de demência, destes, a doença de Alzheimer representou 59,8% dos casos [Bottino *et al.*, 2008]. Em um relatório publicado pela *Alzheimer's Association* (2010), a doença de Alzheimer aparece na sétima posição de causa de todas as mortes nos Estados Unidos. Dentre as diversas formas de se estagiar o curso da doença, uma delas divide da doença de Alzheimer em dois estágios [Dubois *et al.*, 2007]: o estágio pré-demencial ou pré-clinico e o estágio demencial.

Para se chegar ao diagnóstico, o clínico primeiro avalia sinais e sintomas vinculados à demência utilizando como base os critérios publicados pela DSM-IV. Na suspeita da doença de Alzheimer, o paciente é avaliado aplicando critérios mais específicos. O critério mais utilizado para diagnóstico de provável Alzheimer é NINCDS-ADRDA *(National Institute of Neurological and Communicative Disorders and Stroke - Alzheimer's Disease and Related Disorders)* documentado em Mckann *et al.* (1984). Há outros critérios na literatura, dentre eles, os principais são os critérios do CERAD *(Consortium to Establish a Registry for Alzheimer's Disease)* encontrado em Mirra *et al.* (1991), e a revisão publicada por Dubois *et al.* (2007).

O presente artigo apresenta um sistema de suporte à decisão clínica aplicado ao diagnóstico da doença de Alzheimer. Um sistema de apoio à decisão clínica é definido como um sistema computacional utilizado com o objetivo de buscar dados clínicos mais relevantes e prover a médicos e pacientes informações mais efetivas, conforme o contexto clínico presente [Sirajuddin *et al.*, 2009].

Shwe *et al.* (1991) utilizaram um modelo baseado em probabilidade e rede bayesiana para apoiar o diagnóstico clínico, mapeando aproximadamente 500 doenças e 4000 sintomas. Amaral *et al.* (1995) propuseram um sistema de apoio ao diagnóstico baseado em regras de produção, classificando 30 grupos e aproximadamente 300 categorias de doenças psiquiátricas. Em ambos os trabalhos, não houve relatos da implantação do sistema em uma rotina clínica: o desempenho foi medido em função de simulações e dados retrospectivos publicados na literatura científica. Lindgren (2007) mostrou um sistema de apoio à decisão baseado em regras de produção, visando apoiar o raciocínio clínico para o diagnóstico da demência e subtipos, incluindo Alzheimer. O estudo focou primeiramente no modelo de conhecimento, processo de raciocínio e a interface do sistema; o desempenho não foi apresentado devido a dificuldades de sua medida. Outros sistemas dão suporte ao processo de diagnóstico utilizando métodos de classificação dado um conjunto completo de dados clínicos, oriundos principalmente de exames de imagem [Mueller *et al.*, 2010].

Este artigo apresenta um sistema visando apoiar o raciocínio clínico, utilizando um modelo de conhecimento baseado em rede bayesiana [Korb e Nicholson, 2004]. Em contraste com os trabalhos relacionados, este modelo considera a causalidade de eventos e a representação de incertezas, bem como observações parciais da realidade, características presentes no contexto clínico. Este artigo está organizado da seguinte forma: na Seção 2, definimos os componentes da arquitetura do sistema de suporte à decisão clínica, mostramos a estrutura da rede bayesiana e o método de aprendizagem computacional para obtenção dos parâmetros da rede bayesiana a partir da base de treinamento. Na Seção 3, mostramos as ferramentas utilizadas na implementação e os resultados obtidos utilizando a análise de sensibilidade. As conclusões e próximos trabalhos são apresentados na Seção 4.

2. Sistema de suporte à decisão clínica

A Figura 1 mostra os principais componentes de um sistema de apoio à decisão clínica. O histórico e informações clínicas do paciente são armazenados no repositório de dados clínicos (Item 1 da Figura 1). Este repositório é consultado e atualizado pelo médico responsável pelo diagnóstico clínico através de uma interface de comunicação (Item 2). A interface de comunicação também é responsável pela correspondência dos itens clínicos observados do paciente ao motor de inferência (Item 3). O objetivo do motor de inferência é, através de um método iterativo, calcular a máxima probabilidade marginal a *posteriori* da variável aleatória ou nó da doença de interesse. O motor de inferência utiliza a base de conhecimento (Item 4) contendo os critérios de diagnóstico e conhecimentos clínicos acerca da doença. O módulo de aquisição do conhecimento (Item 5) é responsável pela atualização da base de conhecimento, normalmente, tarefa efetuada por especialistas daquele domínio de conhecimento. O modelo de conhecimento adotado é representado por uma rede bayesiana. A estrutura da rede bayesiana é representada graficamente por um grafo unidirecional acíclico. Cada componente do grafo indica a relação causal entre dois nós (ou duas variáveis aleatórias), o nó pai representando a causa e o nó filho o efeito. Os parâmetros se referem às tabelas de distribuição de probabilidades *a priori* e condicionais. O método de aprendizagem computacional tem por objetivo atualizar continuamente os parâmetros da rede bayesiana de acordo com a decisão clínica tomada pelo médico, dados os itens clínicos apresentados pelo paciente.

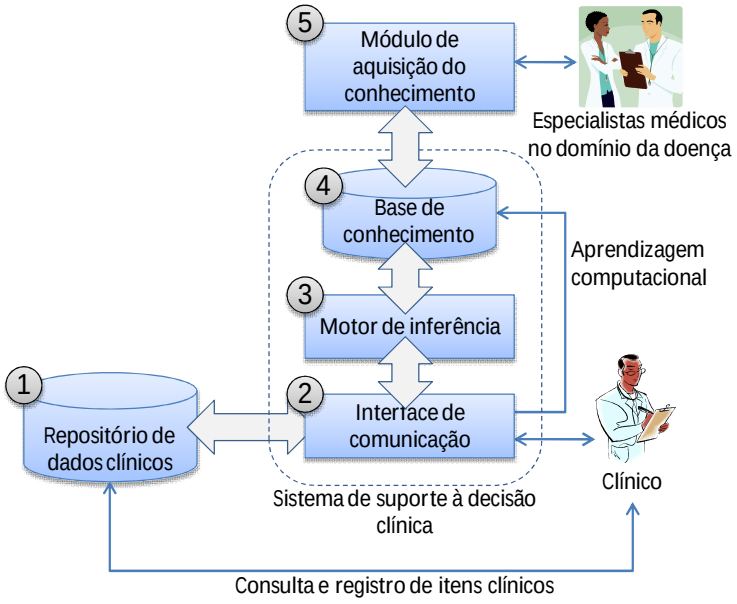


Figura 1. Componentes de um sistema de suporte à decisão clínica

Na Subseção 2.1 iremos detalhar os componentes estruturais da rede bayesiana.

2.1. Estrutura da rede bayesiana

Utilizamos uma estrutura da rede bayesiana multinível, com variáveis aleatórias distribuídas de forma discreta ou multinomial, conforme mostra a Figura 2. O primeiro nível abriga variáveis aleatórias B_i *(Background information)* representando os fatores de pré-disposição ou de exclusão da doença. O segundo representa a doença de interesse

D (Disease) formado por um único nó central. O terceiro nível contém as variáveis aleatórias F_i *(Findings)* representando os efeitos da doença (sintomas, sinais, resultados de exames psico-neurológicos, e outros).

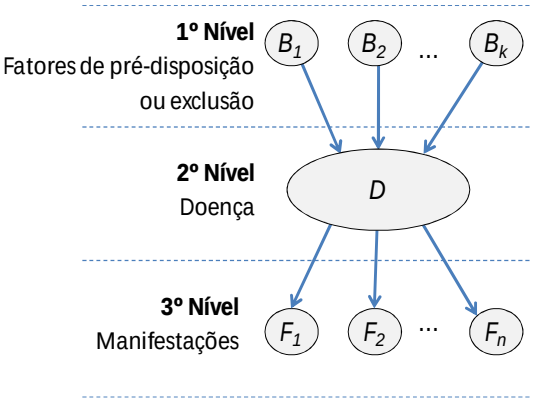


Figura 2. Estrutura da rede bayesiana multinível e multinomial.

O objetivo do motor de inferência é calcular a probabilidade marginal ou *a posteriori* do nó central ou variável aleatória que representa a doença de interesse. A base do motor de inferência é o teorema de Bayes, conforme mostra a Equação 1.

$$\Pr(h \mid e) = \frac{\Pr(e \mid h) \cdot \Pr(h)}{\Pr(e)} \qquad (1)$$

onde Pr(e|h) é a função densidade probabilidade condicional das evidências dadas as hipóteses, Pr(h) a função densidade probabilidade das hipóteses e Pr(e) a função densidade probabilidade das evidências. O teorema de Bayes relaciona, dadas as evidências ou itens clínicos, qual a função probabilidade da hipótese, no caso, o acometimento do paciente com a doença de Alzheimer, obtido pela função probabilidade condicional Pr(h|e).

As variáveis aleatórias foram definidas com base nos critérios de diagnóstico de provável doença de Alzheimer, e posicionadas no nível da rede bayesiana correspondente. Quanto à natureza, todas as variáveis aleatórias são discretas, podendo ser binomiais (ex.: presente/ausente), ou multinomiais, indicando, por exemplo, a gravidade ou intensidade de uma manifestação (ex.: suave/moderada/grave). Algumas variáveis aleatórias podem contribuir ou refutar o grau de crença da doença. A Tabela 1 mostra uma listagem das principais variáveis aleatórias obtidas conforme os critérios de diagnóstico da doença de Alzheimer e sua respectiva classe ou nível na rede bayesiana. No total, foram mapeados 15 fatores de pré-disposição/exclusão e 43 manifestações, considerando principalmente os critérios publicados em NINCDS-ADRDA, DSM-IV e outras publicações.

Os parâmetros da rede bayesiana foram obtidos através de um método de aprendizagem computacional conhecido por *Expectation Maximization* [Friedman, 1997], com base em uma amostra de dados clínicos observados parcialmente de uma amostra reunindo pacientes e indivíduos de controle normal. Este tema será detalhado na próxima subseção.

Tabela 1. Fatores de pré-disposição ou exclusão, e manifestações relacionados ao diagnóstico da doença de Alzheimer

Classe (Nível)	Itens
Fatores de pré-disposição ou exclusão	Comprometimento social e ocupacional, doença cerebrovascular, doença de Parkinson, doença de Huntigton, hematoma subdural, hidrocefalia de pressão normal, tumor cerebral, hipotireoidismo, deficiência de niacina, hipercalcemia, neurossífilis, infecção com HIV, histórico familiar de Alzheimer.
Manifestações	Déficit de memória, afasia, apraxia, agnosia, perturbação do funcionamento executivo, declínio cognitivo significativo e contínuo, depressão, insônia, delírios, ilusões, alucinações, explosões catastróficas, verbais, emocionais ou físicas, distúrbios sexuais, perda de peso, hipertonia, mioclonia, distúrbio de marcha, crises convulsivas, hemiparesia, perda sensorial, Mini-Exame de Estado Mental (MMSE), escala de demência de Blessed, punção lombar normal, EEG, atrofia de amígdala com progressão, atrofia de hipocampo com progressão, e testes psico-neurológicos aplicados pelo CERAD.

2.2. Aprendizagem dos parâmetros da rede bayesiana

Com a estrutura definida e fixada, aplicamos métodos de aprendizagem computacional para obter os parâmetros da rede bayesiana. Utilizamos os dados clínicos disponibilizados pela base CERAD (http://cerad.mc.duke.edu/) como base de treinamento. O CERAD foi fundado pelo *National Institute on Aging* em 1986 com o objetivo de desenvolver medidas e padrões utilizados para o diagnóstico da doença de Alzheimer. Suas principais contribuições ao longo de 20 anos de pesquisa foram: proposição de baterias de testes clínicos, neuropsicológicos, neuropatológicos, protocolos de avaliação de neuroimagens, definição de escalas e procedimentos para diagnóstico e avaliação do estágio da doença, incluindo a confirmação por exames *postmortem*, abrangendo pacientes e indivíduos de controle normal [Fillenbaum *et al.*, 2008]. Os dados de cada bateria de testes e avaliações clínicas foram coletados através de formulários e disponibilizados eletronicamente em uma base de dados relacional.

A base de treinamento envolveu 1557 indivíduos (1094 pacientes e 463 controles). Para cada variável aleatória da rede bayesiana, procuramos identificar uma ou mais correspondências com atributos na base de treinamento. A Tabela 2 mostra um exemplo do trabalho de correspondência da variável aleatória "Déficit de memória". O primeiro passo é selecionar os atributos da base de treinamento que possuem correspondência semântica com a variável aleatória em questão. Se mais de um atributo for selecionado, é necessário aplicar uma função de priorização, definindo, por exemplo, como evidência da variável aleatória, o estado do primeiro atributo que aparecer preenchido na base de treinamento.

Para os atributos numéricos (ex.: MMSE), é necessário aplicar algum método de discretização. Foi adotado o método de discretização supervisionado proposto por Fayyad e Irani (1993), utilizando como classe o atributo CDR *(Clinical Dementia Rating).* O CDR foi inicialmente proposto pela Escola de Medicina da Universidade de Washington com objetivo de avaliar o estágio da demência. Representa uma escala de

zero a três, zero indicando sem demência ou controle normal, e três, demência severa, sendo utilizado principalmente para avaliação do estágio da doença de Alzheimer devido a sua forte correlação com o diagnóstico da doença [Maia *et al.*, 2006]. Ainda na etapa de treinamento, consideramos o critério exposto na Equação 2 para representação da variável aleatória *D*. Atribuímos o valor zero à variável aleatória do diagnóstico negativo da doença de Alzheimer, e valor um ao diagnóstico positivo.

$$\begin{cases} D = 0, \text{se CDR} < 1; \\ D = 1, \text{caso contrário}; \end{cases} \quad (2)$$

Tabela 2. Exemplo de correspondência de uma variável aleatória da rede bayesiana com os dados presentes na base CERAD

Variável aleatória (VA) avaliada	Nome do formulário na base	Atributos identificados na base com correspondência semântica
Déficit de memória	Histórico clínico	Problemas com memória
		Problemas com memória com prejuízos para atividades da rotina diária
	Observações clínicas	Comprometimento de memória por mais de 12 meses
	Observações clínicas quanto às funções cognitivas	Déficit de memória
		Outros déficits de memória

Aproximadamente 49% dos dados da base de treinamento estão ausentes ou não avaliados. Foi identificada a correspondência de 80% das variáveis aleatórias da rede bayesiana em relação aos atributos da base de treinamento, 11 fatores de pré-disposição e 31 sintomas, sinais e testes. Adotamos o algoritmo EM *(Expectation Maximization)* como método de aprendizagem dos parâmetros da rede bayesiana.

O algoritmo EM, descrito primeiramente por Dempster *et al.* (1977), é utilizado quando há dados faltantes ou observações parciais na base de treinamento. Dado o modelo estatístico consistindo em um conjunto *X* de dados observados, um conjunto *Z* de dados latentes, e um vetor θ de parâmetros desconhecidos, juntamente com a função verossimilhança $L(\theta; X, Z) = \Pr(X, Z \mid \theta)$, a estimativa de máxima verossimilhança dos parâmetros desconhecidos é determinada pela probabilidade marginal dos dados observados, conforme mostra Equação 3.

$$L(\theta; X) = \Pr(X \mid \theta) = \sum_{Z} \Pr(X, Z \mid \theta) \quad (3)$$

Por tratar-se de uma função complexa, o algoritmo EM busca iterativamente o ponto de máximo da função aplicando dois passos. O primeiro é o passo E *(Expectation):* consiste em calcular o valor esperado do logaritmo da função probabilidade condicional da distribuição de *Z* dado *X*, sobre as estimativas atuais dos parâmetros $\theta^{(t)}$, conforme mostra a Equação 4.

$$Q(\theta \mid \theta^{(t)}) = E_{Z|X,\theta^{(t)}}[\log L(\theta; X, Z)] \quad (4)$$

A Equação 4 também é denominada *log-likelihood.* O passo M *(Maximization)* busca os parâmetros que maximizam a Equação 5.

$$\theta^{(t+1)} = \arg\max_{\theta} Q(\theta \mid \theta^{(t)}) \qquad (5)$$

O passo E é repetido *n* vezes, os valores de *Z* recalculados para obtenção de uma melhor estimativa dos parâmetros θ, até atingir a convergência do vetor θ ou o número máximo de iterações.

3. Implementação e resultados

Todas as funções relacionadas à rede bayesiana, bem como o algoritmo de aprendizagem, EM, foram implementadas utilizando a biblioteca BNT *(Bayes Network Toolbox)*, disponibilizada para o ambiente Mathworks MATLAB® [Murphy, 2001]. O método de discretização dos atributos numéricos foi implementado pelo aplicativo WEKA da Universidade de Waikato (http://www.cs.waikato.ac.nz/ml/weka/).

Os parâmetros da rede bayesiana foram representados em tabelas de distribuição de probabilidades condicionais. As probabilidades *a priori* da rede bayesiana foram definidas considerando a distribuição de Dirichlet [Gopalan e Berry, 1998]. A distribuição Dirichlet refere-se à família de distribuições de probabilidade multivariadas, parametrizadas pelo vetor α. Sua função densidade de probabilidade retorna a probabilidade de *K* eventos x_i, dado que cada item foi observado α_i*-1* vezes.

O algoritmo EM foi inicializado considerando distribuição uniforme das variáveis aleatórias. O algoritmo atingiu o critério de convergência em 18 iterações. A Figura 3 mostra a curva de aprendizagem, o valor esperado do vetor da função logaritmo da probabilidade condicional dos parâmetros, ou *log-likelihood.*

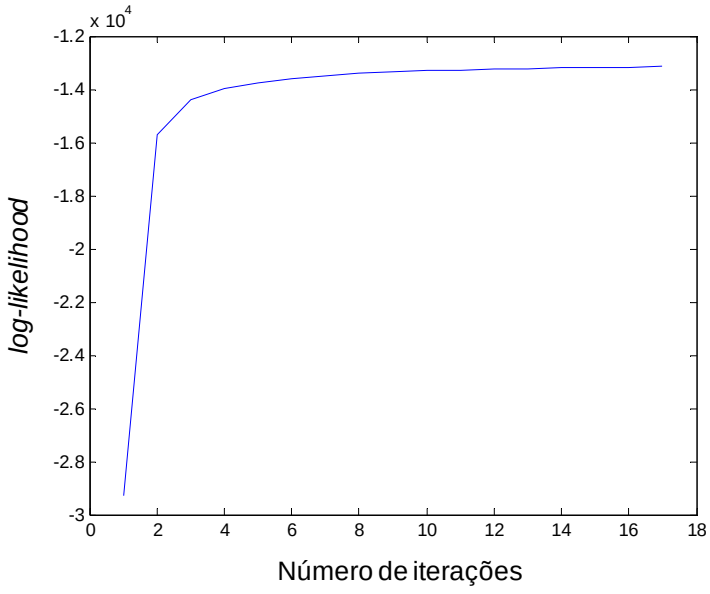


Figura 3. Curva de aprendizagem obtida do algoritmo EM *(Expectation Maximization)*

Para avaliação dos resultados, foram utilizadas a entropia como métrica e análise de sensibilidade [Korb e Nicholson, 2004]. A entropia foi utilizada como medida de incerteza. A entropia é obtida conforme Equação 6, onde *H(X)* representa a entropia, Pr(x_i) a probabilidade de ocorrência de x_i. Uma medida próxima de 1 significa grau de

incerteza máximo; ao contrário, próximo de zero, grau de incerteza mínimo. A expectativa é que, dada uma combinação de evidências, a entropia obtida no nó de consulta ou variável aleatória da doença de Alzheimer seja reduzida.

$$H(X) = -\sum_{i=1}^{n} \Pr(x_i) \cdot \log_n \Pr(x_i), \quad H(X) \in [0;1] \qquad (6)$$

A técnica de análise de sensibilidade é utilizada para identificar quais evidências mais reduzem as medidas de entropia. Em um sistema de suporte à decisão clínica, esta análise pode ser utilizada para indicar ao clínico quais observações devem ser coletadas do paciente naquele momento, de modo a obter maior redução da entropia e, assim, o grau de incerteza do diagnóstico clínico. Neste trabalho, o principal propósito da análise de sensibilidade é simular as evidências, mostrando uma validação e desempenho da rede bayesiana.

A Tabela 3 mostra 17 evidências ordenadas de forma decrescente pela redução da medida de entropia. As evidências foram filtradas considerando um percentual de dados faltantes na base de treinamento menor ou igual a 50%. A entropia é avaliada pontualmente na variável aleatória que representa a doença de Alzheimer. A redução da entropia é obtida calculando a diferença com e sem a identificação da evidência. Assim, decorrida a aprendizagem dos parâmetros da rede bayesiana, notamos que a identificação de crises convulsivas produz a maior redução da medida de incerteza. Após a 17ª posição, obtivemos redução da entropia menor que 0,001.

Tabela 3. Evidências em ordem decrescente pela redução da entropia

Posição	Variável aleatória presente na rede bayesiana	Redução da entropia	% de dados faltantes
1	Crises convulsivas	0,634	27,87
2	Insônia	0,433	12,40
3	Doença cerebrovascular	0,044	50,10
4	Déficit de campo visual	0,035	12,59
5	Teste de Nomeação de Boston	0,029	11,30
6	Depressão	0,027	17,92
7	Teste: *Word List Memory* (CERAD)	0,022	13,42
8	Mini-Exame de Estado Mental (MMSE)	0,020	1,03
9	Perda de peso	0,009	27,36
10	Escala de demência de Blessed	0,007	1,48
11	Neurossífilis	0,006	46,82
12	Déficit de memória	0,006	0,90
13	Convulsões e distúrbios de marcha	0,005	22,61
14	Teste: *Constructional Praxis* (CERAD)	0,003	13,23
15	Afasia	0,001	50,42
16	Teste: *Word List Recall* (CERAD)	0,001	14,45
17	Teste de fluência verbal semântica	0,001	12,72

4. Conclusões

Este artigo descreveu um sistema de apoio à decisão clínica com o modelo de conhecimento baseado em rede bayesiana, com aplicações para o diagnóstico da doença de Alzheimer. A base de treinamento contém 1557 instâncias (pacientes). Contudo, há muitos dados faltantes, conforme mencionado em parágrafos anteriores. Esse fato dificulta uma conclusão mais efetiva do desempenho da rede bayesiana. A melhor validação do sistema de apoio à decisão é a sua implantação em uma rotina clínica controlada, monitorando o grau de acerto ou relevância, dado o contexto clínico apresentado.

As principais contribuições do artigo foram: (1) a aplicação do modelo probabilístico e causal para representação do conhecimento clínico; (2) formalização dos critérios clínicos da doença de Alzheimer em uma estrutura computacional; (3) verificação das observações ou evidências a serem coletadas mais relevantes, resultante da aprendizagem utilizando uma base de treinamento.

Como trabalho futuro pretende-se: desenvolver ou aplicar um método de atualização da rede bayesiana dada indicação de novas evidências e a decisão tomada pelo clínico; incorporar ao processo de aprendizagem uma base de treinamento mais completa, com menor percentual de dados faltantes; implantar o sistema na rotina clínica.

Agradecimentos

Este trabalho foi parcialmente financiado por CNPQ, FAPERJ e INCT-MACC.

Referências

APA (1994). Diagnostic and Statistical Manual of Mental Disorders DSM-IV. Washington, DC, American Psychiatric Association.

Bottino, C. M. C., D. Azevedo, M. Tatsch, S. R. Hototian, M. A. Moscoso, J. Folquitto, A. Z. Scalco, M. C. Bazzarella, M. A. Lopes and J. Litvoc (2008). "Estimate of dementia prevalence in a community sample from São Paulo." Dementia and Geriatric Cognitive Disorders 26(4): 291-299.

Dempster, A. P., N. M. Laird and D. B. Rubin (1977). "Maximum likelihood from incomplete data via the EM algorithm." Journal of the Royal Statistical Society. Series B (Methodological) 39(1): 1-38.

do Amaral, M. B., Y. Satomura, M. Honda and T. Sato (1995). "A psychiatric diagnostic system integrating probabilistic and categorical reasoning." Methods Inf Med 34(3): 232-243.

Dubois, B., H. H. Feldman, C. Jacova, S. T. DeKosky, P. Barberger-Gateau, J. Cummings, A. Delacourte, D. Galasko, S. Gauthier, K. Meguro, J. O'Brien, F. Pasquier, P. Robert, M. Rossor, S. Salloway, Y. Stern, P. J. Visser and P. Scheltens (2007). "Research criteria for the diagnosis of Alzheimer's Disease: revising the NINCDS-ADRDA criteria." Lancet Neurology 6: 734-746.

Fayyad, U. M. and K. B. Irani (1993). Multi-interval discretisation of continuous-valued attributes for classification. Proceedings of the Thirteenth Internatinal Joint Conference on Artificial Intelligence, San Fracisco CA, Morgan Kaufmann.

Fillenbaum, G. G., G. van Belle, J. C. Morris, R. C. Mohs, S. S. Mirra, P. C. Davis, P. N. Tariot, J. M. Silverman, C. M. Clark and K. A. Welsh-Bohmer (2008). "Consortium to Establish a Registry for Alzheimer's Disease (CERAD): The first twenty years." Alzheimer's & dementia: the journal of the Alzheimer's Association 4(2): 96-109.

Friedman, N. (1997). Learning belief networks in the presence of missing values and hidden variables. Proceedings of the Fourteenth International Conference on Machine Learning, San Francisco CA, Morgan Kaufmann.

Gopalan, R. and D. A. Berry (1998). "Bayesian multiple comparisons using Dirichlet process priors." Journal of the American Statistical Association 93(443): 1130-1139.

Korb, K. and A. Nicholson (2004). Bayesian artificial intelligence. Clayton, Victoria, Australia, Chapman & Hall/CRC.

Lindgren, H. (2007). Decision support in dementia care: developing systems for interactive reasoning, Umeå University, Faculty of Science and Technology, Computing Science.

Maia, A. L. G., C. Godinho, E. D. Ferreira, V. Almeida, A. Schuh, J. Kaye and M. L. F. Chaves (2006). "Application of the Brazilian version of the CDR scale in samples of dementia patients." Arquivos de Neuro-Psiquiatria 64(2B): 485-489.

McKhann, G., D. Drachman, M. Folstein, R. Katzman, D. Price and E. M. Stadlan (1984). "Clinical diagnosis of Alzheimer's disease: Report of the NINCDS-ADRDA Work Group under the auspices of Department of Health and Human Services Task Force on Alzheimer's Disease." Neurology 34(7): 939-944.

Mirra, S. S., A. Heyman, D. McKeel, S. M. Sumi, B. J. Crain, L. M. Brownlee, F. S. Vogel, J. P. Hughes, G. van Belle and L. Berg (1991). "The Consortium to Establish a Registry for Alzheimer's Disease (CERAD). Part II. Standardization of the neuropathologic assessment of Alzheimer's disease." Neurology 41(4): 479-486.

Mueller, S. G., N. Schuff, K. Yaffe, C. Madison, B. Miller and M. W. Weiner (2010). "Hippocampal atrophy patterns in mild cognitive impairment and Alzheimer's disease." Hum Brain Mapp 31(9): 1339-1347.

Murphy, K. (2001). "The bayes net toolbox for matlab." Computing science and statistics 33(2): 1024-1034.

Shwe, M. A., B. Middleton, D. Heckerman, M. Henrion, E. Horvitz, H. Lehmann and G. Cooper (1991). "Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base." Methods of information in medicine 30(4): 241-255.

Sirajuddin, A. M., J. A. Osheroff, D. F. Sittig, J. Chuo, F. Velasco and D. A. Collins (2009). "Implementation pearls from a new guidebook on improving medication use and outcomes with clinical decision support. Effective CDS is essential for addressing healthcare performance improvement imperatives." J Healthc Inf Manag 23(4): 38-45.

Intelimed - Sistema de Apoio ao Diagnóstico Médico Baseado em Técnicas de Inteligência Computacional

Caio C. M. Davi[1], Marcelo G. P. Lacerda[1], Flavio R. S. Oliveira[2], Fernando B. de Lima Neto[1], Raul A. Morais Melo[2]

[1]Escola Politécnica de Pernambuco
Universidade de Pernambuco (UPE)
Recife, PE, Brasil.

[2]Núcleo de Telessaúde (NUTES)
Universidade de Pernambuco (UPE)
Recife, PE, Brasil.

{caiocmd,flaros}@gmail.com, {mgpl,fbln}@ecomp.poli.br, melo.raul@hotmail.com

Abstract. *Intelimed System is a solution for supporting medical diagnosis based on intelligent computing techniques, it's under development by a multi-disciplinary team composed of doctors, engineers and computer students from both areas. This paper reports the progress of the team responsible for developing the intelligent engine application. The results of simulations so far were satisfactory although there is ample room for improvement.*

Resumo. *O Sistema Intelimed é uma solução para apoio ao diagnostico médico baseado em técnicas de Computação inteligente em desenvolvimento por uma equipe multi-disciplinar composta por médicos, engenheiros da Computação e estudantes de ambas as áreas. Este artigo relata os progressos da equipe responsável pelo desenvolvimento do motor inteligente da aplicação. Os resultados das simulações até o momento mostraram-se satisfatórios embora haja amplo espaço para melhorias.*

1. Introdução

A Semiologia Médica é o ramo da Medicina relacionado ao estudo dos sinais e sintomas das doenças. Utilizando-se dos conceitos e técnicas adquiridos nessa área do conhecimento médico o profissional de saúde é capaz de criar hipóteses de diagnósticos e posteriormente diagnosticar corretamente um paciente [Porto 2005]. Todavia o domínio da semiologia é muito complexo, e de aquisição demorada e trabalhosa [Ducla Soares 2007].

Desta forma este trabalho propõe um sistema de apoio ao diagnóstico médico utilizando técnicas de inteligência computacional, como mecanismo para reduzir a imprecisão inerente ao processo de diagnose. Todavia o domínio da Medicina é um campo muito extenso forçando os autores a resumir o escopo a ser tratado em uma sub-área da Medicina: o diagnóstico da asma.

O trabalho aqui descrito relata os resultados preliminares de um projeto em curso e integra um projeto maior, que vem sendo desenvolvido pelo Núcleo de Telessaúde da Universidade de Pernambuco (NUTES-UPE).

2. Referencial Teórico

Em oposição ao panorama acima descrito (de imprecisão inerente ao diagnóstico médico) estão os Sistemas de Apoio a Decisão (SADs) [Laudon and Laudon 2004], com a capacidade de criar mecanismos capazes de armazenar informações de maneira ativa sob a forma de memória organizacional. A partir destas informações previamente registradas os SADs podem auxiliar a resolver, auxiliados de técnicas de Inteligência Computacional, diversos tipos de problemas, tais como problemas de classificação, regressão e otimização.

Neste contexto surgem os Sistemas de Apoio à Decisão Clínica (SADCs): como o resultado da aplicação de Sistemas de Apoio a Decisão na área de saúde, mais especificamente no apoio na elaboração de diagnósticos médicos. A concepção de um diagnóstico médico é o resultado da análise das informações a cerca do paciente, informações clínicas e informações patológicas. A elaboração de um diagnóstico, assim como a obtenção de respostas a questões respeitantes à etiologia de determinadas doenças, pode ser assistida por SADCs [Vasconcelos et al. 2003].

Sistemas informatizados de apoio à decisão baseados em algoritmos matemáticos e Inteligência Computacional tentam simular o raciocínio do médico em suas tomadas de decisão [Andrade Zoby 2006], utilizando a capacidade organizacional provida pelos Sistemas de Informação em Medicina.

3. Sistema Intelimed

O sistema aqui descrito está na intersecção entre Medicina e engenharia e é constituído por três principais áreas de pesquisa: semiologia (em Medicina), engenharia de software e inteligência computacional (ambas em Computação). O objetivo principal do sistema é fornecer ao profissional médico suporte a decisão remoto. Este suporte a decisão deve advir de dados recolhidos e processados utilizando-se técnicas de Computação Inteligente.

Fisicamente o sistema é distribuido da seguinte forma: em um servidor central estão armazenados os casos-base que são diagnosticados por médicos especializados na área em que a patologia em questão se encontra. Estes casos-base virão a compor a base de dados utilizada pelo motor inteligente da aplicação. Com auxílio de técnicas inteligentes esta base de dados será transformada em uma estrutura pronta para auxiliar em diagnósticos médicos (uma árvore de decisão treinada, por exemplo). Estas estruturas, por sua vez, serão enviadas para servidores intermediários, localizados fisicamente em PSFs (postos de saúde da família), com isso médicos podem acessar remotamente esta estrutura que guarda o conhecimento do especialista em determinada área médica. Esta organização física é exemplificada na Figura 1 : Visão de Implantação (retirada de [de Menezes Junior et al. 2011]).

A base de dados utilizada durante a primeira etepa (para diagnosticar a asma) foi levantada seguindo um questionário voltado para pacientes com idade entre 12 e 14 anos e com suspeita de asma (ou para seus responsáveis). Este questionário foi elaborado com base no *International Study of Asthma and Allergies in Childhood* - ISAAC [ISAAC 1992]. Dentre muitas técnicas existentes dentro do domínio da Computação Inteligente duas foram selecionadas para a primeira etapa desta pesquisa: Sistemas Bayesianos e Árvores de Decisão. O principal motivo da escolha por estas duas técnicas em especial foi a capacidade de auditoria que ambas oferecem.

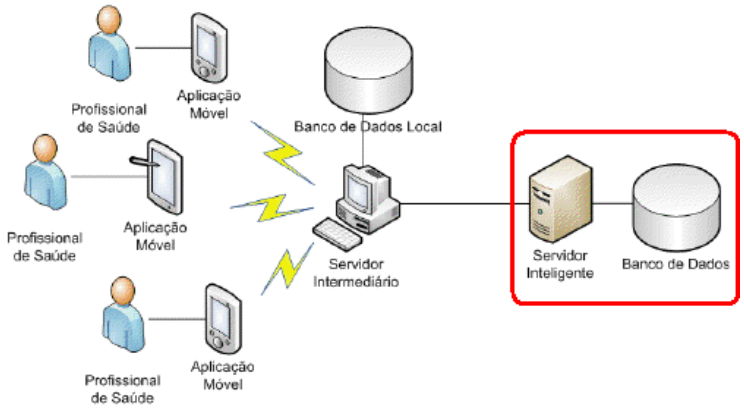


Figura 1. Visão de Implantação com destaque no Servidor Principal, local onde se encontra o motor inteligente.

4. Resultados

Foi utilizado o software de Mineração de Dados Orange [University of Ljubljana Slovenia 2011]. A base de dados resultante do pré-processamento foi testada com os classificadores: (i) Naïve Bayes [Friedman et al. 1997], (ii) k-NN [Friedman et al. 1997], (iii) C4.5 [Quinlan 1993] e (iv) Árvore de Classificação com a métrica Ganho de Informação [Quinlan 1986], todos utilizando a configuração padrão do software.

Em se tratando de um experimento na área de saúde, as métricas consideradas foram Acurácia, Sensibilidade e Especificidade [Fletcher and Fletcher 2006]. Por conta da quantidade de dados disponível, optou-se pela validação com o método *Leave-one-out* que visa reduzir tendências na avaliação do erro. Os resultados da simulação podem ser vistos na Tabela 1:

Tabela 1. Resultados das primeiras simulações

Classificador	Acurácia	Sensibilidade	Especificidade
kNN	0.6190	0.7200	0.4706
Árvore de Classificação	***0.7619***	0.7200	***0.8235***
Naïve Bayes	0.7143	***0.8000***	0.5882
C4.5	0.5952	0.6800	0.4706

A base de dados foi construida através de um questionário baseado nos estudos do ISAAC, este foi concebido por profissionais especialistas na área e aplicado no HUOC - Hospital Universitário Oswaldo Cruz. No levantamento de dados inicial os seguintes fatores foram utilizados para determinar o diagnóstico da asma: se o paciente apresenta sibilos ou teve o resultado do exame de espirometria positivo este é considerado como positivo, caso não apresente nenhum destes fatores é considerado negativo.

5. Conclusão

Apesar de ser um projeto ainda em andamento, o presente trabalho já apresenta resultados animadores. Mesmo com uma base de dados reduzida, os experimentos comprovam a existência das condições de se utilizar técnicas de inteligência computacional para o auxílio ao diagnóstico médico. Além disso, ambas as técnicas escolhidas para serem

implementadas em um primeiro momento obtiveram os melhores resultados nos três quesitos (acurácia, sensibilidade e especificidade) entre as demais técnicas avaliadas, o que aponta para um provável sucesso na primeira implantação do sistema, principalmente se for levado em conta o fato de que, durante a vida do sistema, as bases de dados utilizadas crescerão tendendo a se tornarem cada vez mais representativas, o que, conseqüentemente, permitirá um diagnóstico ainda mais preciso.

Entrevistas com pacientes ainda estão acontecendo. Portanto, em algum tempo, haverá condições de melhor avaliar cada técnica aqui citada. Em seu tempo, o motor inteligente aqui descrito será integrado às soluções móveis do projeto, resultando assim em um sistema remoto de apoio ao diagnóstico médico baseado em técnicas de inteligência computacional.

Os autores agradecem à FINEP (Financiadora de Estudos e Projetos) e ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) pelo apoio financeiro recebido para a execução deste projeto.

Referências

University of Ljubljana Slovenia (2011). Orange. Laboratory of Artificial Intelligence. http://orange.biolab.si/.

Andrade Zoby, E. (2006). Sistema de apoio a decisão para o diagnóstico médico de doenças sexualmente transmissíveis. *Escola Politécnica de Pernambuco, UPE*.

de Menezes Junior, J. V., D'Castro, R. J., Rodrigues, F. M. M., Gusmão, C. M. G., Lyra, N. R. S., and Sarinho, S. W. (2011). Intelimed: Uma experiência de desenvolvimento de sistema móvel de suporte ao diagnóstico médico. Revista Brasileira de Computação Aplicada, Passo Fundo.

Ducla Soares, J. (2007). *Semiologia Médica – Princípios, Métodos e Interpretação*. Lidel.

Fletcher, R. H. and Fletcher, S. W. (2006). *Epidemiologia Clínica: Elementos Essenciais*. Artmed, 4 edition.

Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, 29:131–161.

ISAAC (1992). *International Study of Asthma and Allergies in Childhood – ISAAC Phase One Manual*. Auckland (NZ), Münster (FRG).

Laudon, K. and Laudon, J. (2004). *Sistemas de Informação Gerenciais*. Pearson Prentice Hall,, São Paulo.

Mark, D. B. (2002). Tomadas de decisões em medicina clínica. *Harrison: Medicina Interna*, 13.

Porto, C. C. (2005). *Semiologia Médica*. Guanabara Koogan, 5 edition.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Vasconcelos, J., Henriques, R., and Rocha, Á. (2003). Modelo para o desenvolvimento de sistemas de apoio à decisão clínica para a prática da medicina baseada na evidência. In *CBIS - Congresso Brasileiro de Informática em Saúde*. SBIS.

AGEbot: um chatterbot em AIML voltado para responder questões sobre Epilepsia

Matheus Canali Fossatti[1], Roberto dos S. Rabello[1], Ana Carolina B. De Marchi[1]

[1]Curso de Ciência da Computação – Universidade de Passo Fundo (UPF)
Caixa Postal 611 – 99.052-900 – Passo Fundo – RS – Brazil

fossattiwk@gmail.com, {rabello,carolina}@upf.br

Abstract. *This article presents the process of developing a chat robot in AIML designed to answer questions about the epilepsy area. Some concepts are described about chatterbots, the markup language AIML, the field area of epilepsy, which method was used for the development and what are the features of the simulator.*

Resumo. *Este artigo apresenta o processo de desenvolvimento de um robô de conversação na linguagem AIML destinado a responder dúvidas sobre a área da epilepsia. Nele são descritos conceitos sobre chatterbots, a linguagem de marcação AIML, a área de domínio da epilepsia, qual metodologia foi utilizada para o desenvolvimento e quais são as funcionalidades do simulador.*

1. Introdução

Um dos principais objetivos da Inteligência Artificial (IA) é elaborar sistemas computadorizados para, de alguma forma, expressar características humanas. Atualmente, com a crescente evolução da IA, as tarefas de percepção começam a ganhar mais destaque. Tais tarefas podem ser representadas por uma técnica da inteligência artificial conhecida por Processamento de Linguagem Natural (PLN). O PLN é formado por um conjunto de métodos que tem como objetivo a análise e a compreensão da linguagem humana. Também conhecido por Linguística Computacional, o PLN possui aplicabilidade relacionada a traduções automáticas, ferramentas de auxilio a escrita, perguntas e respostas, categorização textual, recuperação e extração de informações, entre outros [SBC, 2009].

Na década de 50, Alan Turing propôs um desafio que determinaria se um sistema de computador poderia ou não ser inteligente através da realização destas tarefas. O desafio era baseado na interação entre duas pessoas e um computador através da comunicação por mensagens escritas. Se não fosse possível distinguir o ser humano da máquina, então o sistema poderia ser considerado inteligente.

Foi a partir desse desafio que se iniciaram o desenvolvimento dos sistemas simuladores de diálogos que, atualmente, possuem aplicabilidade em diversas áreas. Também conhecidos por *Chatterbots* (*Chatter*: Bate-Papo, *Bot*: Robô) esses sistemas foram elaborados com o intuito de tornar mais familiar a interação entre o homem e os computadores, dando a impressão de que o sistema possui personalidade própria [SGANDERLA, 2003].

Grande parte dos chamados *chatterbots* são implementados através de uma base de conhecimento desenvolvida na linguagem AIML (*Artificial Intelligence Markup Language*). Atualmente, ela é considerada uma das principais linguagens para o desenvolvimento de simuladores de conversação. O objetivo da AIML é analisar as mensagens enviadas pelo usuário e escolher a forma como essas mensagens deverão ser respondidas [LEONHARDT, 2003].

A utilização de *chatterbots* vem sendo muito comum em diversas áreas da saúde. Eles são capazes de responder aos pacientes dúvidas sobre doenças, diagnósticos e tratamentos, e auxiliar médicos na tomada de decisões. Desta forma, foi investigada a necessidade de modelar um *bot* capaz de auxiliar no aprendizado da epilepsia. A epilepsia pode ser definida como uma descarga neural paroxística (início e término abrupto de um evento epilético) que leva à convulsão generalizada, quando o tecido normal é invadido pela atividade da crise [SILVA, 1987].

Diante desse contexto, o objetivo deste trabalho é apresentar o processo de desenvolvimento de um *chatterbot* com uma base de conhecimento AIML específica em epilepsia, através de um estudo dos *chatterbots* e métodos existentes na linguagem, coleta de informações sobre a área de domínio, implementação e treinamento da base de conhecimento, desenvolvimento da sua interface e ao final divulgar validações e resultados do trabalho. Algumas pesquisas na área de computação, como [KUROMOTO, 2007], buscam apoiar o diagnóstico da epilepsia. O agente descrito neste trabalho tem por objetivo sanar dúvidas do usuário leigo, interagindo e tentando responder questões sobre sintomas e tratamentos sobre a doença.

2. Chatterbot: o Estado da Arte

Os *chatterbots* são agentes inteligentes de computadores que usam algumas técnicas de inteligência artificial para simular conversas com pessoas reais [ROTHERMEL, 2006]. A simulação é realizada através de trocas de mensagens de texto, como se o usuário estivesse em uma sala de bate-papo virtual.

O objetivo destes robôs de conversação é ser capaz de responder perguntas feitas por usuários sem que eles percebam que estão conversando com um programa de computador, mas sim com uma pessoa real [TEIXEIRA, 2005].

As principais características destes simuladores de diálogo são a capacidade memorização, atingir variados temas para conversa, robustez a respeito de sentenças não conhecidas, a capacidade de o robô falar por si só e algumas características visuais como animações, imagens de pessoas, gestos e expressões que fornecem um toque de personalidade ao *chatterbot* [ROTHERMEL, 2006].

A maior vantagem dos *chatterbots* é o seu grande poder de interação, que proporciona diálogos interessantes e motivam o usuário a participar da conversação, dando total liberdade ao mesmo. Na área educacional, eles são um método de ensino muito eficaz, onde os alunos despertam um interesse de interagir com a máquina.

É possível destacar três gerações de *chatterbots*, definidas de acordo com suas técnicas [NEVES, 2005]: a) Técnica de Casamento de Padrões e Regras Gramaticais: consiste em um conjunto de combinações de palavras-chaves selecionadas das perguntas e um grupo de respostas pré-programadas; b) Técnica de Inteligência Artificial:

trabalham através de técnicas de regras de produção e redes neurais; e c) Linguagens de Marcação: método mais completo do ponto de vista do processamento natural, baseada na linguagem AIML.

Os *chatterbots* também são classificados de acordo com sua finalidade, sendo divididos em cinco categorias [LAUREANO, 1999]: a) Educacional: destinados a área da educação, trabalham atendendo a alunos e professores, respondendo dúvidas e repassando informações; b) Entretenimento: tipo mais desenvolvido de *chatterbot*. Aqui se encontram os sistemas destinados apenas a divertir e entreter o usuário; c) Perguntas Mais Frequentes: também conhecido por FAQ (*Frequent Asked Questions*), são responsáveis por responder perguntas e esclarecer quaisquer dúvidas que o usuário tenha; d) Comerciais: fazem o papel de atendentes ou suporte a consumidores. Nessa categoria também se enquadram os robôs de marketing, que representam empresas e apresentam propaganda de produtos e serviços; e) Propósitos Gerais: todos os outros *chatterbots* que não se enquadram nos itens anteriores, ou por possuir um funcionamento diferente ou por não se destinaram a um propósito específico.

Após realizado um estudo sobre os *chatterbots* existentes (tanto comercias como acadêmicos), percebeu-se a inexistência de um *bot* destinado a área da saúde, mais especificamente na área da epilepsia. Assim, o *chatterbot* desenvolvido neste trabalho, composto por uma base de conhecimento em AIML, tem a finalidade de um sistema FAQ nessa área. Dentre os programas mais conhecidos desenvolvidos em AIML encontram-se o A.L.I.C.E.[1] e o Sete Zoom[2].

3. Área de Domínio

De acordo com a Liga Internacional Contra a Epilepsia (ILAE), a epilepsia pode ser definida como um distúrbio de origem cerebral, temporária e reversível, onde provoca manifestações motoras, sensitivas, sensoriais neurovegetativas [EPILEPSIA, 2010]. Através da ILAE, foram criadas classificações padronizadas para o diagnóstico das crises e síndromes epilépticas, a partir da busca de critérios nas informações obtidas dos pacientes e características de eletroencefalograma [GUERREIRO, 2000].

3.1. Classificação dos Eventos Epilépticos

As classificações de eventos epilépticos propostas pela ILAE surgiram em 1969 e se desenvolveram até uma nova metodologia apresentada em 2001. O trabalho adota a classificação sugerida em 1981, composta por três divisões: crises parciais, crises generalizadas e crises não-classificáveis [CARVALHO, 2007], conforme apresentado na Tabela 1.

3.2. Diagnóstico

O diagnóstico realizado pelo neurologista é baseado na descrição dos fatos que acontecem com o paciente antes, durante e após uma crise. As descrições devem ser feitas pelo doente ou por testemunhas que acompanharam a situação, pois muitas vezes o paciente não é capaz de lembrar o ocorrido.

[1] http://alicebot.blogspot.com/

[2] http://www.inbot.com.br/novo/cases.php#sete

Para completar o diagnóstico, exames de eletrencefalograma (EGG), hiperpnéia e fotoestimulação são solicitados para auxiliar na investigação da existência de uma lesão cerebral ou podem mostrar anomalias nas ondas cerebrais [EPILEPSIA, 2010].

Tabela 1. Classificação de crises epiléticas segundo do ILAE de 1981

Crises Parciais	Crise Parcial Simples (CPS)	- Sem prejuízo da consciência; - Com sinais motores; - Com sinais sensitivos, somatossensoriais ou especiais; - Com sinais ou sintomas autonômicos; - Com sinais psíquicos.
	Crise Parcial Complexa (CPC)	- Com prejuízo da consciência; - Inicio de CPS seguida por alteração da consciência; - Alteração da consciência no início.
	Crises Secundariamente Generalizadas	- CPS evoluindo para Crises Generalizadas Tônico-Clônicas (CGTC); - CPC evoluindo para CGTC; - CPS evoluindo para CPC e, então, para CGTC.
Crises Generalizadas	CGTC; crise de ausência; Crise de ausência atípica; Crises mioclônicas; Crises tônicas; Crises clônicas; Crises atônicas.	
Crises Não-Classificáveis	Informações incompletas e/ou inadequadas.	

3.3. Tratamento

O tratamento é realizado através de medicações que possam controlar a atividade anormal dos neurônios, diminuindo as cargas cerebrais anormais [ASPESI, 2001]. No entanto, o tratamento depende de pessoa para pessoa e da classificação de crise que ela possui. Algumas crises podem ser controladas e desaparecer com o tempo e a medicação pode ser suspensa; outros pacientes precisam seguir com o tratamento a vida inteira, com ataques reduzidos; alguns casos os pacientes não respondem bem aos medicamentos, ou precisam de uma dose tão alta que será melhor aceitar um controle parcial da doença.

4. Metodologia

4.1. Coleta de dados sobre epilepsia

Inicialmente a coleta de informações sobre epilepsia foi baseada nos assuntos básicos da doença, como seus conceitos, causas, diagnósticos, classificações e tratamentos. Essas informações foram encontradas em diversos livros a fim de explorar de forma diversificada o maior número de definições possíveis.

Em seguida, foi realizada uma pesquisa em *websites* informativos em busca de FAQs a respeito da doença, tentando ampliar o escopo do tema e alcançar o maior número de questões que os mesmos julgavam ser mais importantes e comuns.

Por fim, realizou-se uma busca em fóruns de discussão e *blogs* frequentados por epilépticos a fim de explorar quais são as mais frequentes perguntas e dúvidas e em que formato elas são apresentadas.

4.2. Documentação

Antes das informações coletadas sobre a epilepsia serem implementadas na base de conhecimento em AIML, foi criada uma documentação contendo possíveis perguntas e respostas que o *chatterbot* seria capaz de responder. Estas questões foram separadas por

assuntos e transformadas em categorias na linguagem AIML. Essa documentação serviu para que um especialista validasse o formato definido das perguntas e se as respostas relacionadas estavam completas e corretas, além de corrigir e complementar conceitos.

4.3. Implementação

Para iniciar o processo de implementação da base de conhecimento em AIML foi utilizada a ferramenta de desenvolvimento Pandorabots. Através dela, o sistema pode ser hospedado rapidamente e fica disponível para o acesso dos usuários desde o inicio da sua criação. O sistema oferece a possibilidade de publicar *bots* com nomes específicos, criar documentações, ferramentas para criação e treinamento de conteúdo para usuários sem experiência de programação (Figura 1). A ferramenta também agilizou a etapa de codificação das perguntas e respostas contidas na documentação elaborada (seção 4.2), oferecendo uma estrutura simples e eficiente para treinamento e testes do programa.

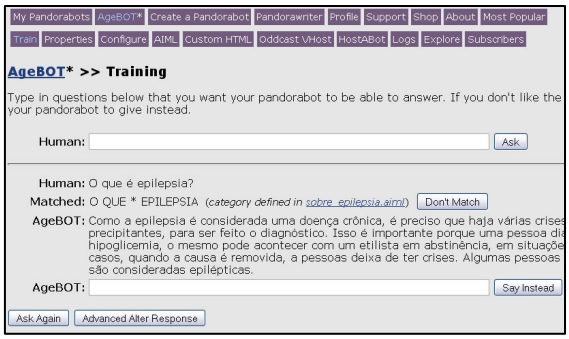


Figura 1. Tela de testes e treinamento da base AIML do sistema Pandorabots

As categorias desenvolvidas na base de conhecimento foram divididas em 16 arquivos .aiml, sendo separados por assuntos, a fim de organizar a estrutura e facilitar o processo de manutenção das questões. Os arquivos foram classificados com temas sobre questões gerais sobre epilepsia, sintomas, causas, tratamentos, diagnósticos, síndromes, prevenções, classificações, riscos, gravidez, questões diversas, informações sobre o *bot*, além de um arquivo genérico para o tratamento de perguntas sem resposta ou não relacionadas à área de domínio. Na Tabela 2 é apresentada a relação dos arquivos AIML desenvolvidos, mostrando a quantidade de categorias e perguntas elaboradas.

Tabela 2. Relação de categorias e perguntas implementadas no AGEbot

Nº	Nome	Qtd. Categorias	Qtd. Perguntas
1	Saudaçoes.aiml	37	X
2	Agebot.aiml	36	X
3	Sobre_epilepsia.aiml	584	18
4	Sobre_sintomas.aiml	794	20
5	Sobre_causas.aiml	450	10
6	Sobre_tratamentos.aiml	1855	35
7	Sobre_diagnosticos.aiml	261	6
8	Sobre_sindormes.aiml	46	5

Nº	Nome	Qtd. Categorias	Qtd. Perguntas
9	Sobre_prevencao.aiml	44	2
10	Sobre_classificacao.aiml	820	20
11	Sobre_riscos.aiml	357	5
12	Sobre_gravidez.aiml	117	7
13	Sobre_estatisticas.aiml	111	2
14	Sobre_oquefazer.aiml	283	5
15	Sobre_diversas.aiml	804	23
16	Generico.aiml	34	X

Assim, o AGEbot passou a contar com mais de seis mil e quinhentas categorias, o que corresponde a um total de 158 perguntas que o *chatterbot* é capaz de responder sobre os diversos temas citados.

Para a etapa de interação com o usuário foi implementada uma interface simples, amigável e de fácil entendimento, através de uma página web desenvolvida em HTML integrada com o interpretador AIML do Pandorabots (Figura 2).

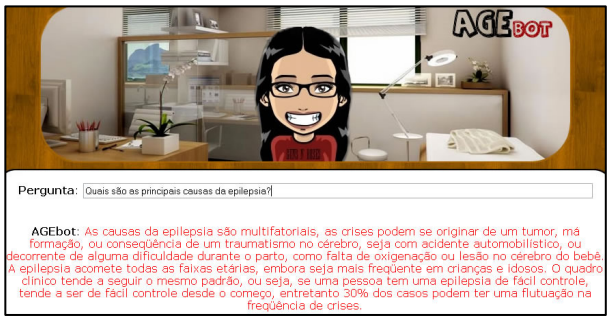


Figura 2. Interface de conversação com a AGEbot

A interface de comunicação do *chatterbot* é composta por um simples formulário separado pelos campos de pergunta e resposta fornecida pelo *bot*. O usuário deve digitar a dúvida, formulada em linguagem natural, no campo da pergunta e pressionar a tela *enter*. Em seguida, o formulário envia a questão para o interpretador do Pandorabots (Program Z, desenvolvido em Common LISP) para que ele busque na base AIML em qual regra de categorias a sentença melhor se enquadra. Ao final, o interpretador retorna a resposta encontrada em poucos segundos e imprime na tela.

4.4. Divulgação

A divulgação do *chatterbot* começou através do desenvolvimento de um *website* contendo informações sobre o projeto, como descrição, justificativa, objetivos e um *link* direcionado a página do sistema Pandorabots, onde o *bot* inicialmente foi hospedado. Após a criação da página do projeto, teve início o processo de indexação do site ao sistema de buscas do Google. Através desta indexação, a página do *chatterbot* poderia ser encontrada facilmente pela pesquisa do Google, através de busca por palavras-chave como *chatterbot*, epilepsia, perguntas, questões, FAQ, entre outras. Por último foi realizado uma busca por blogs, sites, comunidades e fóruns a respeito da epilepsia, criando tópicos a fim de publicar o endereço do *bot*.

5. Funcionamento do AGEbot

O AGEbot é um agente de conversação sobre a área de epilepsia e foi desenvolvido através dos modelos da última geração de *chatterbots*, através da linguagem AIML. A escolha desse método deve-se ao fato de ser uma linguagem nova e desenvolvida especialmente para a criação de simuladores de conversação. Além disso, a linguagem AIML pode também funcionar independente de qualquer linguagem de programação.

Nas próximas seções serão apresentadas algumas técnicas e tratamentos utilizados para o desenvolvimento do *chatterbot*.

5.1. Método de Aprendizado

A base de conhecimento inicial foi desenvolvida através de um método chamado estratégia antecipatória, descrito na metodologia através dos itens de coleta e da documentação das dúvidas mais frequentes.

Em seguida, com as perguntas bases já formadas, iniciou-se o processo de aprendizado através da utilização do sistema de *targeting* do sistema Pandorabots, onde são mostradas as entradas dos usuários e suas respectivas respostas. Assim, é possível visualizar qual resposta foi adequada e nos casos em que a reposta não é satisfatória, o desenvolvidor do *chatterbot* (*botmaster*) pode recadastrar as respostas corretas.

Finalmente, o *bot* é colocado em funcionamento para que possa se iniciar a estratégia de análise de *logs*. Nessa etapa, foi utilizado método de aprendizagem por reforço (Figura 3), onde o agente irá aprender e evoluir com a ajuda do ambiente, ou seja, através da interação com os usuários, onde foram guardados registros (*logs*) de toda a conversação realizada.

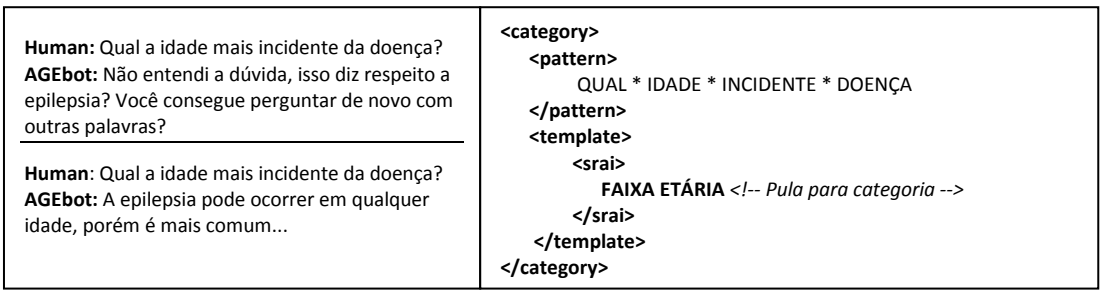


Figura 3. Exemplo da criação de nova categoria através do método de reforço

Com esse histórico, o *botmaster* é capaz de encontrar alguns erros e incoerência de respostas, além de poder criar novas regras sobre as perguntas que não foram respondidas pelo *bot*.

5.2. Tratamentos das respostas

Por se tratar de uma linguagem de marcação, a utilização de comandos HTML dentro do AIML tornam as respostas do *chatterbot* muito mais dinâmicas e criativas, não dependendo apenas de texto para poder se expressar. Com isso, alguns modelos de respostas diferenciados foram implementados, como, por exemplo, a utilização de textos, imagens ilustrativas e links para vídeos e *websites*.

Para que o *bot* tenha a capacidade de responder de forma diferente sempre que lhe for feita a mesma pergunta, foram cadastrados diferentes tipos de definições que apontam para o mesmo assunto. Através da *tag* <random>, o simulador é capaz de escolher de forma randômica uma resposta contida na base e apresentar outros conceitos sobre a mesma dúvida.

5.3. Tratamentos das perguntas

Algumas palavras ou perguntas podem ser feitas de inúmeras formas pelas pessoas, por isso o *bot* deve ser capaz de entender da mesma maneira questões semelhantes. É

através da recursividade que o *chatterbot* reconhece essas semelhanças e realiza o redirecionamento para a resposta em comum.

O AGEbot é um *chatterbot* com características de FAQ, ou seja, foi pensado a fim de responder apenas perguntas sobre um determinado assunto. Porém, também foram implementados alguns tratamentos de sentenças afirmativas realizadas pelos usuários. Através de uma base genérica, o *bot* é capaz de responder alguns afirmações dos usuários, como, por exemplo, quando o usuário não entende a resposta apresentada pelo simulador ou quando discorda de algum conceito.

É comum que no decorrer da conversa o usuário acabe desvirtuando o assunto, realizando perguntas que não fazem parte do escopo do *chatterbot* ou que então faça perguntas que ainda não foram atribuídas à base de conhecimento. Por esse motivo, o *chatterbot* também utiliza uma base genérica, onde ele dará uma resposta de forma aleatória tentando estimular o usuário a reformular sua pergunta ou voltando ao foco do tema. Dessa forma, é possível dar continuidade ao diálogo, sem que *chatterbot* fique sem respostas, o que pode acarretar no desinteresse do usuário.

6. Resultados

Para monitorar os acessos de usuários a página do *chatterbot* foram utilizados recursos da ferramenta *Google Analytics*. Com ela pode-se visualizar um total de 94 acessos a página, com mais ênfase nos estados de São Paulo (34%), Rio Grande do Sul (16%) e Minas Gerais (9%). As principais origens dos acessos foram através de sites de referência (53%), acessos diretos a página (24%) e mecanismos de pesquisa (23%). Dessa forma podemos ver que o processo de divulgação teve um bom resultado através de fóruns e comunidades sobre a epilepsia, além de um quarto dos acessos terem sido através da indexação realizada pela Google.

O *chatterbot* foi disponibilizado para interação com usuários em um período de quatro semanas e, durante esse período, foram gerados *logs* (históricos das conversações) para auxiliar no treinamento, desenvolvimento e validação do programa. A partir destes *logs*, obteve-se um total de 399 perguntas realizadas, das quais 46,4% foram respondidas com sucesso, 49,1% não possuíam respostas cadastradas e 4,5% não se enquadravam no escopo da base de conhecimento ou eram muito detalhadas e específicas. Lembrando que, o *bot* nunca deixava o usuário sem respostas, sempre tentando incentivá-lo a realizar outras perguntas relacionadas ao assunto através de repostas genéricas cadastradas na base.

Cabe ressaltar que, nas primeiras semanas de acesso ao robô a base de conhecimento estava em fase inicial de desenvolvimento e não era capaz de responder boa parte das dúvidas dos usuários. Por isso, devemos analisar o desempenho individual de cada semana, para que possamos visualizar o crescimento da quantidade perguntas que puderam ser respondidas. Na Figura 4 é apresentado um gráfico demonstrando tal crescimento a cada semana de *logs* analisados.

É possível verificar, através do gráfico da Figura 4, que a utilização do aprendizado da base de conhecimento pelo método de reforço mostrou-se muito eficiente neste trabalho, pois a cada semana foram realizadas análises sobre os *logs* de conversação, o que possibilitou a criação de novas categorias para enquadrar questões

que não faziam parte da base ou corrigir regras redirecionando para respostas já existentes. Com isso, percebe-se uma crescente na quantidade de questões respondidas corretamente e podemos dizer que o *chatterbot* está aprendendo e evoluindo a cada interação com o usuário. Dessa forma, se ignorarmos as perguntas que não faziam parte do escopo da base, teremos um crescimento ainda maior em cada semana, como ilustrado no gráfico da Figura 5.

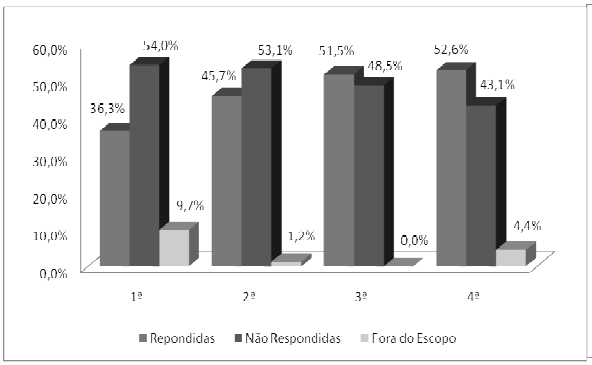


Figura 4. Gráfico da porcentagem de questões respondidas durante cada semana de interações com usuários

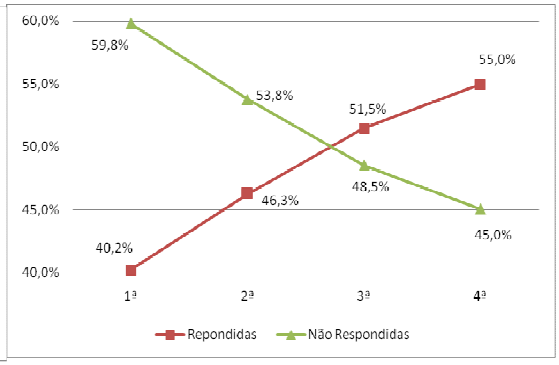


Figura 5. Porcentagem de perguntas respondidas corretamente, ignorando as questões que não faziam parte do escopo da base de conhecimento

Da mesma forma, comparando a primeira e última semana do período proposto para treinamento e validação do *chatterbot*, percebe-se que o *bot* teve um crescimento de 14,8% na quantidade de questões respondidas com sucesso e que a tendência é que cresça mais ainda com o tempo, através da interação com outros usuários e na análise de *logs* para continuação do seu aprendizado.

7. Conclusões

A terceira geração de *chatterbots* trouxe mais destaque e facilidade no desenvolvimento de simuladores de diálogo através da utilização da linguagem AIML, uma linguagem de marcação poderosa e simples de ser usada, possuindo uma base de dados de fácil compreensão e adaptável a diversos idiomas e linguagens de programação.

Através dos resultados obtidos, é possível concluir que o *chatterbot* atingiu seus objetivos, sendo capaz de responder e sanar dúvidas de usuários mais leigos a respeito da epilepsia. O trabalho também demonstrou que *bot* está aprendendo e aumentando sua base de conhecimento a cada interação com o usuário, mostrando ser capaz de atingir um alto índice de respostas no decorrer de seu aprendizado.

Como trabalhos futuros, pretende-se: incrementar a base de conhecimentos para dúvidas mais específicas; ampliar o escopo para outras doenças relacionadas; desenvolver mais categorias com informações a respeito do AGEbot, atribuindo mais personalidade ao sistema; aplicar métodos de "memória de última sentença" e predicados para melhorar o contexto das conversações; integrar o *bot* a um sistema tutor em epilepsia e avaliar o sistema quanto à usabilidade e satisfação do usuário.

Os resultados obtidos apontam indícios de que a utilização de *bots* de conversação possibilita um aumento no interesse e na curiosidade dos usuários, além de

apresentar uma facilidade de aprendizado a partir de uma posição mais ativa por porte do usuário, através de raciocínio para formular perguntas a serem respondidas. Além disso, é possível ampliar e adaptar facilmente seu escopo de conhecimento para diversas áreas, não somente na saúde, mas também para áreas educativas e comerciais.

8. Referências

ASPESI, Nelson V; PERLA, Alexandre S. (2001) Epilepsia/Convulsão – Ataque Epiléptico. Disponível em: <http://www.abcdasaude.com.br/artigo.php?95>. Acessado em: 20 maio 2010.

CARVALHO, Lucimar M. F. (2007) Modelos de Aprendizagem para Sistemas Neuro-difusos. Tese (Doutorado em Engenharia Elétrica) – UFSC, Florianópolis.

EPILEPSIA. Liga Brasileira Contra Epilepsia – Sobre Epilepsia. (2002) Disponível em: <http://www.epilepsia.org.br/epi2002/temas_indice.asp>. Acessado em: 26 maio 2010.

GUERREIRO, Carlos A. M.; GUERREIRO, Marilisa M.; CENDES, Lopes. (2000) Epilepsia. São Paulo: Lemos Editorial.

KUROMOTO, Paula Y. ; MARQUES,Paulo Mazzoncini de A. ; OLIVEIRA, Lucas Ferrari; ANA, Lauro Wichert; FELIPE, Joaquim Cezar. (2007) Sistema Informatizado Amigável para Auxílio ao Diagnóstico em Epilepsia. In: VII Workshop de Informática Médica. Porto de Galinhas.

LAUREANO, Eduardo A. G. C. (1999) ConsulBot – Um Chatterbot Consultor para Ambientes Virtuais de Estudo na Internet. Disponível em: <http://www.di.ufpe.br/~tg/1999-1/eagcl.doc>. Acessado em: 20 nov. 2009.

LEONHARDT, Michelle D.; CASTRO, Daiane D.; DUTRA, Renato L. S.; TAROUCO, Liane M. R. (2003) "ELEKTRA: Um Chatterbot para Uso em Ambiente Educacional". In: *Renote*. Porto Alegre.

NEVES; André M. M.; BARROS, Flávia A. (2005) "iAIML: Um Mecanismo para Tratamento de Intenção em Chatterbots". In: XVIII Encontro Nacional de Inteligência Artificial. São Leopoldo.

SBC. Processamento de Linguagem Natural. Disponível em: < http://www.sbc.org.br/index.php?option=com_content&view=category&layout=blog &id=50&Itemid=71 >. Acessado em: 03 nov. 2009.

SILVA, P. (1987) As Bases Farmacológicas da Terapêutica. Trad Goodman & Gilman, Rio de Janeiro: Guanabara Koogan.

SGANDERLA, Rachele B.; FERRARI, Débora N.; GEYER, Cláudio F. R. (2003) "BonoBOT: Um Chatterbot para Interação com Usuários em um Sistema Tutor Inteligente". In: XIV Simpósio Brasileiro de Informática na Educação. Rio de Janeiro.

TEIXEIRA, Sérgio; RAMIRO, Thiago B.; OLIVEIRA, Elias; MENEZES, Crediné S. (2005) "Chatterbots em ambientes de aprendizagem – uma proposta para a construção de bases de conhecimento". In: XI Workshop de Informática na Escola. São Leopoldo.

Avaliação de habilidades sensório-motoras em Ambientes de Realidade Virtual para treinamento médico: uma Revisão Sistemática

Alexandre M. dos Anjos[1,2], Fátima L. S. Nunes[1,3], Romero Tori[1,3]

[1]Interlab – Laboratório de Tecnologias Interativas – Escola Politécnica/USP
Av. Prof. Luciano Gualberto, nº 380 – 05508-010 – São Paulo – fone (11) 3091-5282

[2] Universidade Aberta do Brasil - Universidade Federal de Mato Grosso (UFMT)
Av. Fernando C. da Costa, nº 2367 – 78060-900 – Cuiabá-MT – fone: (65) 3615-8028

[3]LApIS - Laboratório de Aplicações de Informática em Saúde
Programa de Pós-graduação em Sistemas de Informação – Escola de Artes, Ciências de Humanidades/USP
Av. Arlindo Béttio, 1000 - Ermelino Matarazzo - São Paulo - CEP: 03828-000

{dinteralexandre,fatima.nunes,tori}@usp.br

Abstract. *Studies on the contributions of the Sensory Motor Skills (SMS), are essential to the medical training area. Considering experiences found in this particular context, this proposal will consist of an investigation that uses a process of Systematic Review (SR) to retrieve the state of the art in relation to methodologies for evaluate the skills acquisition. It was developes a revision of the existing literature, identifying metrics that are frequently used to evaluation strategies of acquisition of SMS both in Real Environments (RE) and Virtual Environments (VE). After identifying these metrics an analysis was developed to categorize them according to their similarities and application context. Lastly, the study reveals the possible metrics that will contribute to the creation and formulation of a new evaluation methodology of SMS for Virtual Reality environments (VR).*

Resumo. *Estudos sobre as contribuições oriundas do campo de desenvolvimento de Habilidades Sensório-Motoras (HSM) são imprescindíveis para área de treinamento médico. Considerando experiências encontradas especialmente nesse contexto, a proposta em pauta, consiste em uma investigação que se utiliza de um processo de Revisão Sistemática (RS) para recuperar o estado da arte sobre metodologias de avaliação de aquisição de HSM. Para isso, foi necessário desenvolver uma revisão de literatura existente, identificando métricas que são frequentemente usadas em estratégias de avaliação de aquisição de HSM tanto em Ambientes Reais (AR) quanto Ambientes Virtuais (AV). Após a identificação dessas métricas, desenvolveu-se uma análise para categorizá-las, de acordo com o seu grau de similaridade e, contextos de aplicação. Ao final, o estudo revela possíveis métricas, que contribuirão para criação ou a formulação de uma nova metodologia de avaliação de HSM para ambientes de Realidade Virtual (RV).*

1. Introdução

Nos últimos anos, há relatos de experiências que usam RV para simulações cirúrgicas em áreas de treinamento médico acompanhadas de uma significativa melhora no desempenho das atividades desses profissionais (SEYMOUR *et al*. 2002).

Não obstante a utilização de simulações observa-se também que é possível efetuar a transferência de conhecimentos ou habilidades técnicas, por meio da utilização de salas de operações virtuais com ambientes de aprendizagem voltados para área de treinamento médico (SEYMOUR *et al*. 2002).

Nesse sentido, com o crescimento de aplicações da área de Realidade Virtual (RV) para aquisição de habilidades por meio de estratégias de simulação, configura-se um novo método de apoio a aprendizagem que utiliza recursos de educação virtual para a área de treinamento médico.

Essa nova maneira de ensinar trouxe à tona desafios inéditos no sentido de refletir sobre a implementação de modelos que pudessem ser alternativos a contextos reais de aprendizagem.

Essas alternativas merecem destaque quando se observa que a motivação para estratégias virtuais de ensino-aprendizagem é notória, especialmente por apresentar soluções que minimizam a necessidade de experiências com alto custo, como aquelas em que são exigidos materiais e recursos físicos em ambientes educacionais presenciais e reais (TORI; NUNES; GOMES E TOKUNAGA, 2009).

Em experiências que têm como ênfase o desenvolvimento de atividades sensório-motoras incluem-se também contextos que envolvem estratégias de ensino-aprendizagem para a aquisição de habilidades. Uma das reflexões a tecer nesse contexto refere-se à compreensão sobre os processos de avaliação de aquisição de habilidades em ambientes tridimensionais de acordo com concepções presentes na área de RV. Avaliar habilidades, por exemplo, passa a ser uma discussão de caráter complexo, especialmente quando atravessa um espaço contínuo entre a realidade e a virtualidade.

Para avaliar essas habilidades é necessário capturar as informações de interação, armazená-las e reproduzi-las, de alguma forma, em modelos experimentais. Em alguns momentos, porém, é preciso resgatar experiências, oriundas não apenas de contextos virtuais, mas também aquelas ênfases em experiências reais ou provenientes de aplicações virtuais bidimensionais.

A partir do momento em que é possível identificar variáveis presentes tanto em contextos reais de avaliação de habilidades como em contextos virtuais, torna-se possível estudar os fatores que devem ser considerados para a criação de uma nova metodologia de avaliação.

No contexto de Ambientes Virtuais (AV), a RV e a Realidade Aumentada (RA) envolvem um controle altamente interativo de processos computacionais em ambientes tridimensionais. Um usuário (na condição de aprendiz) pode aceder a espaços virtuais, visualizando, manipulando e explorando os dados da aplicação em tempo real, utilizando seus sentidos, particularmente os movimentos naturais do seu corpo. A grande vantagem desse tipo de interface é que o conhecimento intuitivo do usuário a respeito do mundo real pode ser transferido para manipular informações em AVs.

Este trabalho se insere em um contexto que tem como objetivo desenvolver um estudo sobre os parâmetros e variáveis encontrados em metodologias para avaliação de aquisição de Habilidades Sensório-Motoras (HSM) em Ambientes de RV ou RA. Investigou-se também, metodologias de ambientes reais que pudessem ser adaptadas para AVs, compondo, ao final, uma categorização a partir do grau de similaridade entre os parâmetros encontrados. Assim, o objetivo principal deste artigo é apresentar o estado da arte sobre avaliação de HSM em ambientes reais e virtuais, a partir de uma Revisão Sistemática da literatura, discutindo-se os parâmetros encontrados a fim de distinguir indícios que podem levar à definição de uma metodologia adequada para a avaliação da aquisição dessas habilidades em ambientes de RV e RA.

A segunda e a terceira seção do artigo apresentam, respectivamente, conceitos sobre HSM e Revisão Sistemática. Na quarta seção apresenta-se o processo metodológico e, ao final, resultados e discussões encontrados a partir das métricas, parâmetros e reflexões são disponibilizados.

2. Habilidades sensório-motoras

Para delimitar a concepção de treinamento educacional, adotar-se-á, como referência, a terminologia de HSM, a partir de definições da área de comportamento motor, que discute atividades sensoriais motoras desenvolvidas por seres humanos. Nesse sentido Magill (2000 Apud PÓVOAS e RODRIGUES, 2007), apresentam o conceito de habilidades motoras como “tarefas com uma finalidade específica a ser atingida e que relacionam-se a aspectos de movimento podendo subdividir-se em habilidades grossas ou finas”.

Póvoas e Rodrigues (2007) definem as habilidades motoras grossas a partir do pressuposto de que “[...] utilizam os grandes grupos musculares do corpo para produzir uma ação. Fazem parte dessas habilidades as atividades como pular e caminhar, nas quais a precisão de refinamento dos movimentos é menor”, enquanto que as habilidades finas são caracterizadas pelos movimentos de músculos pequenos.

Em outro contexto, Go Tani *et al.* (2004) definem a importância da informação sensorial durante a execução de movimentos, uma vez que ela é comparada como uma referência de avaliação em memória humana em um processo contínuo e sequenciado de ações.

Nesse sentido, a terminologia "sensório-motora" é utilizada com o objetivo de indicar a compreensão da percepção sensorial humana associada a aspectos de interação em RV envolvendo fatores de percepção humana, sejam eles visuais, sonoros ou táteis.

3. Revisão Sistemática

Como estratégia metodológica para verificar as contribuições e o estado da arte da pesquisa, procurou-se alicerçar esse processo a partir das concepções de Kitchenham (2004) e Biolchini *et al*. (2005) sobre Revisão Sistemática (RS), conforme descreve-se a seguir.

De acordo com Kitchenham (2004), uma revisão sistemática de literatura é uma forma de identificar, avaliar e interpretar todas as pesquisas disponíveis relevantes para uma questão específica, área temática, ou fenômeno de interesse. Para Biolchini *et al*. (2005) RS trata-se de um método de pesquisa explícito e rigoroso, que procura

identificar o conhecimento científico em uma determinada área, por meio da coleta, combinação e avaliação crítica de descobertas de diversas abordagens já realizadas. Segundo esses autores, a importância do estabelecimento de um processo de RS, é também notória, quando verifica-se a definição de fases, a saber: a) Planejamento da revisão; b) Procedimentos de condução e extração da RS e, ao final, c) Procedimentos de elaboração de relatórios da RS.

Durante a fase de planejamento da RS define-se primeiramente um protocolo, no qual constam o problema de pesquisa, de forma a delimitar a natureza essencial e a eficiência de determinadas práticas ou métodos no domínio em pauta, bem como aspectos como fontes de pesquisa, *Strings* de busca utilizadas (sequência ou combinação de palavras-chaves para a pesquisa), critérios de inclusão e exclusão de trabalhos encontrados na literatura e forma de sumarização dos resultados.

Na fase de condução da RS executa-se a formalização de um processo de pesquisa conforme o protocolo estabelecido. Nessa etapa, de forma sumarizada, devem ser listados todos os trabalhos encontrados na literatura e aplicados os critérios de inclusão e exclusão definidos no protocolo. O resultado final é uma lista de artigos que devem ser incluídos na fase seguinte.

Inicia-se então a fase de extração que consiste em um detalhamento das informações por meio da extração de variáveis e anotações específicas de referências que podem agregar valor ou alterar os rumos do processo de pesquisa desenvolvido até então. Ao final, desenvolve-se um relatório incluindo-se todas as etapas do processo de RS, conclusões e resultados encontrados.

4. Metodologia

A questão de pesquisa que norteou o presente estudo teve como objetivo investigar parâmetros e variáveis que são encontrados em metodologias para avaliação de HSM em Ambientes de RV e RA ou em metodologias de ambientes reais que possam ser adaptadas para ambientes virtuais. Foi definido o planejamento da RS, conforme apresentado a seguir.

Inicialmente, delinearam-se fontes de buscas a partir de pesquisas no Portal de Periódicos da CAPES (*IEEE journals, IEEE conferences, ACM journals, ACM conferences*); *Village*, *SCOPUS*, Biblioteca Digital da Sociedade Brasileira de Computação, Referências acadêmicas presentes no *Google Scholar* e Biblioteca de teses da Universidade de São Paulo.

Nessa etapa, consolidaram-se *strings* de busca formuladas em inglês, a partir de palavras ou sentenças, como **Metodologia, Estratégias, Aulas, Práticas, Treinamentos.** ("*methodology" or "strategy" or "pratice" or "classes" or "training*"); **Avaliação.** ("*assessment" or "evaluation*"); **Realidade Virtual.** ("*Virtual Reality*") e **Aquisição de Habilidades.** ("*skill acquisition*").

A fim de estabelecer os interesses da pesquisa, o protocolo construído definiu os seguintes critérios de inclusão de trabalhos: artigos publicados nos últimos 5 anos; artigos que apresentem metodologias específicas para avaliação de aquisição de HSM em ambientes de Realidade Virtual (RV) ou em Ambientes Reais (AR); artigos que utilizam alguma metodologia específica de avaliação de aquisição de HSM em AV ou

em AR e artigos que executam alguma avaliação de aquisição de HSM sem especificar metodologias.

De maneira semelhante, foram elaborados os seguintes critérios de exclusão: artigos que não contenham estratégias ou metodologias de avaliação HSM e artigos já encontrados e considerados para análise em outra base. Esses passos finalizaram a etapa de planejamento, possibilitando a condução da RS e a extração de dados.

4.1. Condução da Revisão e Extração de Dados

Nessa etapa as *Strings* elaboradas no processo de planejamento foram executadas em cada uma das máquinas de busca das fontes selecionadas, documentando-se e registrando-se os trabalhos em um formulário de condução da revisão mediante a leitura dos resumos das obras, aplicando-se os critérios de inclusão e exclusão previamente estabelecidos totalizando 45 artigos das seguintes bases: ACM (13), Village (2), IEEE (10), ISI (6) e SCOPUS (14). O processo foi realizado no período de 01/07/2010 a 30/10/2010.

5. Resultados e discussões

Para melhor subsidiar as análises e discussões da presente pesquisa, a partir dos 8 artigos selecionados, procurou-se classificar os experimentos que aconteciam em contextos virtuais ou reais para avaliação de habilidades HSM conforme disponibilizado na Tabela 1.

Outro fator a ser destacado avaliando-se o artigo 4 (Tzafestas *et al* 2006.) do Quadro 1, é que apenas uma das experiências utiliza métodos de avaliação convencionais, como provas, questionários e outros, que não são adaptados especificamente para avaliação de HSM.

Além disso, observa-se que 4 (50%) das experiências, são especificas da área de treinamento médico – artigos 1, 2, 3 e 8 (Panait et al. 2009; Van der Meijden; Schijven, 2009; Madan; Frantzides, 2007; Kolesnikov, *et al.* 2009) apresentados na Tabela 1.

Tabela 1. Análise dos critérios de inclusão atendidos

Citação	Ambientes		**Apresentam** metodologias **específicas** de avaliação de aquisição de HSM em ambientes Reais ou de RV.	**Utilizam** alguma metodologia **específica** de avaliação de aquisição de HSM em ambientes Reais ou de RV.	**Executam** alguma avaliação de aquisição de HSM sem especificar metodologias
	Real	Virtual			
1-Panait et *al*. (2009)		•		•	
2-Van der Meijden; Schijven, (2009).		•		•	
3-Madan; Frantzides, (2007).		•		•	
4-Tzafestas; Palaiologou; Alifragis, (2006).		•			•
5-Dawei; Asim; Saeid, (2009).	•	•	•		
6-Chowriappa *et al.* (2009).		•	•		
7-Watanabe; Katsura, (2010).		•		•	
8-Kolesnikov *et al.* (2009).		•	•		

Após a leitura dos artigos de forma mais detalhada, foi possível constatar a existência de um número considerável de variáveis que poderiam ser candidatas para a

construção de uma nova metodologia de avaliação de HSM virtuais, considerando especificamente a área de treinamento médico virtual. A fim de melhorar a apresentação dos resultados da RS, adotou-se como estratégia inicial uma categorização ou agrupamento de métricas, de acordo com o respectivo grau de similaridade encontrado, conforme pode ser observado nas Tabelas 2 a 7.

A primeira categoria de métrica frequente em estratégias de avaliação de HSM foi encontrada em 5 artigos selecionados (62,5%), e refere-se aos erros cometidos por aprendizes durante o desenvolvimento de suas atividades de treinamento. (Tabela 2).

Tabela 2. Métricas de Erros.

Detalhamento da dimensão	Explicação
Erros genéricos	Métricas quantitativas para análise do número de erros presentes em simuladores (Físicos ou virtuais) (DAWEI; ASIM; SAEID, 2009).
Erros de baixo nível de competência técnica	Número de erros apresentados em tarefas de complexidade simples, desenvolvidas em simuladores reais ou virtuais (TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006).
Erros de alto nível de competência técnica	Número de erros apresentados em tarefas de complexidade (média ou alta) desenvolvidas tanto em simuladores físicos como reais. (TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006).
Erro de posicionamento	Erros de posicionamento em atividades que requerem manipulação e movimentação tanto em simuladores físicos como reais. (CHOWRIAPPA, SUBRAHMANIYAN, SRIMATHVEERAVALLI e KESAVADAS, 2009)
Perda de objetos (agarramento e desprendimento involuntário)	Métrica que considera o erro ao agarrar objetos tanto em simuladores físicos como reais. O Desprendimento acidental é um exemplo (PANAIT *et al.* 2009).
Danos causados, consequentes do contato ou colisão de simuladores	Métrica que considera colisões acidentais na manipulação ou movimentação de objetos tanto em simuladores físicos como virtuais (PANAIT *et al.* 2009; VAN DER MEIJDEN e SHIJVEN, 2009).

Conforme observado na Tabela 3, outra categoria encontrada considerou também o tempo total ou parcial utilizado pelos aprendizes durante as atividades de treinamento, cujas métricas foram encontradas em 5 artigos (62,5%), conforme descreve-se a seguir:

Tabela 3. Métricas de tempo.

Detalhamento da dimensão	Explicação
Tempo para **simulação – usando RV** (*Conclusão da tarefa*)	Métrica com o objetivo de medir o tempo para conclusão de tarefas ou habilidades, usando aplicações de (RV). (MADAN e FRANTZIDES, 2007; VAN DER MEIJDEN e SHIJVEN, 2009; TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006; DAWEI; ASIM; SAEID, 2009).
Tempo para **simulação – usando RV** (*Por etapa ou fração da tarefa*)	Métrica com o objetivo de medir o tempo para frações de tarefas ou habilidades, usando aplicações de (RV) (TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006; CHOWRIAPPA, SUBRAHMANIYAN, SRIMATHVEERAVALLI e KESAVADAS, 2009; DAWEI, ASSIM e SAEID, 2009).
Tempo para simulação: usando **simuladores físicos** (*Conclusão da tarefa*)	Métrica com o objetivo de medir o tempo para conclusão de tarefas ou habilidades, usando simuladores físicos. Exemplo: Box de laparoscopia manual (TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006; MADAN e FRANTZIDES, 2007).
Tempo para simulação: usando **simuladores físicos** *(Por etapa ou fração da tarefa)*)	Métrica com o objetivo de medir o tempo para frações de tarefas ou habilidades, usando simuladores físicos. Exemplo: Box de laparoscopia manual (TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006).
Tempo real (*Conclusão da tarefa*)	Considera-se apenas o tempo de conclusão da tarefa, independente do tipo de ambiente utilizado (real ou virtual). Exemplo: Sistemas híbridos que utilizam simuladores físicos ou aplicações de RV (TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006).
Tempo real (*por etapa ou fração da tarefa*)	Considera-se apenas o tempo de fração da tarefa, independente do tipo de ambiente utilizado (real ou virtual). Exemplo: Sistemas híbridos que utilizam simuladores físicos ou aplicações de RV (TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006).

Considerando a necessidade de capturar informações para avaliação, especialmente em dispositivos que trabalham com RV, foi possível notar também a presença de indicadores que revelaram medidas de esforço físico e orientação em processos de treinamento com 5 artigos (62,5%) encontrados, conforme descreve-se na Tabela 4 a seguir:

Tabela 4. Métricas de esforço físico e orientação.

Detalhamento da dimensão	Explicação
Medida de **força** aplicada	Métrica que registra a quantidade de força aplicada aos diferentes graus de liberdades de aplicações de (RV) (VAN DER MEIJDEN e SHIJVEN, 2009).
Força de tensão	Métrica que registra a quantidade de força aplicada aos diferentes graus de liberdades de aplicações de (RV) (PANAIT *et al.* 2009).
Força de apreensão	Métrica que registra a quantidade de força aplicada a dispositivos que têm como objetivo apreender um determinado objeto (PANAIT *et al.* 2009).
Força	Métrica que registra a quantidade de força aplicada aos diferentes graus de liberdades de aplicações de (RV) (CHOWRIAPPA, SUBRAHMANIYAN, SRIMATHVEERAVALLI e KESAVADAS, 2009; WATANABE e KATSURA, 2010; KOLESNIKOV, ZEFRAN, STEINBERG, e BASHOOK, 2009)
Posição	Métrica que registra os diferentes graus de liberdades de aplicações de (RV) (CHOWRIAPPA, SUBRAHMANIYAN, SRIMATHVEERAVALLI e KESAVADAS, 2009; WATANABE e KATSURA, 2010; KOLESNIKOV, ZEFRAN, STEINBERG, e BASHOOK, 2009)
Velocidade	Métrica que registra a velocidade de realização de procedimentos simulados. Podem ser considerados tanto em ambientes reais como em ambientes de RV (CHOWRIAPPA, SUBRAHMANIYAN, SRIMATHVEERAVALLI e KESAVADAS, 2009; WATANABE e KATSURA, 2010)
Massa	Medida associada ao peso dos objetos que serão manipulados (WATANABE e KATSURA, 2010)

Além do esforço físico e da orientação, foi possível encontrar também experiências que permitiam a comparação de comportamentos entre aprendizes durante a realização de atividades de treinamento. Nesse sentido, foram encontrados 4 artigos de relevância (50%), conforme descreve-se na Tabela 5 a seguir:

Tabela 5. Métricas de registro de percursos e procedimentos.

Detalhamento da dimensão	Explicação
Gravação de padrões de comportamento	Métrica utilizada para gravação de padrões de comportamentos em ambientes de RV. A captura normalmente é desenvolvida com o objetivo de identificar padrões. Exemplo: Padrões de especialistas e padrões de novatos (WATANABE e KATSURA, 2010)
Gravação de trajetórias	Métrica utilizada para gravar trajetória de usuários perante simuladores de RV. Algumas experiências usam a gravação de trajetos para estudos ou orientações de novos usuários por meio de diversas técnicas. Exemplo: *Feedback* de força (MADAN e FRANTZIDES, 2007; KOLESNIKOV, ZEFRAN, STEINBERG, e BASHOOK, 2009; DAWEI; ASIM; SAEID, 2009)

De forma mais simplificada, apenas três dos artigos (37,5%) de relevância considera processos de simetria de destreza, sucesso ou insucesso, conforme se apresenta na Tabela 6.

Tabela 6: Métricas de destreza, sucesso ou insucesso de procedimentos.

Detalhamento da dimensão	Explicação
Destreza	Métrica que mede a eficiência, sucesso ou insucesso para realização de determinados procedimentos, seja em simuladores físicos ou virtuais (PANAIT *et al.* 2009).
Sucesso ou insucesso	Métrica que mede a eficiência, sucesso ou insucesso para realização de determinados procedimentos, seja em simuladores físicos ou virtuais (MADAN e FRANTZIDES, 2007; TZAFESTAS, PALAIOLOGOU e ALIFRAGIS, 2006).

Especialmente em artigos que consideram capacidade dos usuários adquirirem habilidades a partir de medidas de usabilidade, foram encontradas métricas que consideraram o grau de facilidade e memória de aprendizes durante a realização de suas atividades de treinamento.

De forma destacada, observou-se também medidas muito reveladoras em processos de avaliação de HSM, como as métricas de automaticidade que são capazes de revelar a capacidade ou a desenvoltura dos usuários na realização de suas atividades, encontradas em 4 artigos selecionados (50%), conforme observa-se na Tabela 7.

Tabela 7. Métricas de usabilidade ou de avaliação de grau de experiência do usuário.

Detalhamento da dimensão	Explicação
Medida de automaticidade da habilidade	Medida de automaticidade é proporcional ao tempo que determinado aprendiz dedica para a aprendizagem de determinada tarefa, a ponto de fazê-la de forma automática ou sem necessidade de grandes esforços cognitivos. (WATANABE e KATSURA, 2010)
Medida de facilidade da habilidade	Utilizando-se de questionários ou outros instrumentos, de forma subjetiva alguns avaliadores tentam dimensionar o grau de facilidade para desenvolvimento de determinadas tarefas, seja em experiências físicas reais ou virtuais (MADAN e FRANTZIDES, 2007)
Capacidade de o usuário recordar	Medida de aprendizagem de habilidades pela capacidade de que o usuário gravar percursos parciais ou totais de determinadas tarefas, sejam elas em experiências físicas reais ou virtuais) (DAWEI; ASIM; SAEID, 2009)
Experiência dos usuários	Alguns autores tentam medir a aprendizagem de habilidades levando em consideração a experiência do usuário em determinado domínio da informação. Teoricamente, os usuários trabalham a hipótese de que um usuário mais experiente no domínio de determinada aplicação pode apresentar melhores resultados em relação àquele que não possui grau de experiência de mesmo nível (VAN DER MEIJDEN e SHIJVEN, 2009).

As categorias evidenciadas nas tabelas apresentadas indicam que há métricas para avaliação aplicadas em áreas de domínio específico, como por exemplo, treinamento médico. Porém não foram encontrados indícios de aplicações dessas métricas de forma mais generalizada, ou seja, como uma metodologia única para avaliação de HSM.

Uma das métricas que mereceram destaque nas tabelas anteriores foi a medida de automaticidade, que tornou evidente a necessidade de um aprofundamento dos estudos de avaliação de HSM em RV aproveitando as contribuições originadas partir de uma área que estuda o desenvolvimento de habilidades motoras humanas.

6. Conclusão

O presente artigo apresenta os resultados de um processo de RS, com o objetivo de levantar métricas ou variáveis comumente utilizadas para avaliação de HSM. As evidências encontradas foram categorizadas por sua similaridade. As discussões dos autores revelaram a existência de estratégias especificas para avaliação de HSM e especialmente na área de treinamento médico, que foram identificadas em 50% dos artigos selecionados.

Observam-se também experiências que se utilizam de simuladores virtuais e físicos, e outras que revelam a preocupação com estratégias de avaliação HSM em ambientes reais, o que traz como desafio a generalização de uma proposta que possa transitar entre espaços reais, físicos ou virtuais de avaliação de habilidades.

Dessa forma, o processo de RS revelou a presença de experiências aplicadas apenas em avaliações especificas de HSM, o que dificulta uma generalização de uma metodologia que possa adequar-se a diferentes estratégias de treinamento.

Para isso, um dos grandes legados do estudo em questão é a possibilidade de verificar estágios gradativos de aquisição de HSM em diversas variáveis como a automaticidade de aprendizagem e outras métricas específicas que podem contribuir para a criação de uma metodologia para avaliação de HSM e especialmente aplicações na área de treinamento médico que se utiliza de RA e RV.

7. Agradecimentos

Agradecemos a FAPEMAT, aos integrantes do INTERLAB – Laboratório de Tecnologia Interativas - Escola Politécnica/USP e LapIS – Laboratório de Aplicações de Informática em Saúde – Escola de Artes, Ciências e Humanidade/USP e INCT-MACC Instituto Nacional de Ciência e Tecnologia em Medicina Assistida por Computação Científica.

8. Referências

Biolchini, J.; Mian, P. G.; Natali, A. C. C.; Travassos, G. H. (2005) Sytematic review in software engineering. Relatório Técnico, RT-ES 679/05 System Engineering and Computer Science Dept., COOPE/UFRJ.

Chowriappa, A.; Subrahmaniyan, N.; Srimathveeravalli, G.; Bisantz, A.; Kesavadas, T.; , "Modeling and defining expert handwriting behavior," Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on , vol., no., pp.2918-2923, 11-14 Oct. 2009. Disponível em: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5346085&isnumber=5345886>. Acesso em: 10 jun. 2010.

Dawei, J; Asim B, Saeid N. (2009), "MUSTe method for quantifying virtual environment training system efficacy". OZCHI '09: Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24. November 2009.

Go Tani; Freudenheim, A. M; Meira Júnior, C, M; Corrêa, U. C. (2004) "Aprendizagem motora: tendências, perspectivas e aplicações. Ver. Paul. Educ. Fís., São Paulo, v.18, pg.55-72, Ago.

Kitchenham, B. (2004) "Procedures for Permorming Systematic Reviews", Joint Technical Report Software Engineering Group, Departament of Computer Sicence keele University, United king and Empirical Software Engineering, Nacional ICT Australia Ltd, Australia.

Kolesnikov, M.; Zefran, M.; Steinberg, A.D.; Bashook, P. G.;(2009) "PerioSim: Haptic virtual reality simulator for sensorimotor skill acquisition in dentistry," Robotics and Automation, 2009. ICRA'09. IEEE International Conference , vol., no., pp.689-694, 12-17 May 2009. Disponível em: <http://ieeexplore.ieee.org>. Acesso em: 02 de Out. 2010.

Madan, A K; Frantzides, C T. (2007), Prospective randomized controlled trial of laparoscopic trainers for basic laparoscopic skills acquisition. SURGICAL ENDOSCOPY AND OTHER INTERVENTIONAL TECHNIQUES 21 (2): 209-21.

Magill, R. (2000) "Aprendizagem Motora: conceitos e Aplicações". São Paulo: Edgard Blucher.

Panait, L., Akkary, E., Bell, R.L., Roberts, K.E., Dudrick, S.J., Duffy, A.J. (2009), "The Role of Haptic Feedback in Laparoscopic Simulation Training". Journal of Surgical Research, 156 (2), pp. 312-316. Disponível em: <http://www.scopus.com >. Acesso em: 15 mar. 2010.

Póvoas, M. B. C.; Rodrigues, F. M. (2007) "Habilidades Motoras e tipos de prática: Uma reflexão visando o aprimoramento e conscientização do movimento na prática

pianística". Disponível em: In: <http://www.ceart.udesc.br/revista_dapesquisa/volume3/numero1/musica/>. Acesso: 25 Nov. 2010.

Seymour, Neal E.; Gallagher, Anthony G.; Roman, Sanziana A.; O'Brien, Michael K.; Bansal, Vipin K.; Andersen, Dana K.; Satava, Richard M.; (2002) Annals of Surgery. Vol. 236, No. 4, 458 - 464

Tori, R.; Nunes, F. L. S.; Gomes, Victor, H. P., Tokunaga, D. M. VIDA: Atlas Anatômico 3D Interativo para Treinamento a Distância. In: Anais do XXIX Congresso da Sociedade Brasileira de Computação, (WIE 2009 XV Workshop Sobre Informática na Escola, Bento Gonçalves), Porto Alegre: Sociedade Brasileira de Computação, 2009. p. 1–10. Disponível em: <http://bibliotecadigital.sbc.org.br/?module=Public&action=SearchResult&author=476>. Acesso em: 20 Jul. 2010.

Tzafestas, C.S.; Palaiologou, N.; Alifragis, M. (2006) "Virtual and remote robotic laboratory: comparative experimental evaluation," Education, IEEE Transactions on, vol.49, no. 3, pp.360-369, Aug. Disponível em: <http://ieeexplore.ieee.org>. Acesso em: 25 jul. 2010.

Van Der Meijden, O A J; Schijven, MP. (2009), "The value of haptic feedback in conventional and robot-assisted minimal invasive surgery and virtual reality training: a current review". Surgical Endoscopy and Other Interventional Techniques 23 (6):1180-1190.

Watanabe, T.; Katsura, S. A support method for haptic skill acquisition using graph theory, Advanced Motion Control, 2010 11th IEEE International Workshop on , vol., no., pp.589-594, 21-24 March 2010.

IEEE XPLORE DIGITAL LIBRARY. Disponível em: <http://ieeexplore.ieee.org/xpl/periodicals.jsp>. Acesso em: 10 jun. 2010.

ACM DIGITAL LIBRARY. Disponível em: <www.acm.org/publications/journals>. Acesso em: 22 Jul. 2010.

SCOPUS. Base de Dados. Disponível. Em: <http://www.scopus.com/home.url>. Acesso em: 25 Jul. 2010.

ENGINEERING VILLAGE. Base de Dados. Disponível em: <http://www.ei.org/>. Acesso em: 22 Ago. 2010.

UNIVERSIDADE DE SÃO PAULO. Base de Dados. Disponível em: <www.iee.usp.br/biblioteca/basededados.htm>. Acesso em: 10 Jul. 2010.

ISI WEB OF KNOWLEDGE. Disponível em: <http://www.isiwebofknowledge.com/>. Acesso em: 20 Jul. 2010.

SBC BIBILIOTECA DIGITAL. Disponível em: <http://bibliotecadigital.sbc.org.br/>. Acesso em: 13 Out. 2010.

GOOGLE ACADÊMICO. Disponível em: <http://www..scholar.google.com.br>. Acesso em: 15 Ago. 2010.

CAPES. Banco de Dados da Capes. Disponível em: <http://www.periodicos.capes.gov.br>. Em: 05 Out. 2010.

Aplicação móvel como ferramenta de coleta e geração de dados da área epidemiológica

Henrique Dezani[1], Lucas Garrute do Carmo[1], Aline Cristina Y. Y.Chen[1], Carlos Magnus Carlson Filho[1], Luciene Cavalcanti Rodrigues[1], Mônica Regina Bocchi[1]

[1]Faculdade de Tecnologia de São José do Rio Preto (FATEC)
São José do Rio Preto – SP – Brasil

{dezani,prof.carlos,prof.luciene,monica}@fatecriopreto.edu.br,
lucas.garrute@gmail.com, linechen_06@hotmail.com

Abstract. *This paper describes a mobile application to collect data on outbreaks of dengue by Community Health Workers in order to accelerate the transmission of these data to the Environmental Health Surveillance, allowing them to take preventive and corrective actions in real time before a situation risk. For this work it was defined São José do Rio Preto as the city for the case study.*

Resumo. *Neste artigo é descrita uma aplicação móvel para a coleta de dados de focos de dengue pelos Agentes Comunitários de Saúde de maneira a acelerar a transmissão destes para a Vigilância Ambiental em Saúde, permitindo que sejam tomadas ações preventivas e corretivas em tempo real diante de uma situação de risco. Para este trabalho foi definida São José do Rio Preto como a cidade para o estudo de caso.*

1. Introdução

A Vigilância Ambiental em Saúde tem como alguns dos objetivos principais prevenir e controlar os fatores de riscos de doenças e de outros agravos à saúde, decorrentes do ambiente e das atividades produtivas, além de interferir com ações diretas para eliminar os principais fatores ambientais de riscos à saúde humana [Ministério da Saúde 2009].

No município de São José do Rio Preto, no estado de São Paulo, de acordo com o Sistema de Informação de Agravos de Notificação (SINAN), houve um grande número de casos de dengue notificados em 2010, conforme mostrado na Figura 1, cabendo à Vigilância Ambiental em Saúde juntamente com a Vigilância Epidemiológica tomar medidas como, por exemplo, controle do vetor e conscientização da população.

Uma das ações tomadas pela Secretaria de Saúde consiste em visitas realizadas pelos Agentes Comunitários de Saúde nos domicílios, conscientizando a população e procurando focos da dengue que, caso existam, são eliminados pelos agentes. Durante as visitas é preenchido manualmente um boletim, registrando a situação do imóvel, se há focos de dengue e o tipo de intervenção levada a cabo para eliminação dos criadouros.

Esse tipo de preenchimento gera atrasos no envio dos dados para a Vigilância Ambiental em Saúde, pois sua digitação é feita posteriormente. Além disso, o preenchimento manual contém problemas como caligrafias ilegíveis e rasuras, o que

acarreta inserção errônea dos dados no sistema. Todo o processo, desde o preenchimento do boletim até a análise dos dados digitalizados, dura 30 dias.

A agilidade na obtenção dos dados é importante, uma vez que serão armazenados e analisados de maneira a gerar informações relevantes para a tomada de decisões sobre ações preventivas.

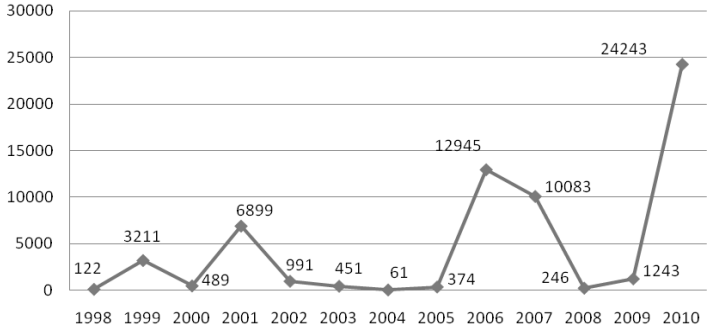


Figura 1. Casos de dengue confirmados na cidade de São José do Rio Preto. Fonte: adaptado de SINAN Web (2011)

Visto que o *Aedes aegypti*, mosquito transmissor da dengue, tem em torno de 45 dias de vida e nesse período pode transmitir a doença para 300 pessoas [Ministério da Saúde 2011], propõe-se neste projeto o desenvolvimento de uma aplicação móvel a ser utilizada pelos agentes para armazenar e transmitir as informações dos focos de dengue em tempo real, permitindo assim a tomada rápida de ações pelos órgãos responsáveis.

2. Desenvolvimento da aplicação

O sistema desenvolvido neste projeto consiste em uma aplicação móvel e em outra servidora. A aplicação móvel é responsável pela captura dos dados pelos Agentes Comunitários de Saúde e envio desses para a aplicação servidora, a qual recupera esses dados em tempo real e processa-os no sistema de informações geográficas.

2.1. Aplicação Móvel

A aplicação móvel foi desenvolvida utilizando-se a tecnologia J2ME (Java 2 Micro Edition), criada pela empresa Sun Microsystems com o intuito de permitir o desenvolvimento de aplicações para dispositivos consumidores com processamento, vídeo e memória limitados [Muchow 2004]. A escolha desta tecnologia é justificada devido à compatibilidade com uma grande quantidade de dispositivos móveis, diminuindo, desse modo, custos de implantação e manutenção.

Visto que esta aplicação será utilizada por diversos agentes e em diversos dispositivos, a facilidade para manuseá-la foi um dos principais requisitos levados em consideração para o desenvolvimento do projeto. Conforme pode ser observado na Figura 2(a), a busca do logradouro é facilitada para os teclados alfanuméricos, pois o usuário pode digitar os primeiros caracteres do nome da rua, por exemplo, e o sistema apresenta possíveis correspondentes que poderão ser selecionados. É importante ressaltar que os nomes definidos na Figura 2(a) são logradouros da cidade de São José do Rio Preto e estão armazenados na aplicação. Há também possibilidade de atualização dessa listagem por meio de um pedido de serviço à aplicação servidora.

Depois de selecionar o logradouro, o agente informará o número do estabelecimento e se esse foi trabalhado, ou seja, se foi vistoriado ou se estava fechado. Após término da vistoria, o sistema apresenta a tela contendo o logradouro selecionado anteriormente, tornando necessária somente a inserção do número do novo imóvel.

Se o estabelecimento for vistoriado, o agente selecionará os tipos de recipientes encontrados, informando a quantidade, quantos possuíam água e destes, quantos possuíam larvas. O agente poderá informar também a quantidade de amostras coletadas para a análise e se houve o controle mecânico. Posteriormente, o sistema conecta-se automaticamente com a aplicação servidora e envia os dados coletados, que serão armazenados e analisados pelo sistema de informações geográficas. Caso não haja conexão disponível no momento da conclusão do preenchimento, o sistema armazena os dados na memória do dispositivo para que possam ser enviados assim que a conexão com a Internet for restabelecida ou para que sejam gravados localmente no servidor, a partir de uma conexão USB ou Bluetooth.

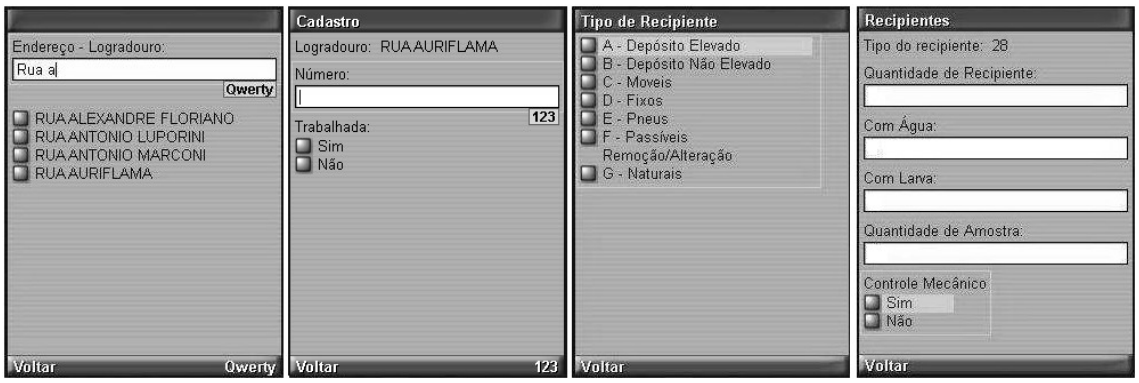


Figura 2. Principais telas da aplicação móvel: (a) Pesquisa de logradouro (b) Cadastro do endereço visitado (c) Escolha dos recipientes encontrados (d) Cadastro das análises dos recipientes

2.2. Aplicação servidora

A aplicação servidora consiste em uma aplicação *web* que foi desenvolvida utilizando a tecnologia J2EE (Java 2 Enterprise Edition) [Oracle 2011]. Esta aplicação contém vários serviços *web* que efetuam o recebimento dos dados advindos da aplicação móvel e armazenam-nos em uma base de dados, para que seja possível tratá-los por meio de um sistema de informações geográficas. Esta aplicação também fornece serviços de autenticação do agente e de envio de dados, tal como os logradouros que um agente deve vistoriar num determinado período.

3. Resultados Preliminares

Após o processo de transferência dos dados para o servidor e de alteração dos arquivos de texto, quando necessária, para o formato do sistema de informações geográficas, processou-se os dados utilizando o programa MapInfo Professional [Pitney 2011], instalado na prefeitura municipal de São José do Rio Preto. Este programa permite processar os dados de várias formas, conforme mostrado na Figura 3. Como exemplos, apresentam-se um mapa temático dos estabelecimentos pendentes (Figura 3-a), ou seja, que não foram possíveis vistoriar, e um mapa temático de interpolação (Figura 3-b),

onde é possível identificar as áreas de maior risco a partir das tonalidades de cinza exibidas no mapa, por exemplo, a área com 33 recipientes infectados possui a cor mais escura e as áreas com menor quantidade de recipientes possuem cores mais claras.

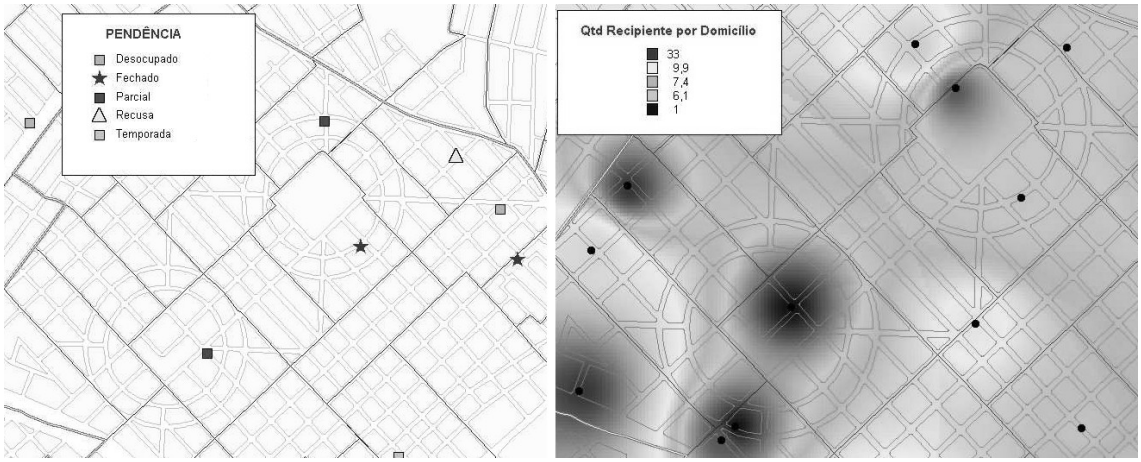


Figura 3. Mapas temáticos (a) Estabelecimentos Pendentes (b) Interpolação

4. Conclusão

A partir deste projeto inicial, os dados transmitidos imediatamente para uma base de dados única podem ser utilizados nos mais diversos sistemas de informações geográficas. A análise espacial dos focos do mosquito determina as ações de eliminação de criadouros imediatas e são fundamentais para interrupção da transmissão viral.

A coleta de informações em tempo real permite a tomada rápida de decisões como: transferência de agentes para áreas de maior risco, para eliminação de larvas do mosquito, bloqueio químico por meio de nebulização para eliminação do inseto adulto, com redução imediata da transmissão e alerta à população para buscar assistência médica, evitando casos graves. Para uma divulgação mais ampla, a distribuição dos dados em formato XML (eXtensible Markup Language) tem sido estudada.

Por utilizar uma tecnologia capaz de ser executada em diversos dispositivos móveis de baixo custo (preço médio de duzentos reais) e com requisitos mínimos de *hardware*, esta aplicação pode ser utilizada em diversas localidades do país.

Referências

Muchow, John W. (2004), Core J2ME: Tecnologia & MIDP. São Paulo: Pearson Makron Books, 2004.

Ministério da Saúde. (2009) Diretrizes Nacionais para a Prevenção e Controle de Epidemias de Dengue. Brasília – DF.

Ministério da Saúde. (2009) "Campanha de combate a dengue", http://www.combatadengue.com.br, Março de 2011.

Oracle (2011) "Java 2 Enteprise Edition (J2EE)", http://www.oracle.com/technetwork/java/javaee/overview, Janeiro de 2011.

Pitney Bowes Software Inc. (2011) "MapInfo Professional" http://www.pbinsight.com/welcome/mapinfo, Março de 2011.

Interação com Dados Médicos Usando Acelerômetros

Vinicius Rogério Araujo Silva[1], **Héber de Padua Sousa**[1], **Anselmo Cardoso de Paiva**[1], **Aristofanes Corrêa Silva**[1]

[1]Núcleo de Computação Aplicada – NCA UFMA – Universidade Federal do Maranhão (UFMA) – São Luís – MA – Brazil

{vinicius.ras, hebersousa}@gmail.com, {paiva, ari}@deinf.ufma.br

Abstract. *The visualization of volumetric data has important applications on medicine, such as diagnosing diseases and surgery planning. Apart from the possibility of visualizing volumes which were obtained from data which came from medical examinations, it is also possible to manipulate the visualized data by using techniques of volume sculpting, which allows for the isolation of interesting areas or for the removal of volume parts. This work presents an analysis of the use of accelerometers as interation devices which enable the manipulation, through volume sculpting techniques, of volumetric data which was obtained from medical exams.*

Resumo. *A visualização de dados volumétricos possui importantes aplicações na medicina, tais como o diagnóstico de doenças e o planejamento de cirurgias. Além da possibilidade de visualizar volumes obtidos através de dados de exames médicos, também é possível manipular os dados visualizados através de recortes que permitem isolar áreas de interesse ou remover partes do volume. Este trabalho apresenta uma análise do uso de acelerômetros como dispositivos de interação que permitem que um volume obtido a partir de dados volumétricos de exames médicos seja manipulado, através da aplicação de técnicas de recorte neste volume.*

1. Introdução

A visualização de dados volumétricos é atualmente uma área científica de grande importância e seus diversos aspectos vêm sendo pesquisados e aprimorados há alguns anos. No campo da medicina, podemos citar como exemplo a visualização de dados volumétricos para a análise de órgãos e estruturas internas do corpo humano, possibilitando, por exemplo, o diagnóstico de determinadas doenças ou o planejamento de cirurgias [Paiva, Seixas and Gattass 1999]. A obtenção destes dados pode ser feita de diversas maneiras, dentre as quais podemos destacar exames médicos como Ressonância Magnética e Tomografia Computadorizada.

Até recentemente, o uso de acelerômetros [Figueiredo et al 2007] estava condicionado a áreas bastante específicas e onde seu custo era irrelevante, como em sistemas militares e aeroespaciais. Porém, os custos vêm se tornando cada vez mais suportáveis, de forma que a utilização dos acelerômetros passou a ser menos limitada. Atualmente, podemos encontrar os mais variados exemplos de utilização, incluindo desde sistemas de ativação de cintos de segurança, até a utilização dos acelerômetros como dispositivos de entrada para jogos eletrônicos populares.

O objetivo deste trabalho é analizar a utilização de acelerômetros quando empregados para a visualização e manipulação de dados volumétricos, obtidos através de exames médicos, tais como a Ressonância Magnética e a Tomografia Computadorizada, de modo que na sequência seja possível desenvolver técnicas de segmentação interativa de dados médicos [Bornik, Beichel and Schmalstieg 2006] e simulação de procedimentos cirúrgicos.

2. Fundamentação teórica

Tradicionalmente, na medicina, imagens bidimensionais do corpo humano são utilizadas para auxiliar no diagnóstico clínico, no planejamento cirúrgico e no acompanhamento do tratamento de pacientes. Contudo, muitas vezes a compreensão destas imagens bidimensionais se restringe a profissionais experientes, capazes de analizar e interpretar estas imagens. A visualização volumétrica oferece uma forma mais intuitiva de observar estas imagens. Através das técnicas de visualização volumétrica, é possível gerar uma visualização tridimensional (Figura 1) a partir de um conjunto de várias imagens bidimensionais obtidas em um exame médico. Esta representação tridimensional dos dados do exame é comumente denominada ”volume”.

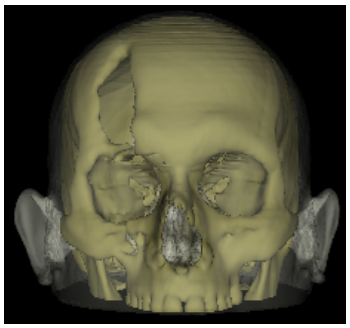


Figura 1. Volume gerado através de uma série de fatias.

Para certos tipos de aplicações, torna-se necessário explorar internamente o volume adquirido através dos dados volumétricos. Nestes casos, é necessário que o usuário que está visualizando os dados tenha a capacidade de manipular o volume, realizando operações para remover partes indesejadas do volume ou isolar áreas de interesse no mesmo. Para estas finalidades, existem as chamadas Técnicas de Escultura de Volumes ou Técnicas de Recorte [Wang and Kaufman 1995], que permitem que partes do volume visualizado sejam selecionadas pelo usuário para serem removidas ou isoladas.

Em geral, as Técnicas de Recorte são utilizadas na implementação das chamadas Ferramentas de Recorte [Huff 2006]. Nas aplicações que permitem a visualização e manipulação de volumes, estas ferramentas são oferecidas para o usuário e permitem que ele(a) especifique a maneira como quer recortar o volume. Existem diversas ferramentas neste sentido, e cada uma resulta em recortes diferentes no volume. No presente trabalho, foram utilizadas as ferramentas de recorte denominadas Borracha e Guilhotina. Cada uma foi implementada em duas versões: uma que se utiliza de dispositivos apontadores (como o mouse) e outra que se utiliza dos acelerômetros.

A ferramenta da Borracha permite que o usuário controle um cilindro que irá definir qual área do volume deverá ser excluida. As partes do volume que estiverem dentro

da região definida pelo cilindro serão removidas. Em sua implementação com dispositivos apontadores, o usuário controla, através do mouse, um circulo na tela, o qual representa o cilindro que será removido do volume, alinhado ao plano de visualização do usuário. Já na implementação com acelerômetros, o usuário deve movimentar um controle em suas mãos. Através de seus movimentos, o usuário controla um cilindro tridimensional no aplicativo, podendo rotacionar e posicionar este cilindro da forma desejada para o recorte. A Figura 2 ilustra o funcionamento da Borracha, em sua versão com acelerômetros, no aplicativo desenvolvido para este trabalho.

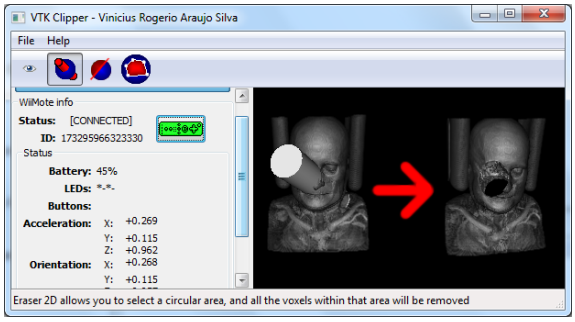


Figura 2. Ferramenta de recorte Borracha.

A ferramenta da Guilhotina permite que o usuário controle um plano. Após posicionar o plano em relação ao volume que deve ser recortado, o recorte é feito de forma que todos os pontos do volume que estejam acima do plano sejam removidos. Em sua implementação com dispositivos apontadores, o usuário deve clicar em dois pontos da tela, que serão utilizados para definir o plano de recorte. Já em sua implementação com acelerômetros, o usuário movimenta um controle com acelerômetro em suas mãos e controla o posicionamento e a rotação do plano de recorte, de maneira semelhante à forma como controla o cilindro de recorte na ferramenta da Borracha. A Figura 3 ilustra o funcionamento da Guilhotina, em sua versão com apontadores.

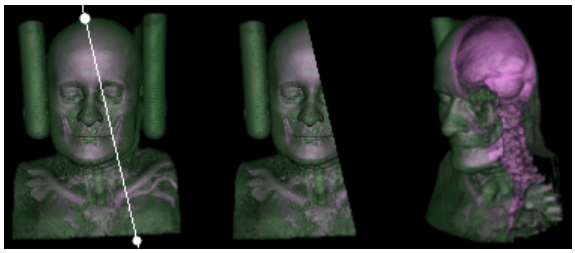


Figura 3. Ferramenta de recorte Guilhotina.

Uma característica importante de um aplicativo que permite a um usuário visualizar volumes e manipular seus dados através de Técnicas de Recorte é a forma como o aplicativo interage com o usuário. Os métodos usados pelo usuário para executar ações no aplicativo vão impactar diretamente na sua usabilidade, de forma que alguns métodos podem ser considerados mais intuitivos do que outros. Existem dispositivos de entrada considerados mais tradicionais para a maioria dos usuários, tais como o mouse e o teclado. Neste contexto, o uso de acelerômetros para permitir que o usuário execute técnicas de recorte sobre volumes é uma novidade – e objeto de estudo do presente trabalho. O uso

dos acelerômetros permite novas formas de interação do usuário com a aplicação, através da captação de certos movimentos, o que não era possível pelo uso de dispositivos mais convencionais.

3. A proposta

A proposta do presente trabalho é realizar uma investigação acerca de uma aplicação protótipo desenvolvida, que utiliza acelerômetros como dispositivos de entrada durante a visualização e manipulação de dados volumétricos. A aplicação é capaz de visualizar dados volumétricos obtidos a partir de exames médicos e manipular estes dados através do uso de dispositivos de entradas convencionais (mouse e teclado) e também através do uso de dispositivos de entrada com acelerômetros incorporados. Nesta aplicação, foram implementadas algumas ferramentas de recorte, cujas funcionalidades seriam acessíveis ao usuário através do uso dos dispositivos de entrada citados.

As ferramentas de recorte Guilhotina e Borracha foram implementadas com a utilização de dispositivos apontadores e de acelerômetros. O dispositivo utilizado para os testes foi um controle utilizado em jogos eletrônicos populares, chamado WiiMote, que se assemelha a um controle remoto e possui um acelerômetro incorporado.

4. Conclusão

Este trabalho apresentou uma introdução à visualização de dados volumétricos e a manipulação destes dados através de dispositivos de entrada baseados no uso de acelerômetros. Devido às suas características de captação de movimentos, o uso de acelerômetros garante um grau de controle e imersão ao usuário que não é possível com dispositivos de entrada mais tradicionais, tais como o mouse e o teclado. Pretende-se continuar trabalhando implementando o suporte a novas ferramentas de recorte de volumes, bem como experimentar novos tipos de dispositivos de interação, como o PSMove, que possui um acelerômetro mais preciso que o WiiMote, e o Kinect, o qual funciona através do reconhecimento de movimentos do usuário através de câmeras especiais. Estes novos dipositivos podem ser mais ou menos precisos que os dispositivos até agora usados, e podem levar a experiências diferenciadas de interação.

Referências

Bornik, A., Beichel, R., Schmalstieg, D. Interactive Editing of Segmented Volumetric Datasets in a Hybrid 2D/3D Virtual Environment. In: VRST 06: Proceedings of the ACM symposium on Virtual reality software and technology. (2006)

Figueiredo, L. J, Gafaniz, A. R., Lopes, G. S., Pereira, R. Aplicações de Acelerómetros. Monografia. Lisboa, Portugal, 2007.

Huff, R. Recorte Volumétrico Usando Técnicas de Interao 2D e 3D. Dissertação (Pós-Graduação em Computação). PPGC da UFRGS, Porto Alegre, 2006.

Paiva, A. C., Seixas, R. B., Gattass, M. Introdução à Visualização Cientifica. Monografia em Ciência da Computação, no 3/99, Departamento de Informática, PUC-Rio, 1999.

Wang, S.W. and Kaufman, A.E. Volume sculpting. In Proceedings of the Symposium on Interactive 3D Graphics, SI3D, pages 151-156, Monterey, CA, USA, 1995. New York, ACM Press.

Avaliação da Aquisição de Conhecimento em Ambientes Virtuais de Aprendizagem 3D para Treinamento Médico

Eunice P. dos Santos Nunes[1,2], Fátima L. S. Nunes[1,3], Romero Tori[1,3]

[1]Interlab – Laboratório de Tecnologias Interativas – Escola Politécnica/USP
Av. Prof. Luciano Gualberto, nº 380 – 05508-010 – São Paulo – fone (11) 3091-5282

[2]Instituto de Computação – Universidade Federal de Mato Grosso (UFMT)
Av. Fernando C. da Costa, nº 2367 – 78060-900 – Cuiabá-MT – fone: (65) 3615-8791

[3]LApIS - Laboratório de Aplicações de Informática em Saúde
Programa de Pós-graduação em Sistemas de Informação – Escola de Artes, Ciências de Humanidades/USP
Av. Arlindo Béttio, 1000 - Ermelino Matarazzo - São Paulo - CEP: 03828-000

{eunice.poli,fatima.nunes,tori}@usp.br

Abstract. *The development of tools supporting teaching, such as simulators and serious games through 3D Learning Virtual Environments (3D LVEs), is becoming an increasingly common alternative in the medical training area. In this context, this work presents the initial results from an experimental study conducted with students in the health field, interested in learning of Human Anatomy, taking VIDA (Virtual Interactive Distance-Learning on Anatomy) as a tool to support the teaching environment. These results are part of a larger context aiming at establishing a method to evaluate knowledge acquisition in 3D LVEs.*

Resumo. *O desenvolvimento de ferramentas de apoio ao ensino, como simuladores e jogos sérios por meio de Ambientes Virtuais de Aprendizagem tridimensionais (AVAs 3D) está se tornando uma alternativa cada vez mais comum na área de treinamento médico. Nesse contexto, este trabalho apresenta os primeiros resultados obtidos a partir de um estudo experimental realizado com estudantes da área de saúde, interessados no aprendizado de Anatomia Humana, tendo como ferramenta de apoio ao ensino o ambiente VIDA (Virtual and Interactive Distance-Learning on Anatomy). Os resultados desta pesquisa fazem parte de um contexto maior que visa estabelecer um método para avaliar a aquisição de conhecimento por meio de AVAs 3D.*

1. Introdução

O uso do computador e das tecnologias da informação e comunicação tem crescido em diversas áreas do conhecimento, especialmente na área médica. No âmbito educacional o desenvolvimento de ferramentas de apoio ao ensino, como simuladores e jogos sérios por meio de Ambientes Virtuais de Aprendizagem (AVAs), está se tornando uma alternativa cada vez mais comum na área de treinamento médico. Essas ferramentas buscam melhorar o desempenho dos estudantes em conteúdos difíceis de serem absorvidos apenas com aulas expositivas, que são geralmente associadas com textos e imagens de livros.

Além disso, o elevado número de estudantes que ingressam nos cursos da área de saúde, a falta de laboratórios físicos suficientes para atender a esta demanda e o alto custo dos modelos anatômicos sintéticos, têm motivado pesquisadores a produzirem materiais de *e-learning* que visam a simular modelos anatômicos e cenários clínicos da vida real, para fins de treinamento médico, educação ou avaliação.

Com o advento da *Web* 2.0, os AVAs evoluíram com os recursos proporcionados pelas tecnologias de Realidade Virtual (RV) e Realidade Aumentada (RA), surgindo os AVAs tridimensionais (3D), que estão sendo amplamente adotados em diversos domínios de conhecimento, destacando-se a área de educação médica.

Nunes *et al*. (2007) afirmam que: "A tecnologia de Realidade Virtual vem despertando grande interesse neste domínio, pois amplia as possibilidades de estudo e prática de variadas técnicas e procedimentos médicos".

Existem diversas definições conceituais para AVAs, sendo que no contexto deste trabalho AVA é um *software* baseado em recursos tecnológicos diversos com propósito de auxiliar o processo de ensino-aprendizagem, portanto, qualquer ambiente de Realidade Virtual e Realidade Aumentada utilizado com propósito de aprendizagem, pode ser considerado um AVA 3D.

Um exemplo de ambiente virtual 3D que tem sido empregado como AVA é o *Second Life*, que apesar de ter sido desenvolvido, inicialmente, para relacionar as diversas pessoas do mundo usando uma interface intuitiva e semelhante ao *The Sims* (uma série de jogos eletrônicos de simulação de vida), evoluiu e passou a ser amplamente adotado no meio acadêmico, profissional e social. Este ambiente tem sido empregado, por exemplo, em experiências ligadas à Biologia (como genética) e medicina, aproveitando o poder de simulação da ferramenta (VALENTE & MATTAR, 2007).

Por meio dos AVAs 3D, o processo de ensino-aprendizagem tem potencial para tornar-se mais ativo, dinâmico e personalizado. Porém, uma discussão que é delineada neste contexto se refere à contribuição que tais ambientes efetivamente oferecem para a aquisição de conhecimento do aprendiz e como é possível medir tal aprendizado.

Nesse sentido, o presente trabalho que faz parte de um contexto maior que visa estabelecer um método para avaliar a aquisição de conhecimento do aprendiz obtida por meio de AVAs 3D, tem como objetivo apresentar os primeiros resultados obtidos a partir de um estudo experimental realizado com estudantes da área de saúde, interessados no aprendizado de Anatomia Humana.

O Ambiente Virtual de Aprendizagem 3D utilizado neste experimento foi o ambiente VIDA (*Virtual and Interactive Distance-Learning on Anatomy*), desenvolvido pelo Interlab/USP[1], que consiste de uma ferramenta voltada para o ensino de Anatomia Humana e utiliza técnicas de estereoscopia baseada em anaglifos para produzir o efeito 3D (TORI *et al*., 2009).

O presente artigo está estruturado da seguinte forma: a seção 2 apresenta uma contextualização das tecnologias de Realidade Virtual e Realidade Aumentada aplicadas na área educacional, a seção 3 apresenta uma breve descrição do projeto VIDA e sua

[1] http://www.interlab.pcs.poli.usp.br/

contribuição para o ensino de Anatomia Humana, a seção 4 descreve a condução do estudo experimental, a seção 5 apresenta os resultados e discussões, e a seção 6 apresenta as conclusões deste trabalho.

2. Realidade Virtual e Aumentada na Educação

Muitos pesquisadores têm defendido que sistemas de Realidade Virtual e Realidade Aumentada são tendências na área de treinamento educacional. Isso porque os sistemas de RV e RA oferecem oportunidades para experiências imersivas e não imersivas, contextos e atividades realistas para o aprendizado experimental, simulações de treinamento, modelagem de cenários complexos, e a colaboração de múltiplos usuários interagindo entre si dentro do mesmo mundo virtual (SHERMAN & CRAIG, 2003; VALENTE & MATTAR, 2007).

Os sistemas de RV e RA permitem extrapolar os limites espaço-temporais, possibilitando, por exemplo, a visitação "virtual" a lugares que sejam muito pequenos para se explorar na vida real, ou lugares de extensão muito grande, difíceis de serem visualizados como um todo (VENDRUSCOLO *et al*., 2005).

Observa-se na literatura que aplicações de RV e RA são encontradas em larga escala em sistemas de treinamento virtual 3D nas mais diversas áreas de domínio, como Medicina, Engenharias, Indústria, Ciências e Matemática, entre outras.

Nas áreas da Indústria e da Engenharia encontram-se diversos sistemas de RV e RA voltados para treinamento virtual, como mostram os trabalhos de Watanuki & Kojima, (2007), Angelov & Styczynski, (2007) e Wenju & Guangyae, (2010).

Na área de Medicina, um exemplo de aplicação de sistemas de RV e RA envolvendo treinamento médico é o sistema "*Virtual Patients*" de Troconis *et al*. (2010) que apresenta uma simulação em 3D de cenários clínicos da vida real para fins de treinamento médico, educação ou avaliação. Este sistema foi desenvolvido no ambiente virtual *Second Life* e existem várias formas de usá-lo para simular cenários clínicos. Por exemplo, o aluno pode assumir diversos papéis no ambiente, como de médico, paciente etc. O estudante pode agir independentemente ou ser guiado por um instrutor e pode, ainda, interagir por meio de um cenário colaborativo.

Outro trabalho na área de educação médica é de Gutierrez *et al*. (2007) que investiga o efeito do grau de imersão sob a *performance* da aprendizagem dos estudantes de medicina em uma simulação de um cenário de acidente automobilístico, desenvolvido com tecnologia de Realidade Virtual, que tem como objeto de estudo um paciente virtual.

O trabalho de Soler *et al*. (2008) apresenta um conjunto de ferramentas para melhorar o procedimento de laparoscopia, que são utilizadas como planejamento cirúrgico (pré-operatório) e também como simuladores cirúrgicos.

Atividades de aprendizagem baseadas em jogos também têm sido alvo de pesquisas, principalmente com a conexão dos "jogos sérios" às tecnologias de RV e RA, aplicadas na área de educação médica. Esta categoria de jogos, que extrapola a ideia de ser apenas entretenimento, visa a simular situações práticas do dia-a-dia e tem como alvo a "geração jogador" que tem um estilo cognitivo caracterizado pela multitarefa enquanto aprendem, apresenta curto período de atenção durante a aprendizagem e uma abordagem exploratória e de descoberta (YUCHUL *et al*., 2007).

Considerando este cenário, pode-se afirmar que os sistemas de RV e RA são recursos bastante viáveis de serem aplicados em diferentes contextos educacionais, principalmente na área de treinamento médico.

No entanto, nota-se que apesar da comunidade científica mostrar interesse em desenvolver AVAs com uso de tecnologias interativas, poucos estudos têm medido os reais efeitos na aprendizagem. Dessa forma há uma lacuna a ser preenchida, ao se estudar a avaliação da aquisição de conhecimento do estudante por meio das AVAs 3D, a fim de estabelecer uma metodologia de avaliação do aprendizado

3. Projeto VIDA

O projeto VIDA (*Virtual and Interactive Distance-learning on Anatomy*) visa ao desenvolvimento de um Ambiente Virtual de Aprendizagem 3D, baseado em Atlas Anatômico Virtual, voltado para o ensino de Anatomia Humana. Tem o objetivo de suportar tanto interação com dispositivos comuns, como mouse e teclado, como também manipulação direta de objetos virtuais tridimensionais (TORI *et al*., 2009).

O sistema VIDA se baseia na visão estereoscópica para oferecer maior realismo na visualização das estruturas anatômicas tridimensionais. Os parâmetros da projeção estereoscópica são calibrados de forma que cada usuário tenha a sensação de estar visualizando a estrutura anatômica no ar, ou seja, no espaço físico entre o seu corpo e o monitor, produzindo um efeito de profundidade. A visualização estereoscópica pode ser gerada e visualizada por diversos dispositivos e técnicas, como CAVE, capacete HMD, óculos polarizados ou anaglifos (TORI, KIRNER & SISCOUTTO, 2006).

Neste experimento, a versão utilizada suporta apenas a interação com dispositivos comuns, como *mouse* e teclado e a forma de visualização estereoscópica foi o anaglifo, que não exige monitores ou projetores especiais, apenas um par de óculos descartável de duas cores com filtros coloridos (cada lente com uma cor diferente), produzindo o efeito 3D desejado. O uso mais comum de filtragem é utilizar o vermelho para o olho esquerdo e o ciano (uma combinação de azul e verde) para o olho direito.

O desenvolvimento de um Atlas Anatômico Virtual 3D vem de encontro às dificuldades dos alunos em memorizar a elevada quantidade de termos das estruturas anatômicas. Em entrevista com professores da área de Anatomia Humana, estes relatam que muitos alunos apresentam dificuldades para o aprendizado das estruturas anatômicas por motivos diversos, como o excesso de terminologias, o pequeno tamanho de algumas estruturas e outros fatores individuais, como falta de atenção e motivação. Adicionalmente, para alguns há ainda o receio de se deparar com os cadáveres humanos para dissecação.

Frente às dificuldades existentes, principalmente no que diz respeito à memorização, o objetivo do projeto VIDA é contribuir para o aprendizado de Anatomia, favorecendo a aquisição de conhecimento por meio da participação ativa do aluno no ambiente virtual.

O ambiente VIDA pode também ser empregado em uma fase de pré-estudo, antes de o aluno usar o laboratório físico, tornando as estruturas mais familiares, ao serem apresentadas fisicamente. Adicionalmente, pode-se, com isso reduzir horas de estudo em laboratórios físicos, diminuindo, em médio prazo, o custo do processo de ensino-aprendizagem nesta área.

4. Estudo Experimental

O estudo experimental foi realizado com 60 estudantes da área de saúde. Deste total, tem-se 37 alunos do primeiro período do curso de Medicina e 23 alunos do primeiro período do curso de Nutrição.

Os participantes do estudo experimental possuem um baixo nível de conhecimento em Anatomia Humana, pois são alunos que acabaram de iniciar o primeiro ano do curso e haviam frequentado poucas aulas de Anatomia até a realização do experimento. Os 60 alunos foram divididos igualmente em dois grupos: grupo de controle e grupo experimental. Salienta-se que a adesão aos experimentos realizados foi voluntária.

Neste estudo, dois experimentos idênticos foram conduzidos para memorização de partes da estrutura óssea da pelve feminina, diferenciando apenas uma variável (meio de estudo).

Para tal, o grupo de controle realizou o estudo de memorização por meio de uma imagem 2D convencional encontrada em livros de Anatomia, apresentada aos alunos de forma impressa em uma folha de papel, acompanhada de legendas nas partes a serem memorizadas, como ilustra a Figura 1. Por outro lado, o grupo experimental realizou o mesmo estudo de memorização, utilizando a mesma imagem, porém por meio de estruturas anatômicas virtuais disponíveis no ambiente VIDA. Além da visualização, o aluno teve a oportunidade de interagir diretamente com o objeto em estudo, executando operações de rotação, translação e *zoom*. As imagens 3D capturadas do sistema VIDA estão ilustradas nas Figuras 2 e 3.

Os dois grupos tiveram o mesmo tempo de estudo: cinco minutos para memorizar as dez legendas colocadas na imagem da estrutura óssea da pelve feminina. Após a fase de estudo, os grupos foram submetidos a uma mesma avaliação escrita, na qual deveriam escrever os nomes das partes indicadas pelas setas, com o objetivo de medir o nível de aprendizado das partes da estrutura pélvica feminina apresentada.

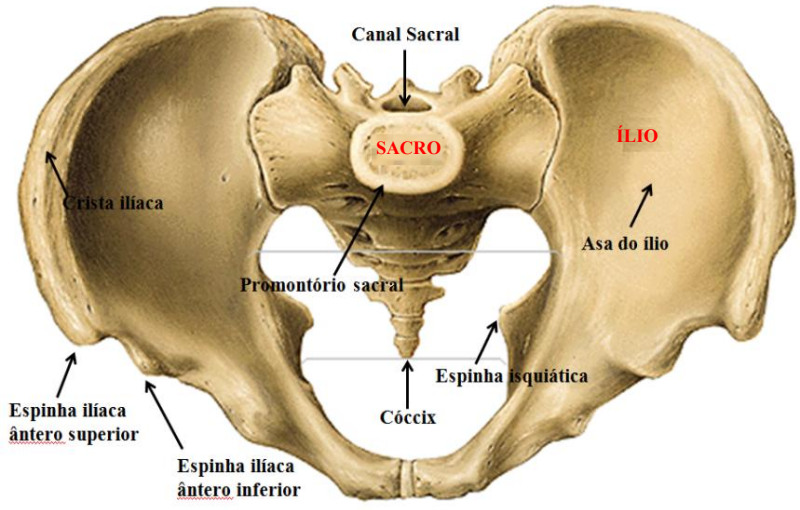


Figura 1. Pelve Feminina – imagem 2D adaptada (SOBOTTA, 2000)

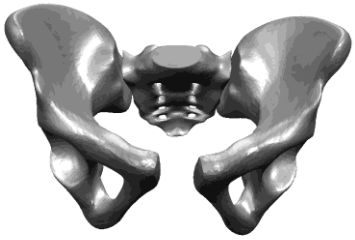


Figura 2. Pelve Feminina sem as legendas – parte óssea (imagem 3D – VIDA)

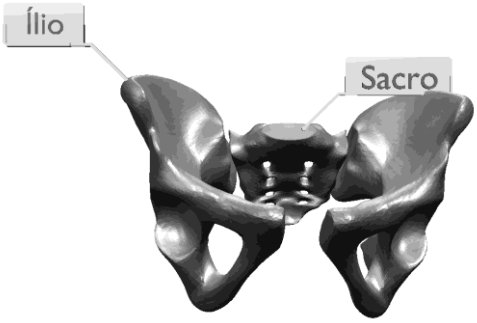


Figura 3. Pelve Feminina com parte das legendas (imagem 3D – VIDA)

5. Resultados e Discussões

Após os experimentos, as provas escritas foram corrigidas a fim de verificar se o meio de aprendizagem poderia exercer alguma influência nos resultados obtidos.

Verificou-se uma diferença significativa na média geral da prova escrita de Anatomia Humana - especificamente estrutura óssea da pelve feminina, ao se comparar os dois grupos.

O grupo de controle apresentou uma média de 81,25% de acertos, enquanto no grupo experimental a média foi de 95,25%, sendo possível observar que a média das notas da prova do grupo experimental foi superior à média das notas do grupo de controle, como ilustra a Figura 4.

Ao observar a diferença entre as médias dos dois grupos, pode-se supor que o estudo realizado apenas no papel com imagens similares àquelas encontradas nos livros de Anatomia, nas quais o aluno apenas vê e aplica dicas mnemônicas para facilitar a memorização das estruturas anatômicas não é tão eficiente se comparada ao estudo realizado no computador, por meio de um AVA 3D, no qual o aluno vê, interage e executa.

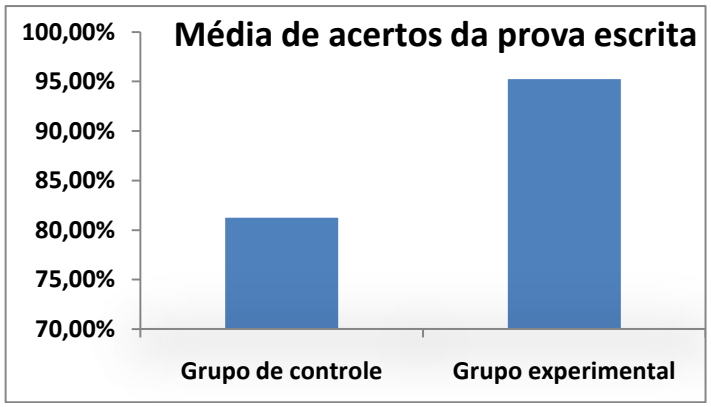


Figura 4. Média de acertos da prova escrita

Entretanto, apesar da diferença nas médias entre os dois grupos, é difícil medir com precisão o nível de aquisição de conhecimento do aprendiz por meio de AVAs 3D,

já que não há na literatura o estabelecimento de uma metodologia para executar a avaliação de forma autônoma.

Dessa forma, Nunes *et al*. (2010) definiram um conjunto de parâmetros a fim de propor um método para avaliar de forma autônoma a aquisição de conhecimento por meio dos AVAs 3D, e neste estudo experimental foram considerados, inicialmente, alguns desses parâmetros que são apresentados na Tabela 1.

Tabela 1. Parâmetros selecionados para avaliar o conhecimento adquirido (NUNES *et al*., 2010)

Parâmetros do usuário	Descrição	Referências
Tempo para completar a tarefa	Tempo estimado para completar a tarefa.	(Troconis *et al*. 2010)
Ativo ou passivo	Capturar a intensidade de participação do usuário durante o treinamento.	(Ayala 2010), (Troconis *et al*. 2010)
Diferenças individuais	Devem ser identificadas para correlacionar com as características e habilidades individuais	(Lee 2007)

Em relação ao parâmetro - tempo para completar a tarefa, o especialista da área estimou cinco minutos, sendo que neste estudo experimental a maior parte dos participantes não usou todo o tempo estimado e mesmo assim obtiveram resultados melhores na prova escrita. Dessa forma, este é um parâmetro importante a ser avaliado pelo especialista do domínio, para não se tornar excessivo e nem insuficiente e, também deve estar alinhado a outros parâmetros, como por exemplo, ao parâmetro diferenças individuais.

Quanto ao parâmetro - ativo ou passivo, pode-se afirmar que os participantes foram muito ativos no ambiente, pois apresentaram um número de interações consideráveis, incluindo translações, rotações e *zoom*. Portanto, a interação com o objeto em estudo pode ter sido um fator motivador que contribuiu para o melhor aprendizado do grupo experimental.

Quanto ao parâmetro - diferenças individuais, nota-se que os participantes apresentaram características e habilidades semelhantes, como pode ser observado na Tabela 2.

Salienta-se que apesar de cada parâmetro apresentar a sua contribuição para a avaliação do aprendizado, só é possível diferenciar o desempenho entre os participantes, analisando o conjunto de parâmetros como um todo, pois aliada a estes parâmetros, têm-se uma questão importante a salientar, o fato da forma de aprendizagem (aquisição de conhecimento) depender de cada pessoa, já que algumas aprendem visualmente, outras verbalmente, algumas pessoas preferem explorar e outras deduzir.

Após aplicação da prova escrita, os alunos do grupo experimental responderam a um questionário *on-line*, contendo dez perguntas, elaborado seguindo a escala Likert[2]. A Tabela 2 sintetiza os resultados obtidos com a aplicação do questionário.

[2] É um tipo de escala de resposta psicométrica usada comumente em questionários, principalmente em pesquisas de opinião. Ao responderem a um questionário baseado nesta escala, os perguntados especificam seu nível de concordância com uma afirmação (http://pt.wikipedia.org/wiki/Escala_Likert).

Tabela 2. Avaliação geral do questionário

Itens avaliados	1	2	3	4	5
1. Nível de experiência em RV	17%	23%	40%	20%	0%
2. Nível de conhecimento em anatomia humana	3%	67%	23%	7%	0%
3. Nível de conhecimento da pelve feminina	3%	57%	27%	13%	0%
4. Sensação do efeito 3D (o objeto se projetou à frente da tela?)	0%	0%	3%	0%	97%
5. Os recursos de interação proporcionaram a sensação de estar manipulando o esqueleto pélvico feminino	0%	0%	7%	40%	53%
6. O AV retratou com fidelidade a experiência do mundo real	0%	0%	20%	53%	27%
7. Os recursos de interação foram suficientes para o aprendizado	0%	0%	0%	20%	80%
8. Legendas bem posicionadas favorecendo a memorização	0%	7%	17%	67%	10%
9. O AVA 3D foi motivador para os estudos de Anatomia Humana	0%	0%	0%	10%	90%
10. O AVA 3D proposto é útil para uma fase de pré-estudo antes de usar o laboratório físico	0%	0%	0%	10%	90%

Os resultados do questionário mostraram que o nível de experiência dos participantes em sistemas de Realidade Virtual foi mediano, isto se justifica pelo fato dos participantes serem da "geração jogador". Já o nível de conhecimento de Anatomia Humana foi considerado baixo, confirmado pelo fato dos alunos estarem iniciando o 1º. ano do curso e terem frequentado poucas aulas de Anatomia até a data do experimento.

Em relação às funcionalidades do sistema, como os recursos de interação e o *design* dos modelos anatômicos do objeto em estudo, pode-se afirmar que foram satisfatórias. Quanto ao posicionamento das legendas, alguns participantes não se sentiram confortáveis, sugerindo alterações, como por exemplo, o ambiente não exibir todas as legendas do objeto de uma única vez no *display*, isto é, separá-las em camadas, e conforme a interação do usuário com uma parte da estrutura, a legenda poderia aparecer ou desaparecer do *display*.

Vale ressaltar que alguns participantes não conseguiram obter de forma plena, a sensação do objeto se projetando à frente da tela, isto pode ter ocorrido pelo fato de portadores de daltonismo terem dificuldade de visualização do efeito 3D, quando se utiliza a técnica do anaglifo para visualização estereoscópica.

6. Conclusões

Analisando os resultados obtidos com este trabalho, nota-se que várias ferramentas de apoio ao ensino, como simuladores e jogos sérios por meio de Ambientes Virtuais de Aprendizagem 3D, têm se tornado uma alternativa cada vez mais comum na área de treinamento médico, mas nenhum dos estudos selecionados apresenta um método para avaliar de forma autônoma a aquisição de conhecimento do aprendiz.

Dessa forma há uma lacuna a ser preenchida, ao se estudar a avaliação da aquisição de conhecimento em AVAs que usam tecnologias emergentes, como RV e RA e, que ainda não têm estabelecidas diretrizes de avaliação ou modelos cognitivos.

A partir dos resultados obtidos neste estudo experimental pode-se afirmar que os AVAs 3D propiciam maior motivação para o aprendizado na área de educação médica. Porém, não basta apenas ter motivação, uma questão importante a ressaltar é o quanto um AVA 3D pode contribuir para o ensino e como é possível avaliar o conhecimento adquirido.

Salienta-se que vários outros fatores, como a novidade em usar o ambiente tridimensional, a excitação de estar participando de um experimento e a curiosidade, possam ter influenciado positivamente na atenção e motivação dos participantes. Portanto, futuros experimentos de mais longo prazo, em que os efeitos "novidade" e "curiosidade" possam ser minimizados poderão dar a exata medida da diferença entre os dois meios de estudo (convencional e AVA 3D).

Agradecimentos

Os autores agradecem ao Instituto Nacional de Ciência e Tecnologia - Medicina Assistida por Computação Científica (INCT-MACC), pelo apoio financeiro (Processo 573710/2008-2 Edital MCT/CNPq Nº 015/2008 - Institutos Nacionais de Ciência e Tecnologia), à Fapesp pelo apoio financeiro, à Profª Célia Regina Maganha e Melo da EACH-USP, ao Prof. Flávio Silva Tampelini da Faculdade de Medicina da UFMT e ao doutorando Cleber Gimenez Corrêa da Poli-USP.

Referências

Angelov, A. N. & Z. A. Styczynski. Computer-aided 3D Virtual Training in Power System Education. In *Power Engineering Society General Meeting, IEEE*, 1-4. Proc. Power Engineering Society General Meeting, IEEE, 2007.

Ayala, A. P. Acquisition, representation and management of user knowledge. *Expert Systems with Applications - Elsevier,* 37**,** 2255-2264, 2010.

Gutierrez, F., J. Pierce, V. M. Vergara, R. Coulter, L. Saland, T. P. Caudell, T. E. Goldsmith & D. C. Alverson. The Effect of Degree of Immersion upon Learning Performance in Virtual Reality Simulations for Medical Education. *Medicine Meets Virtual Reality 15,* 125**,** 155-160, 2007.

Lee, J.The effects of visual metaphor and cognitive style for mental modeling in a hypermedia-based environment. *Interact. Comput.,* 19**,** 614-629, 2007.

Nunes, E. P. S., F. L. S. Nunes, R. Tori & C. Nunes. 2010. Definindo Parâmetros para Avaliação da Aquisição de Conhecimento em Ambientes Virtuais 3D. In *III Workshop sobre Avaliação e Acompanhamento da Aprendizagem em Ambientes Virtuais*. João Pessoa, PB: XXI Simpósio Brasileiro de Informática na Educação - SBIE, 2010.

Nunes, F. L. S., R. M. E. M. Costa, A. C. M. T. G. Oliveira, S. R. Delfino, L. Pavarini, I. A. Rodello, J. R. F. Brega & A. C. Sementille. Aplicações Médicas usando Realidade Virtual e Realidade Aumentada. In *Realidade Virtual e Aumentada - Conceitos, Projeto e Aplicações,* eds. C. Kirner & R. Siscoutto. Petrópolis-RJ: SBC - Sociedade Brasileira de Computação, Porto Alegre, 2007.

Preece, J. *Design de Interação: além da interação homem-computador*. Porto Alegre: Bookman, 2005.

Sherman, W. R. & A. B. Craig. *Understanding Virtual Reality: interface, application, and design*. San Francisco, CA: Morgan Kaufmann Publishers, 2003.

Sobotta, J. *Atlas da Anatomia Humana*. Rio de Janeiro: Guanabara Koogan, 2000.

Soler, L., S. Nicolau, J. Fasquel, V. Agnus, A. Charnoz, A. Hostettler, J. Moreau, C. Forest, D. Mutter & J. Marescaux. Virtual Reality and Augmented Reality Applied to Laparoscopic and Notes Procedures. In *5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro - ISBI 2008*, 1399-1402. Proc. (ISBI 2008).

Tori, R., C. Kirner & R. Siscoutto. *Fundamentos e Tecnologia de Realidade Virtual e Aumentada*. Belém, PA: VIII Symposium on Virtual Reality, 2006.

Tori, R., F. L. S. Nunes, V. H. P. Gomes & D. M. Tokunaga. VIDA: Atlas Anatômico 3D Interativo para Treinamento a Distância. In *XXIX Congresso da Sociedade Brasileira de Computação*, 1-10. Bento Gonçalves, Porto Alegre: SBC, 2009.

Troconis, M.-T., K. Meeran, J. Higham, U. Millstrom & M. Partridge. Design and Delivery of Game-Based Learning for Virtual Patients in Second Life: Initial Findings. In *Researching Learning in Virtual Worlds,* Springer-London, 111-138, 2010.

Valente, C. & J. Mattar. *Second Life e WEB 2.0 na Educação: o potencial revolucionário das novas tecnologias*. São Paulo: Novatec, 2007.

Vendruscolo, F., J. A. Dias, G. Bernardi & M. Cassal. Escola TRI-Legal - Um Ambiente Virtual como Ferramenta de Apoio ao Ensino Fundamental Através de Jogos Educacionais. In *Colabor@ - Revista Digital da CVA - RICESU, 2005*.

Watanuki, K. & K. Kojima. Knowledge Acquisition and Job Training for Advanced Technical Skills Using Immersive Virtual Environment. *Journal of Advanced Mechanical Design Systems and Manufacturing,* 1**,** 48-57, 2007.

Wenju, W. & L. Guangyae. Virtual Reality in the Substation Training Simulator. In *Computer Supported Cooperative Work in Design (CSCWD)*, 438-443. Shanghai, China, 14th Conference International on, 2010.

Yuchul, J., P. Hogun, C. Yoonjung & M. Sung-Hyon. Designing a Cognitive Case-Based Planning Framework for Home Service Robots. In *16th IEEE International Conference on Robot & Human Interactive*, Jeju, Korea, 827-832, 2007.

A web-based application to help on assessment of respiratory system

Fabrício Ferracioli[1], Maria Angélica de O. Camargo-Brunetto[1]

[1]Departamento de Computação – Universidade Estadual de Londrina (UEL)
Caixa Postal 6001 – 86.051-990 – Londrina -PR – Brazil

fabricioferracioli@gmail.com , angelica@uel.br

Abstract. *We describe a Web-based application to help health care professionals on treatment of patients with chronic respiratory disease. The software addresses clinical issues and research/teaching ones. At the present the software provides an Electronic Patient Record (EPR) that allows integrating different exams used to assess respiratory conditions of a subject. It is also possible to retrieve the history of assessments, with graphic view for each kind of exam. Recent contributions of the software includes retrieving information using dynamic filters and authoring of medical knowledge, by means of XML documents that add metadata to exams. The information retrieving module helps in research issues, allowing identification of groups of patients that satisfy conditions defined by the user. The medical knowledge authoring module has contributed to improve the process of software development, increasing the integration between health care expert, knowledge engineering and software designer.*

1. Introduction

Health care professionals at university hospital must be concerned to clinical issues as well as teaching and scientific ones. Frequently, clinical data can be a rich source of study and should be structured, in such way that it can be easily retrieved. Even thought there are many alternatives of open source Electronic Patient Record (EPR), it is common that many health care units[1] have difficulties to keep integrated the clinical data and results from a lot of exams, as well as to keep a temporal record of the patient's assessments performed. At scientific and teaching issues, it is desirable for the teacher/researcher not only a fast way to retrieve information, but also to get an explanation about each exam, including: how it is performed, what variables are collected, what variables are computed, details about computation, how to interpret results, and so on. Surely, clinical and/or researchers have methods to reach their tags, but it is observed that naive procedures are commonly performed. A typical example is the generation of a lot of worksheets that can be easily updated, and do not require the definition of a structure to organize the data. This approach is a great source of human errors.

[1] Along the text, health unit means one of the following: a hospital, a clinic, a laboratory, a research group.

In our university, there is a health care unit responsible for the physical rehabilitation of chronic respiratory diseases patients Chronic diseases in general require continuous attention to control the occurrence of crisis. Aiming to help health care professionals (mainly in cardiac and respiratory rehabilitation) to accompany their patients individually as well as to study groups of patients, according to specific features, it was designed, developed and implemented the software SACAR-Web (Software for Assessment of Cardiac and Respiratory System for WEB). The assessment can be made through several kinds of exams, which have specific material and methods for both, execution techniques and analysis of results.

One of the great challenges of the proposed software is to construct a large database with shared clinical data from several health care units. The idea of sharing database of clinical data has emerged from the need to get a large set of cases of study for both issues (teaching and research). Such approach may be a fast way to reach the target, instead of considering the data of only one health care unit. It was though in a collaborative work, where each health care unit has a local administrator that is responsible for the data of his/her patients, with full access (reading/writing) on them, allowing (or no) to share his/her patients data with local administrators of other health care units (with permissions of data reading but with blinded names of patients). However, when the first prototype was ready, it was observed that some health care researches had a certain resistance on sharing all the data. An alternative approach proposed by the potential users was the creation of research groups that could share some kind of clinical data, but only for specific studies. That alternative approach is in progress, but facing this problem we decide to give priority on working hardly in adding some features to the application in such way that the health care professionals feel more trustful. Such decision has implied in review the requirements established in the software design. The details are described in section 2.

Some features of the web-based application had already presented in early paper [Camargo-Brunetto, 2005]. So, here we briefly present such features and outline the main additional modules developed at present, which includes: (i) relevant information retrieving using dynamic filters on the variables of interest of the health care professional; (ii) module of authoring medical knowledge that provides explanation about procedures, variables and calculus involved in each exam. In previous version of Sacar-Web application, information about the exams was available as static content including tables or explaining texts. In fact, such explanations are the metadata of the exams. So, it was observed the need to organize such metadata and to isolate this from the program code. The use of XML Schema was essential to attend this requirement. The remaining of the article is organized in the following sections: Section 2 presents a system overview, information retrieving by means dynamic filters is presented in section 3. Section 4 presents the methodology employed to add metadata on the exams. In section 5 are presented related works. Section 6 concluding remarks and future works.

2. System Overview

In this section we describe some design considerations and present the architecture of the system with a brief description of its modules.

2.1. Design Considerations

During the design of the application, it has been identified some user's needs, that given support to establish the software requirements:

(i) The software can be used from different locations, and should be easy to learn how to use it.

(ii) The software should attend different interests of health care professionals, including clinical and/or teaching/researching ones.

(iii) The software provides a large variety of exams, but should allow that health care professionals select only ones matching with their interests.

(iv) The software should provide information about the exams, including how the results have been reached, how the exam has been executed. It is important that health care professionals know how the software provides the results of the exams.

(v) Different health care unit databases can be shared for teaching/research issues.

(vi) The knowledge domain must be extensible (today it addresses cardiac and respiratory systems, but other systems are associated, as muscular system, nervous system, and others).

Based on these requirements, we have established the following decisions:
For the requirement (i), the simple way to provide wide access to the application is through a web-based application. We had decided to develop the software based on open source languages and standards. For the requirement (ii), to attend different issues (clinical and teaching/research), the most appropriate strategy was to define different levels of users with specific roles and permissions that are explained in the following:

The system has three kinds of users: **local administrator**, **collaborator** and **knowledge expert**. The **local administrator** is the person responsible for a health care unit (normally an experienced health care professional, including teachers and researchers) and has the following permissions: (i) to customize the system for his/her unit care which includes: to select the exams of his/her interest and to select the idiom to be used; (ii) full access to the Electronic Patient Record (EPR); (iii) to perform information retrieving using dynamical filters; (iv) to see the evaluation of a specific patient according to graphic view of different exams and (v) to register his/her collaborators that will take the role of filling the EPR. The **Collaborator user** has permission to use only the Patient Assessment module (to register patients, introduce or view exams) and must report his actions to the local administrator. The **knowledge expert** has the role to add information about each exam, including details on execution procedure, variables considered, interpretation rules of the results, bibliography and so on. Such activities are related to the requirement (iv).
For the requirement (iii), it was established a user profile, so that the web application provides an adapted interface. The requirement (iv) was established after some observations: several biomedical software systems are embedded in specific devices that deliver a large variety of graphics, table results, provides a set of standards of normal values, complex analysis, and so on; it is possible that one health care unit has different devices and each one has its own software; the user does not know how the result

provided by the software device was obtained; also it is possible that the health care unit does not have some special kind of device (for example, a gas analyzer, that is expensive), but uses exams that provides indirect measures. Such facts have been motivated to include in our software additional information about the exams, including procedures of execution, as well as possible interpretation of the results. For requirement (v), the strategy is to work with shared databases, and different visions of the union of them, according to the user profile. Confidential information, as the patient name for example is not shared. It is essential that requirement (iv) be fully reached.

The requirement (vi), domain extensibility will be attended by a work in progress, by means XML structured documents describing the different systems (cardiac, respiratory, muscular, nervous and so on), with their structure and assessment techniques possible.

2.2 Architecture of the System

Figure 1 shows the architecture of the application.

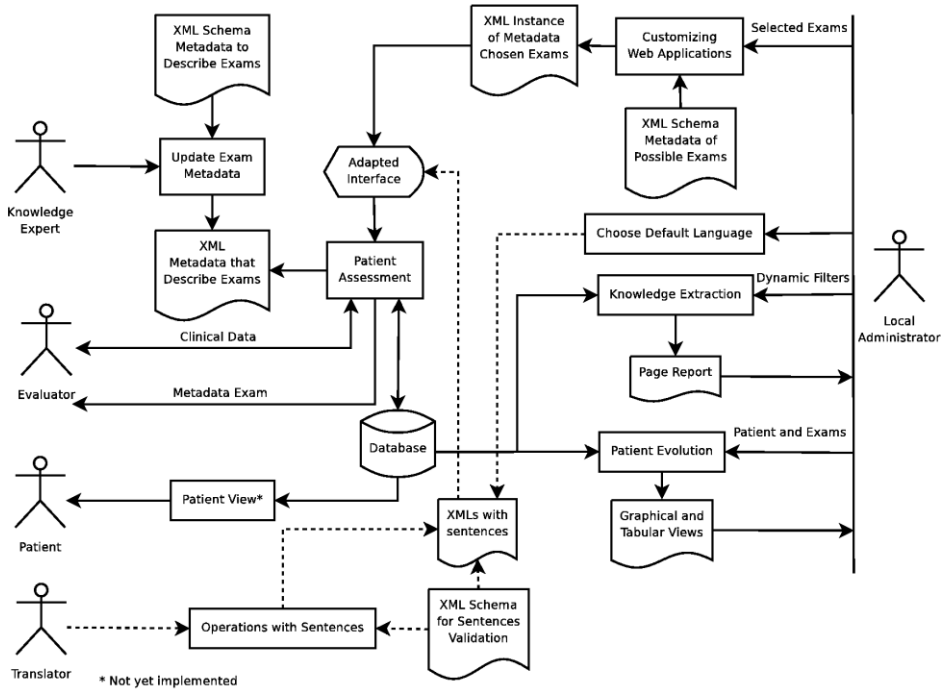


Figure 1: Sacar-Web system architecture

By means the module **Patient Assessment** it is possible to perform data collecting and analysis of patient exams. Each assessment is represented by a set of exams. The exams available are grouped by category: **Questionnaires to assess quality of life:** Chronic Respiratory Disease Questionnaire (CRQ) [Guyatt, 1987], adapted to brazilian population, and St. George's Respiratory Questionnaire Paul Jones (SGRQ) [Jones, 1991]; **Tests of pulmonary function:** The system provides the use of different equations of normal values, including Brazilian standard [SBPT, 1996], American standard due to [Knudson, 1976] and European standard by the Official Statement of the European Respiratory Society [ERS, 1993]; **Exams to test exercise capacity:** Shuttle Walk Test [Payne, 1996], the 6 Minutes Walking Test [Troosters, 1999] and Test of

Cardiac and Pulmonary Effort (TCPE) [Casaburi, 2005]. For TCPE, the system accepts the user attaches files containing graphics generated by other software developed in our laboratory that analyses physiologic signs. This software runs on client-side using Scilab (open source scientific software) and generates a set of 9 graphics grouped and aligned in a 3 x 3 matrix of graphics.

The module **customizing the web application** allows that the local administrator selects what exams he/she intends to use. The possible exams that the system provides are represented by a XML of the possible exams. When the local administrator selects his/her preferences, the system generates a XML document with an instance of exams of that health care unit.

2.3 Hardware and Software specifications

Sacar-Web is hosted on the Web-server Apache 2-2.2.4 [Apache, 2009] and has been developed using PHP language (5.2.1) with coding standard. The code documentation is made using the PHPDoc. Interfaces have been designed using CSS (Cascading Style Sheet) and Javascript for the user interaction with interface. For managing database it has been used MySQL 5.0.27 [Oracle, 2010] with tables InnoDB, that provides safe storage. XML documents were used to store the structure of an assessment that is a set of specific exams). XML [Bray, 2008] was also used do organize metadata of the exams. This approach helps to isolate content, structure and presentation, giving more flexibility, simplicity and easy reading. Furthermore such language allows include metadata that can be useful to aggregate semantic to the content of the document. To define the structure of XML documents it has been used XML Schema. In order to establish an interface between the computer program in PHP and XML documents it has been used the API (Application Programming Interface) DOM (Document Object Model).

3. Information retrieving by applying dynamic filters

An important feature of the proposed web application is the possibility to extract groups of subjects that satisfy different conditions, according to the interest of the health care professional. There are some variables that are commonly used to filter data, as gender and age range. Besides these static filters, it was introduced dynamic filters that the user (local administrator) defines at execution time. A filter consists of a set of rules. Each rule includes: to select an exam, to select a variable of this exam and to select a relational operator specifying the range of values of select variable. The user can define so many rules so desire. This procedure will help the researcher to discover interesting features about different groups of patients.

3.1 Application of dynamic filters example

Suppose the health care professional needs to identify subjects with limited capacity of exercise. A subset of subjects can be searched analyzing variables from different exams. For example, the variable distance achieved in the six-minutes walking test and the volume of used up oxygen estimated by the Shuttle-walking test.

One instance of such search could be the following: Select gender female, age between 30 and 68, total distance achieved in 6 minutes-walking test greater than 550 meters, and estimative of Volume of used up Oxygen given by the Shuttle Walk Test greater than 13. In this case we have a rule defined by two conditions that must be satisfied simultaneously (by using *and* operator). Figure 2 shows a screen shot of the example of application of dynamic filter. Each line of the filter composes a part of the SQL query that will be submitted to database server.

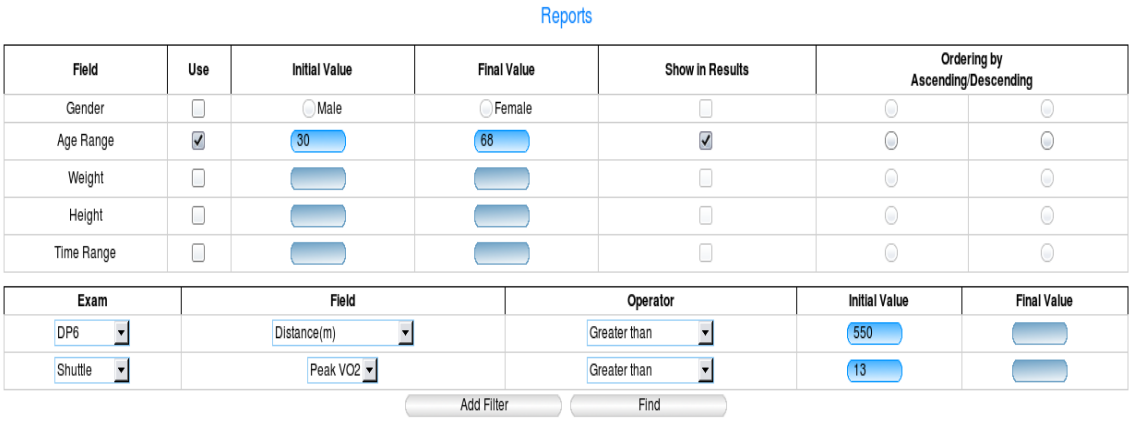
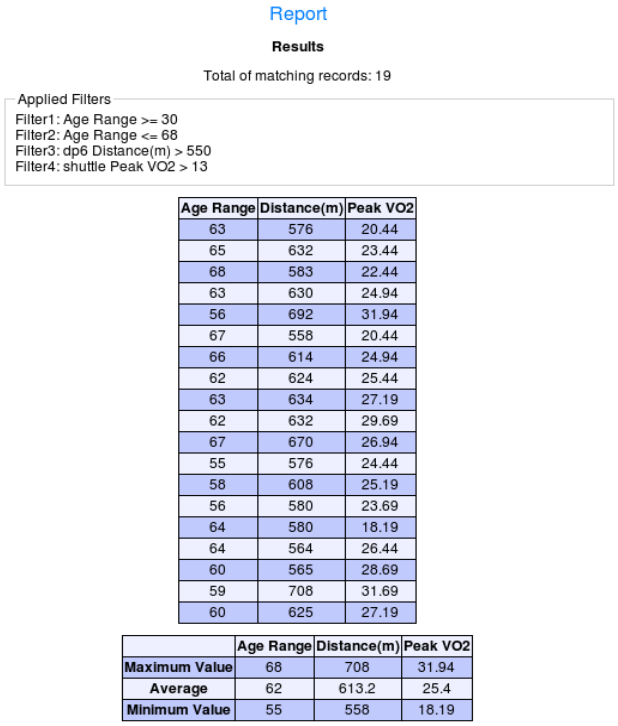


Figure 2: Applying dynamic filters for information retrieving

The result of applying the filter is browsed by a report page that displays the results in a tabular format, where each row is a subject and columns include the values of the variables used in the filter. Figure 3 shows the report page of the example above. In this information retrieving, the data of all health care units are searched. At the moment we are working with two health care units.



Age Range	Distance(m)	Peak VO2
63	576	20.44
65	632	23.44
68	583	22.44
63	630	24.94
56	692	31.94
67	558	20.44
66	614	24.94
62	624	25.44
63	634	27.19
62	632	29.69
67	670	26.94
55	576	24.44
58	608	25.19
56	580	23.69
64	580	18.19
64	564	26.44
60	565	28.69
59	708	31.69
60	625	27.19

	Age Range	Distance(m)	Peak VO2
Maximum Value	68	708	31.94
Average	62	613.2	25.4
Minimum Value	55	558	18.19

Figure 3 – Result of applying dynamic filters

Metadata of exams using XML

As argued early, it is important that the user of the system know how each exam is considered in the software. Each exam has a set of information available to the users, in order to give clarity about the main features of each exam. They include: (i) how the exam is performed (protocol), (ii) Identification of the input variables and calculated variables, including the mathematical expressions used to obtain them, (iii) Unit of measure used in each input variable, (iv) domain of valid values of input variables, (v) mathematical expression used to give the interpretation of the result, (vi) the standard of normality used (if it is the case), (vii) the bibliography that support the information and procedures given. The first idea was to delivery metadata of the exams, but in a very naive way. This idea has emerged from the need that new users (local administrators) had to be sure about the protocols of the exams. If the health care professional knows about all the conditions on what each exam is performed, he/she can be sure about the suitability of the web application for him/her. Another point to be outlined is the fact that students can take advantage, once the system presents readily the details of each exam. It is common that health care professionals use devices that normally have embedded software with a lot of results, but most of times, the user does not know how the results were obtained. Figure 4 shows a screenshot of a window with the metadata of DP6 exam. Firstly, such information was provided by means static pages. In the following, it was designed XML document, but the content was still filled by the software designer, based on the information delivered by the knowledge expert.

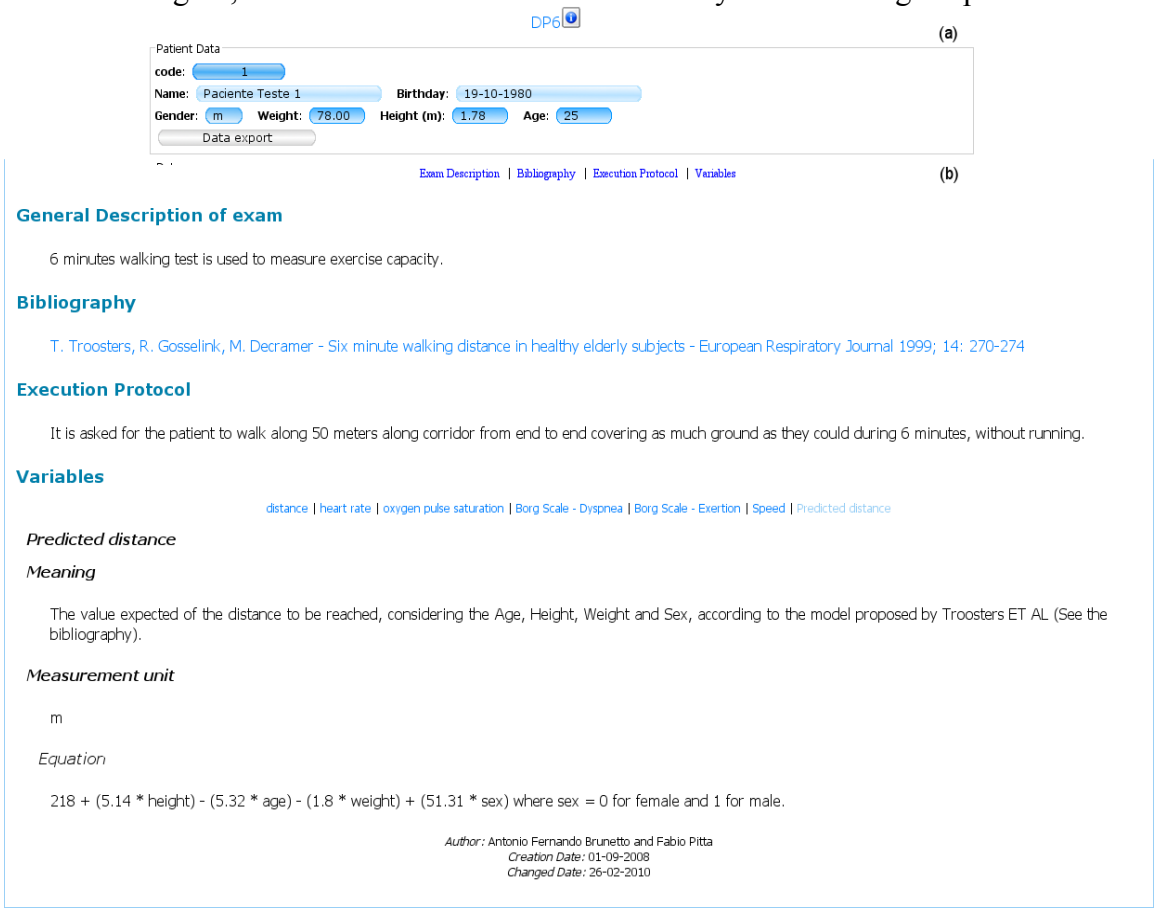


Figure 4 – Metadata of the exam DP-6 in browsing view

Considering some difficulty to get all this information from the experts on the exam, it was developed an authoring module that allows the creation and/or updating of content of metadata of the exams. In this module, the knowledge expert user should select a specific exam and fill a form with the metadata about the exam. The metadata of the exams are stored in RDF [Manola, 2004] and XML documents. Each exam has its XML-Schema [Fallside, 2004] and XML document associated. In this way, the metadata of the exams is independent of the program code, assuring more flexibility of the system and more reliability on the information.

4. Related Works

In SACAR-web we use technologies of open source Software. The set of technologies named LAMP (Linux + Apache + MySQL + PHP) has been used. McDonald (2003) has presented the advances in open source medical software that was revisited in our work. OpenEMed (1999) is a "virtual patient record" based on two CORBAmed specifications: Patient Identification Data Service (PIDS) and Clinical Observation Access Service (COAS). Open Source Clinical Applications & Resources [OSCAR, 1991] is a medical record system developed and used at McMaster University. The software provides registration, scheduling, medical record and billing services. Also, the Good Electronic Health Record (GEHR) based on openEHR [Leslie, 2008] is a model and framework for developing medical record systems. Other works address the issue of medical contents using XML. Hulse et al (2005) proposes a knowledge authoring environment to make easy the development of medical knowledge documents based on XML schema, the KAT. The key features of KAT include quick access to content for authors; flexible, generic addition of new content types; support for content reuse and rapid availability of knowledge content in clinical applications. In this context, Sacar-Web has the module of authoring the metadata of the exams, as described in section 4. PropeRWeb [van der Linden, 2005] is a multidisciplinary electronic health record (HER) system used in extramural patient care for stroke patients. The system uses open source components and is based on open standards. It is a Web application that uses servlets and Java Server Pages (JSP's) with CORBA connection to the database servers. This work, which is similar to ours, uses archetypes. Archetypes are defined by medical experts separated from the software. In Sacar-Web, we have defined a XML document storing all the possible exams and a XML document storing the exams selected by the health expert. In this way, the acceptability of the software by the medical users tends to be greater. The work of McKenna et al (2007) refers to the system PAS (Physiology analysis system), which is a resource to support efficient warehousing, management and analysis of physiologic data, particularly, continuous time-series data that may be extensive, of variable quality and distributed across many files. These works have given support to our project, once the development of software has been started based on the health care professional user needs reported by the knowledge engineer and software designer.

5. Concluding Remarks and future works

The main contributions of this work are: to provide a web-based application addressing different issues about assessment of cardiac and respiratory conditions; to propose a set of strategies to attend the main health care user needs. Among such strategies it is

included: to define different roles for each user type, to establish a user profile that is used by the application to provide an adapted interface, to separate knowledge expert from coding, to include the authoring module of metadata exams, to include dynamical filters in order to retrieve relevant information. Separating knowledge expert from coding has shown to be a good practice in software development, once the maintenance is easer and the system tends to be more reliable. As future work it is being designed several improvements, mainly concerning to usability of the software.

Acknowledgments

Our thanks to CNPq by the financial support and In Memoriam to the knowledge expert in Cardiac and Respiratory Physiology, professor Dr Antonio Fernando Brunetto.

References

The Apache Software Foundation. The Apache HTTP Server Project [Internet]. c2009 Available from: http://httpd.apache.org

Bray, T. et al. (2008). Extensible Markup Language (XML) 1.0 (Fifth Edition) [Internet]. Massachusetts:W3C Recommendation; [revised 2008 Nov 26**] Available from: http://www.w3.org/TR/xml/**

Camargo-Brunetto, M. A. O. and Rossi, A. D. (2005). Uma aplicação Web para análise de exames cardiorrespiratórios. Anais do V Workshop de Informática Médica; Porto Alegre, Brazil. Portuguese.

Casaburi, R. et al. (2005). Principles of Exercise, *Testing and Interpretation*. 4th edition. Philadelphia: Lippincott Williams & Wilkins; 568 p.

ERS (1993). Standardized lung function testing: Official statement of the European Respiratory Society. *European Respiratory Journal*. 16(Suppl):1-100.

Fallside, D. C. and Walmsley, P. (2004). XML Schema Part 0: Primer Second Edition [Internet]. Massachusetts:W3C Recommendation; c2004 [revised 2004 Oct 28], Available from: http://www.w3.org/TR/xmlschema-0/

Guyatt, G. H. e al. (1987). A measure of quality of life for clinical trails in chronic lung disease. *Thorax. 42(10):*773-778.

Hulse, N. C. et al. (2005). KAT: A flexible XML-based Knowledge Authoring Environment. *Journal of American Medical Informatic Association*. 12(4):418-430.

Jones, P.W., Quirk, F.H., Baveystock, C.M. (1991) The St. George's Respiratory Questionnaire. *Respir Med. Sep;85(Suppl)*:25-31.

Knudson, R.J. et al (1976) The maximal expiratory flow-volume curve. Normal standards, variability and effects of age. *Am Rev Respir Dis*. 113(5):587-130.

Leslie, H. OpenEHR (2008): The World's Record [Internet]. Australia:Pulse+IT Magazine. Available from: http://www.pulseitmagazine.com.au

McKenna, T. M. et al. (2007). The physiology analysis system: An integrated approach for warehousing, management and analysis of time-series physiology data. *Computer Methods and programs in Biomedicine. 86(1)*:62-72.

Manola, F., Miller, E., McBride, B.(2004). RDF Primer [Internet]. Massachusetts:W3C Recommendation; c2004 [revised 2004 Feb 10]. Available from: http://www.w3.org/TR/rdf-primer/

McDonald, C.J. et al (2003). Open Source Software in medical informatics – why, how and what. *International Journal of Medical Informatics. 69(2-3)*:175-184.

Oracle Corporation. MySQL (2010): the world most popular open source database [Internet]. Available from: http://www.mysql.com

OSCAR (1991). Open Source Clinical Application & Resources (OSCAR) [Internet]. Available from: http://www.oscarcanada.com

Payne,G. E., Skehan, J. D. (1996). Shuttle walking test: a new approach for evaluating patients with pacemakers. *Heart. 75(4):*414-418.

SBPT (1996). I Consenso Brasileiro sobre Espirometria. *Jornal de Pneumologia. 22(Suppl)*:99-165.

Troosters, T., Gosselink, R., Decramer, M. (1999). Six minute walking distance in healthy elderly subjects. *Eur Respir J. 14*:270-274.

OpenEMed (2001) [Internet]. Available from: http://www.openemed.org/

van der Linden et al (2005). A. PropeR revisited. *International Journal of Medical Informatics. 74(2-4)*:235-244.

e-kids: Uma Ferramenta no Auxílio da Aprendizagem de Crianças Portadoras de Disfunção Global do Desenvolvimento (Autista), baseada no método ABA

Otilio Paulo da Silva Neto[1], Paulo Tarcilio de Sousa e Carvalho[2], Aislan Rafael Rodrigues de Sousa[1]

[1]IFPI – Instituto Federal do Piauí – Campos Picos - CEP 64.600-000 Picos (PI)
[2]Instituto de Ensino Superior R.Sá – 64.600-000 – Picos (PI)

{otiliopaulo, aislanrafael}@ifpicos.edu.br, paulo.multicom@gmail.com

Abstract: *This paper presents a prototype application to help professionals in the field of psychology that use ABA methods in teaching learning of children with global developmental dysfunction, autism or simply, as it is popularly known. The e-kids is a prototype that will be described in detail to show how it is important to use new technologies in the construction process of knowledge of the alphabet, numbers, colors, images and more.*

Resumo: *Este artigo apresenta um protótipo de aplicação para auxiliar profissionais da área de psicologia que utilizam o método ABA no ensino da aprendizagem de crianças portadoras de disfunção global do desenvolvimento, ou simplesmente autismo, como é popularmente conhecida. O e-kids é um protótipo que será descrito de forma detalhada para mostrar quão é importante o uso de novas tecnologias no processo da construção do conhecimento do alfabeto, números, cores, imagens entre outros.*

1 A Tecnologia e o Autismo

O crescente avanço dos recursos tecnológicos contribui constantemente para aumentar as formas de aprendizagem dos indivíduos, como a melhoria no processamento da informação desde a recepção (audição e visão), até a conversão destas novas informações até a armazenagem neural (aprendizagem) conforme estudos que envolveram os softwares com esta finalidade, sendo descritos por Fiolhais e Trindade (2003) e Balen et al (2008).

A proporção estimada de autistas é aproximadamente de 1 a 5 casos em cada 1000 crianças no Brasil, Klin (2006). Em texto de Paiva e Ribeiro (2010)**,** eles citam que, de acordo com a ONU, acredita-se ter mais de 70 milhões de pessoas com autismo em todo o mundo, e que só nos Estados Unidos essa estimativa fica em torno de 1:110. Nesse contexto, descrevemos o protótipo **e-*kids***, uma ferramenta de auxilio aos psicólogos e psicopedagogos que atuam na alfabetização de crianças portadoras de disfunção global do desenvolvimento utilizando o método ABA.

2 Análise do Comportamento Aplicado (ABA)

Análise do Comportamento Aplicado (Applied Behavior Analysis) – ABA, advêm da abordagem científica **Behaviorismo**, que tem como objeto de estudo a analise objetiva do comportamento observável e mensurável. Observa, analisa e explica a relação que existe entre o comportamento e o meio ambiente onde ocorre, Lear (2004).

A metodologia de ensino da Análise do Comportamento Aplicada, normalmente é individualizado, em situações de um-para-um e se propõem a ensinar à criança habilidades que ela não possui.

2.1 Metodologia

Utiliza-se ensino por Tentativas Discretas (*Discrete Trial Teaching* – DTT), consiste em um método de aprendizagem estruturado, comandado por outrem, e caracteriza-se por dividir sequências complicadas de aprendizado em passos muito pequenos ou discretos (separados) ensinados um de cada vez durante uma série de tentativas (*trials*), seguindo de reforços positivos (prêmios) e o grau de ajuda que for necessário para alcançar o objetivo/objeto. Lear (2004).

3 O Protótipo

Para a obtenção do objetivo proposto por este trabalho, foi necessário o desenvolvimento de um software que terá o papel de interagir com a criança e a construção de um hardware que será acoplado ao microcomputador, pelo o qual a criança irá interagir com o referido software, como mostra a **Figura 1**.

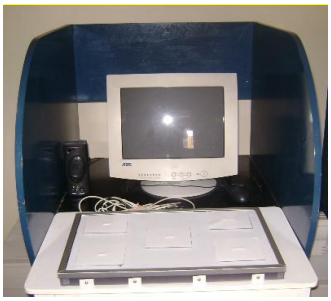


Figura 1: Modelo do Protótipo e-kids

3.1 O Ambiente

O ambiente proposto no ***e-kids*** é composto de um profissional de psicologia ou psicopedagoga que conduzirá o processo de aprendizagem, de uma criança que utilizará a aplicação, de um microcomputador onde a aplicação estará rodando, de um software desenvolvido para gerenciar a aplicação e um hardware construído que permite a interação da criança com o software. A **Figura 2** demonstra o ambiente do protótipo.



Figura 2: Ambiente do protótipo

3.2 O Software

O software foi desenvolvido na linguagem PHP com o uso da *Framework Spaghetti*, para a plataforma Web, permitindo que seja acessado de qualquer computador através da Internet, Utilizou-se o banco de dados Mysql para armazenamento dos dados e para a interação uma melhor interação criança-software. Utilizamos o *Flash* devido aos seus recursos de animação para Web.

3.3 O Hardware

Inicialmente notou-se que as crianças tinham muitas dificuldades em utilizar o teclado convencional devido as suas coordenações motoras, daí surgiu à idéia de construirmos um teclado específico para o uso do ***e-kids***.

Na construção do Hardware, **Figura 3**, utilizou-se fita durex para fixar os fios evitando curto, caneletas para fixar as laterais com a base, alumínio na base superior, plástico adesivo branco cobrindo a parte superior, parafusos nas bases laterais das caneletas, adesivo dupla face fixando a película de plástico com as trilhas, cola para as borrachas nos cantos de cada botão para servir de base, etiquetas adesivas identificando as teclas, fio de cobre fino interligando as trilha, placa com circuito e cabo USB ou PS2, película de plástico com trilhas e circuitos elétricos.



Figura 3: Teclado de Cores

3.4 Utilização

Na utilização do e-kids, o profissional de psicologia monta as cartas no teclado de cores e orienta a criança para teclar na cor desejada, conforme ilustra a **Figura 3**. A criança por sua vez deverá tocar em uma das cartas distribuídas no teclado, caso tecle na cor correta o software responderá com mensagens de incentivo: “parabéns você acertou!”, “muito bem nome da criança” e “outros incentivos”, e então as cartas são trocadas de lugar e o processo se repete, conforme a evolução da criança.

DIEGO R. et al (2010) afirma que a utilização de aplicações em ambientes virtuais objetiva formar profissionais mais capacitados para o futuro.

4 Resultados Preliminares

Submetemos o uso do ***e-kids*** em 8 (oito) crianças, sendo 3 (três) crianças normais e 5 (cinco) autistas, as primeiras não tiveram nenhuma dificuldade quanto a utilização do protótipo, como já era esperado. As segundas, surpreendentemente não tiveram dificuldade quanto à utilização da ferramenta, mas observou-se preliminarmente que tais crianças despertavam um interesse maior em utilizar o ***e-kids***.

Notou-se ainda, mesmo que precocemente, as crianças autistas assimilavam as atividades mais rapidamente do que as crianças que não utilizavam o ***e-kids*.**

5 Conclusões

O uso do ***e-kids*** em crianças portadoras de disfunção global do desenvolvimento obteve um resultado satisfatório. De forma preliminar, as referidas crianças conseguiram assimilar as cores de forma simples, como se estivessem brincando.

Atribui-se tal resultado devido ao protótipo despertar nas crianças uma nova forma de aprender, diferente dos métodos tradicionais.

Concluímos que o protótipo ***e-kids*** está pronto para ser aplicado em crianças portadoras de tais disfunções e que o mesmo as auxiliam na aprendizagem.

Pretende-se futuramente aplicar o ***e-kids*** para auxiliar em outros tipos de atividades curricular para crianças com dificuldades de aprendizagem, facilitando assim, o trabalho de psicólogos, psicopedagogos e profissionais da saúde e educação.

6 Referências

BALEN, Sheila Andreoli; MASSIGNANI, Rosiana; SCHILLO, Raquel. Aplicabilidade do software Fast Forword na reabilitação dos distúrbios do processamento auditivo: resultados iniciais. **Rev. CEFAC**, São Paulo, v. 10, n. 4, dez. 2008. Disponível em <http://www.scielo.br/>. acesso em 19 jan. 2011.

COSTA, Maria Ione Ferreira da; NUNESMAIA, Henrique Gil da Silva. Diagnóstico genético e clínico do autismo infantil. **Arq. Neuro-Psiquiatr.**, São Paulo, v. 56, n. 1, mar. 1998 . Disponível em <http://www.scielo.br/>. acesso em 19 jan 2011.

DIEGO R. Colombo Dias, JOSÉ R. Ferreira Brega, ILDEBERTO A. Rodello, JOSÉ R. Pereira Lauris. Sistema de Multiprojeção para o Auxílio ao Ensino de Odontologia. WIM – X Workshop de Informática Médica, 2010, Belo Horizonte. XXX Congresso da Sociedade Brasileira de Computação, 2010.

FIOLHAIS, Carlos; TRINDADE, Jorge. Física no computador: o computador como uma ferramenta no ensino e na aprendizagem das ciências físicas. **Rev. Bras. Ensino Fís.**, São Paulo, v. 25, n. 3, set. 2003 . Disponível em <http://www.scielo.br/>. acesso em 19 jan. 2011.

KLIN, Ami. Autismo e síndrome de Asperger: uma visão geral. **Rev. Bras. Psiquiatr.**, São Paulo, 2006 . Disponível em <http://www.scielo.br/>. acessos em 27 jan. 2011.

PAIVA Junior; RIBEIRO, Sabrina. Pesquisa do CDC revela número alto de prevalência de autismo nos EUA em crianças de oito anos, além de grande aumento em relação a pesquisa anterior. **Revista Autismo:** Informação gerando ação. Ed 0, Set 2010. Disponível em: <http://revistaautismo.com.br/> Acesso em 19 Jan 2011

LEAR, K. Ajude-nos a Aprender. (Help us Learn: A Self-Paced Training Program for ABA Part 1:Training Manual). Traduzido por Windholz, M.H.; Vatavuk, M.C.; Dias, I. S.;Garcia Filho, A.P. e Esmeraldo, A.V.Canadá, 2004.

Serious Game para o tratamento de torcicolo

Jean Vitor de Paulo[1], Luis Fernando Braga. Lopes[2], José Barbosa Dias Jr[2], Eduardo Filgueiras Damasceno[2]

[1] FESURV – Universidade de Rio Verde (GO)
[2] Instituto Federal Goiano Campus de Rio Verde (GO)

jean.exe@live.com, edamasceno.rv@ifgoiano.edu.br, lfbraga.rv@ifgoiano.edu.br

Abstract. *This work shows the implementation of a rehabilitation Digital Game (Rehabilitation Serious Game) developed in Augmented Reality in order to achieve the therapeutic procedures for treatment of torticollis, where the user patient can enjoy the virtual environment while doing rehabilitation exercises. This work shows only the techniques and methods of computational implementation.*

Resumo. *Este trabalho mostra a implementação de um Jogo Digital Reabilitador (Rehabilitation Serious Game) desenvolvido em Realidade Aumentada com o intuito de realizar os procedimentos terapêuticos de tratamento de torcicolo, onde o usuário paciente pode usufruir do ambiente virtual ao mesmo tempo em que faz os exercícios reabilitadores. Este trabalho mostra apenas as técnicas e os modos de implementação computacional.*

1. Introdução

Com o advento da Realidade Virtual e Aumentada (RVA) um novo paradigma começou a ser postulado, criando assim um cerne voltado ao usuário e não mais ao dispositivo. E partindo-se deste princípio houve um fortalecimento na aliança entre as áreas de saúde e de computação, permitindo um progresso colossal para as aplicações médicas.

E neste constante avanço as aplicações se destacam pelo uso da tecnologia de RVA para as atividades de reabilitação, pois possuem a flexibilidade na edição de ambientes virtual de aplicação e de simulação para saúde (PIRON *et al.*, 2005).

Dentre novas interações e interfaces apresentadas, é a utilização de jogos virtuais no ambiente terapêutico, na qual os pacientes a serem tratados tem um maior envolvimento com a reabilitação (GEIGER *et al.*, 2002), produzindo assim, resultados mais satisfatórios.

Este trabalho mostra o desenvolvimento de um protótipo de um jogo digital baseado em realidade aumentada para atenuar um quadro álgico do torcicolo.

O torcicolo é uma dorsopatia que pode ser definida como um distúrbio da coluna cervical (na região do pescoço), relacionada diretamente a contração espástica ou não, de modo que a cabeça permanece inclinada para um dos lados e o queixo chega a projetar-se para o lado oposto, fazendo com que os movimentos da região cervical tornem-se às vezes dolorosos e acabe limitando o movimento da cabeça. O torcicolo também pode ser um quadro encontrado nos indivíduos que mantém uma Posição Viciosa da Cabeça (PVC), o que é muito comum da área de informática.

Um movimento brusco da cabeça, ou mesmo um mau jeito durante o sono podem provocar um leve ferimento em uma das articulações presentes na região do pescoço (REDDER, 2005), conforme apresentado na Figura 1a.



Figura 1a - Distinção de um torcicolo.

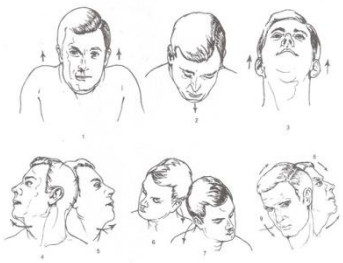


Figura 1b - Exercícios para torcicolo.

Exercícios de alongamento regulares são benéficos para a prevenção da recorrência do torcicolo. Nos casos de cervicalgia crônica, são utilizadas medicações para dor crônica, associada a um programa de reabilitação que visa melhora dos sintomas e prevenção da recorrência dos sintomas.

O tratamento fisioterápico é baseado nos exercícios regulares de alongamento e flexão do pescoço, onde o paciente executa movimentos circulares e posicionados com a cabeça Figura 1b.

2. Serious Games

O Conceito de Serious Games parte do foco de simular virtualmente situações práticas do dia-a-dia, com o objetivo de proporcionar o treinamento de profissionais, situações críticas em empresas, conscientização para crianças, jovens e adultos e até mesmo para situações corriqueiras, através de jogos digitais (ABORDAGENS..., 2009). Existem de acordo com (STAPLETON, 2004), sete focos da abrangência dos Serious Games, sendo eles: a) Educação Primária; b) Educação Superior; c) Tratamento de Saúde; d) Corporativo; e) Militar; f) Não-Governamentais; g) Outros;

Dentre estes o setor de saúde em especial, tem se beneficiado largamente com o advento dos Serious Games. Os Jogos Sérios aplicados a essa respectiva área tem de acordo com (ABORDAGENS..., 2009), quatro áreas principais. a)Auxiliares de Terapia; b)Promoção de saúde e condicionamento físico; c)Monitoramento de Saúde; d) Treinamento. Esse artigo apresenta um jogo baseado em RVA que promove um auxílio de terapia.

3. Trabalhos Relevantes

As aplicações de RVA formam um conjunto no qual são criados ambientes de simulações tridimensionais da realidade e podem ser exploradas por técnicas reabilitativas supervisionadas por um terapeuta (RIVA; MANTOVANI; GAGGIOLI, 2004).

Existem diversas vantagens nas aplicações virtuais de reabilitação, pois os elementos que o paciente visualizará são virtualmente simulados, portanto eles não podem se ferir durante a realização da atividade e, além disso, todos os estímulos podem ser controlados pelo terapeuta (ASSIS; CORREA; LOPES, s.d.).

Utilizando-se dessas vantagens, existem sistemas completamente baseados nessas tecnologias de reabilitação. Dentre as terapias que se destacam para a reabilitação fisioterápica que utilizam as técnicas de captura de imagens por câmeras digitais é possível destacar o projeto ARPhysio (LIMA, 2006) que mostra uma aplicação de RA que registra os dados dos marcadores como forma de avaliar os dados antropométricos e cinesiológicos para se determinar a validade dos exercícios para o tornozelo. Este sistema utiliza tecnologia interativa para fornecer aos pacientes dicas visuais para direcionar sua

atenção para a estrutura dinâmica de um movimento, facilitando assim a reabilitação do músculo lesionado.

4. Materiais e Método

Para chegar-se a um objetivo aplicando a tecnologia de realidade aumentada nesse tipo de reabilitação, foi necessário que o jogo que seguisse os movimentos de uma reabilitação. Para isso foi utilizado o framework de realidade aumentada SudaRA (Suporte ao Desenvolvimento de Realidade Aumentada) que é baseado no ARtoolkit, para o retorno do posicionamento da cabeça, em conjunto com o ambiente de desenvolvimento Microsoft Visual Studio 2010 e enfim para a modelagem 3D foi utilizado o software Blender.

5. Descrição do protótipo

O rastreamento implementado no framework SudaRA estima a pose de marcadores, tornando possível o desenvolvimento de aplicações que necessitem conhecer a posição e orientação de elementos, que são representados na cena por estes marcadores. Como o padrão da Realidade Aumentada, é utilizado marcadores em formas quadradas, para o rastreamento e posicionamento da cabeça do paciente, que é identificada devido a utilização de uma faixa na região da testa com os respectivos marcadores. O controle da motocicleta é realizado com movimentos da cabeça. A localização, rotação e identificação dos marcadores são a base dos pontos de referência utilizados para a condução da aplicação.

Logo ao identificar o marcador, é plotado em tela, um guidão, que referencia a posição da cabeça em relação à aplicação, e é através do movimento da cabeça, que ocasionará uma alteração na rotação do marcador e a moto será então guiada pelo percurso. A frequência dos movimentos feitos com o pescoço depende basicamente da velocidade da moto. A velocidade, assim como a movimentação, pode ser controlada através dos movimentos da cabeça. O ato de ir com a cabeça para frente, aumenta-a, assim como o movimento de ir com a cabeça para trás realiza o processo reverso. É necessário logo após realizar uma ação que altera a velocidade, retornar a cabeça à posição original.

6. Análise Comparativa

O Jogo aqui desenvolvido prezou por igualar ao máximo os movimentos executados em uma reabilitação devido a dores na área cervical proveniente de um torcicolo. Ele induz, através de uma pista virtual, movimentos que são feitos em uma reabilitação terapêutica. O paciente deve conduzir uma moto através de um circuito fictício, realizando toda a terapia.

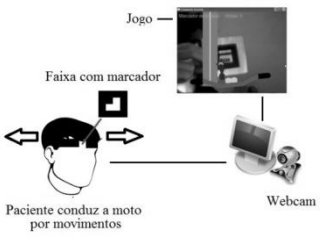


Figura 2 a -funcionamento.

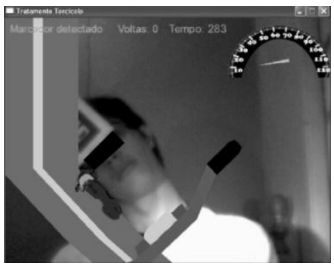


Figura 2 b - Aplicação em andamento

Ao inicializar a aplicação, é necessário um prévio posicionamento da cabeça em diversas posições. E a cada posicionamento será salva a respectiva imagem a cada confirmação, para análise futura do terapeuta, e ao final do tempo de aplicação, esse processo irá se repetir.

8. Considerações Finais

Aplicações para a área de saúde **possibilitam** uma imersão sem precedentes a pessoa que está sendo tratada e também uma forma alternativa de realizar atividades que antes eram realizadas de uma forma única e sem muito envolvimento.

A computação aplicada à reabilitação de pacientes já foi demonstrada em diversos estudos realizados anteriormente, e já é possível encontrá-la em uso em um grande número clínicas de reabilitação tanto física quanto cognitiva.

Os tipos de movimentos necessários para um tratamento cinesiológico do torcicolo são totalmente aplicáveis a meios computacionais que realizam uma detecção da posição e filtrem esses movimentos para uma futura análise. Os estudos aplicados para o desenvolvimento desse artigo, só comprovam o fato, de que a Realidade Aumentada pode ser aplicada em praticamente todas as áreas de reabilitação.

Referências

PIRON, L., TONIN, P., PICCIONE, F., IAIA, V., TRIVELLO, E., AND DAM. M. “Virtual environment training therapy for arm motor rehabilitation”, in Presence, Vol. 14 p. 732-740, 2005.

CHRISTIAN GEIGER, CHRISTIAN REIMANN, JÖRG STÖCKLEIN, E VOLKER PAELKE, “JARToolKit – A Java Binding for ARToolKit”, Proceedings of the Augmented Reality ToolKit Workshop, Darmstadt, 5 pp., 29 de Setembro, 2002

EDDER REDDER, Infográfico "Saúde é vida" – “O que é um torcicolo” Disponível em http://saude.abril.com.br/edicoes/0259/infograficos/conteudo_88238, 2005 (acesso em fevereiro de 2011)

"Abordagens práticas de Realidade Virtual e Aumentada", In: SVR 2009 IX Symposium on Virtual and Augmented Reality, Porto Alegre, maio 2009.

ANDEW J. STAPLETON, "Serious Games: Serious Opportunities", Swinburne University of Technology, Hawthorn, Australia, 2004

RIVA, GIUSEPPE; MANTOVANI, FABRIZIA; GAGGIOLI, ANDREA." Presence and rehabilitation: toward second-generation virtual reality applications in neuropsychology". In: Journal of NeuroEngineering and Rehabilitation, 2004.

GILDA APARECIDA DE ASSIS, ANA GRASIELLE DIONÍSIO CORREA, CÍCERO JOSE NUNES, ROSELI DE DEUS LOPES. "Neuror: Sistema de Realidade Aumentada para Reabilitação Física de Pacientes Vítimas de Acidente Vascular Encefálico", Escola Politécnica, Universidade de São Paulo, Brasil

LIMA, J. P. S. do M.; LIMA, A. S. do M.; TEICHRIEB, V.; KELNER, J.; MOURA FILHO, A. G. de. “ARPhysio: Usando Realidade Aumentada para Análise do Movimento Humano.” In: III Workshop de Realidade Aumentada, Rio de Janeiro, RJ, Brasil, Setembro 2006.

WIT - V Women in Information Technology

PREFÁCIO

O WIT é uma iniciativa da SBC para discutir os assuntos relacionados a questões de gênero e a Tecnologia de Informação (TI) no Brasil - histórias de sucesso, políticas de incentivo e formas de engajamento e atração de jovens, especialmente mulheres, para as carreiras associadas à TI.

Organizado na forma de palestras convidadas e painéis, o workshop estará centrado em debater problemas relacionados à mulher e ao seu acesso à TI, tanto do ponto de vista de mercado de trabalho quanto de inclusão e alfabetização digital. Os temas abordados se concentrarão na necessidade de educar, recrutar e treinar mulheres, como uma política estratégica para o desenvolvimento e competitividade nacional e regional.

O objetivo principal do WIT é criar um fórum que promova estratégias para aumentar a participação de mulheres em TI no Brasil. Os temas abordados incluirão:aspectos críticos que impactam o acesso pleno das mulheres, treinamento, participação e liderança na área; estratégias para aumentar a visibilidade, no Brasil, dos problemas relativos a gênero e TI, com a conscientização de todos os segmentos da nossa sociedade; políticas empresariais nacionais e internacionais para fazer face a tais desafios e apresentação de histórias de sucesso.

FORUM MENINAS DIGITAIS - EVENTO SATELITE do WIT 2011

O Fórum Meninas Digitais faz parte das atividades de um programa direcionado às alunas do ensino médio/tecnológico, para que conheçam melhor a área de informática e das Tecnologias da Informação e Comunicação, de forma a motivá-las a seguir carreira nessas áreas. Também, o evento tem colaboração de multiplicadores desta proposta, que aproveitam a oportunidade discutir projetos e parcerias, de forma a disseminar esta ideia no território nacional.

COMISSÕES

Coordenação Geral

Claudia Bauzer Medeiros, IC-Unicamp (coordenadora geral)
Karin Breitman, DI-PUC Rio (coordenadora geral)
Roberta de Souza Coelho, DIMAp-UFRN (coordenadora local)

WPerformance - X Workshop em Desempenho de Sistemas Computacionais e de Comunicação

PREFÁCIO

O Workshop em Desempenho de Sistemas Computacionais e de Comunicação, conhecido como WPerformance, é um fórum para discussão e disseminação de idéias sobre métodos, técnicas e ferramentas para avaliação de desempenho de sistemas computacionais e de comunicação. Em sua décima edição, o WPerformance 2011 será realizado nos dias 21 e 22 de julho como parte do XXXI Congresso da SBC, em Natal-RN.

Na edição deste ano, um total de 42 artigos completos foram submetidos. Todos passaram pelo processo de revisão pelo comitê de programa formado por 27 pesquisadores de instituições nacionais e internacionais. Dos artigos submetidos, 13 foram selecionados para publicação nos anais do evento, dando uma taxa de aceitação de 31%. Esta taxa de aceitação indica a alta competitividade do evento e nos leva a crer que teremos uma programação de alta qualidade técnica.

Entre as novidades para o evento deste ano, o WPerformance 2011 traz o retorno da premiação do melhor artigo do workshop. Dentre os 13 artigos aceitos para publicação, os três trabalhos mais bem avaliados pelo comitê de programa foram selecionados como candidatos ao prêmio de melhor artigo. Esses três trabalhos foram repassados ao comitê de premiação, formado por professores de renome que atuam na área de avaliação e desempenho, que indicaram o melhor artigo do WPerformance 2011. O resultado será anunciado na abertura oficial do WPerformance e a primeira sessão técnica da programação será composta pelos três trabalhos indicados.

Além das sessões técnicas, teremos este ano dois "keynote speakers": Prof. Matthias Grossglauser do EPFL, Suíça e o Prof. Paulo Fernandes da PUC-RS. Também teremos um painel onde pesquisadores ligados ao tema de avaliação desempenho discutirão os atuais “Hot Topics” da área.

A coordenação do X WPerformance agradece a dedicação de todo o comitê de programa e do comitê de seleção do melhor artigo pelo trabalho no processo de revisão dos artigos, o que viabilizou uma programação técnica de qualidade. Agradecemos, também, todo o apoio da organização geral do CSBC 2011. Por fim, convidamos a todos para participar e enriquecer a qualidade técnica das discussões do workshop. Sejam muito bem vindos às sessões técnicas do WPerformance 2011!

Cordialmente,

Antonio Augusto de Aragão Rocha (IC-UFF)
Daniel Ratton Figueiredo (PESC-COPPE/UFRJ)
Coordenadores do WPerformance 2011

COMISSÕES

Coordenação Geral

Antonio Augusto de Aragão Rocha, IC-UFF (coordenador geral)
Daniel Ratton Figueiredo, PESC-COPPE-UFRJ (coordenador geral)
Adilson Barboza Lopes, DIMAp-UFRN (coordenador local)

Comitê de Programa

Aline Viana - INRIA (França)
Ana Paula Couto Silva - UFJF
Antônio Abelém - UFPA
Artur Ziviani - LNCC
Bruno Ribeiro - UMASS (EUA)
Carlos Fisch de Brito - UFC
Carlos Alberto Vieira Campos - UNIRIO
Cristina Murta - IFMG
Daniel Batista - USP
Edmundo Madeira - UNICAMP
Eduardo Nakamura - FUCAPI/UFAM
Esteban Clua - IC-UFF
Fernando Silveira - Technicolor
Gustavo Figueiredo - UFBA
José Ferreira de Rezende - PEE/UFRJ
José Augusto Suruagy Monteiro - UNIFACS
Jussara Almeida - UFMG
Katia Obraczka - UCSC
Magnos Martinello - UFES
Marcelo Dias de Amorim - LIP6/CNRS, UPMC Sorbonne Universités (França)
Marinho Barcellos - UFRGS
Moises Ribeiro - UFES
Morganna Diniz - UNIRIO
Paulo Fernandes - PUCRS
Rosa Leão - PESC/UFRJ
Vanessa Braganholo Murta - IC/UFF
Wagner Meira Jr. - UFMG

Comitê de premiação dos melhores artigos

Edmundo de Souza e Silva - PESC/UFRJ
Nelson Fonseca - UNICAMP
Valmir Barbosa - PESC/UFRJ
Virgilio A. F. de Almeida - UFMG

ARTIGOS

Caracterização e Análise de Redes de Remetentes e Destinatários de SPAMs

Thaína Amélia de Oliveira Alves[1], **Humberto T. Marques-Neto**[1]

[1] Departamento de Ciência da Computação
Pontifícia Universidade Católica de Minas Gerais (PUC Minas)
30.535-901 - Belo Horizonte - Brasil

thaina.alves@sga.pucminas.br, humberto@pucminas.br

Abstract. *The interaction between senders and receivers of electronic messages (emails) can be studied as a complex network of information exchange that represents the process of communication between people involved. This article characterizes the networks formed from this exchange of emails recorded in a log generated by the real filter spam from an email provider. The results show that typical metrics of complex networks, such as popularity and connectivity can be used to assist the identification of malicious users (spammers). We observed that few spammers have high popularity and high connectivity in the network. This characterizes users that can affect the performance of electronic mail service, disseminating large quantities of unwanted messages that are probably to be processed and discarded at their destination.*

Resumo. *A interação entre remetentes e destinatários de mensagens eletrônicas (e-mails) pode ser estudada como uma rede complexa de troca de informações que representa o processo de comunicação entre as pessoas envolvidas. Este artigo caracteriza as redes formadas a partir dessa troca de e-mails registrada em um log real gerado pelo filtro de spam de um provedor de correio eletrônico. Os resultados mostram que métricas típicas de redes complexas, tais como popularidade e conectividade, podem ser utilizadas para auxiliar a identificação de usuários maliciosos (spammers). Observou-se que poucos spammers possuem alta popularidade e alta conectividade na rede, caracterizando usuários que podem afetar o desempenho do serviço de correio eletrônico ao disseminarem grande quantidade de mensagens indesejadas que, provavelmente, serão processadas e descartadas no seu destino.*

1. Introdução

Cada vez mais a Web vem se transformando em um meio para a disseminação de conteúdo entre pessoas relacionadas direta ou indiretamente. Juntamente com esse crescimento, percebe-se também um aumento no volume de e-mails indesejados e não solicitados (spams). Esforços dedicados ao bloqueio de spams também ocorrem de maneira significativa. Apesar desses esforços, e-mails indesejados ainda representam cerca de 90% de todos e-mails trafegados na Internet [Symantec 2011].

Relatórios recentes mostram que o Brasil está presente entre os três países que mais disseminam spams na Internet, juntamente com o Estados Unidos e a Índia [IronPort Email and Web Security 2011]. No relatório do [Message Labs 2011], pode ser

visto que em fevereiro de 2011 a taxa de spams por e-mail trafegado aumentou 2,7% comparado ao mês anterior (1 em cada 1.23 mensagens de e-mail). Apesar da existência de mecanismos dedicados ao bloqueio de spam, ainda nota-se a chegada de muitas mensagens indesejadas na caixa de entrada dos destinatários.

A melhoria dos mecanismos antispam pode trazer muitos benefícios para os usuários finais que deixariam de ter suas caixas de entrada repletas de e-mails indesejados. Entretanto, a filtragem realizada por esses mecanismos pode ser ineficiente devido a falta de conhecimento do comportamento dinâmico de spammers nas redes de e-mail. Tal fato, é evidenciado no relatório da [Nucleus Research 2007], que mostra que o spam não filtrado, ou seja, que precisa ser filtrado pelo próprio usuário, acarreta uma queda de produtividade em cerca de 1,2% por empregado ao ano, causando prejuízos para as empresas. Essa grande quantidade de spams que circulam na Internet acaba afetando o desempenho dos provedores de e-mail e, consequentemente, causa desperdício de recursos para o tratamento do tráfego de spam na rede [Dan Twining 2004]. Prejuízos desse tipo poderiam ser minimizados com a análise dos spammers para sua efetiva identificação.

A troca de informações entre pessoas através do uso de um serviço de correio eletrônico tem estrutura semelhante a uma rede complexa, que é definida em [Easley and Jon 2009] como uma rede que possui uma topologia com caracaterísticas específicas, tais como, alto coeficiente de agrupamento, caudas pesadas em distribuições de probabilidade e reciprocidade. Sendo assim, sugere-se que a rede de remetentes e destinatários de spams desta pesquisa seja estudada como uma rede complexa, a partir da análise de características como reciprocidade, conectividade e popularidade dos nós da rede. A reciprocidade será discutida na Seção 2, as duas últimas características serão analisadas mais especificamente na Seção 4.

O aumento das redes formadas por usuários de e-mails e do conteúdo malicioso existente nelas, motiva a identificação de possíveis efeitos negativos na atuação dos correios eletrônicos. Através de algumas métricas de redes complexas tais como, popularidade e conectividade, é possível identificar características desse processo de troca de informações [Easley and Jon 2009]. A popularidade é um fenômeno caracterizado por desequilíbrios extremos, no contexto dessa pesquisa seria poucos destinatários recebendo muitos spams enquanto a maioria dos destinatários recebendo poucos spams. Já a conectividade representa se os remetentes da rede ao enviarem spams atingem os destinatários de forma regular, possuindo conexão com vários destinatários. Ao analisar tais métricas, pode ser possível identificar focos de spams.

Este artigo caracteriza, através do uso de métricas de redes complexas, uma rede de troca de informações formada pelos remetentes e destinatários de spams presentes em um log real gerado pelo filtro antispam de um provedor de e-mails. Tal análise busca identificar possíveis usuários focos de recepção de tráfego malicioso, pois eles podem causar problemas de desempenho dos serviços de e-mail a partir do desperdício de recursos para o tratamento de mensagens que, provavelmente, serão descartadas no seu destino.

O restante deste trabalho está organizado em 5 seções. Os trabalhos relacionados são discutidos na Seção 2. Na Seção 3 é descrita a metodologia de caracterização. A Seção 4 apresenta e analisa os resultados relevantes para este trabalho e na Seção 5 é apresentada a conclusão e possíveis trabalhos futuros.

2. Trabalhos Relacionados

Diversas pesquisas na área de caracterização de tráfego malicioso buscam identificar as estratégias e comportamento dos spammers. O trabalho de [Ramachandran and Feamster 2006] analisa as estratégias dos spammers no nível de infraestrutura da rede. Os autores apresentam características da origem das mensagens e concluem que o envio de spams acontece em faixas restritas de endereços de IPs. Em outra pesquisa nesta mesma corrente realizada por [Li and Hsieh 2006], observou-se o surgimento de grandes grupos de IPs que enviam spams referenciando repetidas URLs contidas no conteúdo das mensagems de spam.

Na busca por diferenciar comportamentos maliciosos de comportamentos legítimos, a pesquisa de [Gomes et al. 2004] caracteriza e analisa uma carga de trabalho com base em alguns critérios, tais como, tamanho das mensagens e processo de chegada. Foi possível identificar que o comportamento malicioso (spam) e o comportamento legítimo se diferenciam em muitos dos aspectos. Em [Gomes et al. 2009] foi observado que e-mails legítimos tipicamente possuem um direcionamento social, ou seja, as mensagens possuem relação bilateral, pois, geralmente são respondidas. Por outro lado, spams não possuem reciprocidade, ou seja, são mensagens unilaterais que atingem uma grande quantidade de destinatários.

O trabalho de [Pu and Webb 2006] faz uma análise de mensagens enviadas, avaliando as técnicas que os spammers utilizam para construir as mensagens. Os autores mostraram que algumas técnicas deixam de ser usadas devido a fatores como correção na segurança de programas. Por outro lado, outras estratégias persistem por mais tempo. Levando em conta este comportamento dinâmico dos spammers, o trabalho de [Gomes et al. 2005] analisa quantitativamente um tráfego de e-mail verificando várias características, entre elas, a entropia dos fluxos de entrada e de saída de cada nó da rede, como proposto por Shannon [C.E Shannon 1948]. Essa pesquisa mostrou que o tráfego legítimo apresenta menor entropia do que o tráfego oportunista gerado pelo envio de spams.

Ainda em questão a natureza dinâmica dos spammers a pesquisa de [Guerra et al. 2010] avalia filtros antispam para a análise de tendências de spam, pois, eles são os agentes que podem forçar os spammers a mudar suas táticas. Para esta pesquisa foram utilizados filtros antigos e recentes do Open Source filter Spam Assassin [Project 2010] para comparar spams dos últimos 12 anos. Os resultados mostram que técnicas de spammers mudam a medida que filtros começam a detectá-las.

As pesquisas citadas acima apontam um grande dinamismo do comportamento dos spammers, o que é um fator agravante para os servidores de e-mails que gastam cada vez mais recursos no combate às mensagens não legítimas [Pathak et al. 2009]. Quanto aos prejuízos que os spams podem causar, o estudo de [Dan Twining 2004] verifica que spammers são prejudiciais para o desempenho de servidores de e-mail, uma vez que o tratamento de spams podem congestionar a chegada de e-mails válidos para os usuários finais.

A pesquisa de [Dan Twining 2004] se relaciona ao presente artigo, porém aqui busca-se identificar e analisar os usuários que podem prejudicar o desempenho dos provedores de e-mail. Além disso, este trabalho também se baseia em métricas de redes complexas, como a popularidade e a conectividade, afim de verificar a existência de usuá-

rios focos de spam da rede de informações formada, onde os remetentes e destinatários representam os nós e a relação entre eles determinada pelo envio de e-mails são os *links*.

Segundo [Easley and Jon 2009], a popularidade de uma rede complexa é representa por uma lei de potência [M. E. J. Newman 2006]. Quando uma função decresce a medida que um valor K é elevado, tal função segue uma lei de potência. Em outros termos, poucos usuários possuem alta frequência de utilização da rede, e quando essa frequência de uso diminui a quantidade de usuários aumenta, mostrando que "poucos usuários utilizam muitos recursos da rede". Além da popularidade, a conectividade entre os remetentes e destinatários também será analisada para os spammers da rede.

3. Metodologia de Caracterização

Esta Seção apresenta a metodologia de caracterização que delineia o processo de análise do comportamento dos usuários de e-mail, quantificando e qualificando os registros do log gerado pelo filtro de spam de um provedor de serviços de correio eletrônico. Entender as características do comportamento dos remetentes e destinatários de spams é uma tarefa que pode contribuir para o desenvolvimento e evolução de técnicas para detectar spammers, possibilitando assim estabelecer uma melhoria nos serviços de e-mails.

O conjunto de dados reais utilizado neste estudo foi coletado na infraestrutura de um provedor de e-mails através do filtro antispam denominado "*InterScan Messaging Security Suite 7.0 for Windows*" [Trend Micro 2007]. Este filtro foi desenvolvido pela empresa *Trend Micro* a qual possui grande conceito em segurança na troca de informações digitais [NSS Labs 2010]. O log contém o tráfego de e-mails que chegaram ao provedor e foram considerados maliciosos (ou não legítimos) e, portanto, foram bloqueados pelo filtro de spam. O conteúdo deste log foi coletado por um período de 2 meses (julho e agosto de 2010) e possui informações tais como: os remetentes e destinatários das mensagens, IPs e domínios dos usuários, data e hora, classificação do spam, bem como outros dados do cabeçalho da mensagem. É válido ressaltar que o log está anonimizado não sendo possível identificar tais usuários (remetentes ou destinatários) na sociedade.

3.1. Visão Geral da Carga de Trabalho

Inicialmente foi realizada uma contabilização geral do conteúdo do log. No período de 2 meses consecutivos, aproximadamente 6 milhões dos e-mails que chegaram ao provedor foram considerados maliciosos e, por isso, foram bloqueados pelo filtro de spam. Esses e-mails contêm cerca de 400 mil remetentes distintos, os quais estão relacionados a aproximadamente 37 mil domínios diferentes. Pode ser observado de acordo com a quantidade de remetentes e domínios, a proporção de 1 domínio para cada 10 remetentes distintos, o que mostra uma quantidade elevada de spammers distribuídos em poucos domínios. Tais valores, podem ser justificados por estudos anteriores como [Gomes et al. 2007] e [Guerra et al. 2010] que apontam a existência de um comportamento dinâmico dos spammers, pois, eles normalmente alteram a identificação dos remetentes ou dados do corpo da mensagem, mesmo permanecendo no mesmo domínio.

O provedor de e-mail possui cerca de 4.000 contas de usuários distintos, sendo estes distribuídos nos 18 domínios diferentes gerenciados pelo provedor de serviços. Novamente podemos observar uma grande concentração de usuários receptores de spams para um mesmo domínio. Além disso, tendo em vista que tais usuários em média receberam

aproximadamente 1.000 spams no período de 2 meses da coleta do log, foi possível verificar a existência de usuários focos, ou seja, usuários que recebem uma grande quantidade de spams em comparação a média geral. Particulamente, existe um único destinatário que recebeu mais de 100 mil spams no período, enquanto outros receberam apenas uma ou duas mensagens classificadas como spam.

Tais remetentes e destinatários podem ser responsáveis por prejudicar o desempenho de provedores de e-mails, uma vez que uma grande quantidade de spams se concentram em um número restrito de usuários de e-mail. Uma visão geral da distribuição dos remetentes e destinatários de spam são apresentados na Tabela 1. Contabilizando a quantidade de domínios nos 2 meses, é possível observar que dos 37 mil domínios distintos dos remetentes, cerca de 10 mil domínios se repetem entre os meses de julho e agosto.

Tabela 1. Contabilização dos remetentes e destinatários distintos de Spams

Log	Remetentes		Destinatários		# E-mails
	# Usuários	# Domínios	# Usuários	# Domínios	
Julho	285.028	25.799	4.129	18	2.404.025
Agosto	139.771	20.853	4.070	18	3.358.509
				Total de E-mails	5.762.534

Para modelar a rede complexa os remetentes foram identificados através dos IPs *Senders* contidos no campo IP de cada registro, e os destinatários foram identificados pelos endereços de e-mails contidos no campo *Receivers*. Os remetentes foram identificados através de seus IPs devido ao seu comportamento dinâmico (vide Seção 2). Além desses campos utilizados para identificação da rede complexa, o campo classificação do spam também é relevante, pois, é responsável por informar a gravidade do spam, de acordo com configurações pré-definidas pelo filtro antispam da *Trend Micro*. A Figura 1 apresenta a estrutura do campo classificação presente nos registros do log analisado e em seguida, apresenta-se a definição de cada campo desta classificação:

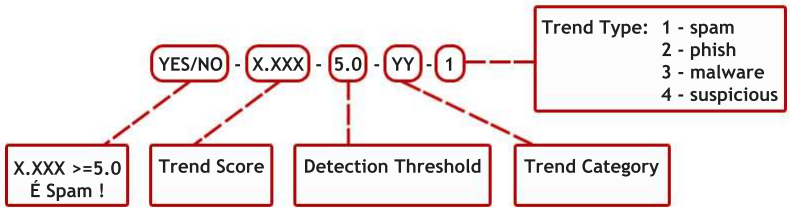


Figura 1. Campo Classificação do Spam

- O primeiro campo informa se o e-mail foi considerado spam (*Yes* ou *No*).
- *Trend Score*: representa a "pontuação" do spam utilizando regras pré-definidas.
- *Detection Threshold*: corresponde ao *Administration Console Settings* configurado para o filtro de spam; este pode possuir os níveis baixo (4.0), médio (5.0) e alto (6.0); para a coleta deste log foi utilizada a configuração com o nível médio.
- *Trend Category*: define qual categoria o spam pertence; não é utilizado para esta coleta, ou seja, todos spams são de uma mesma categoria.
- *Trend Type*: tem o valor 1 (spam) como padrão; os outros possíveis valores são: 2-phish, 3-malware e 4-suspicious; registros com estes últimos valores não foram capturados nessa coleta.

Segundo as regras definidas no *Administration Console Settings* do filtro de antispam, sempre que o *Trend Score* for maior ou igual ao *Detection Threshold* temos um e-mail considerado spam. Com essa classificação dos spams foi possível verificar quais são os e-mails mais "graves", ou seja, aqueles e-mails que possuem maior *Trend Score*.

Após a contabilização geral, foram realizadas análises quantitativas nos dados da rede formada pelos usuários presentes no log. As distribuições de frequência acumulada e probabilidade dos usuários (remetentes e destinatários) foram calculadas para a análise da popularidade e da conectividade dos spams na rede, afim de identificar os usuários focos de spam. Na Seção 4 serão apresentadas essas análises.

3.2. Seleção e Classificação dos dados da Carga de Trabalho

Para obter uma melhor visualização dos dados e identificação das características dos usuários, a rede de remetentes e destinatários de spams foi analisada separada por semanas. Para isso, foram selecionadas 4 semanas entre julho e agosto consideradas quantitativamente relevantes para o estudo, ou seja, as semanas em que houve uma maior quantidade de e-mails trocados entre os usuários desse tipo de serviço. A análise dessas semanas foram realizadas considerando os dois pontos de vista:

- *Remetentes* de spams: registros agrupados por cada IP *sender* distinto, contabilizando a quantidade de destinatários para cada IP; este conjunto de dados representa todos remetentes de spams presentes na rede.
- *Destinatários* de spams: registros agrupados por cada destinatário distinto, contabilizando a quantidade de spams para cada destinatário; este conjunto de dados representa todos os destinatários de spams presentes na rede.

Durante a organização dos registros foram armazenados também os valores máximo, mínimo e médio do *Trend Score* dos spams trafegados na rede. A partir desses valores foi possível identificar a relação entre a quantidade de spams recebidos e enviados pelos usuários com o valor médio do *Trend Score*, o que representa a gravidade do spam que foi filtrado como citado na Subseção anterior. Com isso é possível identificar usuários que recebem ou disseminam spams considerados mais graves e que, por isso, podem causar um dano maior tanto ao usuário destinatário da mensagem quanto ao provedor de serviços de correio eletrônico.

4. Resultados

Esta Seção apresenta e discute os principais resultados encontrados na caracterização da rede de remetentes e destinatários de spams em estudo. Inicialmente apresentamos a análise das métricas de redes complexas relevantes para o artigo e, em seguida, é apresentada uma classificação realizada a partir dos usuários de trocas de e-mail (remetentes e destinatários). Então, foi realizada as análises dos dados separadamente para cada tipo de classificação dos usuários.

4.1. Seleção das métricas da teoria de Redes Complexas

A partir da rede formada pelos usuários de troca de e-mails, foi possível analisar a popularidade e a conectividade dos usuários, porém, a conectividade foi analisada apenas para os remetentes, pois, como vimos na Seção 2 os spams não possuem reciprocidade.

Através dessas métricas podemos identificar a existência dos usuários que mais enviam e que mais recebem spams na rede, ou seja, usuários mais "populares" na rede. Estes usuários podem prejudicar o desempenho do serviço de e-mails ao disseminarem grande quantidade de spams.

Como apresentado na Seção 2 a popularidade é representada por uma lei de potência [M. E. J. Newman 2006], ou seja, poucos usuários possuem alta frequência de utilização da rede, e quando essa frequência de uso diminui a quantidade de usuários aumenta. Assim como a popularidade, a conectividade dos usuários na rede também é uma importante métrica, pois, através dela, pode-se avaliar se a disseminação de conteúdo malicioso na rede ocorre a partir dos usuários com alta conectividade.

4.2. Análise da rede de Remetentes e Destinatários de Spams

Com o intuito de entender a propagação dos spams na rede formada a partir dos usuários de troca de e-mails, a análise dos usuários da rede foi realizada considerando os dois pontos de vista (Remetentes de spams e Destinatários de spams) citados na Seção 3.2. Nas Subseções a seguir são apresentadas para cada grupo definido acima, as análises dos usuários da rede e das métricas de redes complexas selecionadas.

4.2.1. Análise dos IPs Remetentes de Spam

A análise dos IPs Remetentes foi realizada para cada uma das 4 semanas selecionadas, (vide Seção 3.2). Contabilizando os spams enviados de cada semana e relacionando-os aos IPs Remetentes responsáveis, na semana 1 cerca de 4% dos IPs Remetentes foram responsáveis pelo envio de 50% dos spams da rede, sendo 5% na semana 2 e 6% na semana 3 e 4, apresentando uma média de 5% durante as 4 semanas.

Observa-se que a quantidade de remetentes responsáveis pelo envio de uma grande quantidade de spams não varia muito de uma semana para outra. Tal fato pode ser justificado pela contabilização dos dados ter sido realizada com base no campo IP dos remetentes de spam. Como visto na Seção 2 desse artigo, spammers possuem um comportamento dinâmico. Porém, ao analisar o volume de tráfego gerado pelo IP foi possível identificar usuários com remetentes distintos partindo de um mesmo IP em semanas diferentes. Tal ocorrência é apresentada na Tabela 2 a seguir.

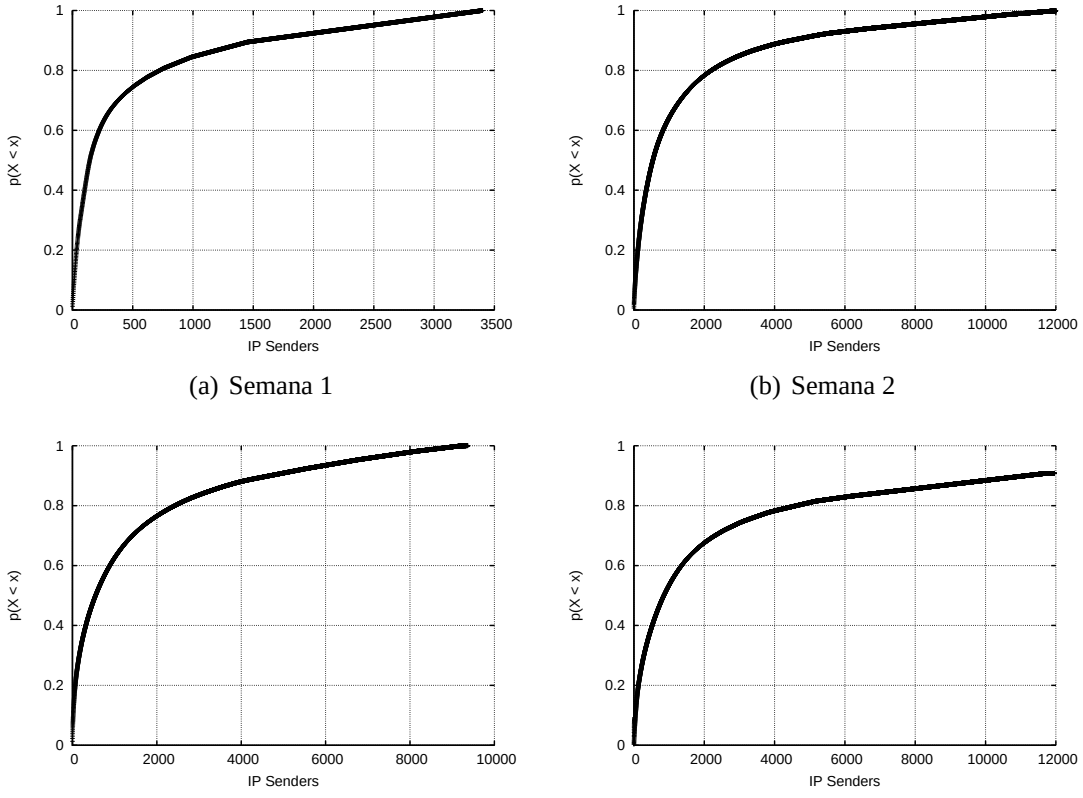
Tabela 2. Modificação dos spammers de um mesmo IP durante as semanas.

Semana	IPs	Remetentes	# Spams Enviados
1	63.201.115.98	rmt23987@dmn13.com.br	20
2	63.201.115.98	rmt134298@dmn13.com.br	193
3	63.201.115.98	rmt237893@dmn13.com.br	381
4	63.201.115.98	rmt365278@dmn13.com.br	386

Para analisar a conectividade dos spammers da rede foram contabilizados os e-mails trocados entre os usuários da rede. Verificou-se que para o período analisado existe uma quantidade alta de remetentes de spam (400 mil) para um número relativamente baixo de destinatários (4 mil). Avaliando a quantidade de spams enviados para os usuários finais (aproximadamente 6 milhões de spams), é possível identificar se tais spammers possuem alta conectividade entre os usuários destinatários. Para isso, verificamos que dentre os 6 milhões de spams bloqueados pelo filtro, estes se espalham regularmente

entre os destinatários da rede, onde em média os usuários destinatários recebem cerca de 1.700 e-mails. Isso mostra que os *senders* atingem constantemente todos os destinatários da rede. Entretanto, foi visto a existência de usuários que concentram mais spams que outros, chegando a um máximo de 100 mil spams para único usuário destinatário, tais usuários podem ser considerados usuários focos.

Para analisar a popularidade dos nós (usuários) dessa rede foram calculadas as CDFs [1] e as PDFs[2] dos usuários separados por semanas. Na Figura 2 é possível visualizar a disposição da curva CDF dos IPs Remetentes de spams durante as 4 semanas. Pode ser visto nas CDFs que a curva da distribuição está sempre bem acentuada e próxima ao eixo y, mostrando que os IPs Remetentes de spams se concentram em poucos usuários, mais precisamente menos que 2 mil spammers. Com isso, observa-se que o envio de spam de fato se concentra em poucos IPs Remetentes, mostrando uma maior popularidade de certos spammers que podem ser considerados focos da disseminação de spams na rede.



(a) Semana 1 (b) Semana 2

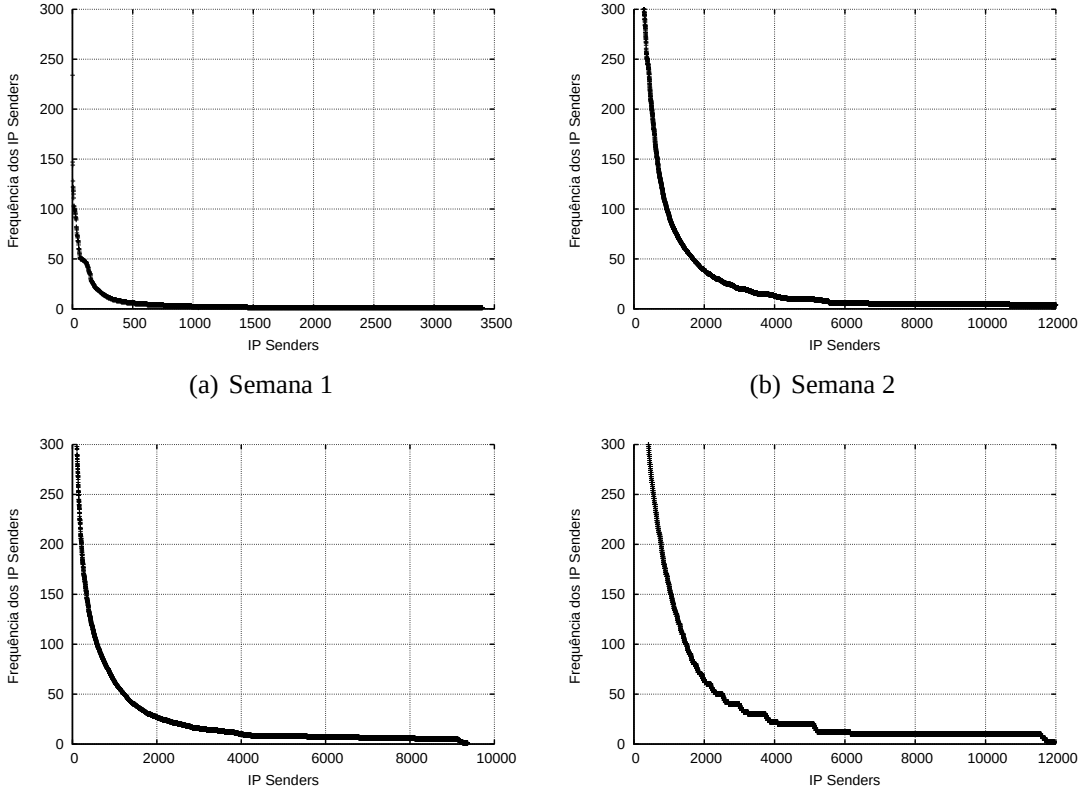
(c) Semana 3 (d) Semana 4

Figura 2. CDFs para os *IPs Remetentes* durante as 4 semanas

Além das CDFs, que apresenta a distribuição de frequência acumulada dos remetentes na rede, podemos também análisar as PDFs da rede formada. A Figura 3 apresenta a disposição das curvas das PDFs dos IPs Remetentes de spam. Pode ser visualizado nas PDFs um comportamento típico de uma lei de potência [M. E. J. Newman 2006], pois implica que poucos remetentes enviam spam para uma grande quantidade de usuários finais,

[1] *Cumulative Distribution Frequency (CDF)*
[2] *Probability Distribution Frequency (PDF)*

enquanto a maioria dos spammers envia mensagens maliciosas para poucos destinatários. Em outras palavras, poucos nós da rede de remetentes de spam têm alta popularidade entre os usuários.



(a) Semana 1 (b) Semana 2

(c) Semana 3 (d) Semana 4

Figura 3. PDFs para os *IPs Remetentes* durante as 4 semanas

Para melhor visualizar tal fato, foram selecionados dentre os IPs Remetentes aqueles que enviaram 50% dos spams da rede para cada semana selecionada. Estes usuários foram denominados como *Heavy Senders*. É possível ver na Tabela 3 que mesmo com a quantidade de remetentes aumentando como mostra na semana 2, atingindo aproximadamente 12 mil IPs, o percentual dos que enviam 50% dos spams não é superior a 8%, mostrando que estes IPs podem estar associados a usuários focos de envio de spams.

Tabela 3. Visão geral dos remetentes de spam durante as semanas

Semana	**# IPs Remententes**	**# Heavy Senders (enviam 50% spams)**	**% Remetentes que enviam 50% dos spams**
1	3.404	150	4,41%
2	12.167	558	4,60%
3	9.369	548	5,87%
4	11.975	868	7,27%

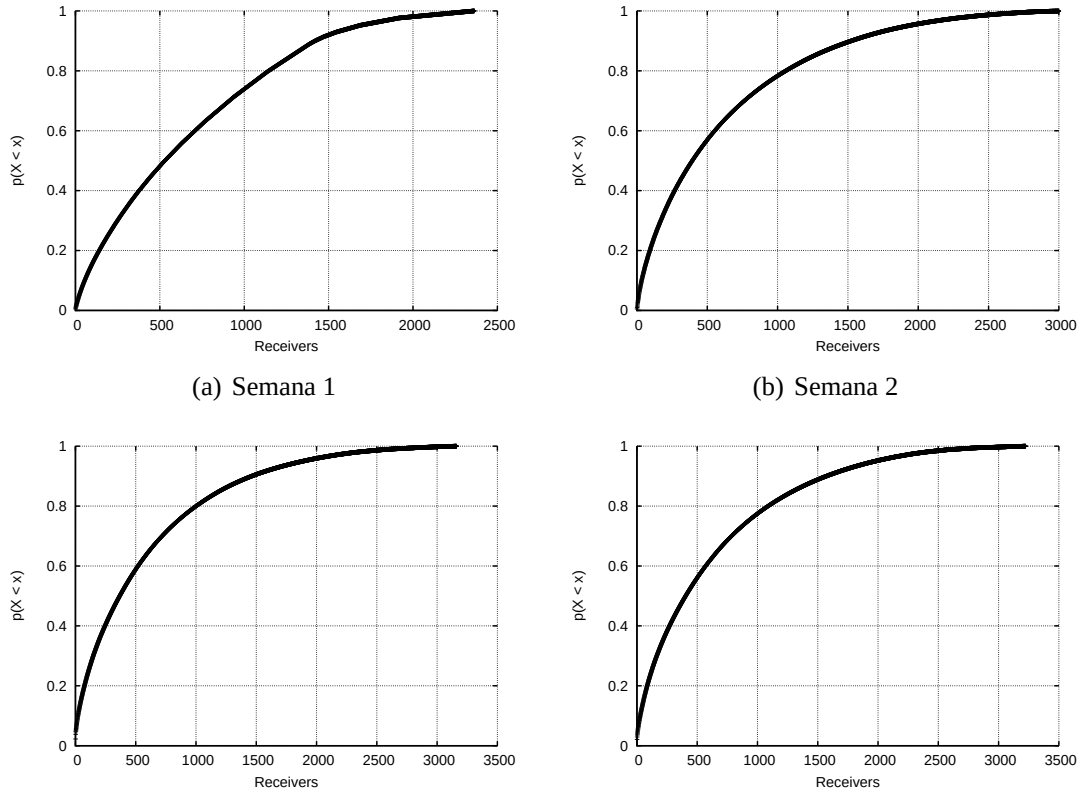
A descoberta de dados desse tipo, como a concentração de envio de spam em poucos remetentes é útil para possíveis melhorias na rede de correio eletrônico, pois, ao tratar tais usuários focos de spam, pode-se evitar o desperdício de recursos para combater os spammers e consequentemente melhorar o desempenho de serviços de e-mail. Isso pode acontecer também com destinatários focos de spams, que são analisados a seguir.

4.2.2. Análise dos Destinatários de Spam

A análise dos destinatários de spams também foi realizada para cada uma das 4 semanas selecionadas. Ao contabilizar a quantidade de spams enviados para um determinado usuários final, encontrou-se focos de destinatários em cada semana analisada. Na primeira semana, 22% dos destinatários foram alvos de 50% do total de spams, nas semanas 2 e 4, apenas 13% dos destinatários foram alvos deste mesmo percentual de spams enviados, e na terceira semana 12% dos usuários receberam metade dos spams.

Os valores mostram uma variação maior ocorrendo da semana 1 para semana 2. Porém, tal valor pode ser justificado devido à semana 1 possuir menor quantidade de usuários finais do que a semana 2, e desta forma, os spams se concentram em um número menor de destinatários. Para as outras semanas (3 e 4) percebe-se que há uma concentração de muitos spams para poucos usuários destinatários.

A popularidade de certos destinatários de spam, assim como os remententes, também se concentram em poucos usuários. A Figura 4 apresenta as curvas CDF dos destinatários de spam em cada semana destacada para o estudo. Observa-se que essas curvas se concentram sempre próximo ao eixo y, o que representa menores quantidades de usuários. Isso mostra uma maior popularidade de certos destinatários entre os usuários finais de spams. Tal fato pode ser observado também na Figura 5 que apresenta as curvas PDFs dos destinatários de spams durante as 4 semanas selecionadas.

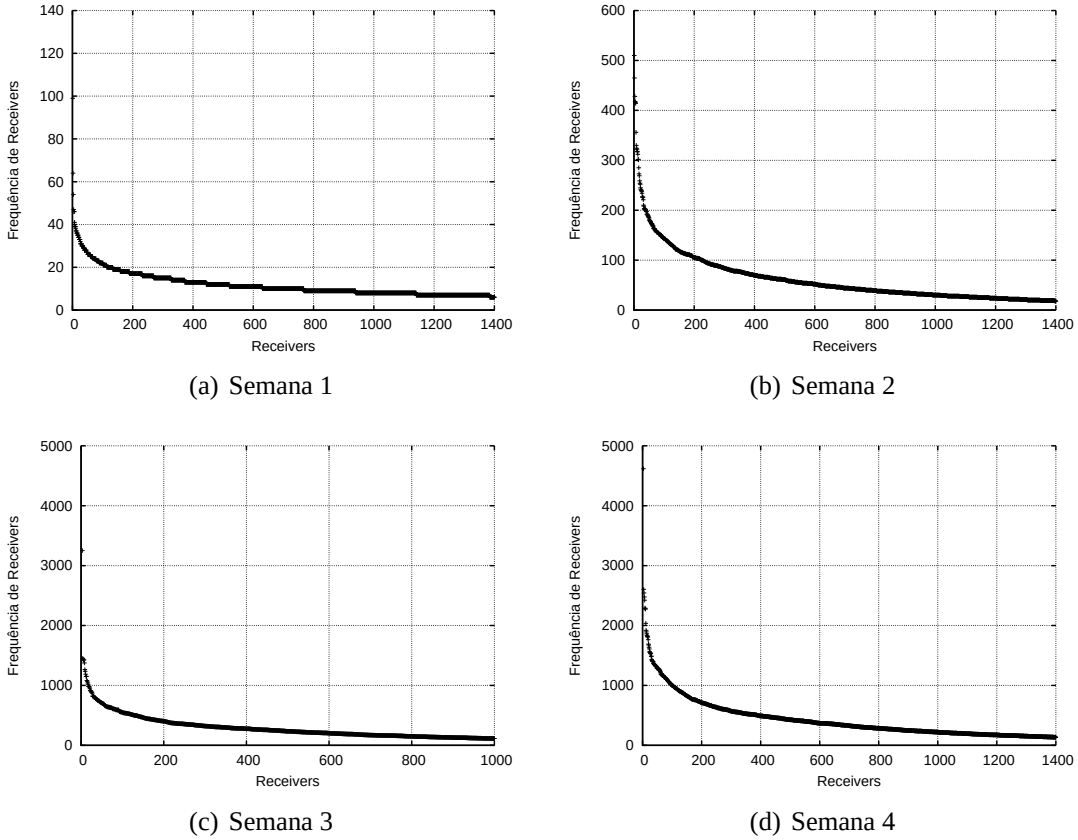


(a) Semana 1 (b) Semana 2

(c) Semana 3 (d) Semana 4

Figura 4. CDFs para os destinatários durante as 4 semanas

Assim como nos remetentes analisados na Subseção anterior, as PDFs dos destinatários de spam também apresentam um comportamento típico de uma lei de potência. Isto



(a) Semana 1 (b) Semana 2

(c) Semana 3 (d) Semana 4

Figura 5. PDFs para os destinatários durante as 4 semanas

aponta que poucos usuários receberem muito spam, enquanto a maioria dos usuários destinatários recebem menor quantidade de spams. Pode-se concluir também que poucos nós da rede de destinatários de spam têm alta popularidade entre os IPs Remetentes. Na Tabela 4 tal fato pode ser melhor visualizado. Observa-se que apesar de ter uma quantidade de destinatários maior como mostra na semana 4, o percentual dos usuários que recebem pelo menos 50% dos spams não é superior a 14% nas semanas 2 a 4. Assim como foi feito para os remetentes, estes usuários foram denominados como *Heavy Receivers*.

Tabela 4. Visão geral dos Destinatários de Spam durante as semanas

Semana	**Total *Receivers***	**Quantidade Heavy *Receivers* (recebem 50% spams)**	**% *Receivers* que recebem 50% dos spams**
1	2.365	150	22,41%
2	3.036	558	13,18%
3	3.163	548	11,70%
4	3.226	868	12,59%

Analisando pela visão do administrador de redes, a descoberta de dados desse tipo, como a concentração de envio de spam para poucos destinatários pode ser útil na identificação de usuários que atraem spams. Ao serem identificados e tratados devidamente, poderia facilitar o serviço do provedor de e-mails no bloqueio aos spams que tais usuários finais iriam atrair para suas caixas de entrada. Como consequência, seria possível alcançar um melhor desempenho nos serviços de e-mails, não congestionando o envio de mensagens válidas, como apresentado na Seção 2 na pesquisa de [Dan Twining 2004].

4.3. Movimentação dos Spams na rede de Remetentes e Destinatários

Como visto na Seção 2, as pesquisas apontam um grande dinamismo do comportamento dos spammers. Tal característica também está presente nos remetentes de spam analisado neste trabalho. Pode ser visualizada para cada uma das 4 semanas selecionadas para esta pesquisa, a alta movimentação dos spammers na rede. A cada semana os remetentes de spam são alterados continuando poucos com o mesmo IP. Na Tabela 5 é apresentada a movimentação de entrada e saída de IPs durante este período de 4 semanas.

Tabela 5. Evolução Temporal dos remetentes durante as semanas

Semana	# IPs que continuaram	# IPs que saíram	# IPs novos
1	0	0	0
2	3	145	555
3	147	411	401
4	0	548	868

Observa-se que a quantidade de IPs novos ou que saem é sempre maior que a quantidade de IPs que continuam nas semanas seguintes. Isto pode ser visualizado durante as semanas 2, 3 e com ênfase na semana 4, quando todos 568 IPs saíram e 868 novos IPs entraram. Tais dados mostram a constante alteração dos remetentes ao enviarem spams para seus destinatários.

4.4. Classificação dos Spams da rede de Remetentes e Destinatários

Outro fator interessante da análise dos spammers neste estudo é a classificação do spam que foi enviado. Esta classificação, como visto na Seção 3.1, é realizada pelo filtro antispam utilizado pelo provedor de e-mails no qual o log foi coletado. Através desse campo de classificação é possível identificar a gravidade de cada spam recebido pelo provedor. Como citado anteriormente na Seção 3.2, foram armazenados os valores máximos, mínimos e as médias dos *Trend Scores* dos e-mails da rede de remetentes e destinatários em estudo. Com esses valores verificou-se a relação entre a quantidade de spams recebidos e enviados pelos usuários com o valor médio do *Trend Score*.

Na Tabela 6 são apresentados os valores médios dos *Trend Scores* contabilizados de forma geral para a rede. Ela mostra também a média dos *Trend Scores* para os *Heavy* spams e a média dos *Trend Scores* para "*Light*" spams, esse último grupo representa os demais usuários da rede que não foram considerados focos de spam.

Tabela 6. *Trend Score* IPs *Senders*

Semana	Média Geral		Média Heavy		Média Light	
	Senders	***Receivers***	***Senders***	***Receivers***	***Senders***	***Receivers***
1	67,244	35,127	737,077	67,163	36,769	25,705
2	5,605	6,297	33,774	7,104	4,250	6,176
3	48,798	8,870	19,880	27,501	5,059	6,415
4	0,041	14,007	3,282	47,657	0,018	9,186

Pode ser visto que para os *Heavy* spams a média do *Trend Score* é sempre mais alta, como, por exemplo, a semana 4, em que a média geral do *Trend Score* dos *Receivers* é 14,0 e a média dos *Heavy* e *Light receivers* são respectivamente 47,6 e 9,18. Tais dados mostram a maior gravidade dos spams recebidos pelos *Heavy receivers* e enviados pelo *Heavy senders*.

Este é mais um fator importante entre as características da rede de remetentes e destinatários deste artigo, pois como foi visto, a quantidade de *Heavy senders* é pequena (concentra em poucos usuários), mas os spams disseminados por estes usuários são mais graves. Portanto, ao identificar e tratar os usuários que enviam ou recebem spams considerados mais graves, o desempenho de filtros antispam e serviços de e-mail serão diretamente beneficiados.

5. Conclusão

Neste trabalho foi apresentada e aplicada uma metodologia de caracterização da rede de remetentes e destinatários de spams de um provedor de mensagens eletrônicas. A metodologia consiste na análise de um conjunto de dados reais coletados na infraestrutura de um provedor de e-mails através de um filtro antispam, de forma a identificar características relevantes no comportamento de spammers e destinatários de spams.

A análise das métricas de redes complexas, popularidade e conectividade, permitiu identificar que alguns grupos de remetentes e destinatários de spams possuem alta popularidade na rede. Além disso, foi visto que os remetentes da rede de spam tem uma conectividade alta, ou seja, os spammers se conectam a vários destinatários. Quanto às características dos spams enviados, analisando o campo de classificação de spams existente nos registros, percebeu-se que spams enviados pelos *Heavy senders* e recebidos pelos *Heavy receivers* são considerados mais graves pelo filtro de spam. Logo, apenas com o esforço de identificar esses remetentes e destinatários focos de spam, é possível alcançar resultados significativos para contribuição na economia de recursos e melhoria do desempenho dos provedores de serviços de e-mail durante o tratamento de spams.

Como trabalho futuro pretende-se analisar os impacto da eliminação de usuáros focos de spams que são identificados com a metodologia apresentada neste artigo. Além disso, com posse do log de e-mails legítimos coletado pelo mesmo provedor no mesmo período, pretende-se sobrepor as duas redes formadas (rede de spams e de e-mails legítimos) afim de identificar o motivo porque alguns usuários são mais atacados do que outros e, também, se isso é influenciado por como os spammers decidem os seus alvos, bem como se os usuários se tornam vulneráveis. Tais análises podem ser relevantes para melhoria no desempenho de serviços de mensagens eletrônicas.

Referências

C.E Shannon (1948). A Mathematical Theory of Communication. 27:379–423.

Dan Twining, Matthew M. Williamson, M. J. F. M. M. R. (2004). Email prioritization: reducing delays on legitimate mail caused by junk mail. In *Distributed Computing Systems 2009 ICDCS '09 29th IEEE International Conference on*.

Easley, D. and Jon, K. (2009). *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*. Cambridge University Press, 1nd edition.

Gomes, L. H., Almeida, R. B., Bettencourt, L. M. A., Almeida, V., and Almeida, J. M. (2005). Comparative Graph Theoretical Characterization of Networks of Spam and Legitimate Email. In *Proceedings of the Second Conference on Email and Anti-Spam - CEAS 2005*, Stanford, CA, USA. CEAS.

Gomes, L. H., Almeida, V. A. F., Almeida, J. M., Castro, F. D. O., , and Bettencourt, L. M. A. (2009). Quantifying Social And Opportunistic Behavior In Email Networks. *Advances in Complex Systems*, 12(1):99–112.

Gomes, L. H., Cazita, C., Almeida, J. M., Almeida, V., and Meira, J. W. (2007). Workload models of spam and legitimate e-mails. *Perform. Eval.*, 64(7-8):690–714.

Gomes, L. H., Cazita, C., Almeida, J. M., Almeida, V., and Meira, Jr., W. (2004). Characterizing a Spam Traffic. In *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pages 356–369, New York, NY, USA. ACM.

Guerra, P. H. C., Guedes, D., Wagner Meira, J., Hoepers, C., Chaves, M. H. P. C., and Steding-Jessen, K. (2010). Exploring the spam arms race to characterize spam evolution. In *Proceedings of the 7th Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, Redmond, WA.

IronPort Email and Web Security (2011). Cisco 2010 annual security report.

Li, F. and Hsieh, M.-H. (2006). An empirical study of clustering behavior of spammers and group-based anti-spam strategies. In *CEAS'06*.

M. E. J. Newman (2006). Power laws, Pareto distributions and Zipfs law. page 28. Department of Physics and Center for the Study of Complex Systems.

Message Labs (2011). Message labs intelligence.

NSS Labs (2010). Consumer anti-malware products group test report.

Nucleus Research (2007). Spam, the repeat offender.

Pathak, A., Jafri, S., and Hu, Y. (2009). The case for spam-aware high performance mail server architecture. In *Distributed Computing Systems, 2009. ICDCS '09. 29th IEEE International Conference on*, pages 155 –164.

Project, T. A. S. (2010). Spamassassin.

Pu, C. and Webb, S. (2006). Observed trends in spam construction techniques: A case study of spam evolution. In *CEAS*.

Ramachandran, A. and Feamster, N. (2006). Understanding the network-level behavior of spammers. *SIGCOMM Comput. Commun. Rev.*, 36:291–302.

Symantec (2011). State of spam e phishing-a monthly report. Technical report.

Trend Micro (2007). InterScan Messaging Security Suite.

Caracterização das Propriedades Dinâmicas da Estrutura da Rede Sobreposta do SopCast *

Kênia Carolina Gonçalves[1], Alex Borges[2], Jussara Almeida[1], Ana Paula Silva[2], Humberto Marques-Neto[3], Sérgio Vale Campos[1]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais
Belo Horizonte – MG – Brazil

[2]Departamento de Ciência da Computação – Universidade Federal de Juiz de Fora
Juiz de Fora – MG – Brazil

[3]Pontifícia Universidade Católica de Minas Gerais
Belo Horizonte – MG – Brazil

keniacarolina@dcc.ufmg.br, alex.borges@ufjf.edu.br, jussara@dcc.ufmg.br

anapaula.silva@ufjf.edu.br,humberto@pucminas.br,scampos@dcc.ufmg.br

Resumo. *Sistemas Peer-to-Peer (P2P) de transmissão de vídeo ao vivo estão se tornando cada vez mais populares. Apesar de vários estudos sobre o comportamento dos usuários e sobre as propriedades da rede sobreposta formada em tais sistemas, pouco é o conhecimento sobre como a estrutura desta rede evolui com o tempo durante uma transmissão de conteúdo. Nesse trabalho apresentamos uma caracterização das propriedades dinâmicas da estrutura desta rede sob dois pontos de vista: características individuais dos nós e características da rede como um todo. Para isso, foram utilizadas métricas de redes que avaliam a centralidade do nó e a topologia da rede. Os resultados mostram três perfis de nós com métricas de centralidade distintas na rede e pouca variabilidade dos nós entre esses perfis. Além disso, as medidas da topologia da rede também tendem a permanecer estáveis. Finalmente, os resultados obtidos neste trabalho podem ser usados por pesquisas futuras com simulações mais realistas com estratégias otimizadas que representem o dinamismo real da rede sobreposta.*

Abstract. *Peer-to-Peer video streaming systems are becoming increasingly popular. Despite several studies on network overlay and user behavior, little is still known about how the overlay network evolves during a live streaming transmission. In this work we present a network dynamic characterization under two different perspectives: the individual nodes and the overlay network characteristics. We have used complex network metrics such as node centric metrics and network topology properties. Our results show that network nodes may be grouped into 3 distinct clusters. Moverover, network topology metrics tend to be stable during the transmission. The characterization we show may be used as a substrate to future research and system simulation, as it can be used to generate a more realistic synthetic workload.*

*Esta pesquisa é parcialmente financiada pelo Instituto Nacional de Ciência e Tecnologia para a Web - INCTWeb (MCT/CNPq 573871/2008-6), CNPq, FAPEMIG e CAPES.

1. Introdução

A popularidade de aplicações de transmissão de vídeos na Internet cresce a cada dia. As pesquisas recentes mostram que em 2013 o número de usuários desse tipo de aplicação pode chegar a 83 milhões [Borges et al. 2010a]. Uma das arquiteturas de rede utilizadas por essas aplicações na transmissão de vídeo ao vivo pela Internet é a rede Par-a-Par (P2P). O SopCast[1], o PPLive[2] e o PPStream[3] são exemplos deste tipo de aplicação. O SopCast tem apresentado uma grande popularidade e em 2010 teve um maior volume de tráfego mundial se comparado com as aplicações PPLive e PPStream [Google 2010]. Por isso, foi escolhido para ser analisado neste trabalho.

No SopCast, a transmissão de vídeo ocorre em uma rede P2P, na qual os clientes recebem o conteúdo transmitido por outros participantes, criando assim o que é conhecido como rede sobreposta, ou seja, uma rede lógica de transmissão de dados. Nesta rede, os nós representam clientes e uma aresta direcionada entre dois nós representa a troca de conteúdo entre clientes, ou seja, eles estabelecem uma parceria. Estas parcerias são estabelecidas e desfeitas dinamicamente em resposta tanto ao comportamento dinâmico dos pares quanto ao desempenho (qualidade do vídeo recebido) observado por um par.

Embora existam na literatura alguns trabalhos que tenham caracterizados diferentes aspectos de aplicações de transmissão de vídeo ao vivo baseada em redes P2P [Hei et al. 2007, Borges et al. 2010b, Huang et al. 2008], incluindo alguns estudos sobre as propriedades da rede sobreposta criada em tal aplicação [Silverston and Fourmaux 2007, Oliveira et al. 2009], nenhum deles caracterizou as propriedades dinâmicas desta rede. Dentre alguns aspectos que já foram analisados destacam-se o número e a duração das parcerias estabelecidas por um par [Tang et al. 2009, Stutzbach et al. 2008]. Embora relevantes, estes aspectos fornecem apenas uma visão parcial sobre como a rede de parcerias evolui ao longo do tempo. Uma outra questão em aberto é o quão dinâmica é a estrutura da rede ao longo do tempo de uma transmissão. Uma rede muito dinâmica pode ocasionar em problemas de imprevisibilidade, dificuldade de manutenção, alocação de recursos e entre outros.

O objetivo desse trabalho é prover uma caracterização inicial das propriedades dinâmicas da rede sobreposta de um sistema de transmissão P2P de vídeo ao vivo, isto é, de como as características dos nós e da rede evoluem com o tempo. Essa caracterização busca apresentar as características individuais dos nós e da topologia da rede. Para analisar os nós individualmente foram utilizadas métricas de centralidade que permitem entender o comportamento do nó dentro da estrutura da rede. São elas: Grau, *Betweenness* e *Closeness*. Já para a topologia da rede foram utilizadas as seguintes métricas para compreender a estrutura da rede no qual os nós estão inseridos: Diâmetro, Coeficiente de Agrupamento, Grau Máximo e Caminho Minímo Médio.

A caracterização foi realizada com base nos resultados de experimentos que utilizaram entre 200 e 465 clientes SopCast executados no PlanetLab [4]. Estes clientes foram conectados a um canal de grande populatidade na China, o CCTV-1, e monitorados durante 40 minutos em um horário de pico local (20 horas). Os resultados encontra-

[1]www.sopcast.org
[2]www.pptv.com
[3]www.ppstream.com
[4]www.planet-lab.org

dos mostram que os nós podem ser agrupados em três perfis com valores de métricas de centralidade distintos na rede: *Centralidade Alta* (CA), *Centralidade Intermediária* (CI) e *Centralidade Baixa* (CB). O perfil *CA* contém nós mais centrais dentro da estrutura da rede com medidas altas para as métricas avaliadas. Já os nós do perfil *CB* são mais periféricos na visão da rede sobreposta coletada com baixos valores de centralidade. Sendo os nós do perfil *CI* aqueles com valores intermediários entre o *CA* e o *CB*. Apesar de terem sido encontrados três perfis de nós diferentes, os nós apresentam pouca dinamicidade entre os perfis. O perfil individual do nó é definido pela estrutura da rede em que ele se conecta. Além disso, os nós tendem a manter em média 70% das parcerias estabelecidas ao longo do tempo. As características estruturais da rede não apresentam variações significativas ao longo do tempo em relação ao Diâmetro e Caminho Mínimo Médio. Os resultados dessa caracterização podem ser explorados por pesquisas futuras em simulações mais realistas com estratégias otimizadas que representem a real dinâmica da rede.

O restante desse trabalho está organizado da seguinte forma: a Seção 2 discute os trabalhos relacionados da literatura e a Seção 3 apresenta o processo de execução dos experimentos. Já a Seção 4 traz um *overview* das métricas de centralidade dos nós e da estrutura da rede que foram utilizadas neste trabalho. Finalmente, a Seção 5 apresenta os resultados obtidos seguida pela Seção 6 com a conclusão desses resultados.

2. Trabalhos Relacionados

Sistemas P2P de transmissão de vídeo ao vivo já foram estudados sob diferentes pontos de vista ao longo dos últimos anos. Um dos principais objetivos é entender o comportamento da rede e de seus usuários a fim de melhorar o desempenho dos sistemas, propondo, por exemplo, novos algoritmos para o estabelecimento de uma rede sobreposta que diminua o atraso de difusão de conteúdo em uma transmissão.

Em Hei et. al. [Hei et al. 2007], os autores mostram que a aplicação PPLive atinge um bom desempenho usando a Internet para a transmissão de vídeo ao vivo. Além disso, eles caracterizam o comportamento do usuário, bem como as informações que são trocadas entre os nós. No entanto, o dinamismo da rede sobreposta não foi explorado. Em outro trabalho dos mesmos autores [Hei et al. 2008], foi utilizada uma metodologia para analisar o comportamento dos usuários. Esta análise baseou-se no desempenho do sistema e na qualidade percebida pelos usuários finais, sem considerar as características da estrutura topológia da rede sobreposta.

O trabalho de Siverston et. al. [Silverston and Fourmaux 2007] analisa as características do tráfego gerado pelas aplicações SopCast, PPLive, TVAnts e PPStream, utilizando um conjunto de dados coletados durante a Copa do Mundo de 2006. Apesar de analisarem algumas métricas, como a taxa de *upload* e *download*, bem como o tamanho médio dos pacotes trocados, os autores não abordaram as características da rede sobreposta formada para a transmissão de conteúdo e como elas evoluem com o tempo.

Em [Stutzbach et al. 2008], Stutzbach et. al. propõem uma metodologia para analisar a evolução da rede sobreposta em uma aplicação de compartilhamento de arquivos em redes P2P. Porém, neste trabalho é aplicada a metodologia em redes P2P de transmissão de vídeo ao vivo. A contribuição deste trabalho é diferente da apresentada em [Stutzbach et al. 2008] dado que a transmissão ao vivo de vídeo em aplicações

P2P têm características diferentes, como prazo de entrega de pacotes, que influenciam na metodologia de caracterização.

A evolução das propriedades de redes também foi estudada por Watts [Watts and Strogatz 1998] em que ele apresenta o fenômeno do mundo pequeno como um dos resultados de experimentos em redes dinâmicas. Para realizar tal análise Watts usou métricas de redes como o diâmetro, grau dos nós e coeficiente de agrupamento. O livro recente de Easley et. al. [Easley and Kleinberg 2010] também apresenta vários trabalhos que buscam compreender a dinamicidade do crescimento e da redução de clientes em redes populares. No entanto, não é abordado como é a evolução da estrutura de uma rede P2P para a transmissão de vídeo ao vivo.

O trabalho que está sendo apresentado aqui segue o trabalho de Borges et al. [Borges et al. 2010b], onde os autores estudaram o comportamento dos usuários na aplicação SopCast. Os resultados obtidos dos autores foram estendidos neste trabalho, porém novos dados e novos resultados foram gerados sobre a perspectiva da estrutura da rede sobreposta e seu dinamismo. A rede sobreposta do SopCast é estudada pelo autores Tang et. al. [Tang et al. 2009] com *logs* de clientes SopCast executados e monitorados no PlanetLab. Os autores mostram a correlação entre o grau de saída e a média de *upload* em fotografias da rede. Eles mostram a relação entre pacotes de controle e de dado transmitidos entre os parceiros. Os dados utilizados são coletados de 51 nós do PlanetLab com duração de 40 minutos em apenas um dia.

Em comparação com a caracterização de Tang et. al. [Tang et al. 2009] e as outras de sistemas P2P anteriores [Silverston and Fourmaux 2007, Hei et al. 2007, Huang et al. 2008, Tang et al. 2009, Borges et al. 2010b, Stutzbach et al. 2008], neste trabalho foi utilizado um número maior de clientes SopCast (trabalhos anteriores como [Fallica et al. 2008] utilizaram no máximo 70 clientes) na tentativa de se obter uma representatividade maior da rede sobreposta do SopCast. Além disso, este trabalho apresenta uma análise diferente da rede que busca representar o dinamismo das propriedades sob dois pontos de vista: os nós com diferentes perfis de centralidade e a estrutura da rede.

3. Base de Dados

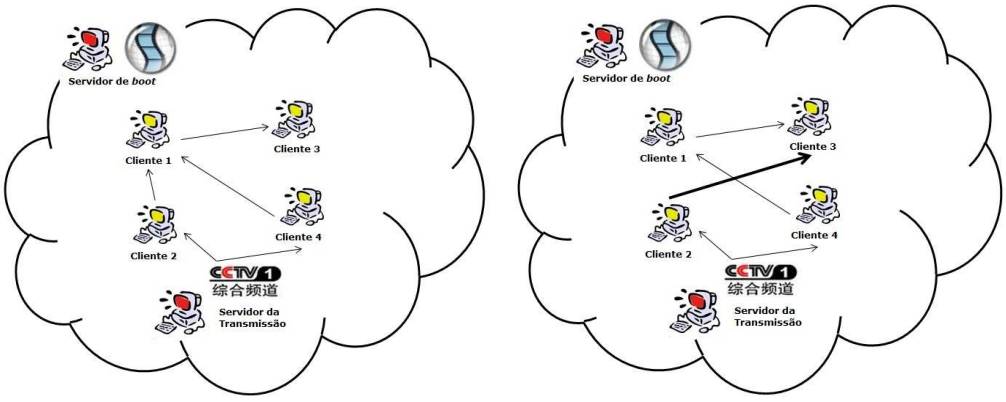
A base de dados utilizada neste trabalho foi obtida através de experimentos realizados no PlanetLab com clientes da aplicação SopCast conectados a um canal aberto, o CCTV-1. Na Seção 3.1 é apresentada um breve descrição sobre a aplicação SopCast e seu funcionamento. Na Seção 3.2 apresentamos a descrição de como os experimentos foram feitos. Os dados utilizados estão sumarizados na Seção 3.3.

3.1. O Sistema SopCast

O SopCast é um das aplicações mais populares de transmissão de vídeo ao vivo que utiliza redes P2P, apresentando um maior volume de tráfego em relação as aplicações PPLive e PPStream em 2010 [Google 2010]. Para que ocorra transmissões de vídeo na rede P2P do SopCast são utilizadas redes sobrepostas. Em cada rede sobreposta ocorre a transmissão de um determinado vídeo, ou seja, trata-se de uma rede lógica formada exclusivamente para a transmissão de conteúdos específicos. A rede sobreposta de transmissão de um vídeo está relacionada na aplicação por *canal* em que o cliente está conectado.

Os clientes SopCast ao usarem a aplicação se conectam a um determinado canal, passando a fazer parte da rede sobreposta que está sendo transmitindo o vídeo. Os canais do SopCast podem ser abertos ou fechados. Um canal aberto é aquele que ele está disponível na aplicação para que qualquer cliente se conecte a rede sobreposta de transmissão. Por outro lado, o canal fechado é aquele que para o cliente ter acesso à rede que está sendo transmitido o vídeo é necessário que ele tenha permissão. Neste trabalho os clientes SopCast foram conectados a um dos mais populares canais abertos da China, o CCTV-1. A China é um dos países com o maior número de clientes SopCast [Google 2010].

As redes sobrepostas para a transmissão do conteúdo de um canal SopCast é composta por um servidor de transmissão ao vivo, servidor de *boot* (*bootstrap*) e clientes(nós) da rede. Neste contexto, o servidor de transmissão ao vivo é a fonte inicial do conteúdo e participa da rede sobreposta como um cliente especial do canal que ele está transmitindo. Já o servidor de *boot* é responsável por manter de forma centralizada o registro dos clientes por rede sobreposta. Quando um novo cliente se conecta a um determinado canal, o servidor de *boot* envia uma lista de parceiros candidatos para esse cliente. Os parceiros são clientes que já estão na rede sobreposta do canal desejado e que podem estabelecer uma parceria para a troca de dados. Uma parceria é, portanto, definida pelo fato de um cliente receber dados de outro cliente. Estas parcerias podem ser estabelecidas e desfeitas dinamicamente de acordo com o comportamento dos clientes e de acordo com a qualidade do vídeo transmitido entre parceiros.



(a) Fotografia A (b) Fotografia B

Figure 1. Representação da rede sobreposta do SopCast em duas fotografias

Para representar a dinamicidade das parcerias, a Figura 1 mostra duas fotografias consecutivas (*A* ocorre antes de *B*) de uma rede sobreposta hipotética formada para a transmissão do canal CCTV-1 no SopCast. Na Figura 1 é possível observar que no momento *B* o *Cliente 1* não possui mais parceria com o *Cliente 2* e que o *Cliente 2* estabelece uma nova parceria com o *Cliente 3* diferentemente do momento *A*. Entender a formação das propriedades dinâmicas da rede sobreposta e como suas parcerias evoluem com o tempo é o principal objetivo desta caracterização.

3.2. Metodologia para Realização dos Experimentos

Os resultados deste trabalho são baseados em um conjunto de E experimentos realizados com o SopCast. Cada experimento foi realizado utilizando um número N de computadores do PlanetLab. Cada computador foi configurado para executar um cliente SopCast

cada um e armazenar todas as trocas de dados realizadas ao longo de uma transmissão. Atuando como clientes SopCast eles se conectaram ao canal aberto da China, o CCTV-1, durante o horário de pico local (20 horas). O CCTV-1 é um canal muito popular da emissora estatal chinesa e possui uma qualidade de transmissão de vídeo (600 kbps) maior do que quase todos os outros canais do SopCast. A metodologia para a realização dos experimentos deste trabalho é composta basicamente por duas etapas: configuração dos computadores do PlanetLab e monitoramento dos mesmos durante a transmissão de vídeo no SopCast.

Durante a etapa de configuração dos computadores não foram impostas quaisquer restrições, sejam de armazenamento, taxa de *upload* e *download*. Todas as versões dos softwares utilizados foram atualizadas para a mesma versão em todos os computadores. Além do próprio SopCast, em particular, foi utilizado o Wireshark[5] (*tcpdump*) para capturar o tráfego de rede observado durante o monitoramento de uma transmissão de vídeo.

O Wireshark foi configurado para capturar apenas o tráfego com porta de origem e destino iguais as portas configuradas para o sistema SopCast (por exemplo, porta UDP/TCP 3908). Essa captura é armazenada em um arquivo de *log* e contém a data e a hora (na granularidade de 1 segundo) de cada pacote enviado pelo próprio computador ou recebido por ele na porta configurada ao longo da transmissão de vídeo. Para que todos os computadores estivessem sincronizados nos experimentos foi utilizado o *Network Time Protocolo* [6] (NTP) com a configuração do mesmo servidor entre eles [Pathak et al. 2008]. Essa sincronização garante que a diferença de fuso horário e de tempo entre os computadores seja muito pequena (menos de um segundo).

Após a etapa de configuração de todos os computadores, eles passaram a executar o SopCast e a ferramenta de monitoramento Wireshark. De forma distribuída ao longo do tempo inicial T esses computadores, agora clientes SopCast, foram conectados ao canal CCTV-1. Como clientes comuns do SopCast, estes passaram a monitorar e registrar nos seus respectivos arquivos de *log* todas as parcerias realizadas, ou seja, todas as trocas de dados entre eles mesmos e entre outros clientes. A duração de um experimento é dada pela soma do tempo inicial T e a duração da transmissão monitorada D. Nas análises deste trabalho, foi considerado apenas os registros dos *logs* realizados após todos os clientes do SopCast do experimento já terem se conectado ao canal CCTV-1, ou seja, depois o tempo inical T.

Finalizado o período D de monitoramento, todos os arquivos de *logs* dos clientes SopCast foram organizados em um único arquivo de *log* usado para reconstruir a dinâmica da rede sobreposta do SopCast durante a transmissão de vídeo. Baseando-se nas informações da data, da hora, do IP de origem e do IP de destino foram retiradas S fotografias (*snapshots*) da rede em intervalos de I segundos com janelas deslizantes de L segundos entre cada uma delas. Dado que a caracterização deste trabalho em relação as propriedades dos nós e da rede considera uma granularidade de tempo de um segundo, a diferença de tempo entre os dados dos Cliente SopCast pode ser considerada insignificante.

A rede deste trabalho foi reconstruída utilizando clientes executados no PlanetLab

[5]www.wireshark.org
[6]www.ntp.org

que foram configurados para armazenar todas as trocas de pacotes entre clientes SopCast, trata-se, portanto, de uma visão parcial da rede SopCast do canal aberto em que os clientes foram conectados. Todas as métricas analisadas neste trabalho são em função dessa visão da redesobreposta. A abordagem utilizada neste trabalho não é nova e muitas vezes é aplicada na literatura [Vu et al. 2007, Fallica et al. 2008]. No entanto, uma das principais contribuições deste trabalho está no número de clientes SopCast conectados na rede ao mesmo tempo em relação aos trabalhos da literatura que utilizaram no máximo 70 clientes [Fallica et al. 2008]. Além disso, essa abordagem busca por uma reconstrução da rede mais realista por apresentar uma maior probabilidade de uma visão completa da rede analisando a visão dos nós deste trabalho ao mesmo tempo em pontos diferentes do mundo.

3.3. Sumário dos Experimentos

Neste trabalho são apresentados 7 experimentos realizados entre os dias 28/10/2010 e 17/11/2010 com duração de transmissão de vídeos de 40 minutos em cada dia. Foram utilizados em cada experimento entre 200 a 465 clientes SopCast. Essa variação do número de clientes SopCast acontece devido a instabilidade do ambiente de testes, no entanto a diferença entre o número de participantes nestes experimentos é de aproximadamente 1%. Estes clientes foram conectados ao canal aberto da China (CCTV-1) no horário de pico local (20 horas) ao longo de um tempo inicial de 10 minutos. E então, monitoraram os tráfegos de rede após todos os clientes já estarem conectados na rede durante 40 minutos. Os *logs* de cada cliente SopCast foram agrupados por experimento e retirados 118 fotografias (*snapshots*) em que cada uma delas representa 60 segundos e a diferença entre elas de 20 segundos. Estes dados estão sumarizados na Tabela 1.

Table 1. *Sumário dos Experimentos Realizados*

Número de experimentos realizados (E)	7
Período dos experimentos	28/10/2010 a 17/11/2010
Número de computadores por experimento (N)	200 a 465
Canal aberto utilizado	CCTV-1
Horário de pico local	20:00 hrs
Tempo inicial de conexão (T)	10 minutos
Duração da transmissão (D)	40 minutos
Número de fotografias da rede por experimento (S)	118
Duração de cada fotografia (L)	60 segundos
Janelas deslizantes entre as fotografias (I)	20 segundos

4. Métricas Utilizadas

A rede sobreposta formada pelos clientes conectados no SopCast durante um experimento pode ser representada por um grafo direcionado $G = (V, E)$, onde V é o conjunto de $|V|$ vértices e E é o conjunto de arestas que conectam dois elementos de V. Uma aresta direcionada $(v_i, v_j) \in E$ representa uma parceria estabelecida entre dois clientes representados pelos nós v_i e v_j $(v_i, v_j) \in V$ para a transmissão de dados de v_i para v_j. As características deste grafo podem variar ao longo do tempo à medida que as parcerias são feitas e desfeitas.

O objetivo deste trabalho é exatamente caraterizar como as características deste grafo evoluem com o tempo durante uma transmissão. Essa caracterização é feita sobre

duas perspectivas: características dos nós e características da rede como um togo. Para tanto, foram utilizadas métricas de nós(Seção 4.1) e métricas da rede(Seção 4.2). Por simplificação nas subseções seguintes uma fotografia da rede será abordada como instante t, a rede sobreposta desse instante como grafo G, os nós como vértices v e as parcerias estabelecidas por estes nós como arestas e.

4.1. Métricas dos Nós

Para analisar os nós dentro da estrutura da rede foram utilizadas métricas que avaliam a centralidade do nó nessa estrutura, são elas: grau, *betweenness* e *closeness*.

Grau: O grau g de um vértice $v_i(t)$ no instante t é definido pela soma do grau de entrada e saída de $v_i(t)$. O grau de entrada de v_i é o número de v para os quais ele aponta, enquanto o grau de saída é o número de v que apontam para v_i. [Easley and Kleinberg 2010]. O grau de um vértice é uma das propriedades estruturais mais básicas que avalia o número de arestas adjacentes.

***Betweenness*:** O *betweenness* de um vértice v_i é a razão de $\sigma_{st}(v_i, t)$ por σ_{sw}, em que $\sigma_{sw}(v_i, t)$ representa o número de caminhos mínimos de v_s a v_w que passam por v_i e σ_{sw} o número de caminhos mínimos entre v_s e v_w. O *betweenness* de um vértice é a probabilidade dele estar no caminho mínimo entre dois outros vértices quaisquer.

$$\textit{Betweenness}(v, t) = \sum_{s \neq w \neq i \in V} \frac{\sigma_{st}(v_i, t)}{\sigma_{sw}}$$

***Closeness*:** O *closeness* de um vértice $v_i(t)$ no instante t é usado avaliar o caminho mínimo médio, definido por l, entre o vértice v_i e todo vértice v_w alcançável a partir de v_i tal que (v_i,v_w) $\in V$. Com essa métrica é possível identificar o quão próximo um vértice está dos demais vértices da rede. Em outras palavras, no SopCast, se um nó com um valor alto de *closeness* possui alguma informação na rede sobreposta, essa informação será encontrada pelos demais nós rapidamente.

$$\textit{Closeness}(v, t) = \frac{|V| - 1}{\sum_{i \neq w \setminus i} l(v_i, v_w)}$$

4.2. Métricas da Rede

A estrutura da rede sobreposta como um todo foi avaliada utilizando as seguintes métricas: caminho mínimo médio, diâmetro, coeficiente de agrupamento e grau máximo da rede.

Diâmetro: O diâmetro de um grafo é definido pela distância máxima entre quaisquer dois nós, considerando que não haja ciclo. Em outras palavras, o diâmetro de um grafo é a maior distância m definida pelo número de arestas entre dois vértices (v_i, v_j), considerando todos os pares de vértices no grafo G do instânte t. Com essa métrica é possível ter uma idéia da dispersão do grafo. Logo, ela pode ser utilizada para fazer avaliações aproximadas sobre latência, impactos na rede, entre outros.

$$Di\hat{a}metro(G, t) = max_{\forall_{i,j \in V}} m(v_i, v_j)$$

Caminho Mínimo Médio: O caminho mínimo médio de um grafo G no instante t é definido pelo menor caminho l de um vértice v_i para todos os outros vértices da rede utilizando busca em largura para calcular l.

$$CaminhoM\acute{i}nimoM\acute{e}dio(G, t) = \frac{\sum_{i \neq w \in V} l(v_i, v_w)}{|V|}$$

Coeficiente de Agrupamento: O coeficiente de agrupamento de um grafo G no instante t é dado pela média do coeficiente de agrupamento de todos os vértices Cv. Dados três vértices v_i, v_j e v_w com arestas definidas entre (v_i,v_j) $\in E$ e (v_j,v_w) $\in E$, o coeficiente de agrupamento de um vértice Cv é definido pela probabilidade de existir uma aresta entre (v_i,v_w) $\in E$.

$$CoeficienteDeAgrupamento(G,t) = \frac{\sum_{v \in V} Cv(v,t)}{|V|}$$

Grau Máximo: Por definição o grau máximo de um grafo G no instante t é dado pelo maior grau de um vértice $g(v)$ é dado por:

$$GrauMáximo(G,t) = max_{\forall_{i \in V}} g(v_i)$$

5. Resultados

Para cada experimento realizado foram caracterizadas as propriedades da rede e as centralidades individuais dos nós na estrutura desta rede. Na Seção 5.1 são apresentados os resultados obtidos pela caracterização dos nós enquanto que os resultados das propriedades estruturais da rede estão na Seção 5.2.

5.1. Caracterização dos Nós

A caracterização dos nós decorrentes de suas respectivas centralidades foi feita extraindo as medidas de Grau(g), *Betweenness*(b) e *Closeness*(c) de cada nó por fotografia da rede. Com isso em cada fotografia de um experimento, um nó v_i possui um vetor $v^{vi} = (g, b, c)$ com três posições correspondentes aos valores medidos da centralidade dele. Para identificar perfis de nós com centralidades distintas, os vetores de todos os nós de um experimento foram submetidos a um algoritmo de agrupamento, o *k-means* [Wan et al. 1988]. Este algoritmo atua nos dados de modo não supervisionado utilizando a distância Euclediana como medida de distância entre dois vetores v^{vi} e v^{vj}. O número de perfis em cada experimento foi definido através da relação β_{cv} que avalia a variação da separação dos vetores em perfis. Esta avaliação é feita através da razão da distância média dos vetores dos perfis e seus centróides pela distância média entre os próprios centróides [Menasce and Almeida 2007].

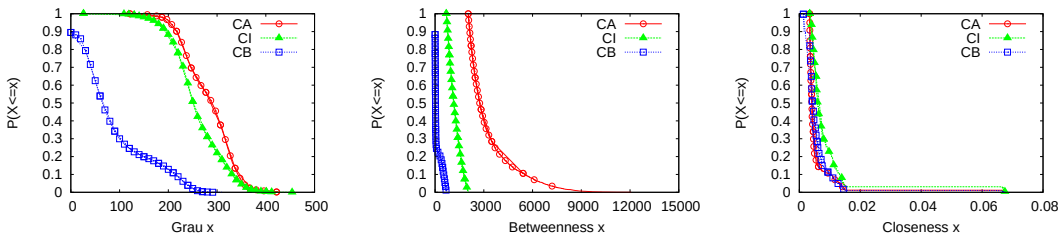
Utilizando esta estratégia foram encontrados em todos os experimentos três perfis de centralidade dos nós. Os perfis foram denominados de *Centralidade Alta* (CA), *Centralidade Intermediária* (CI) e *Centralidade Baixa* (CB). A Tabela 2 sumariza os valores médios e os coeficientes de variação(CV) das medidas de centralidade dos nós calculados sobre todas as fotografias de cada experimento realizado. Note que, apesar de encontrados sempre três perfis(centróides), os valores exatos que definem cada centróide variam de acordo com o experimento.

Um nó do perfil *CA* é aquele que possui Grau e *Betweenness* mais elevado do que os demais, em contrapartida o *Closeness* entre os perfis não apresenta uma variação significativa. Os nós do perfil *CA* são aqueles que dentro da estrutura topológica da rede possuem maior centralidade. Já os nós do perfil *CB* são aqueles que apresentam baixos valores de Grau e *Betweenness*, podendo ser considerados nós com menor centralidade. Já os nós do *CI* são aqueles com características intermediárias ao *CA* e ao *CB*.

Table 2. *Definição dos perfis dos nós*

			% de Nós	Grau		*Betweenness*		*Closeness*	
				Média	CV	Média	CV	Média	CV
Experimento	1	*CA*	4.76%	282.83	0.17	3312.52	0.45	0.005	1.25
		CI	32.69%	257.99	0.20	1212.52	0.30	0.008	1.20
		CB	62.53%	86.94	0.82	129.45	1.60	0.005	1.16
	2	*CA*	6.98%	334.89	0.21	6585.63	0.33	0.005	0.61
		CI	45.85%	224.13	0.17	2258.33	0.32	0.006	0.64
		CB	47.16%	36.10	1.22	58.73	3.22	0.003	0.88
	3	*CA*	3.81%	361.95	0.18	8604.33	0.30	0.003	0.52
		CI	17.55%	240.43	0.21	2556.77	0.36	0.004	1.06
		CB	78.62%	56.83	1.14	135.47	2.30	0.003	0.90
	4	*CA*	2.92%	298.33	0.24	11430.93	0.43	0.004	0.56
		CI	14.61%	230.04	0.20	3443.58	0.38	0.007	0.82
		CB	82.46%	56.83	1.14	157.44	2.64	0.004	0.97
	5	*CA*	6.76%	322.39	0.24	6898.68	0.56	0.004	0.79
		CI	40.97%	230.42	0.19	2134.54	0.36	0.006	0.89
		CB	52.25%	56.83	1.14	120.73	2.24	0.003	1.04
	6	*CA*	2.69%	241.23	0.20	11865.11	0.63	0.004	0.12
		CI	8.99%	252.241	0.24	3159.50	0.42	0.007	0.75
		CB	88.31%	86.20	1.04	282.39	1.61	0.007	1.08
	7	*CA*	19.79%	298.42	0.16	1751.81	0.31	0.006	1.42
		CI	30.03%	261.86	0.16	833.93	0.22	0.007	1.37
		CB	50.17%	123.47	0.68	159.52	1.13	0.005	1.61

As diferenças entre os perfis de centralidade são apresentadas na Figura 2 pelas distribuições acumuladas(CDFs) encontradas para o Experimento 1 (vide Tabela 2), escolhido apenas para exemplificação. Nesta figura é possível observar que nós do perfil *CA* apresentam uma maior probabilidade de terem valores de Grau maior do que os demais perfis. Cerca de 20% dos nós do perfil *CB* possuem Grau aproximadamente igual a 0. Para a métrica *Betweenness*, as três curvas seguem também apresentam uma clara diferença entre os perfis, ou seja, a distribuição para *CA* tem uma tendência maior para valores mais altos seguido por *CI* e então por *CB*. Já o *Closeness* não segue o mesmo comportamento apresentado variações de valores entre os perfis o que não permite uma diferenciação clara entre os perfis.

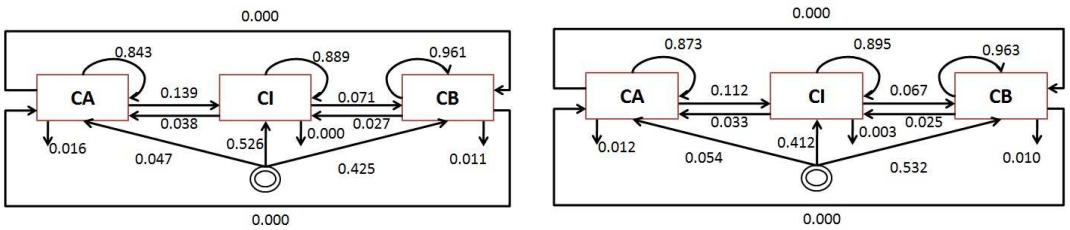


(a) Grau (b) Betweenness (c) Closeness

Figure 2. Métricas dos nós por perfil

Ao longo de uma transmissão, a medida que as parcerias entre os nós se alteram,

um mesmo nó pode apresentar diferentes perfis. Ou seja, em um dado momento, o nó pode ter uma centralidade mais alta e mais adiante, a medida que suas parcerias se alteram, ele pode ter uma centralidade mais baixa. Por exemplo: um nó que no início de uma transmissão é caracterizado pelo perfil *CA*, no decorrer do tempo, suas parcerias se alteram e ocorre a transição para perfil *CI*. O interesse deste trabalho é caracterizar este comportamento dinâmico. Para tal, foi utilizado uma técnica de modelagem denominada *Customer Behavior Model Graph* (CBMG) [Menasce and Almeida 2007]. Um CBMG é um grafo utilizado para representar a dinamicidade em um modelo de comportamento. No contexto deste trabalho, o uso de um CBMG tem por objetivo mapear as alterações das centralidades dos nós ao longo do tempo. Para isso, no CBMG cada estado (retângulo) representa um perfil de centralidade do nó e as arestas entre os estados representam a probabilidade de um nó ter sua centralidade alterada de um perfil para o outro. A Figura 3 (a) apresenta o CBMG encontrado para o Experimento 1 (Tabela 2) em que o mesmo padrão de transição entre os perfis pode ser observado para os outros experimentos. A Figura 3 (b) mostra o CBMG médio de todos os sete experimentos realizados.



(a) Experimento 1 (b) Todos os experimentos

Figure 3. CBMG

Nos dois CBMGs da Figura 3 é possível visualizar a transição dos nós entre os perfis de centralidade observados nos experimentos. A probabilidade de um nó permanecer no mesmo pefil tende a ser bem alta. Por exemplo, considerando o CBMG médio (Figura 3 (b)) um nó que inicialmente possuia centralidade baixa|alta tende a permanecer no perfil *CB*|*CA* com probabilidade de 0.963|0.873. As transições entre dois perfis extremos como o *CA* e *CB* é zero dentro da precisão de três casas adotada neste trabalho, indicando que não há uma redução brusca de conectividade de um nó. Outro fato que pode ser observado é que a probabilidade de um nó diminuir sua centralidade é maior que a probabilidade dele aumentar, visto que de *CA* para *CI* a probabilidade é de 0.112, *CI* para *CB* de 0.067 enquanto que de *CB* para *CI* é 0.025 e *CI* para *CA* 0.033. Logo, os nós ao se conectarem em uma rede sobreposta possuem maior probabilidade de permanecerem no mesmo perfil.

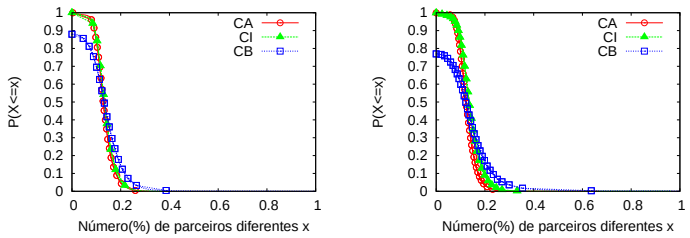
Sendo assim, outra questão que pode ser considerada é que apesar de tenderem a manter o mesmo perfil de centralidade, os nós tendem a variar seus parceiros ao longo do tempo. Note que isto pode acontecer sem que as propriedades dos nós sejam alteradas. A Figura 4 (a) apresenta para cada perfil a CDF da porcentagem p de parceiros diferentes de um mesmo nó i em duas fotografias consecutivas $t1$ e $t2$ da rede. Em outras palavras:

$$p_i = \frac{p_i^{t1} \cap p_i^{t2}}{p_i^{t1} \cup p_i^{t2}}$$

Table 3. *Medidas das métricas da rede*

		Diâmetro		Caminho Mínimo Médio		Coeficiente de Agrupamento		Grau Máximo	
		Média	CV	Média	CV	Média	CV	Média	CV
Experimento	1	4.11	0.07	1.98	0.03	0.24	0.34	361.47	0.08
	2	4.13	0.09	2.08	0.03	0.15	0.28	431.51	0.11
	3	4.33	0.13	2.09	0.02	0.15	0.46	429.71	0.06
	4	4.27	0.10	2.16	0.03	0.12	0.56	389.88	0.09
	5	4.22	0.09	2.14	0.02	0.17	0.21	398.68	0.12
	6	4.23	0.10	2.10	0.07	0.18	0.64	339.83	0.11
	7	4.05	0.05	2.00	0.02	0.32	0.11	377.00	0.05

A porcentagem de parceiros diferentes entre fotografias consecutivas tende a ser muito baixa nos três perfis. Em particular, para os perfis *CA* e *CI* a porcentagem máxima observada foi de 20%. A Figura 4 (b) mostra para cada perfil a CDF correspondem considerando todos os experimentos. Assim como o Experimento 1, a porcentagem de parceiros diferentes entre fotografias consecutivas para todos os experimentos também tende a ser baixa.



(a) Experimento 1 (b) Todos os experimentos

Figure 4. Probabilidade de novos parceiros por perfil de nó

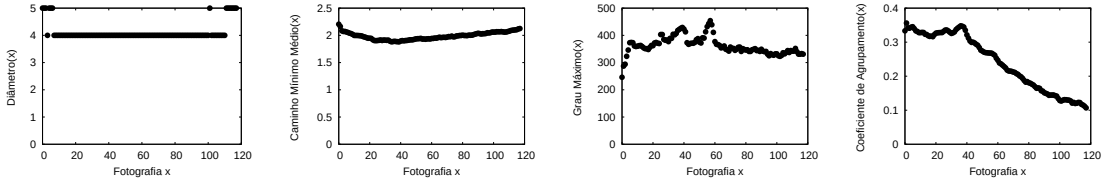
Portanto, os nós possuem poucos parceiros diferentes entre fotografias consecutivas da rede e além disso tendem a se manter em um mesmo perfil. Os perfis foram encontrados pelas métricas de Grau e *Betweenness*. O *Closeness* apesar de ser uma métrica do nó não apresentou uma variação significativa entre os perfis. A definição de cada perfil foi feita de acordo com o experimento, foi possível identificar que em todos os experimentos foram encontrados três perfis distintos de centralidade.

5.2. Caracterização da Rede Sobreposta

Sobre a perspectiva das propriedades estruturais da rede sobreposta, as métricas de rede foram extraídas de cada fotografia dos experimentos. Foi feita uma análise primeiramente do valor médio medido em todas as fotografias de cada experimento. A Tabela 3 sumariza as medidas encontradas por experimento com as médias e os coeficientes de variação (CV) dos valores encontrados.

Em todos os experimentos realizados os valores do Diâmetro e Caminho Mínimo Médio apresentam valores em torno de 4 e 2 respectivamente. Os Graus Máximos dos experimentos apresentam valores médio variando entre 360 e 430 e Coeficientes de Agrupamento entre 0.12 e 0.32. Para entender essa variação do Grau Máximo e do Coeficiente de Agrupamento da rede foi estudada como ocorre a evolução desses valores ao longo do

tempo, ou seja, nas fotografias de um experimento. A Figura 5 mostra as quatro métricas da rede ao longo do tempo do Experimento 1. Nesta análise é possível observar que o Coeficiente de Agrupamento da rede decresce ao longo do tempo, Figura 5(d). Sendo que o número de nós analisados é o mesmo por todas as fotografias, o coeficiente de agrupamento decresce devido a novas parcerias estabelecidas entre os nós ao longo de todo o tempo.



(a) Diâmetro (b) Caminho Mínimo Médio (c) Grau Máximo (d) Coeficiente de Agrupamento

Figure 5. Métricas de rede ao longo do tempo do Experimento 1

Assim, as métricas da rede Diâmetro (Figura 5(a)) e Caminho Mínimo Médio (Figura 5(b)) tendem a permanecer estáveis. Já o Coeficiente de Agrupamento descrece ao longo do tempo e o Grau Máximo (Figura 5(c)) possui variações porém dentro do Grau observado pelo perfil *CA* dos nós. Não foi encontrada uma relação direta dos perfis dos nós com as métricas da rede, exceto pela comportamento estável dos nós nos perfis e pelas métricas Diâmetro e Caminho Mínimo Médio.

6. Conclusões e Trabalhos Futuros

Neste trabalho foi apresentado uma caracterização das propriedades dinâmicas da estrutura de redes sobrepostas do SopCast. Foram relizados uma série de sete experimentos utilizando computadores do PlanetLab que se conectaram ao canal aberto CCTV-1 do SopCast e passaram a atuar na rede de transmissão do canal como clientes comuns. Estes computadores monitoraram o tráfego de rede realizado para a transmissão do vídeo e armazenaram as informações em arquivos de *log*. Estes arquivos foram unificados em um único arquivo que pode ser utilizado para representar uma visão da rede sobreposta para a transmissão de vídeo. Analisando os nós individualmente em cada experimento, foram retiradas as métricas de centralidade dos clientes SopCast.

Os valores destas métricas de cada experimento foram submetidos ao algoritmo de agrupamento *k-means*, onde foi possível identificar três perfis distintos de centralidade: *Centralidade Alta*, *Centralidade Intermediária* e *Centralidade Baixa*. Além de gerar as distribuições acumuladas dos valores obtidos pelas métricas por perfil, foram gerados CBMGs (*Customer Behavior Model Graph*) para representar a transição de um nó com um dado perfil de centralidade para outro perfil ao longo do tempo. Com o CBMG foi possível concluir que os nós possuem baixa probabilidade de transição entre os perfis de centralidade. Além disso, foi analisado entre fotografias consecutivas de um experimento a porcentagem de parceiros diferentes realizados pelos nós. Já na perspectiva da estrutura da rede foi possível observar que o Diâmetro e o Caminho Mínimo Médio da rede tendem a serem mais estáveis que o Grau Máximo e o Coeficiente de Agrupamento.

Com os resultados obtidos neste trabalho foi possível identificar *insights* da dinamicidade da estrutura da rede sobreposta de um sistema de transmissão de vídeo ao

vivo. Estes *insights* podem ser utilizados em pesquisas futuras e na simulação de sistemas para que cargas de trabalho sintéticas ao serem geradas reflitam o real dinamismo da estrutura da rede sobreposta para a transmissão de vídeo em redes P2P.

References

Borges, A., Almeida, J., and Campos, S. (2010a). Can malicious attacks slow peer-to-peer live iptv systems adoption? *Submetido para periódico*.

Borges, A., Gomes, P., Nacif, J., Almeida, J. M., and Campos, S. (2010b). Characterizing sopcast client behavior. *Submetido para periódico*.

Easley, D. and Kleinberg, J. (2010). *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, New York, NY, USA.

Fallica, B., Lu, Y., Kuipers, F., Kooij, R., and Mieghem, P. V. (2008). On the quality of experience of sopcast. *International Conference on Next Generation Mobile Applications, Services and Technologies*.

Google (2010). Google trends.

Hei, X., Liang, C., Liang, J., Liu, Y., and Ross, K. (2007). A measurement study of a large-scale p2p iptv system. *IEEE Transactions on Multimedia*.

Hei, X., Liu, Y., and Ross, K. (2008). Iptv over p2p streaming networks: the mesh-pull approach. *IEEE Communications Magazine*.

Huang, Y., Fu, T. Z., Chiu, D.-M., Lui, J. C., and Huang, C. (2008). Challenges, design and analysis of a large-scale p2p-vod system. *SIGCOMM Comput. Commun. Rev.*

Menasce, D. and Almeida, V. (2007). *Scaling for E-Business: Technologies, Models, Performance, and Capacity Planning*. Prentice Hall.

Oliveira, J. F. A., de C. Gomes, P., Vieira, A. B., and Campos, S. V. A. (2009). Centralidade em redes p2p de transmissão ao vivo. *SBRC-09*.

Pathak, A., Pucha, H., Zhang, Y., Hu, Y. C., and Mao, Z. M. (2008). A measurement study of internet delay asymmetry. In *PAM'08*, Berlin, Heidelberg.

Silverston, T. and Fourmaux, O. (2007). Measuring p2p iptv systems. *NOSSDAV'07: Proceedings of Network Operating Systems Support for Digital Audio Video*.

Stutzbach, D., Rejaie, R., and Sen, S. (2008). Characterizing unstructured overlay topologies in modern p2p file-sharing systems. *IEEE/ACM Trans. Netw.*

Tang, S., Lu, Y., Hernández, J. M., Kuipers, F., and Mieghem, P. (2009). Topology dynamics in a p2ptv network. In *NETWORKING '09*, Berlin, Heidelberg.

Vu, L. H., Gupta, I., Liang, J., and Nahrstedt, K. (2007). Measurement and modeling of a large-scale overlay for multimedia streaming. In *QShine International Conference*, Vancouver, Canada.

Wan, S. J., Wong, S. K. M., and Prusinkiewicz, P. (1988). An algorithm for multidimensional data clustering. *ACM Trans. Math. Softw.*

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*.

Um Novo Algoritmo Distribuído para Avaliação e Localização de Centralidade de Rede

Klaus Wehmuth e Artur Ziviani

Laboratório Nacional de Computação Científica (LNCC)
Av, Getúlio Vargas, 333 – 25651-075 – Petrópolis, RJ, Brasil
Email: {klaus,ziviani}@lncc.br

Abstract. *We propose a new distributed algorithm for assessing centrality in complex networks. This algorithm uses only local information at each node, not needing the knowledge of the network's full topology. Besides calculating the centrality value of each node, our proposal also provides a way for locating the most central nodes, also using only local information at each node. We experimentally evaluate the proposed algorithm by comparing it to a recent algorithm and conclude that our algorithm achieves similar results but with a significant reduction on the associated applicability costs. This outcome allows our algorithm to be applied to large-scale networks as we demonstrate by applying our proposal to a real-world large-scale network trace.*

Resumo. *Nós propomos um novo algoritmo distribuído para avaliação de centralidade em redes complexas. Esse algoritmo usa somente informação localizada em cada nó, não necessitando do conhecimento da topologia completa da rede. Além do cálculo do valor de centralidade em cada nó, nossa proposta também provê um meio para a localização dos nós mais centrais, também somente usando informação localizada em cada nó. Nós avaliamos experimentalmente o algoritmo proposto comparando-o a um algoritmo recente e concluímos que nosso algoritmo alcança resultados similares, porém com custo para sua aplicabilidade prática significativamente menor. Esse resultado permite ao nosso algoritmo ser aplicado a redes reais de larga-escala, o que demonstramos aplicando-o a um registro (trace) de uma rede real de roteadores.*

1. Introdução

O conceito de centralidade é uma importante métrica para se avaliar a importância relativa de nós ou arestas individuais na estrutura de uma rede complexa. Devido à sua generalidade e a existência de diferentes formas de interpretar esse conceito, tem-se várias maneiras de definir centralidade de rede [Freeman, 1977; Borgatti, 2005]. O conceito de centralidade pode consequentemente ser utilizado com diversos propósitos para a avaliação de desempenho e comportamento em uma rede complexa, tais como a inferência de informações sobre o nível de conectividade em uma rede, a importância de pessoas e seu papel em uma rede social, a probabilidade de sobrecarga de um determinado nó ou aresta, o impacto em termos de re-roteamento devido à retirada de um nó, entre outras.

Apesar de sua vasta utilidade, o cálculo de centralidade é em geral computacionalmente bastante custoso e requer conhecimento total da topologia, inviabilizando sua aplicação em redes de larga escala, como as muitas encontradas atualmente no cenário

de redes de computadores. Exemplos dessas redes complexas de larga escala atuais incluem diferentes redes interdependentes que compõem a Internet atual, tais como (i) redes centradas em usuários (p.ex., redes sociais online e de microblog); (ii) redes de aplicações (p.ex., redes de email e redes P2P de distribuição de conteúdo); e (iii) redes de infra-estrutura compostas por roteadores. Nesse contexto, há trabalhos propostos visando tornar mais eficiente o cálculo de centralidades tradicionais [Brandes, 2001] ou ainda propor novas formas de centralidade que possam ser obtidas de forma distribuída e que ainda assim possam representar de forma adequada a importância relativa dos diversos nós e arestas da rede [Nanda e Kotz, 2008; Kermarrec et al., 2011].

Neste artigo, nós propomos um algoritmo distribuído para avaliação e localização de centralidade em redes chamado DANCE (*Distributed Assessment of Network CEntrality*). DANCE possui duas contribuições básicas. Como primeira contribuição, DANCE calcula de forma totalmente distribuída valores locais de centralidade de todos os nós de uma rede baseando-se somente na vizinhança de dois saltos de cada nó, sem necessidade de conhecer em qualquer ponto a topologia completa da rede. As demais propostas para avaliação de centralidade de forma distribuída (ver Seção 2) se restringem ao cálculo distribuído dos valores de centralidade, sem fornecer entretanto meios para a localização dos nós de maior centralidade de forma distribuída. Assim, como segunda contribuição, DANCE possui um método para a localização de forma distribuída dos nós de centralidade mais relevante em uma rede – e em particular o nó de maior centralidade de toda rede, o qual nomeamos nó crítico. Essa localização do(s) nó(s) crítico(s) é realizada utilizando-se somente os valores locais de centralidade computados na primeira etapa do DANCE, também sem necessidade do conhecimento da topologia completa da rede.

A nossa proposta DANCE mostra-se simples, eficaz e eficiente. Nós avaliamos o desempenho do DANCE através de sua comparação com outro algoritmo recentemente proposto [Kermarrec et al., 2011] para o mesmo fim, que se baseia no uso de um caminho aleatório para estimação distribuída da centralidade de cada nó da rede. Os resultados experimentais são obtidos em redes sintéticas com diferentes características topológicas. Em termos de efetividade, DANCE alcança resultados similares na avaliação da centralidade ao algoritmo baseado em caminho aleatório, porém com custo para sua aplicabilidade significativamente menor. Isso permite, contrastando com o estado da arte, a aplicação de DANCE para avaliação da centralidade em redes complexas de larga escala, cuja viabilidade é demonstrada pela aplicação do DANCE a um registro (*trace*) referente a uma rede real de larga escala de conectividade de roteadores da Internet.

Este artigo está organizado da seguinte forma. A Seção 2 analisa trabalhos relacionados. Nossa proposta DANCE é apresentada na Seção 3. A Seção 4 discute os resultados experimentais obtidos e a Seção 5 trata das conclusões e trabalhos futuros.

2. Trabalhos Relacionados

Existem várias métricas de centralidade que buscam determinar a importância relativa de um nó em uma rede sob diferentes aspectos, tais como a importância do nó para a conectividade da rede ou para sua capacidade de difusão de informação. Exemplos de métricas de centralidades bem conhecidas são a *betweenness centrality* e centralidade de autovetor. A primeira corresponde em cada nó ao número de caminhos mais curtos entre todos os pares de nós que passam por este nó. A segunda baseia-se no princípio que

conexões de nós com maior grau de conectividade contribuem mais para a centralidade de cada nó que conexões de nós com baixo grau; sendo o PageRank [Brin e Page, 1998] do Google uma variação dessa medida de centralidade.

O cálculo de centralidades tradicionais é tipicamente bastante custoso computacionalmente e requer conhecimento completo da topologia. Nesse contexto, há trabalhos propostos visando tornar mais eficiente o cálculo de centralidades tradicionais [Brandes, 2001]. Dinh et al. [2010] propõem uma nova métrica de centralidade menos custosa computacionalmente, porém ainda requerendo conhecimento completo da topologia.

Alternativamente, como nossa proposta, alguns trabalhos investigam formas descentralizadas de avaliação de centralidade em redes, buscando assim evitar a necessidade de conhecimento completo da topologia. Lehmann e Kaufmann [2003] propõem uma forma decentralizada de cálculo para centralidades baseadas na descoberta e contabilização dos caminhos mínimos entre os nós. Esse trabalho introduz conceitos básicos na obtenção de centralidades de forma distribuída, entretanto o algoritmo proposto em geral tem um custo computacional ainda proibitivamente alto para aplicação em redes de grande porte. Nanda e Kotz [2008] introduzem uma medida de centralidade chamada *Localized Bridging Centrality* (LBC). Essa medida de centralidade é avaliada utilizando apenas os vizinhos diretos de cada nó da rede e dirigida especificamente para localização de pontes (arestas cuja remoção fragmentam a rede), tendo sido aplicada a redes sem fio de pequeno porte apenas.

Recentemente, Kermarrec et al. [2011] apresentam uma nova medida de centralidade chamada *Second Order Centrality* (SOC) que é obtida por meio de um único caminho aleatório perpétuo computando-se o desvio padrão do intervalo de tempo entre visitas consecutivas desse caminho aleatório a cada nó da rede. Esse método distribuído apresenta bons resultados como medida de centralidade, porém o uso de um único caminho aleatório faz com que se torne difícil estabelecer um critério de parada para o cálculo, além da proposta apresentar um alto custo em termos de mensagens e tempo de convergência.

Em comparação com o estado da arte em avaliação distribuída de centralidade, nossa proposta DANCE contribui com uma solução simples cuja efetividade é similar ao SOC [Kermarrec et al., 2011], porém com um custo para sua aplicabilidade significativamente menor (ver Seção 4). Além disso, todas as demais propostas de avaliação distribuída de centralidade se restringem à avaliação da centralidade em cada nó da rede, não possuindo uma solução para a localização de forma distribuída dos nós mais críticos em termos de centralidade, o que constitui a segunda contribuição de nossa proposta.

3. Algoritmo DANCE

Nós propomos um novo algoritmo distribuído chamado DANCE para avaliar a centralidade dos nós de uma rede e localizar os nós mais centrais utilizando apenas informações localizadas de cada nó da rede. A base lógica para isso é a seguinte premissa:

Premissa: se um determinado nó é central para a rede como um todo, ele também será central localmente para uma vizinhança estabelecida em torno desse mesmo nó.

DANCE explora esse conceito simples e intuitivo de forma a determinar a centralidade dos nós da rede de forma distribuída e posteriormente identificar os nós mais centrais. Uma contribuição chave da proposta é justamente mostrar que mesmo baseado nessa premissa simples, pode-se obter uma solução eficaz e eficiente.

3.1. Avaliação distribuída de centralidade

Cada nó da rede considera sua vizinhança de dois saltos, ou seja, a sub-rede formada pelo nó em questão, seus vizinhos diretos e os vizinhos dos seus vizinhos. Essa vizinhança de dois saltos a partir de um determinado nó n será doravante denominada $2_vizinhança_n$. Após a determinação da $2_vizinhança_n$ de cada nó n da rede[1], é utilizada uma métrica calculada para a $2_vizinhança$ de cada nó. O resultado referente a essa métrica é considerado como a medida de centralidade do nó para o qual a $2_vizinhança$ foi estabelecida. A métrica utilizada pode variar, gerando medidas de centralidade diferentes, e a operação geral do DANCE independe da métrica específica. Um exemplo simples de métrica de centralidade que será utilizada neste trabalho é o volume da $2_vizinhança$ considerada. Utilizando essa métrica, a medida de centralidade de cada nó corresponde à soma dos graus de cada nó que compõe sua $2_vizinhança$. O Algoritmo 1 ilustra o mecanismo geral do DANCE para o cálculo da centralidade de cada nó da rede de forma distribuída.

Algoritmo 1 DANCE_DISTRIBUTEDCENTRALITYCOMPUTATION()

```
1: {at each node n in the network}
2: {define shared variable to contain the topology of the 2-hop neighborhood around this node.}
3: define 2_neighborhoodTopology
4: {define shared variable for the centrality value at this node.}
5: define centrality
6: {launch process that sets the shared variable 2_neighborhoodTopology.}
7: fork discover_2_NeighborhoodTopology()
8: {launch process that sets the shared variable centrality.}
9: fork calculateCentrality()
```

3.1.1. Descoberta da topologia da vizinhança a dois saltos de cada nó

O Algoritmo 2 mostra como é realizada no DANCE a descoberta da topologia da sub-rede referente à vizinhança de dois saltos de cada nó da rede. Cada nó envia uma mensagem para cada um de seus vizinhos que por sua vez repetem esta mensagem aos seus vizinhos. Ou seja, o número de mensagens geradas por um nó será igual ao seu grau somado ao grau de cada um de seus vizinhos, de forma que este número de mensagens será também igual ao volume da vizinhança de raio 1 do nó em questão. Assim, a complexidade de mensagens esperada do Algoritmo 2 é $O(n \times d_m^2)$, onde n é o número de nós e d_m é o grau médio da rede. Cada nó pode determinar sua $2_vizinhança$ de forma independente dos demais nós da rede, logo isso pode ocorrer em paralelo. Portanto, o tempo necessário para isso é limitado ao tempo de propagação de dois saltos. Como consequência, a complexidade de tempo esperada para o Algoritmo 2 é $O(1)$ passos discretos.

3.1.2. Cálculo distribuído da métrica de centralidade

O objetivo do Algoritmo 3 é determinar o valor da centralidade de cada nó baseado na sua $2_vizinhança$. Em princípio, a métrica deve ser recalculada toda vez que houver uma mudança na topologia da $2_vizinhança$ percebida pelo nó (ver Algoritmo 2). Como a determinação da $2_vizinhança$ depende do recebimento das mensagens de topologia local enviada pelos demais membros da $2_vizinhança$, um temporizador é usado para

[1]Essa determinação pode ser regular ou reativa a mudanças na topologia, possibilitando lidar com dinamismos da rede, uma possibilidade que pretendemos explorar como trabalho futuro.

Algoritmo 2 DISCOVER_2_NEIGHBORHOODTOPOLOGY()

```
1: 2_neighbohoodTopology ← EMPTY
2: localTopologyChanged ← TRUE
3: while TRUE do
4:     waitFor receivedMSG OR localTopologyChanged
5:     if receivedMSG then
6:         if receivedMSG is NEW then
7:             mark receivedMSG as not NEW
8:             update 2_neighbohoodTopology
9:             signal 2_neighbohoodTopology_updated
10:            if receivedMSG.ttl > 0 then
11:                decrement receivedMSG.ttl
12:                send receivedMSG to All Neighbors
13:            end if
14:        end if
15:    end if
16:    if localTopologyChanged then
17:        MSG ← createMSGAllNeighbors(ttl ← 1)
18:        send MSG to All Neighbors
19:        localTopologyChanged ← FALSE
20:    end if
21: end while
```

Algoritmo 3 CALCULATECENTRALITY()

```
1: Timer ← DISABLE
2: while TRUE do
3:     waitFor 2_neighbohoodTopology_updated OR Timer.done
4:     if 2_neighbohoodTopology_updated then
5:         restart Timer
6:     end if
7:     if Timer.done then
8:         calculate metric
9:         if metric has changed then
10:            signal metric_Changed
11:        end if
12:    end if
13: end while
```

limitar avaliações desnecessárias da métrica antes de todas as mensagens poderem ter sido recebidas. O Algoritmo 3 não envolve geração de mensagens e, portanto, não tem custo do ponto de vista de tráfego de mensagens na rede. Dependendo da definição da métrica utilizada para a centralidade, esta poderá ter um maior ou menor custo computacional. Entretanto, o cálculo da métrica utilizada fica restrito à sub-rede determinada pela $2_vizinhança$ do nó em questão. Isso faz com que em geral o custo computacional seja bastante modesto quando comparado ao custo de avaliar a mesma métrica considerando a rede completa. A determinação precisa da complexidade envolvida depende diretamente da métrica escolhida. Por exemplo, para a métrica de volume da $2_vizinhança$ mencionada anteriormente, a complexidade computacional é $O(|2_vizinhança_n|)$, onde $|2_vizinhança_n|$ é o número de nós existentes na $2_vizinhança$ centrada no nó n.

3.2. Localização distribuída de nós críticos

Uma vez determinados valores de centralidade para cada nó da rede, um desafio é localizar os nós críticos da rede de forma distribuída. Para isso, primeiramente, introduzimos uma definição para *nós críticos* e *nós semi-críticos*. O processo de determinação de nós críticos é baseado na seleção do nó de maior valor de centralidade de uma $2_vizinhança$. Nesse processo (apresentado detalhadamente adiante), cada nó de cada $2_vizinhança$ seleciona o nó de mais alto valor de centralidade de sua $2_vizinhança$. Para que um nó seja considerado um nó crítico, é necessário que todos os nós de sua $2_vizinhança$ o tenham selecionado como o nó de maior valor de centralidade. Como isso implica que mesmo os

nós da periferia da $2_vizinhança$ selecionaram o nó crítico como sendo o nó de maior centralidade visível para eles, ocorre que todo nó crítico é um máximo local de centralidade em um raio de 4 saltos, e em particular que o nó de maior centralidade da rede será necessariamente um nó crítico. Quando um nó seleciona a si próprio como sendo o nó de mais alta centralidade de sua $2_vizinhança$, ele é considerado um nó semi-crítico. Um nó pode ser semi-crítico sem ser um nó crítico. Esses nós semi-críticos possuem um papel importante na localização de nós críticos conforme será detalhado adiante.

Após determinados os nós críticos, cada nó da rede está associado a um e somente um nó critico. Isso ocorre porque cada nó da rede tem um nó em sua $2_vizinhança$ que ele considera como de maior centralidade. Considerando essa relação como transitiva, teremos uma sequência de valores crescentes até chegar a um nó crítico. Podemos então usar o conceito de nó crítico associado a cada nó da rede. Esse conceito é utilizado para fazer com que cada nó crítico da rede possa ter conhecimento de todos os demais nós críticos existentes na rede. Para que isso aconteça, cada nó crítico no momento em que descobre seu *status* de crítico, informa aos nós que o selecionaram como crítico (todos os nós de sua $2_vizinhança$) e cada um desses nós propaga esta informação para os nós que o selecionaram como o nó de maior centralidade. Ao final desse processo, cada nó da rede conhece o nó crítico ao qual está associado.

O Algoritmo 4 apresenta uma visão geral dos passos necessários para localização dos nós críticos da rede. A seguir, apresentamos a parte do algoritmo que permite a cada nó da rede identificar o nó crítico ao qual este está associado e também a parte do algoritmo para que cada nó crítico conheça todos os demais nós críticos de rede.

Algoritmo 4 DANCE_CRITICALNODELOCATION()

```
1: {at each node n in the network}
2: {define shared variable to contain the id of the critical node associated to this node.}
3: define associatedCriticalNode
4: {launch process that identifies the critical node associated to this node.}
5: fork IDENTIFYCRITICALNODE()
6: {launch process that makes critical nodes known to each other.}
7: fork COMPARECRITICALNODE()
```

3.2.1. Identificação de nós críticos

O Algoritmo 5 é responsável pela identificação do nó crítico associado a cada nó da rede. Para tanto, é necessário primeiramente que cada um dos nós críticos e semi-críticos da rede tome conhecimento desse seu *status*, o que é alcançado pelo Algoritmo 6 disparado logo no início (linha 1) do Algoritmo 5. Assim, analisaremos inicialmente o Algoritmo 6 e, em seguida, o restante do Algoritmo 5.

No Algoritmo 6, cada nó envia seu valor de centralidade para os demais nós de sua $2_vizinhança$. Quando cada nó recebe os valores de centralidade de todos os outros nós de sua $2_vizinhança$, o mesmo seleciona o nó com o maior valor de centralidade e notifica este nó de que ele é o nó de maior centralidade visível para si. Caso o nó tenha selecionado a si mesmo como o nó de maior centralidade da sua $2_vizinhança$, ele imediatamente se identifica como um nó semi-crítico. Cada nó então contabiliza o número de seleções recebidas. Se um determinado nó for selecionado por todos os nós de sua $2_vizinhança$ como sendo o de maior centralidade, este nó é identificado como um nó crítico. Caso haja empate no valor de centralidade, a identidade dos nós é usada

Algoritmo 5 IDENTIFYCRITICALNODE()

```
1: fork CRITICALNODESELECTION()
2: while TRUE do
3:     waitFor nodeCritical OR nodeSemiCritical OR Received criticalNodeIdMSG
4:     if nodeCritical then
5:         criticalNodeIdMSG ← createCriticalNodeIdMSG(thisNode)
6:         send criticalNodeIdMSG to All nodes that selected thisNode
7:     end if
8:     if nodeSemiCritical then
9:         criticalNodeId ← locateAssociatedCriticalNode(thisNode)
10:        criticalNodeIdMSG ← createCriticalNodeIdMSG(criticalNodeId)
11:        send criticalNodeIdMSG to All nodes that selected thisNode
12:    end if
13:    if Received criticalNodeIdMSG then
14:        associatedCriticalNode ← criticalNodeIdMSG.Id
15:        send criticalNodeIdMSG to All nodes that selected thisNode
16:        signal associatedCriticalNodeIdentified
17:    end if
18: end while
```

como critério de desempate, garantindo assim a existência de um único nó crítico em certa região. O Algoritmo 6 necessita que cada um dos nós da rede envie seu valor de centralidade para todos os seus $2_vizinhos$ e, portanto, é semelhante ao Algoritmo 2 de descoberta da topologia de $2_vizinhança$ de cada nó. Assim, pela mesma argumentação apresentada na Seção 3.1.1 para o Algoritmo 2, a complexidade de mensagens e complexidade de tempo esperadas para o Algoritmo 6 são $O(n \times d_m^2)$ e $O(1)$, respectivamente, onde n é o número de nós e d_m é o grau médio da rede. Através do Algoritmo 6, cada nó da rede terá ciência se ele mesmo é ou não um nó crítico ou semi-crítico, e necessariamente o nó com a maior centralidade da rede será um nó crítico. Dependendo da topologia da rede, é possível haver mais do que um nó crítico na rede.

Uma vez que cada nó crítico ou semi-crítico esteja ciente de sua condição (Algoritmo 6), podemos analisar o restante do Algoritmo 5, responsável por fazer com que cada nó da rede conheça o nó crítico associado a ele. Quando um nó percebe ser um nó crítico, ele notifica este fato a todos os nós de sua $2_vizinhança$ que o selecionaram como sendo o nó de maior centralidade visível a estes. Por sua vez, quando um nó recebe a notificação com a identidade do nó crítico, ele armazena a identidade do nó crítico como sendo o nó crítico associado a ele e repassa a mensagem de notificação para todos os nós que o selecionaram como nó de maior centralidade, já que necessariamente o mesmo nó crítico também será o nó crítico associado a estes nós. Assim, a associação de nó crítico se dá de forma transitiva através da seleção de nó com maior centralidade. Quando um nó percebe ser semi-crítico, ele localiza o nó crítico associado a ele utilizando o Algoritmo 7 e propaga esta informação para os nós que o selecionaram, que por sua vez irão também repassá-la. É interessante notar que este tratamento diferenciado dos nós semi-críticos é necessário porque a mensagem propagada a partir do nó crítico não chegará até um nó semi-crítico. Isso decorre do nó semi-crítico por definição selecionar a si próprio e, em consequência, não receber a informação propagada pelo nó crítico.

Para analisar a complexidade de mensagens do Algoritmo 5, devemos notar que cada nó receberá no máximo uma mensagem. Isso ocorre porque os nós críticos e semi-críticos não recebem nenhuma mensagem e todos os demais nós recebem uma única mensagem originada pelo nó que cada um deles selecionou como sendo o de maior centralidade. Como cada mensagem passará por no máximo tantos saltos quanto for o diâmetro da rede, podemos portanto concluir que a complexidade de mensagens desse Algoritmo

Algoritmo 6 CRITICALNODESELECTION()

```
1: Timer ← DISABLE
2: while TRUE do
3:     waitFor metric_Changed OR Received metricMSG OR Timer.done OR Received selectMSG
4:     if metric_Changed then
5:         metricMSG ← createMetricMSG(ttl ← 1)
6:         send metricMSG to All Neighbors
7:     end if
8:     if Received metricMSG then
9:         restart Timer
10:        originNode.metric ← metricMSG.metric
11:        if originNode.metric > thisNode.metric AND metricMSG.ttl > 0 then
12:            decrement metricMSG.ttl
13:            send metricMSG to All Neighbors
14:        end if
15:        if Received metricMSG from All Nodes in 2_neighborhood then
16:            Timer ← DISABLE
17:            selectedNode ← node with highest received metric
18:            if selectedNode ≠ thisNode then
19:                selectMSG ← createSelectionMSG(thisNode)
20:                send selectMSG to selectedNode
21:            else
22:                signal nodeSemiCritical
23:            end if
24:        end if
25:    end if
26:    if Timer.done AND Not Received metricMSG from ALL Nodes then
27:        send requestMSG to all missing Nodes
28:    end if
29:    if Received selectMSG then
30:        Account received selectMSG
31:        if Received selectMSG from All Nodes in 2_neighborhood then
32:            signal nodeCritical
33:        end if
34:    end if
35: end while
```

é $O(n \times Dg)$, onde Dg é o diâmetro da rede. Para determinar a complexidade de tempo, podemos notar que também não ocorrem ciclos na propagação da informação de identidade do nó crítico associado. Dessa forma, o caminho mais longo que pode ser percorrido será o diâmetro da rede. Podemos, portanto, concluir que a complexidade de tempo será $O(Dg)$ passos discretos, onde Dg é o diâmetro da rede.

O Algoritmo 7 objetiva localizar o nó crítico associado a qualquer nó da rede. Partindo de um nó qualquer da rede, consideramos o nó de maior centralidade selecionado por este. Se o nó selecionado for diferente dele mesmo, este nó não é um nó semi-crítico e assim o processo de localização continua recursivamente a partir do nó selecionado pelo nó corrente. Caso o nó corrente seja semi-crítico (ou seja, ele percebe a si próprio

Algoritmo 7 LOCATEASSOCIATEDCRITICALNODE($currentNode$)

```
Input: currentNode
Output: criticalNode
1: waitFor node selected by currentNode Is DEFINED
2: highCNode ← node selected by currentNode
3: if currentNode is NOT semi-critical then
4:     return LOCATEASSOCIATEDCRITICALNODE(highCNode)
5: else
6:     if currentNode is critical node then
7:         return currentNode
8:     else
9:         nextNode ← GETNODEKNOWSHIGHERC(currentNode)
10:        return LOCATEASSOCIATEDCRITICALNODE(nextNode)
11:    end if
12: end if
```

como o nó de maior centralidade em sua $2_vizinhança$), este nó também pode ser um nó crítico. Se isso ocorrer, o nó crítico foi localizado e o algoritmo retorna sua identidade. Caso contrário, isso significa que pelo menos um nó na $2_vizinhança$ do nó semi-crítico selecionou um outro nó como o nó de maior centralidade. Assim, esse nó de maior centralidade não está na $2_vizinhança$ do nó semi-crítico, mas está na $2_vizinhança$ de um dos nós que participam da $2_vizinhança$ do nó semi-crítico. Nesse caso, o nó semi-crítico escolhe o nó de maior centralidade que tenha selecionado outro nó que não seja o nó semi-crítico e o algoritmo continua recursivamente a partir deste nó. É importante notar que o caminho percorrido pelo algoritmo é acíclico e totalmente determinístico.

3.2.2. Comparação de nós críticos

No Algoritmo 8, cada nó compara a identidade de seu nó crítico associado com a identidade do nó crítico associado de seus vizinhos. No caso de vizinhos possuírem diferentes nós críticos, o nó corrente envia esta informação ao seu nó crítico associado, que desta forma toma conhecimento da existência de um outro nó crítico na rede. Toda vez que um nó crítico toma conhecimento de um outro nó crítico, ele envia a esse nó crítico a lista de nós críticos que ele já conhecia e notifica esses nós críticos já conhecidos da existência de um outro nó crítico. Considerando a rede conectada, após a convergência do algoritmo cada nó crítico da rede conhecerá a identidade dos demais nós críticos existentes.

Algoritmo 8 COMPARECRITICALNODE()

```
1: while TRUE do
2:     waitFor associatedCriticalNodeIdentified OR Received compareCriticalNodeIdMSG OR Received
       newCriticalNodeIdMSG
3:     if associatedCriticalNodeIdentified then
4:         compareCriticalNodeIdMSG ← createNewCriticalNodeIdMSG(associatedCriticalNode)
5:         send compareCriticalNodeIdMSG to All direct neighboors
6:         for all Enqueued compareCriticalNodeIdMSG do
7:             if associatedCriticalNode ≠ compareCriticalNodeIdMSG.criticalNode then
8:                 newCriticalNodeIdMSG ← createNewCriticalNodeIdMSG(compareCriticalNodeIdMSG.criticalNode)
9:                 send newCriticalNodeIdMSG to associatedCriticalNode
10:            end if
11:        end for
12:    end if
13:    if Received compareCriticalNodeIdMSG then
14:        if associatedCriticalNode is Defined then
15:            if associatedCriticalNodeIdentified ≠ compareCriticalNodeIdMSG.criticalNode then
16:                newCriticalNodeIdMSG ← createNewCriticalNodeIdMSG(compareCriticalNodeIdMSG.criticalNode)
17:                send newCriticalNodeIdMSG to associatedCriticalNode
18:            end if
19:        else
20:            Enqueue compareCriticalNodeIdMSG
21:        end if
22:    end if
23:    if Received newCriticalNodeIdMSG AND thisNode.isCritical then
24:        if newCriticalNodeIdMSG.Id NOT in criticalNodeList then
25:            send criticalNodeList to newCriticalNodeIdMSG.Id
26:            send newCriticalNodeIdMSG to All nodes in criticalNodeList
27:            add newCriticalNodeIdMSG.Id to criticalNodeList
28:        end if
29:    end if
30: end while
```

4. Resultados Experimentais

Nesta seção analisamos o desempenho do DANCE em termos de efetividade para avaliação de centralidade e complexidade para aplicabilidade prática. Também ilustramos a aplicabilidade de nossa proposta DANCE em uma rede de larga escala.

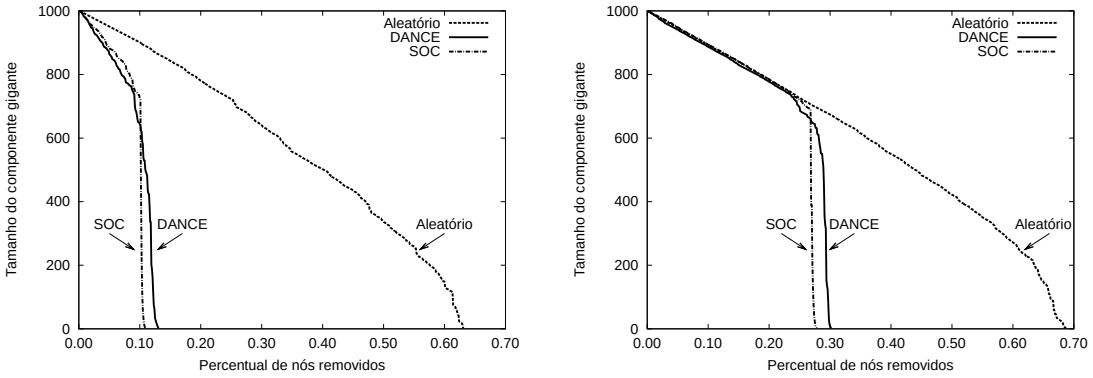
4.1. Efetividade

Nós comparamos a efetividade do DANCE com outro algoritmo recentemente proposto chamado SOC (*Second Order Centrality*) [Kermarrec et al., 2011] para avaliação distribuída de centralidade que pode ser usado em redes onde a topologia é desconhecida. SOC baseia-se em um caminho aleatório perpétuo onde se mede o desvio padrão do intervalo de tempo entre visitas consecutivas do caminho aleatório a cada nó. Apesar de eficaz para avaliação distribuída de centralidade, SOC apresenta dificuldades para determinar o número de passos necessários para que se obtenha valores consistentes de centralidade para todos os nós da rede. O número de passos deve ser grande o suficiente para que o caminho aleatório visite cada nó da rede um número de vezes suficientemente grande para que o desvio padrão do tempo entre as visitas possa ser bem determinado. Como a qualidade da determinação desse parâmetro depende do número de amostras, é de se esperar que sejam necessárias várias visitas do caminho aleatório a cada um dos nós da rede. Também é importante notar que o número de passos necessários para haver convergência depende da topologia da rede.

Na avaliação realizada foram utilizados dois tipos de redes sintéticas com 1000 nós: redes de escala-livre e aleatórias. As redes de escala-livre seguem o modelo Barabási-Albert (BA) [Barabási e Albert, 1999] considerando em sua geração duas conexões por cada novo nó conectado. Essas redes BA por construção sempre tem 1996 arestas, tendo portanto um grau médio igual à 3,992. As redes aleatórias seguem o modelo Erdös-Renyi (ER) [Erdos e Renyi, 1959] com probabilidade de conexão de 0,0045 (para grau médio ser similar ao das redes BA). Nessas redes ER o número de arestas não é fixo e depende de um processo aleatório de conexões entre os nós. Entretanto, devido ao número de nós e a probabilidade utilizada, o grau médio destas redes é aproximadamente 4.

Analisamos a efetividade da avaliação de centralidade dos nós da rede da seguinte maneira. Consideramos 100 redes sintéticas de 1000 nós seguindo o modelo BA e outras 100 redes seguindo o modelo ER, como descrito anteriormente. A partir de cada rede considerada, removemos o nó de maior centralidade indicado por cada algoritmo e avaliamos o impacto dessa remoção sobre a rede, verificando o número de nós restantes no maior componente conexo resultante — chamado de componente gigante. Espera-se que nós de alta centralidade — por definição, os mais importantes da rede ou nós críticos — indicados por cada algoritmo quando removidos causem a uma maior degradação da rede que se reflete na diminuição do componente gigante. O algoritmo DANCE prevê que se possa usar diferentes métricas para avaliação da centralidade na $2_vizinhança$ de cada nó, possibilitando a obtenção de diferentes tipos de centralidade. Nessa avaliação utilizamos por simplicidade, porém sem perda de efetividade, a métrica de calcular o volume de cada $2_vizinhança$ para a determinação da medida de centralidade em cada nó.

Na Figura 1 apresentamos o impacto da remoção de nós sobre o componente gigante de uma rede BA e uma rede ER representativas considerando a avaliação de centralidades oriundas das abordagens DANCE e SOC. Para a rede BA, utilizando o algoritmo DANCE para determinar os nós críticos a serem removidos, o componente gigante ficou reduzido a 20% de seu tamanho original — rede *fragmentada* — após 11,9% dos nós serem removidos. Utilizando o algoritmo SOC, o mesmo resultado foi obtido após 10,7% de nós removidos. No caso da rede ER, foram necessárias 28,9% dos nós re-

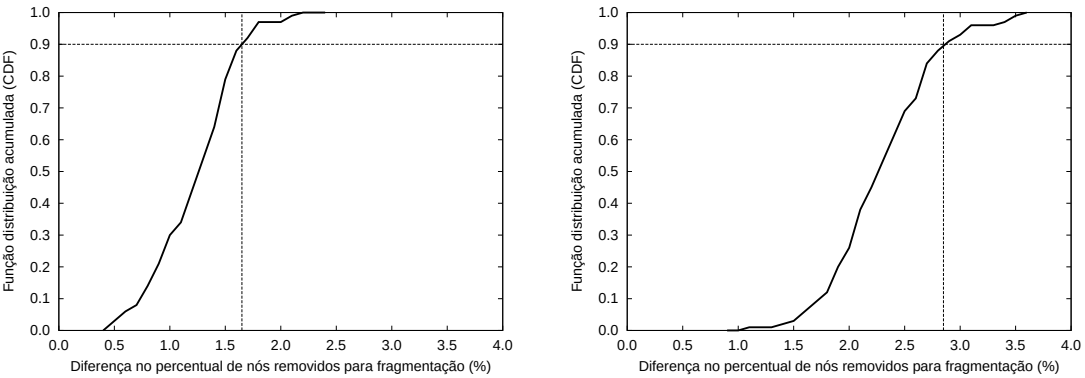


(a) Rede BA (b) Rede ER

Figura 1. Impacto da remoção de nós sobre o componente gigante da rede.

movidos utilizando o DANCE e 27,1% dos nós removidos utilizando o SOC. Para fim de comparação, incluímos ainda o impacto sobre o componente gigante obtido pela remoção aleatória dos nós da rede, demonstrando a efetividade do ataque realizado considerando as centralidades avaliadas tanto pelo SOC e DANCE. É importante notar que a efetividade do DANCE e SOC é bastante similar com DANCE necessitando de um pequeno número de nós removidos a mais para alcançar impacto igual do SOC no componente gigante.

A mesma análise do impacto da remoção de nós sobre o componente gigante foi realizada para 100 redes BA e 100 redes ER. Para todos os casos, os resultados são similares aos apresentados na Figura 1 para as duas redes BA e ER representativas. Na Figura 2 apresentamos a CDF da diferença entre o percentual necessário de nós removidos usando DANCE e SOC para fragmentação da rede (ou seja, redução do componente gigante a 20% do tamanho original). Observamos na Figura 2(a) que em 90% dos casos de redes BA a proposta DANCE necessita de menos de 1,6% de nós removidos a mais que SOC para fragmentar a rede; em 100% dos casos essa diferença é inferior a 2,5%. Para redes ER (Figura 2(b)), essa diferença é inferior a 2,9% e 3,6% em 90% e 100% dos casos, respectivamente. DANCE, portanto, alcança consistentemente uma efetividade bastante similar ao SOC ao avaliar a centralidade de nós da rede distribuidamente. No entanto, na Seção 4.2, mostraremos que essa efetividade similar é alcançada com uma complexidade de aplicabilidade significativamente menor do DANCE em relação ao SOC.



(a) Redes BA (b) Redes ER

Figura 2. Diferença entre DANCE e SOC no percentual de nós removidos para fragmentação.

4.2. Complexidade de aplicabilidade

Nesta subseção analisamos a complexidade para aplicação dos algoritmos SOC e DANCE na obtenção dos resultados de efetividade na avaliação de centralidade apresentados na Seção 4.1. Consideramos a complexidade de mensagens, tempo de convergência e complexidade computacional para obtenção dos resultados:

- **Complexidade de mensagens:** Como o algoritmo SOC somente calcula o valor de centralidade de cada nó da rede sem oferecer uma maneira para localizar nós de alta centralidade, em nossa avaliação consideraremos no DANCE apenas o número de mensagens utilizadas na fase de determinação da centralidade de cada nó. Nesse caso, a complexidade de mensagens esperada é de $O(n \times d_m^2)$ mensagens, onde n é o número de nós e d_m é o grau médio na rede. Consideramos as mesmas 100 redes BA e 100 redes ER da Seção 4.1. Kermarrec et al. [2011] indica que o uso de um único caminho aleatório torna difícil o estabelecimento de um critério de parada para o cálculo da centralidade e analisam que no pior caso $O(n^3)$ passos (mensagens), onde n é o número de nós na rede, são necessários para a avaliação distribuída de centralidade no SOC. Assim, para aplicação de SOC em redes de 1000 nós, Kermarrec et al. [2011] arbitram 2×10^6 passos (mensagens) do caminho aleatório. Logo, nós consideramos em nossas redes de 1000 nós a centralidade do SOC após 2×10^6 passos (mensagens) do caminho aleatório como [Kermarrec et al., 2011]. Em contraste, experimentalmente, o número médio de mensagens utilizadas pelo DANCE para calcular a centralidade de todos os 1000 nós de cada uma dessas redes foi de 47691 mensagens com desvio padrão de 3834 para redes BA e 29913 mensagens com desvio padrão de 1087 para redes ER. A variância apresentada decorre do fato que o número específico de mensagens utilizadas pelo DANCE depende da topologia da rede bem como também explica o maior número de mensagens e a maior variância em redes BA de escala livre. Na Figura 3 apresentamos esses resultados.

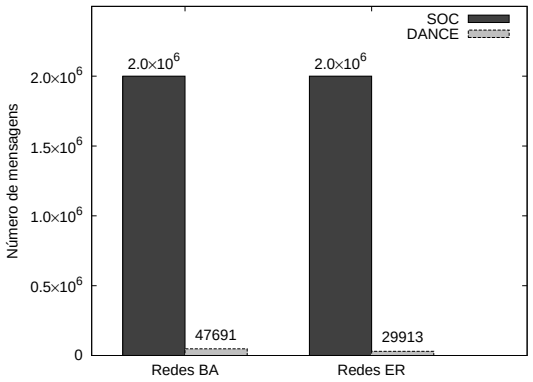


Figura 3. Número de mensagens para avaliação de centralidade.

- **Tempo de convergência:** No algoritmo DANCE o processo de troca de mensagens para determinação dos valores de centralidade é altamente paralelizado, sendo necessário apenas que as mensagens geradas por cada nó se propaguem por dois passos, fazendo com que o tempo necessário para conclusão do algoritmo seja limitado a dois passos ($O(1)$) e independente do número de nós da rede. Como o algoritmo SOC é baseado em um único caminho aleatório, todas as mensagens serão *sequenciais* no tempo, fazendo com que o tempo de execução do algoritmo

seja consideravelmente mais longo e dependente do número de nós. De acordo com Kermarrec et al. [2011], a complexidade de passos do SOC é $O(n^3)$, onde n é o número de nós da rede.

- **Complexidade computacional:** No caso do algoritmo DANCE utilizando a métrica adotada neste trabalho, o cálculo necessário se resume à determinação do volume da $2_vizinhança$ de cada nó da rede. No caso do algoritmo SOC, é necessário calcular o desvio padrão do tempo entre as visitas sucessivas do caminho aleatório a cada nó da rede. Em ambos os casos, podemos notar que o custo computacional é modesto e totalmente distribuído entre os nós da rede.

4.3. Aplicabilidade do DANCE em redes de larga escala

De forma a ilustrar a aplicabilidade do algoritmo DANCE em redes de maior escala realizamos um experimento com um registro (*trace*) de rede real disponibilizado pela CAIDA[2] que representa uma amostra da conectividade ao nível de roteadores da Internet. Essa rede possui 190.914 nós, diâmetro 26, grau médio 6,34 e o nó de maior grau tem 1071 conexões; além de apresentar características de escala livre na distribuição do grau dos nós (ou seja, similar às redes sintéticas seguindo o modelo BA). É importante ressaltar que em uma rede de tal dimensão a aplicabilidade do algoritmo SOC [Kermarrec et al., 2011], apesar de distribuído, se torna inviável por sua alta complexidade de mensagens e elevado tempo de convergência, ambos de ordem $O(n^3)$. Em contraste, para calcular os valores de centralidade de todos os nós da rede utilizando o algoritmo DANCE foram necessárias aproximadamente 250 mensagens em média por nó. A Figura 4 apresenta a fragmentação do componente gigante dessa rede após remoção de nós indicados como de maior centralidade pelo DANCE e também uma remoção aleatória de nós para referência. Observamos comportamento similar ao obtido para redes BA sintéticas (cf. Figura 1(a)). Constatamos, portanto, que é possível aplicar o algoritmo DANCE a redes reais de maior porte obtendo os valores de centralidade efetivos para todos os nós utilizando um número relativamente pequeno de mensagens, em contraste com o estado da arte da literatura.

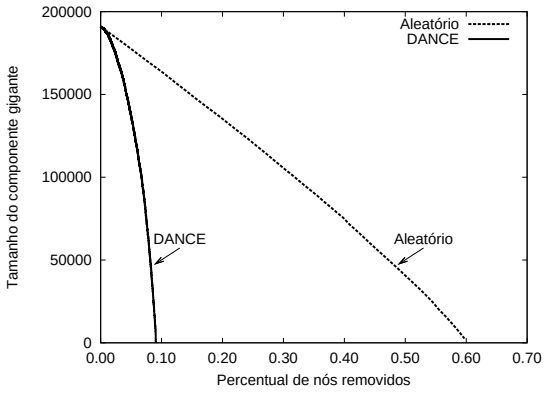


Figura 4. Algoritmo DANCE aplicado à fragmentação de uma rede real.

5. Conclusão

Neste trabalho apresentamos o algoritmo DANCE para avaliação e localização distribuída de centralidade de redes. As principais contribuições de nossa proposta são o cálculo descentralizado de centralidades de forma eficiente, podendo ser facilmente adaptado para

[2] http://www.caida.org/tools/measurement/skitter/router_topology/

obtenção de diferentes tipos de centralidade, e também um método para possibilitar a localização dos nós mais centrais da rede. Ambas as contribuições são efetuadas utilizando apenas informações da vizinhança a 2 saltos de cada nó sem a necessidade de conhecimento da topologia completa da rede. Isso é particularmente relevante considerando-se as dimensões das redes complexas em investigação atualmente. Nós avaliamos o algoritmo DANCE experimentalmente através de sua comparação com outro algoritmo recente de mesmo propósito. DANCE mostra-se simples, eficaz e eficiente, pois alcança uma efetividade na avaliação de centralidade de forma distribuída comparável ao estado da arte, porém com custo para sua aplicabilidade prática significativamente menor.

Dada a contribuição básica do DANCE em permitir avaliação e localização de centralidade de rede de forma distribuída, isso abre perspectivas para diferentes trabalhos futuros. Primeiro, podemos considerar o uso de diferentes definições de centralidade para calcular a centralidade local e avaliar seu impacto ao comparar com centralidades tradicionais. Segundo, dada a possibilidade de aplicar DANCE em redes de maior escala, pretendemos investigar redes reais distintas de larga escala, tais como redes sociais online e redes de distribuição de conteúdo, para analisar como a noção de centralidade pode trazer novos conhecimentos sobre a estrutura e comportamento dessas redes. Terceiro, a grande maioria das redes complexas atuais são dinâmicas. Pretendemos investigar como o algoritmo DANCE pode auxiliar na análise do comportamento dinâmico dessas redes através da avaliação e localização de centralidade.

Agradecimentos

Este trabalho foi parcialmente financiado pela FAPERJ, CNPq e MCT.

Referências

Barabási, A. e Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, 286(5439):509–512.

Borgatti, S. (2005). Centrality and network flow. *Social Networks*, 27(1):55–71.

Brandes, U. (2001). A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25(2):163–177.

Brin, S. e Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. In *Seventh International World-Wide Web Conference (WWW 1998)*.

Dinh, T., Xuan, Y., Thai, M., Park, E., e Znati, T. (2010). On approximation of new optimization methods for assessing network vulnerability. In *Proc. of the IEEE INFOCOM*. IEEE.

Erdos, P. e Renyi, A. (1959). On random graphs I. *Publ. Math. Debrecen*.

Freeman, L. (1977). A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41.

Kermarrec, A.-M., Le Merrer, E., Sericola, B., e Trédan, G. (2011). Second order centrality: Distributed assessment of nodes criticity in complex networks. *Computer Communications*, 34(5):619–628.

Lehmann, K. e Kaufmann, M. (2003). Decentralized algorithms for evaluating centrality in complex networks. Technical Report WSI-2003-10, Wilhelm Schickard Institut.

Nanda, S. e Kotz, D. (2008). Localized Bridging Centrality for Distributed Network Analysis. In *Proc. of Int. Conference on Computer Communications and Networks (ICCCN)*. IEEE.

Roteamento em Múltiplas Rotas com Agregação Dinâmica de Fluxos com Duração Conhecida em Redes WDM

Juliana de Santi[1], **André C. Drummond** [1], **Nelson L. S. da Fonseca**[1]

[1]Instituto de Computação – Universidade Estadual de Campinas (UNICAMP)
Caixa Postal 6167 – 13084-971 – Campinas – SP – Brasil

{santi,andred,nfonseca}@ic.unicamp.br

Abstract. *This paper introduces a scheme that employs holding time aware traffic grooming and multipath routing in WDM networks. High capacity requests require large amounts of bandwidth, which is usually greater than the capacity of a wavelength, hence, the need for routing that can provide multiple paths to meet the requested connection. However, such demands do not necessarily use all the available bandwidth at a wavelength allocated to them. Moreover, in real networks there are also many low speed connections that do not occupy the whole capacity of a wavelength. In this study, we present an algorithm to find multiple routes and aggregate traffic along these routes. To assess the performance of the proposed algorithm, comparisons with a multipath routing and with singlepath with traffic grooming are performed. Results obtained by simulation show improvement on performance when using jointly traffic grooming and multipath routing.*

Resumo. *Neste trabalho, emprega-se, de forma conjunta, agregação de fluxos com duração conhecida e roteamento por múltiplas rotas para atender demandas de alta capacidade em redes WDM. Essas demandas requerem grande quantidade de banda passante, o que é, geralmente, superior a capacidade de um comprimento de onda. Algoritmos de roteamento que possam prover múltiplos caminhos são, portanto, necessários para atender a conexão solicitada. Entretanto, tais demandas não necessariamente utilizam toda a banda disponível nos comprimentos de onda à elas alocados. Além disso, em redes operacionais existem muitas conexões de baixa velocidade. Assim, neste estudo, apresenta-se um algoritmo para encontrar múltiplas rotas e agregar tráfego nessas rotas. Para avaliar o desempenho do algoritmo proposto são realizadas comparações com um algoritmo de roteamento multi caminho sem agregação de tráfego e com um algoritmo de rota única com agregação de tráfego. Resultados obtidos, por meio de simulação, revelam melhoria no desempenho, ao se utilizar de forma conjunta, agregação de tráfego e roteamento por múltiplas rotas.*

1. Introdução

Aplicações de alto desempenho e aplicações científicas podem demandar transferências de quantidade de dados na ordem de *Tera-byte* ou até mesmo *Peta-bytes*. Uma vez que, devido ao volume de transmissão, são pequenas as chances de encontrar um número suficiente de comprimentos de onda no mesmo enlace de fibra óptica em uma única rota (*singlepath*), especialmente em redes ópticas com restrição de comprimento de onda, o

roteamento sobre múltiplas rotas (*multipath*) fornece a solução para atender demandas que requerem grande quantidade de banda passante.

A transmissão dos dados da conexão por vários enlaces de fibra através da abordagem *multipath* tem sido utilizada para minimizar o bloqueio devido a indisponibilidade de banda em uma única rota, i.e., *singlepath* [Chen et al. 2009b]. O roteamento *multipath* tira proveito da disponibilidade de banda passante ao longo de diferentes rotas da origem até o destino para satisfazer as demandas requisitadas. Como exemplo, cita-se que a demanda para 4λ de banda passante na Figura 1 que seria bloqueada na utilização de roteamento *singlepath* pode ser satisfeita através da alocação de 2λ no caminho P_1 e 2λ no caminho P_2, ao se utilizar roteamento *multipath*. Por serem rotas distintas, o tempo de transmissão ao longo de P_1 e P_2 pode ser diferente, sendo necessário armazenar as conexões de cada rota até que todos os dados alcancem o destino para poder re-sequenciá-los. Devido a restrições de disponibilidade em *fiber delay lines* (FDL) e ao custo elevado da buferização eletrônica, rotas distintas implicam em diferentes atrasos. A diferença entre os atrasos dos caminhos utilizados é chamada de atraso diferencial (*differential delay* - DD) [Ahuja et al. 2004].

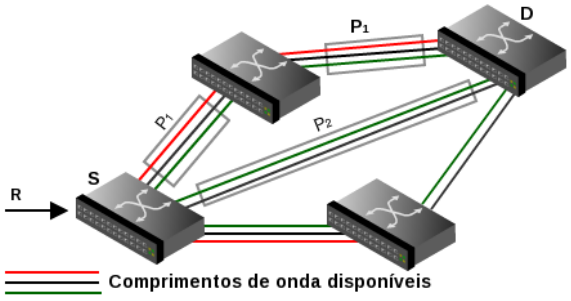


Figura 1. Exemplo de roteamento *Multipath*. R requisitando 4λ é aceita através das rotas P_1 e P_2, cada uma fornecendo dois caminhos ópticos [Chen et al. 2009b].

O mecanismo de roteamento *multipath* introduzido em [Chen et al. 2009b] foi uma das primeiras tentativas de utilizar roteamento por múltiplas rotas para atender demandas de alta capacidade. Mostrou-se que este mecanismo é uma solução atrativa para lidar com demandas que requisitam quantidades extremas de banda passante. Entretanto, as conexões que compõem uma solução *multipath* podem não ocupar a capacidade total dos comprimentos de onda alocados, o que leva à ineficiência na utilização dos recursos. Ademais, em redes ópticas operacionais, há uma quantidade considerável de conexões que demandam quantidade de banda passante menor do que a capacidade de um comprimento de onda, i.e., demandas de baixa velocidade(*sub-wavelengths*) que geram subutilização da capacidade dos caminhos ópticos estabelecidos. Este problema pode ser amenizado pelo uso de técnicas de agregação de tráfego [Dutta and Rouskas 2002], que agregam demandas de baixa velocidade em circuitos ópticos de alta capacidade para maximizar a utilização dos comprimentos de onda e minimizar o custo.

Neste trabalho, investiga-se o potencial de uma solução de roteamento por múltiplas rotas com agregação de tráfego que leva em consideração o conhecimento do tempo de duração das conexões. Assume-se que a duração das conexões é conhecida, uma vez que nos contratos de *Service Level Agreement* para demandas

de circuitos ópticos, disponibiliza-se, usualmente, esta informação [Santi et al. 2010a, Santi et al. 2010b, Tornatore et al. 2008]. Para agregar tráfego em múltiplas rotas, de forma a minimizar o bloqueio de chamadas, o algoritmo proposto é formulado como um problema de programação linear inteira (ILP). Para avaliar os benefícios do algoritmo, compara-se os resultados de simulação com os valores obtido pela abordagem de roteamento *multipath* sem agregação de tráfego bem como comparam-se os resultados com os do algoritmo de roteamento *singlepath* com agregação de tráfego. Os resultados indicam que o algoritmo proposto minimiza tanto a taxa de bloqueio de banda passante quanto o custo de utilização da rede e atende, também, os requisitos de atraso diferencial (*differential delay*).

Este trabalho é organizado da seguinte forma. A Seção 2 apresenta trabalhos correlatos da literatura recente. Na Seção 3, apresenta-se uma breve descrição do algoritmo que agrega conexões com duração conhecida, de forma a balancear a carga na rede. A Seção 4 introduz a formulação ILP para agregação de tráfego por múltiplas rotas (*Multipath*) e a formulação para agregação de tráfego com roteamento de rota única (*Singlepath*), ambas empregando o algoritmo HTBalancing como política de agregação dinâmica de tráfego. Na Seção 5, os resultados numéricos da avaliação do algoritmo proposto. Por fim, a Seção 6 apresenta as conclusões deste trabalho.

2. Trabalhos Relacionados

Em [Tornatore et al. 2008], foi proposto um algoritmo que utiliza o conhecimento do tempo de duração das conexões para agregação de tráfego. A estratégia em [Santi et al. 2010a, Santi et al. 2010b] introduz o balanceamento das conexões com duração conhecida para minimizar a taxa de bloqueio de banda e promover justiça entre as conexões. Em [Cavdar et al. 2009], o tempo de duração das conexões é empregado para estabelecer caminhos alternativos às conexões em caso de falha de enlace (proteção). A informação sobre a duração das conexões é, também utilizada para o aprovisionamento de demandas *multicast* [Munir et al. 2009]. O estudo em [Zhang et al. 2010] emprega demandas cientes de duração para minimizar o consumo de energia no núcleo de redes.

Questões relacionadas a proteção com agregação de tráfego e multiplexação inversa em roteamento com múltiplas rotas são investigadas em [Huang et al. 2010]. Os autores de [Das et al. 2011] propõe uma abordagem de aprovisionamento *multipath* na qual a confiabilidade é determinada em função da expectativa do usuário. Este mecanismo implementa um algoritmo de custo mínimo que utiliza o mínimo possível de banda passante para satisfazer uma conexão. Em [Lee and Modiano 2009], o problema de proteção é investigado, considerando falhas aleatórias em enlaces potencialmente correlacionadas. O problema foi formulado para encontrar um par de caminhos, de forma a minimizar a probabilidade de falha conjunta destes caminhos. Em [Zhang and Perrig 2010], foi proposta uma abordagem de seleção de múltiplos caminhos orientado à disponibilidade. Para tal, o histórico da disponibilidade do caminho é utilizado para descobrir e evitar caminhos com falhas correlacionadas. O objetivo é selecionar um conjunto de caminhos com falhas independentes, de modo a minimizar as chances de que todos os caminhos selecionados falhem simultaneamente.

Em [Chen et al. 2009b], uma abordagem de roteamento *multipath* é empregada para subsidiar aplicações de alto desempenho que requerem banda passante superior

a capacidade de um comprimento de onda. O trabalho em [Chen et al. 2009b] é uma das primeiras tentativas de formular o problema de roteamento *multipath* e o problema de atribuição de comprimento de onda como um modelo de otimização baseado em Programação Linear Inteira (ILP). Apesar das limitações decorrentes da complexidade da ILP, a solução com no máximo duas rotas é capaz de minimizar a probabilidade de bloqueio e aumentar a disponibilidade para as conexões. Em [Chen et al. 2009a], o roteamento *multipath* é estendido para redes multi-domínio, na qual são discutidos os esquemas de roteamento *multipath* fim-a-fim e roteamento *multipath* por domínio. Além dos benefícios do roteamento *multipath* em redes multi-domínio, os autores de [Chen et al. 2009a] analisam, também, o custo do roteamento *multipath* como uma função dos requisitos de buferização por reordenamento de pacotes.

3. Agregação de Tráfego com balanceamento de carga entre conexões com duração conhecida

Contratos de *Service Level Agreements* (SLAs) para efetivação de canais em redes ópticas contém, usualmente, a informação sobre a duração das conexões; informação essa usada para o desenvolvimento de estratégias para a utilização eficiente dos recursos disponíveis em [Fawaz et al. 2004]. O tempo de duração das conexões é empregado em [Tornatore et al. 2008] para a tomada de decisões sobre agregação de tráfego. Em [Santi et al. 2010a, Santi et al. 2010b], os autores introduziram o algoritmo HTBalancing que minimiza a probabilidade de bloqueio das demandas futuras em redes WDM. O algoritmo HTBalancing leva em consideração o tempo de duração das conexões, de forma a proporcionar balanceamento de carga através dos caminhos existentes.

O algoritmo HTBalancing é baseado em dois critérios: o primeiro refere-se a reutilização dos caminhos já estabelecidos na rede, aos quais são atribuídos custos menores do que o custo de novos caminhos. Os custos dos caminhos existentes consideram a sobreposição entre o tempo de vida dos caminhos ópticos e o tempo de duração das chamadas a serem aprovisionadas. O segundo critério contabiliza a disponibilidade de banda passante nos caminhos. Caminhos com menor quantidade de banda passante devem ter custo mais elevado, de forma a evitar sua utilização e, consequentemente, prevenir a formação de gargalos. Caminhos com grande disponibilidade de banda passante, por sua vez, apresentam menor custo, o que os torna bons candidatos para atender as demandas solicitadas.

Para realizar o balanceamento de carga, o algoritmo HTBalancing usa a seguinte função de custo de utilização de caminhos [Santi et al. 2010a, Santi et al. 2010b]:

$$C(p_i) = (c_ht(p_i) \times \alpha) + \left(\frac{1}{bw(p_i)} \times (1-\alpha)\right) \tag{1}$$

$$c_ht(p_i) = \begin{cases} hp_i \times \epsilon & \text{Se } L_i \geq H \\ hp_i \times \epsilon + hp_i \times \Delta_t & \text{Se } L_i < H \\ hp_i \times H & \text{novo caminho} \end{cases} \tag{2}$$

onde:

- $C(p_i)$ é o custo de utilização do caminho p_i;

- $c_ht(p_i)$ determina a contribuição do tempo de duração da conexão para o custo do i-ésimo caminho (Equação 2);
- α define o peso do tempo de duração da conexão para o custo do caminho;
- bw é a banda passante disponível em p_i;
- $(1 - \alpha)$ define o peso da disponibilidade de banda para o custo do caminho;
- H é o tempo de duração da conexão;
- L_i é o tempo de vida do i-ésimo caminho;
- $\Delta_t = H - L_i$;
- $\epsilon = 10^{-5}$ uma constante;
- hp_i e o número de saltos ao longo do i-ésimo caminho.

O termo $\frac{1}{bw(p_i)}$ na Equação 1 é substituído por 1 ao considerar o custo de novos caminhos evitando-se, assim, o estabelecimento desnecessário de novos caminhos, uma vez que a banda disponível neles é a capacidade total de um comprimento de onda.

4. Agregação de Tráfego Baseado em Roteamento por Múltiplas Rotas

Apesar de se mostrar uma abordagem eficiente para atender demandas de alta capacidade através de roteamento por múltiplas rotas, o algoritmo apresentado em [Chen et al. 2009b] não aborda questões relacionadas ao desperdício de banda residual dos caminhos. Na próxima seção, apresenta-se uma abordagem de agregação tráfego baseada em roteamento por múltiplas rotas, a qual considera o tempo de duração das conexões e a banda disponível nos caminhos já estabelecidos na rede [Santi et al. 2010a, Santi et al. 2010b]. O algoritmo proposto deve atender tanto demandas de alta capacidade quanto demandas de baixa velocidade, utilizando de forma eficiente a banda passante disponível. É apresentada, também, a formulação do problema de agregação de tráfego baseado em roteamento de rota única, utilizada para avaliar os benefícios do trabalho proposto.

4.1. Formulação para Múltiplas Rotas

Nesta seção, apresenta-se uma formulação ILP para agregação de tráfego com roteamento por múltiplas rotas. A representação da rede WDM, usada como entrada para a formulação ILP, é dada por um grafo $G(V, E)$, onde V define os nós na rede e E representa o conjunto dos enlaces de fibra. Cada enlace $e \in E$ possui uma capacidade $c_{i,j}$ dada pelo número de comprimentos de onda disponível na fibra, denotada por $W_{i,j}$. Um caminho livre de ciclo p em G é definido por uma lista de nós (s,v_1,...,v_n,d), onde s e d são, respectivamente, a origem e o destino. s, d, $v_i \in V$ e (s,v_i), (v_i,v_{i+1}), (v_n,d) $\in E$. A ILP é alimentada, ainda, pela topologia virtual da rede, a qual descreve os caminhos já estabelecidos e a banda passante disponível nestes caminhos.

O algoritmo proposto tenta encontrar tantas rotas quanto forem necessárias (*multipath*) para atender uma conexão requisitada a um custo mínimo. Para o cálculo do custo leva-se em consideração as informações sobre o tempo de duração da conexão e a quantidade de banda passante disponível no caminho, conforme detalhado na Seção 3.

Para formular a ILP, as seguintes variáveis são definidas:

- $F_{p,i,j,w}$: variável binária que assume o valor 1 se o comprimento de onda w no enlace i, j é utilizado no caminho p; caso contrário seu valor é 0;

- $X_{p/lp,w}$: variável binária que indica se o comprimento de onda w é utilizado ou não em um caminho. O caminho pode ser novo, p, ou já estabelecido, lp;
- C_{lp}: variável de valor real que representa o custo do caminho estabelecido lp definido pela Equação 1;
- B_{lp}: variável inteira que representa a banda disponível no caminho estabelecido lp;
- R: variável inteira que define a banda requisitada pela chamada (OC-3; OC-12; ...OC-979);
- H: variável inteira que determina o tempo de duração de uma conexão;
- λ: representa a capacidade de um comprimento de onda;
- $\alpha = 0.5$: uma constante usada para calcular o custo de um caminho a ser estabelecido ([Santi et al. 2010a, Santi et al. 2010b]).

O algoritmo para agregação de tráfego baseado em roteamento por múltiplas rotas é formulado conforme segue:

$$Min \sum_{lp,w} X_{lp,w} \times C_{lp} + \sum_{p,w} \left(X_{p,w} \times (1-\alpha) + \sum_{i,j} F_{p,i,j,w} \times H \times \alpha \right) \tag{3}$$

$$\sum_{i} F_{p,i,j,w} - \sum_{k} F_{p,j,k,w} = \begin{cases} -X_{p,w} & \text{Se } i = s \\ X_{p,w} & \text{Se } i = d \quad \forall\ j,w,p \\ 0 & \text{caso contrário} \end{cases} \tag{4}$$

$$\sum_{lp,w} X_{lp,w} \times B_{lp} + \sum_{p,w} X_{p,w} \times \lambda \geq R \tag{5}$$

$$\sum_{p} F_{p,i,j,w} \leq 1 \quad \forall\ w,i,j \tag{6}$$

p e $lp \in P$ e P é o conjunto dos caminhos candidatos a atender a conexão.
$F_{p,i,j,w} \in \{0,1\}$.
$X_{p/lp,w} \in \{0,1\}$.
$w = [1,2,...W_{i,j}]$, $i,j \in E$.

A função objetivo na Equação (3) visa encontrar rotas entre a origem s e o destino d com custo mínimo. O custo de utilização de cada caminho que compõe um rota é definido em função do tempo de duração da conexão e da disponibilidade de banda passante naquele caminho. O custo dos caminhos já estabelecidos, representados no primeiro termo da função objetivo, é calculado de acordo com a Equação (1). Rotas com novos caminhos são considerados no segundo termo da Equação (3). Para determinar o custo de um novo caminho, o algoritmo HTBalancing substitui o termo $\frac{1}{bw}$ (Equação (1)) por 1 e o valor de $c_ht(p_i)$ é definido de acordo com a condição três da Equação (2), ou seja, $c_ht(p_i) = hp_i \times H$. Desta forma, o custo de cada novo caminho de uma determinada rota é dado por $C(p_i) = ((1-\alpha) + (hp_i \times H \times \alpha))$. Por sua vez, o número de saltos hp_i em um caminho é representado na função objetivo através da soma de $F_{p,i,j,w}$ sobre seus índices i e j.

A restrição de conservação de fluxo, representada na Equação (4), assegura que a quantidade de dados que entra é igual a quantidade de dados que sai em cada um dos nós selecionados para estabelecer um caminho entre a origem e o destino.

A restrição de capacidade é definida pela Equação (5) e determina que a soma da banda disponível nos caminhos estabelecidos (primeiro termo) mais a banda nos caminhos a serem criados (segundo termo) deve ser maior ou igual a banda requisitada (R) pela conexão a ser atendida.

A restrição de continuidade de comprimento de onda, que determina que um caminho óptico deve usar o mesmo comprimento de onda em todos os enlaces desde a origem até o destino [Rouskas and Perros 2002], é garantida pelos índices w em todas as variáveis. A restrição de comprimento de onda distinto, que estabelece que todos os caminhos ópticos usando um mesmo enlace devem ter diferentes comprimentos de onda associados [Rouskas and Perros 2002], é descrita pela Equação (6).

4.2. Formulação do problema com uma única Rota

Para avaliar as vantagens da utilização de uma abordagem de roteamento por múltiplas rotas ao invés de uma abordagem de roteamento por uma única rota (*singlepath*), apresenta-se, também, uma formulação de roteamento por rota única. Para tal, as seguintes variáveis são definidas:

- $F_{i,j,w}$: variável binária que determina se um comprimento de onda w no enlace i, j é utilizado ou não pela solução;
- T_w: variável binária que define a utilização ou não do comprimento de onda w;
- X_{lp}: variável binária que determina se a rota já estabelecido lp é utilizada ou não na solução;
- C_{lp}: valor real que representa o custo de uma rota já estabelecida lp;
- B_{lp}: variável inteira que determina a banda passante disponível na rota já estabelecida lp;
- Θ: variável inteira que determina o número de comprimentos de onda necessário (R/λ) para atender a chamada requisitada;
- R, H, λ e α são definidos na Seção 4.1.

O Algoritmo de agregação de tráfego com roteamento por rota única é formulado da seguinte forma:

$$Min \sum_{lp} X_{lp} \times C_{lp} + \sum_{w} \left(T_w \times (1-\alpha) + \sum_{i,j} F_{i,j,w} \times H \times \alpha \right) \tag{7}$$

$$\sum_{i} F_{i,j,w} - \sum_{k} F_{j,k,w} = \begin{cases} -T_w & \text{Se } i = s \\ T_w & \text{Se } i = d \quad \forall\ j, w \\ 0 & \text{caso contrário} \end{cases} \tag{8}$$

$$\sum_{lp} X_{lp} \leq 1 \tag{9}$$

$$\sum_{lp} X_{lp} \times B_{lp} \geq \sum_{lp} X_{lp} \times R \tag{10}$$

$$\sum_{w} T_w = \Theta \times (1 - \sum_{lp} X_{lp}) \tag{11}$$

$$\sum_{w} F_{i,j,w} = \left\{0, \sum_{w} T_w\right\} \quad \forall \ i,j \tag{12}$$

p e $lp \in P$ e P é o conjunto das rotas candidatas a atender a conexão solicitada.
$F_{i,j,w} \in \{0,1\}$.
$X_{lp} \in \{0,1\}$.
$T_p \in \{0,1\}$.
$w = [1,2,...W_{i,j}]$, $i,j \in E$.

O objetivo da formulação ILP baseado em uma única rota, Equação (7), é encontrar uma e somente uma rota entre a origem e o destino com custo mínimo. Assim como na formulação por múltiplas rotas, o custo de cada caminho pertencente a uma rota é definido usando a Equação (1). Na Equação (7), o primeiro termo determina o custo das rotas já estabelecidas na topologia virtual e o segundo termo define o custo das potenciais novas rotas.

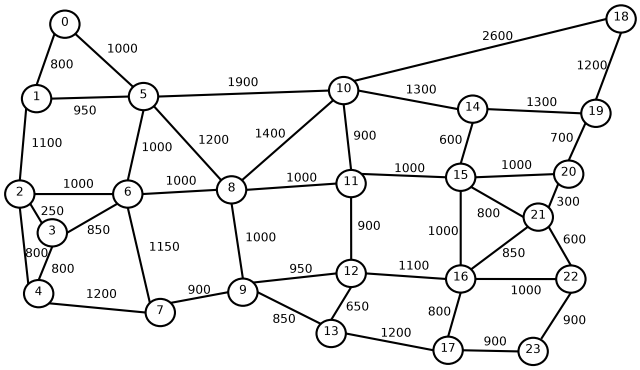
Os índices w em todas as variáveis asseguram a restrição de continuidade de comprimento de onda. A Equação (8) satisfaz a restrição de conservação de fluxo. Uma vez que uma solução de rota única deve ser encontrada, somente uma das rotas já estabelecidas (lp) pode ser escolhida, o que é definido na Restrição (9). Ademais, para fazer parte da solução uma rota já estabelecida precisa-se ter banda passante disponível suficiente para atender a conexão solicitada (Eq. 10). Do mesmo modo, se a solução ILP determina a criação de uma nova rota ($F_{i,j,w}$), a soma da banda passante em todos os caminhos desta rota deve ser maior ou igual a banda requisitada pela demanda a ser atendida (Equação (11)). A restrição (12) garante que todos os caminhos são estabelecidos ao longo da mesma rota.

5. Avaliação de Desempenho

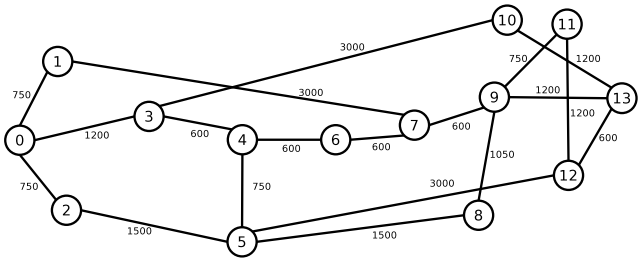
Para avaliar os benefícios do roteamento por múltiplas rotas (*multipath*) com agregação de tráfego orientada ao balanceamento de carga, a solução de roteamento por múltiplas rotas é comparada com a solução de rota única (*singlepath*). Além disso, para avaliar as vantagens da utilização de agregação, a abordagem de roteamento por múltiplas rotas é comparada com uma abordagem de roteamento por múltiplas rotas sem agregação de tráfego [Chen et al. 2009b]. Para estas avaliações, foram realizadas simulações com a ferramenta WDSim [Drummond 2011].

O método de replicação independente foi utilizado para gerar intervalos de confiança com nível de confiança de 95%. Em cada execução, foram consideradas 10.000 requisições. Para cada conexão, a ILP é alimentado com informações sobre a chamada (s,d,B,h) e informações sobre o estado atual da rede. A ILP foi resolvida utilizando a ferramenta Xpress-MP [FICO-Xpress 2011]. Se o resolvedor puder encontrar uma solução ótima, a chamada é aceita, caso contrário ela é bloqueada. No caso de aceitação, os caminhos a serem estabelecidos e os caminhos utilizados para agregação são determinados pela solução ILP, a qual é utilizada para atualizar o estado da rede para a próxima execução da ILP.

As topologias utilizadas na simulação foram a topologia USA, com 24 nós e 43 enlaces bidirecionais (Figura 2(a)) e a topologia NSF, com 16 nós e 25 enlaces bidirecionais (Figura 2(b)). Nestas topologias, cada fibra óptica comporta 16 comprimentos de



(a) USA



(b) NSF

Figura 2. Topologias

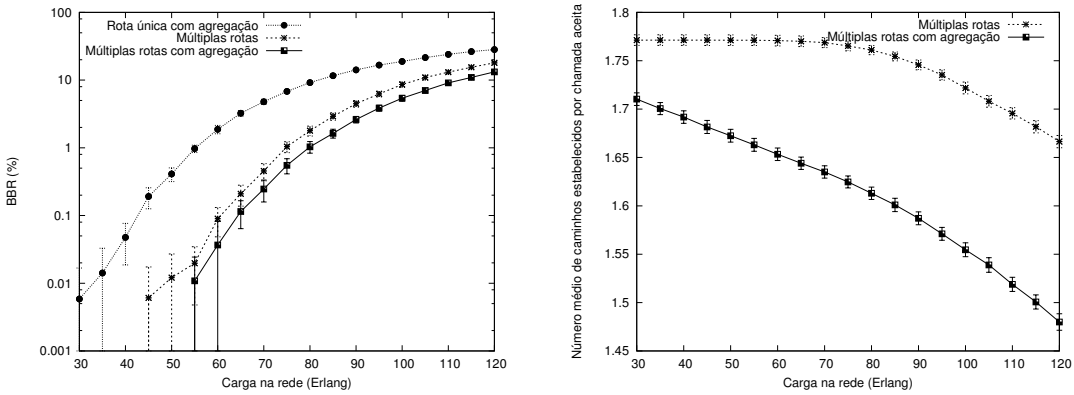
onda, com capacidade de 10 Gbps (OC-192). Os nós utilizados são OXCs sem capacidade de conversão de comprimento de onda e possuem 80 pares (*input*, *output*) de portas de agregação.

A taxa de chegada de requisições de conexão segue uma distribuição de Poisson, i.e., o tempo entre chegadas é exponencialmente distribuído, e o tempo médio de duração de cada chamada também é obtido a partir de uma distribuição exponencial com média de uma unidade de tempo. As conexões de baixa velocidade são distribuídas, de acordo com a seguinte probabilidade de distribuição: $9/45$ para OC-1; $8/45$ para OC-3; $7/45$ para OC-12; $6/45$ para OC-48. A demandas de alta capacidade são distribuídas, de acordo com a seguinte probabilidade de distribuição: $5/45$ para OC-211; $4/45$ para OC-403; $3/45$ para OC-595; $2/45$ para OC-787; $1/45$ para OC-979. As conexões são uniformemente distribuídas através de todos os pares de nós. Variando-se a carga da rede, obtêm-se diferentes taxas de chegada para cada tipo de conexão, o cálculo dessas taxas pode ser realizado utilizando-se a equação de Erlang:

$$A = R \times h \times (B/\lambda) \tag{13}$$

onde R é a taxa média de chegada; h é o tempo de duração da chamada; B é a banda passante requisitada normalizada pela taxa λ (OC-192).

A partir de um valor de carga E dado, do tempo médio de duração da chamada h e da banda passante requisitada B para um determinado tipo de conexão, pode-se obter a taxa média de chegada R e o tempo médio entre chegadas $1/R$. O tempo médio entre chegadas calculado é então utilizado como valor médio de uma distribuição exponencial que define os intervalos de tempo entre chegadas das requisições de conexão.



(a) BBR vs carga na rede

(b) Número médio de caminhos estabelecidos por chamada aceita vs carga na rede

Figura 3. Topologia USA

A métrica empregada para a avaliação do trabalho proposto é a taxa de bloqueio de banda (*bandwidth blocking ratio* - BBR), ou seja, a porcentagem da quantidade de banda bloqueada sobre o total de banda requerida por todas as conexões, durante todo o período de simulação. Para avaliar o custo em termos de componentes eletrônicos e custo de sinalização, foi averiguado o número médio de caminhos ópticos estabelecidos pelo número de chamadas aceitas. Além disso, o atraso diferencial entre os caminhos definidos pela solução ILP foi avaliado conforme segue [Ahuja et al. 2004]:

$$DD = \sum_{p \in P} (d_{\tilde{p}} - d_p) \tag{14}$$

onde: DD é o atraso diferencial para a solução encontrada pela ILP; P é o conjuntos dos caminhos escolhidos pela solução ILP para atender a demanda solicitada; $d_{\tilde{p}}$ é o caminho com o maior atraso no conjunto P.

A Figura 3(a) apresenta os valores de BBR em função da carga para a topologia USA. Para valores de carga baixos, a solução dada pela utilização do roteamento por rota única gera bloqueios, enquanto que as soluções que empregam roteamento por múltiplas rotas bloqueiam chamadas somente sob carga de 45 *Erlang*, quando a estratégia de agregação não está presente, e sob carga de 55 *Erlang*, quando o tráfego pode ser agregado nos caminhos já estabelecidos. Sob cargas de 55 *Erlang*, a abordagem de roteamento de rota única produz valores de BBR 99% maiores do que aqueles dados pela solução de roteamento por múltiplas rotas com agregação. Esta diferença é reduzida para 53% sob cargas de 120 *Erlang*.

Comparando a solução de roteamento por múltiplas rotas com agregação de tráfego com a solução de roteamento por múltiplas rotas sem agregação de tráfego (Figura 3(a)), verifica-se que o uso da banda residual dos caminhos ao empregar agregação de tráfego melhora a utilização da banda passante disponível e, consequentemente, aumenta o número de demandas aceitas. Sob cargas de 60 *Erlang*, a solução com agregação de tráfego gera valores de BBR 59% menores do que aqueles dados pela solução sem agregação de tráfego. A diferença é de 44% sob cargas de 85 *Erlang*. Sob cargas de 120

Erlang, a solução com agregação de tráfego produz valores de BBR 27% menores do que os valores gerados pela abordagem sem agregação de tráfego.

O estabelecimento e a liberação de caminhos ópticos implicam em custo de sinalização. Ademais, quanto maior o número de caminhos, maior o custo em dispositivos eletrônicos (por exemplo: portas de agregação, transceptores para conversão eletro-óptica, terminadores de sinais) para viabilizar a utilização destes caminhos. Na Figura 3(b), compara-se o número médio de caminhos ópticos estabelecidos pelo número de chamadas para a topologia USA. Constata-se que a utilização de agregação de tráfego reduz, de forma considerável, a necessidade de estabelecimento de caminhos ópticos. Sob cargas de 30 *Erlang*, o algoritmo de roteamento por múltiplas rotas com capacidade para agregar fluxos utiliza 3.5% caminhos por chamada aceita a menos do que a abordagem sem a habilidade para agregar tráfego. A medida que a carga na rede aumenta, o número de rotas disponíveis para roteamento é reduzida, ficando, assim, evidente os benefícios da agregação das conexões através dos caminhos já existentes. Sob cargas de 120 *Erlang*, o mecanismo de roteamento por múltiplas rotas demanda uma quantidade de caminhos por chamada 12% maior do que o mecanismo de roteamento por múltiplas rotas com agregação de tráfego.

A Figura 4 apresenta o atraso diferencial médio entre os caminhos escolhidos na solução de roteamento por múltiplas rotas com agregação de tráfego. O atraso diferencial é dirigido pelo algoritmo proposto por meio da distribuição da carga através dos enlaces. Ao evitar enlaces congestionados, o algoritmo da Seção 3 previne o atraso resultante da sobrecarga de enlaces. É possível observar que os valores de atraso diferencial médio dos caminhos para todas as cargas utilizadas nos experimentos estão abaixo de $5ms$, conforme sugerido em [Huang et al. 2008].

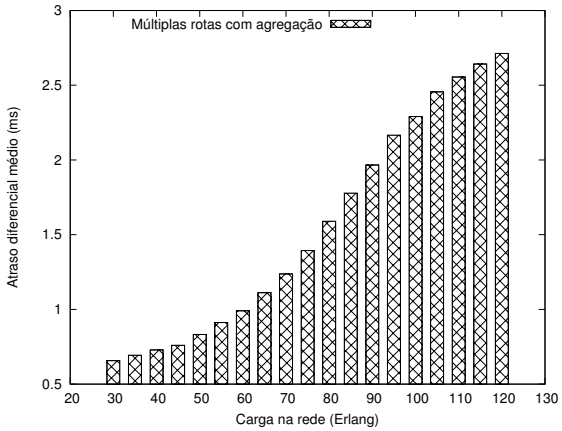
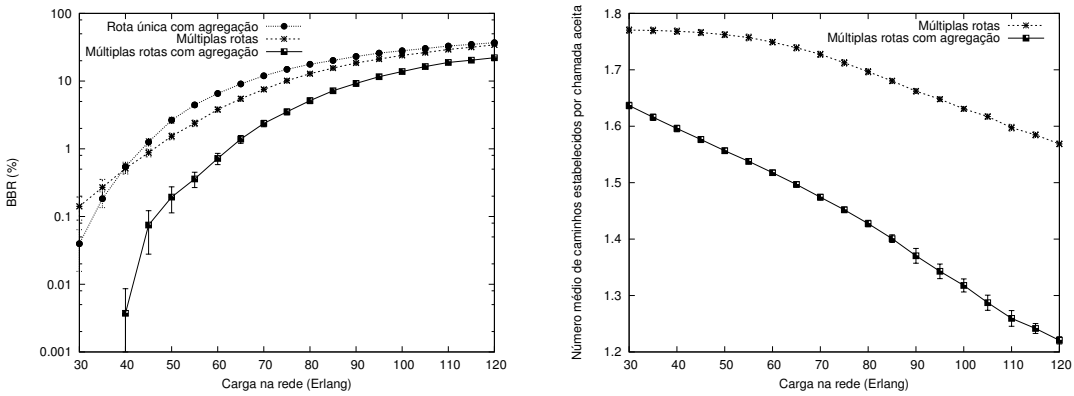


Figura 4. Atraso diferencial médio entre os caminhos escolhidos pela solução ILP vs carga na rede (topologia USA)

A Figura 5(a) apresenta os valores de BBR em função da carga na rede NSF. Diferentemente da solução de roteamento por rota única, o algoritmo que agrega tráfego através de múltiplas rotas começa a bloquear chamadas sob cargas de 40 *Erlang*, e além disso, sua taxa de bloqueio de banda é 99% menor do que os valores gerados pela abordagem de rota única. Sob cargas de 120 *Erlang*, o algoritmo de roteamento por múltiplas



(a) BBR vs carga na rede

(b) Número médio de caminhos estabelecidos por chamada aceita vs carga na rede

Figura 5. Topologia NSF

rotas com agregação de fluxos gera valores de BBR 40% menores do aqueles produzidos no algoritmo de roteamento por rota única com agregação de tráfego. Estes valores evidenciam que a utilização de múltiplas rotas para atender demandas, principalmente aquelas com requisitos de alta capacidade de banda (conforme ilustrada na Figura 1), aumentam a probabilidade de aceitação de conexões.

Comparando os resultados dados pelo algoritmo de roteamento por múltiplas rotas sem agregação fluxos, o algoritmo de roteamento por múltiplas rotas com agregação de tráfego gera valores de BBR 99% menores sob cargas de 40 *Erlang* (Figura 5(a)). Sob cargas de 120 *Erlang*, o algoritmo com habilidade para agregar tráfego através de múltiplas rotas produz valores de BBR 35% menores do que os valores gerados pelo algoritmo de roteamento por múltiplas rotas sem agregação. Para a topologia NSF, a diferença nos valores de BBR produzidos pelos algoritmos de roteamento por múltiplas rotas com e sem agregação de tráfego é ainda maior do que o obtido quando se utiliza a topologia USA. Isto ocorre devido as características da topologia NSF, a qual apresenta uma quantidade menor de caminhos alternativos para os pares origem-destino, o que evidencia a importância da utilização da banda residual para maximizar as chances de aceitação de conexões.

A Figura 5(b) compara o número médio de caminhos ópticos estabelecidos em relação ao número de chamadas aceitas para a topologia NSF. Sob cargas de 30 *Erlang*, o algoritmo com habilidade de agregação de tráfego através das múltiplas rotas utiliza 7.5% caminhos por chamada aceita a menos do que o algoritmo de roteamento por múltiplas rotas sem agregação. Com o aumento da carga e a consequente redução de novas rotas disponíveis, a importância do algoritmo capaz de agregar tráfego em rotas já existentes é ainda maior. Isto é evidenciado nos valores apresentados. Sob cargas de 75 *Erlang* e 120 *Erlang* o algoritmo de roteamento por múltiplas rotas com agregação de tráfego utiliza, respectivamente, 15% e 22% caminhos por chamada aceita a menos do que o algoritmo sem agregação.

Na Figura 6, apresenta-se o atraso diferencial médio para a topologia NSF. Assim como na topologia USA, os valores de atraso diferencial para a topologia NSF são

menores do que $5ms$.

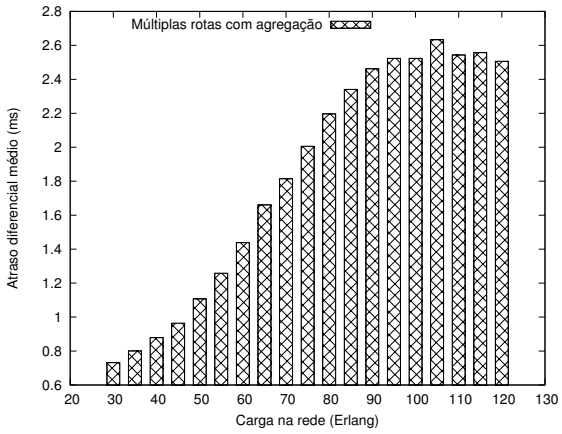


Figura 6. Atraso diferencial médio entre os caminhos escolhidos pela solução ILP vs carga na rede (topologia NSF)

6. Conclusão

A transferência da grande quantidade de dados gerados pelas aplicações de alto desempenho pode ser facilitada pela utilização de múltiplas rotas, cuja capacidade total satisfaz a demanda de banda. Entretanto, a capacidade de um comprimento de onda não necessariamente é completamente utilizada durante as transmissões de alta capacidade. A combinação de uso de múltiplas rotas com agregação de tráfego é, portanto, uma solução atraente em redes operacionais. Este trabalho discorreu sobre a integração de agregação de tráfego com roteamento por múltiplas rotas para atender conexões com duração conhecida. Os resultados indicam que o algoritmo proposto é capaz de reduzir a taxa de bloqueio de banda das demandas de alta capacidade. Além disso, o algoritmo reduz o custo da rede e atende a restrição de atraso diferencial, o que é crítico no roteamento por múltiplas rotas.

Agradecimentos

Este trabalho foi financiado pelo CNPq sob o processo número 141246/2009-0 e pela FAPESP.

Referências

Ahuja, S., Korkmaz, T., and Krunz, M. (2004). Minimizing the differential delay for virtually concatenated ethernet over sonet systems. In *Computer Communications and Networks, 2004. ICCCN 2004. in Proc.*, pages 205 –210.

Cavdar, C., Buzluca, F., Tornatore, M., and Mukherjee, B. (2009). Dynamic scheduling of survivable connections with delay tolerance in wdm networks. In *INFOCOM Workshops 2009, IEEE*, pages 1–6.

Chen, X., Chamania, M., Jukan, A., Drummond, A., and Fonseca, N. L. S. (2009a). On the benefits of multipath routing for distributed data-intensive applications with high bandwidth requirements and multidomain reach. In *IEEE CNSR*, pages 1 –6.

Chen, X., Jukan, A., Drummond, A., and da Fonseca, N. (2009b). A multipath routing mechanism in optical networks with extremely high bandwidth requests. In *Global Telecommunications Conference, 2009. IEEE GLOBECOM 2009. IEEE*, pages 1 –6.

Das, A., Martel, C., Mukherjee, B., and Rai, S. (2011). New approach to reliable multipath provisioning. *J. Opt. Commun. Netw.*, 3(1):95–103.

Drummond, A. (2011). Wdmsim - optical wdm networks simulator.

Dutta, R. and Rouskas, G. (2002). Traffic grooming in wdm networks: past and future. *IEEE Network*, 16(6):46–56.

Fawaz, W., Daheb, B., Audouin, O., Du-Pond, M., and Pujolle, G. (2004). Service level agreement and provisioning in optical networks. *IEEE Communications Magazine*, 42(1):36–43.

FICO-Xpress (2011). Xpress optimization suite.

Huang, S., Mukherjee, B., and Martel, C. (2008). Survivable multipath provisioning with differential delay constraint in telecom mesh networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 191 –195.

Huang, S., Xia, M., Martel, C., and Mukherjee, B. (2010). Survivable multipath traffic grooming in telecom mesh networks with inverse multiplexing. *J. Opt. Commun. Netw.*, 2(8):545–557.

Lee, H.-W. and Modiano, E. (2009). Diverse routing in networks with probabilistic failures. In *INFOCOM 2009, IEEE*, pages 1035 –1043.

Munir, A., Tanwir, S., and Zaidi, S. (2009). Holding time aware dynamic bandwidth allocation algorithm for emerging bandwidth on demand multicast applications. In *HONET, 2009 6th Int. Symposium on*, pages 16 –21.

Rouskas, G. N. and Perros, H. G. (2002). A tutorial on optical networks. pages 155–193.

Santi, J., Drummond, A., da Fonseca, N., and Jukan, A. (2010a). Load balancing for holding-time-aware dynamic traffic grooming. In *IEEE GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference*, pages 1 –5.

Santi, J., Drummond, A. C., and Fonseca, N. L. S. (2010b). Agregação dinâmica de fluxos com duração conhecida em redes wdm através de balanceamento de tráfego. In *XXX Congresso da SBC - Wperformance 2010*, pages 1873–1886.

Tornatore, M., Baruffaldi, A., Zhu, H., Mukherjee, B., and Pattavina, A. (2008). Holding-time-aware dynamic traffic grooming. *IEEE Journal on Selected Areas in Communications*, 26(3):28–35.

Zhang, S., Shen, D., and Chan, C.-K. (2010). Energy efficient time-aware traffic grooming in wavelength routing networks. In *IEEE GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference*, pages 1 –5.

Zhang, X. and Perrig, A. (2010). Correlation-resilient path selection in multi-path routing. In *IEEE GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference*, pages 1 –6.

Análise de desempenho de redes óticas dinâmicas equipadas com EDFA's com saturação de ganho

Victor A. P. Oliveira[1], Iguatemi E. Fonseca[1]

[1]Universidade Federal do Semi-Árido (UFERSA)
Av. Francisco Mota, 572 – 59.625-900 – Mossoró – RN – Brasil

victorcactusti@gmail.com, iguatemi@ufersa.edu.br

Abstract. *This paper presents a study about benefits that the elimination of EDFA gain saturation may have on the performance of a Transparent Optical Network. It is developed an analytical model to estimate the behavior of the EDFA gain variation considering amplifier input power and global network traffic. Impairment Aware RWA algorithm (IA-RWA) is also proposed as a form to evaluate the impact of the amplifiers gain saturation on the performance of the dynamic optical network.*

Resumo. *Este artigo apresenta um estudo sobre benefícios que o tratamento do efeito de saturação dos amplificadores óticos pode trazer para o desempenho de uma rede ótica transparente dinâmica. É apresentado um modelo analítico que pode estimar o comportamento das variações no ganho dos amplificadores com relação a sua potência de entrada e, indiretamente, com a dinâmica do tráfego na rede. Um algoritmo de atribuição de rota e comprimento de onda que leva em conta efeitos da camada física também foi implementado para avaliar o impacto das variações de ganho dos amplificadores sobre o desempenho da rede.*

1. Introdução

Caso o tráfego continue crescendo no ritmo atual, com o surgimento e a consolidação de aplicações como vídeo sob demanda, o esquema de eletrônica tradicional usada nos roteadores no núcleo da rede pode se tornar um limitante para o crescimento da internet (Harai 2008)-(Abbade et al. 2009). Isto acontece não somente por causa da necessidade de altas taxas de transmissão, as quais podem ser inviáveis de serem realizadas pela eletrônica; mas também por causa do significativo consumo de energia que estes roteadores podem empregar (Harai 2008). Redes Óticas Transparentes (TON – Transparent Optical Networks) usando Multiplexação por Divisão de Comprimento de Onda (WDM – Wavelength Division Multiplexing) tem sido encarado como uma estratégia interessante para resolver ambos os problemas (Tonkos et al. 2008)-(Yuang et al. 2008).

Todavia, apesar de todo o avanço alcançado nestes últimos anos, ainda existem desafios a serem vencidos para a implementação em massa de TON`s e, consequentemente, o uso eficiente dos recursos destas redes (Ramaswami 2006). Por exemplo, em uma TON com amplificadores óticos a fibra dopada com Érbium (EDFA – *Erbium-Doped Fiber Amplifier*), a admissão de uma nova conexão na rede pode causar flutuações na Taxa de Erro de Bits (BER – *Bit Erro Rate*) dos caminhos óticos já presentes na rede. Essas flutuações ocorrem devido ao efeito de saturação dos EDFA's, o qual causa variações

nos ganhos dos amplificadores, afetando a potência das conexões e, consequentemente, a relação sinal ruído ótica (OSNR - *Optical Signal Noise Rate*) e, portanto, a BER destas conexões. Perceba que o efeito de saturação dos EDFA's funciona da mesma forma que um efeito não-linear em uma rede ótica. Ou seja, a admissão de uma conexão causa interferência em outras conexões já presentes na rede, portanto, o seu tratamento no processo de admissão de conexões na rede possui considerável complexidade computacional (Fonseca 2005).

Recentemente, algoritmos de atribuição de rota e comprimento de onda (RWA – Routing and Wavelength Assignment) mais sofisticados denominados RWA conscientes de limitações da camada física (IA-RWA – *Impairment Aware RWA*), que levam em conta os efeitos da camada física, têm sido estudados (Ramamurthy 1999)-(Martins-Filho et al. 2003)-(Deng and Subraminiam 2004)-(Lima et al. 2004)-(Fonseca et al. 2004-(Fonseca et al. B 2004)-(Raul et al. 2008)-(Filho et al. 2007)-(Gomes et al. 2009) e (Ali and Tancevski 2002). Em (Ramamurthy 1999)-(Martins-Filho et al. 2003)-(Deng and Subraminiam 2004)-(Lima et al. 2004) a influência do Ruído de Emissão Espontânea (ASE – *Amplified-Spontaneous Emission*) de amplificadores ópticos sobre a BER em uma TON foi investigada. Entretanto, o efeito da saturação de amplificadores tem sido pouco estudado (Deng and Subraminiam 2004). Estratégias para minimizar o impacto de efeitos não-lineares, como Mistura de Quatro Ondas e Modulação (FWM – *Four-wave Mixing*) de Fase Cruzada (XPM – *Cross Phase Modulation*) sobre a qualidade de transmissão (QoT – *Quality of Transmission*) em uma TON, foram apresentadas em (Fonseca et al. 2004), (Fonseca et al. B 2004), (Raul et al. 2008) e (Filho et al. 2007). Em (Gomes et al. 2009) e (Ali and Tancevski 2002) foi averiguado o impacto do efeito de Dispersão de Modo de Polarização (PMD – *Polarization Mode Dispersion*).

O objetivo deste trabalho é estudar o impacto que o efeito de saturação dos EDFA's, o qual também pode ser entendido como um efeito de variação de ganho dos amplificadores, tem sobre uma TON. Para isso é proposto um modelo analítico que pode estimar o comportamento da variação do ganho dos EDFA's com relação a sua potência de entrada e, de maneira indireta, com a dinâmica do tráfego na rede. Além disso, foi implementado um algoritmo IA-RWA que leva em conta, no processo de admissão de uma conexão na rede, o efeito da saturação dos EDFA's. Os resultados das simulações numéricas apontam que o desempenho da rede pode ser melhorado se o controle das variações de ganho na rede é empregado. Foi também verificado que o modelo analítico apresentado captura de maneira satisfatória o comportamento das variações de ganho dos EDFA's.

O restante deste artigo está organizado da seguinte forma. Na Seção II o efeito da saturação dos amplificadores é discutido. A Seção III mostra a proposta de um algoritmo IA-RWA com uma estratégia para o cômputo das variações de ganho dos EDFA's. Na seção IV são apresentados o ambiente de simulação, métricas de desempenho e discutidos os resultados das simulações numéricas. A Seção V traz as conclusões e proposta de trabalhos futuros.

2. Saturação de Amplificadores óticos EDFA's

2.1. Comportamento do ganho

A Figura 1 mostra o ganho do amplificador em função da potência de entrada em um EDFA típico. Observa-se que, para baixas potências de entrada, o ganho do amplificador é seu ganho insaturado. Para altas potências de entrada o ganho tenderá a um ($G \rightarrow 1$), de forma que a potência de saída do amplificador será igual à potência de entrada ($P_{out} = P_{in}$). A Figura 1 possui algumas informações necessárias à compreensão do leitor. Observe que a marca do limiar na Figura 1 delimita a região de variação brusca do ganho. Na Subseção 2.3 este comportamento será analisado analiticamente, através do cálculo da derivada do ganho em função da potência de entrada (detalhe apresentado na Figura 1). O ganho de um EDFA pode ser expresso pela seguinte equação (Ramaswami and Sivarajan 2010)

$$G = 1 + \frac{P_{sat}}{P_{in}} \ln\left(\frac{G_{\max}}{G}\right), \tag{1}$$

em que, P_{sat} é a potência de saturação do amplificador, G_{max} é o ganho insaturado e P_{in} é a potência de entrada do amplificador, ou seja, a soma da potência proveniente de todas as conexões.

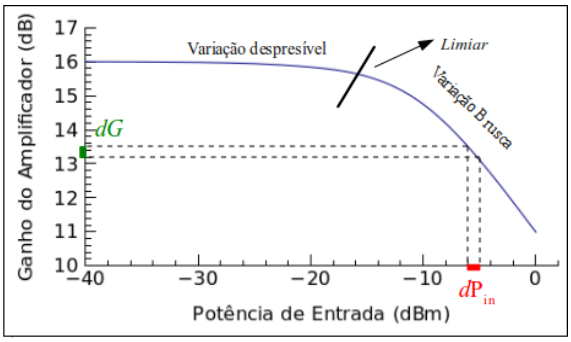


Figura 1: Ganho de um EDFA típico.

2.2. O problema da variação de ganho

O problema da variação do ganho de EDFA's em uma rede ótica pode ser explicado como segue. Considere uma rede, cujo estado inicial é composto somente pela conexão que ocupa o comprimento de onda λ_1, de acordo com a Figura 2 (a). Agora, considere que uma nova conexão é admitida no comprimento de onda λ_2, e que essa conexão compartilha um enlace (L_{25}) com a primeira. Como as duas conexões compartilham o mesmo amplificador (A_{25}^1), o seu ganho tende a diminuir devido ao aumento da potência de entrada no amplificador. Isso significa que a potência de saída nesse mesmo amplificador para a conexão em λ_1 não é mais a mesma. Dessa maneira, todos os amplificadores que sucederem o amplificador A_{25}^1 no caminho da conexão em λ_1 terão seus ganhos alterados, e. g., o amplificador A_{58}^1, que mesmo não estando na rota na qual a nova conexão foi admitida, terá seu ganho alterado. O problema da variação de ganho também pode ocorrer quando uma conexão é retirada da rede, ou seja, se fosse considerado que o caminho ótico no comprimento de onda λ_2 fosse retirado.

Outro problema com relação à variação do ganho é ilustrado através da Figura 2(b). Suponha que exista uma conexão admitida em λ_1 e que seu caminho ótico passe através dos enlaces $L_{87} - L_{74} - L_{41} - L_{12}$. A *condição de looping* ocorre quando uma nova conexão admitida, em λ_2 por exemplo, compartilhar quaisquer dois *enlaces* não consecutivos. Como exemplo, admita que o sentido da conexão em λ_2 é $L_{12} - L_{25} - L_{58} - L_{87}$. Dessa maneira, os enlaces L_{12} e L_{87} serão compartilhados; L_{12} e L_{87} como primeiro e último enlace, respectivamente, da conexão admitida em λ_2 e, L_{87} e L_{12} como primeiro e último enlace, respectivamente, da conexão admitida em λ_1. O grande problema nessa situação é que ao admitir a conexão em λ_2, o amplificador A_{12}^{1} sofrerá diminuição do ganho afetando, portanto, a potência que entra no amplificador A_{12}^{2} do mesmo enlace (análogo ao esquema apresentado na Figura 2(a)). O problema maior nessa situação é a partir do outro enlace (L_{87}), cujo amplificador A_{87}^{1} também sofrerá alteração em seu ganho, mas dessa vez devido à conexão em λ_2. Com o ganho do amplificador A_{87}^{1} sendo alterado, todos os demais amplificadores que seguem a conexão em λ_1 também serão alterados. Como o fim da conexão em λ_1 coincide com o inicio da conexão em λ_2, todo o ciclo se inicia novamente.

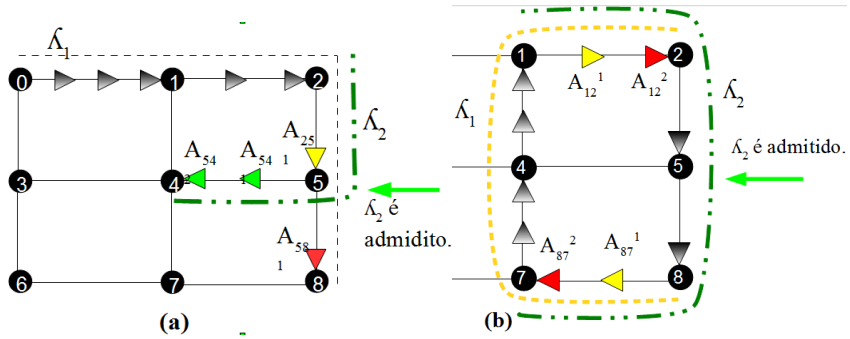


Figura 2: Alterações nos ganhos dos EDFA's ao admitir conexões.

Atente para o fato de que os problemas discutidos anteriormente não possuem o foco somente no nível da simulação computacional. Tais problemas também ocorrem na rede real, mas os próprios amplificadores se estabilizam e a rede entra em estado de equilíbrio depois de um dado tempo, quando uma conexão entra ou sai da rede. Todavia, essa instabilidade momentânea pode levar a perda da informação que está trafegando pela rede no momento da instabilidade. Em simuladores computacionais, esse tipo de problema precisa ser levado em consideração para que os resultados obtidos na simulação sejam o mais próximo possível de uma rede ótica real. Além disso, analisando-se esse problema em uma simulação numérica, pode-se apontar uma solução que minimize o impacto das variações de ganho dos amplificadores.

2.3. Modelagem matemática

À primeira vista, somos tentados a crer que quando o tráfego na rede estiver alto, a variação do ganho nos amplificadores será grande, já que quanto maior o tráfego, maior o número de conexões passando pelos amplificadores e, portanto, maior a potência de entrada. Calculando-se a derivada do ganho em função da potência de entrada, pode-se obter informações sobre o comportamento da variação do ganho do EDFA com relação à potência de entrada.

Note que a variável G está presente em ambos os lados da igualdade (1), significando, portanto, que esta é uma equação transcendental; ou seja, G está em

função de P_{in} e do próprio G. Assim, para o cálculo da derivada, foi utilizado o método de derivação de funções implícitas (Guidorizzi 2008), o qual pode ser expresso por

$$\frac{\partial G}{\partial P_{in}} = -\frac{\frac{\partial f\left(G, P_{in}\right)}{\partial P_{in}}}{\frac{\partial f\left(G, P_{in}\right)}{\partial G}}. \quad (2)$$

A resolução da derivada foge do escopo deste trabalho. Portanto, a equação (3) mostra apenas o resultado final do cálculo da derivada.

$$\frac{\partial G}{\partial P_{in}} = \frac{-GP_{sat}\ln\left(\frac{G_{\max}}{G}\right)}{GP_{in}^2 + P_{sat}P_{in}}. \quad (3)$$

Perceba que a variação do ganho diminui com o quadrado da potência de entrada do EDFA. Dessa forma, pode-se esperar que quando a rede opera com alto tráfego, ou seja, quando potência de entrada dos EDFA's é maior, a instabilidade causada pela variação do ganho do EDFA diminuirá. Isso é confirmado nas simulações apresentadas na Seção 4.

3. IA-RWA's

Com o objetivo de avaliar o impacto da variação de ganho dos EDFA's sobre uma rede ótica dinâmica, dois algoritmos RWA foram implementados.

3.1 IA-RWA em uma rede com saturação de EDFA's

O algoritmo IA-RWA que está sendo proposto nesse trabalho avalia a probabilidade de bloqueio de conexões não somente em função da continuidade, mas também em função de uma métrica de QoT pré estabelecida. Dessa maneira, se uma requisição necessitar de um dado nível de QoT, ela será admitida se e somente se: (a) ela não for bloqueada pela restrição da continuidade; (b) se ela possuir um nível de QoT igual ou superior ao nível de QoT solicitado na requisição; e (c) se a nova conexão não violar a qualidade das conexões já presentes na rede. Na realidade, os quesitos (b) e (c) poderiam ser integrados em um só, uma vez que eles não podem ser medidos separadamente devido à saturação de EDFA's ser um efeito não-linear. Destarte, a nova conexão precisa ser estabelecida temporariamente para poder avaliar o comportamento dela e das outras conexões previamente estabelecidas.

A Figura 3 apresenta o fluxograma do algoritmo proposto. Como pode ser observado, primeiramente é gerada uma requisição de conexão, depois é encontrada a rota considerando o menor caminho, em que o custo do enlace é a distância em quilômetros. Em seguida, o algoritmo realiza o teste da continuidade utilizando a heurística First Fit (Zang et al. 2000). Nesse ponto, se não existir comprimento de onda livre, a conexão é imediatamente rejeitada. Em caso contrário, a conexão é pré admitida apenas para ser computado o QoT e em seguida é averiguado se tal conexão não interferiu de forma a degradar as conexões já estabelecidas na rede. Em caso positivo, a

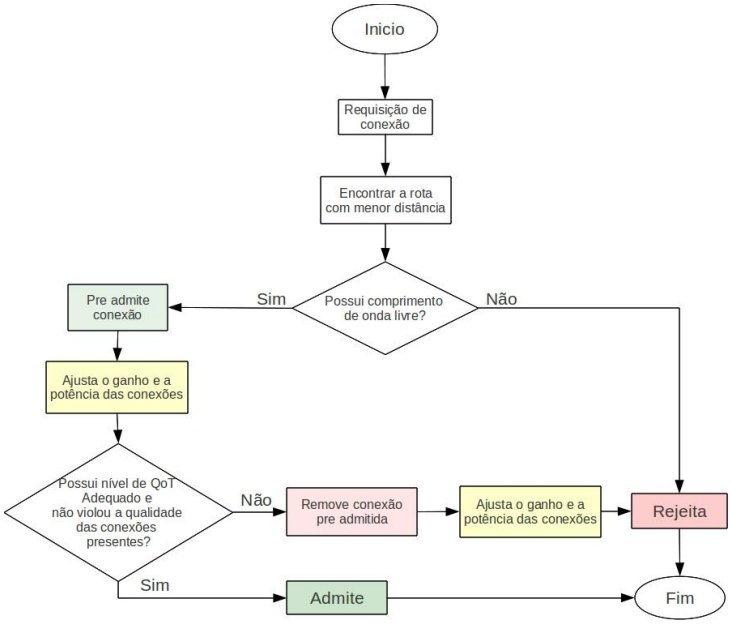


Figura 3: IA-RWA que leva em conta a saturação de EDFA's.

conexão deve ser retirada e rejeitada. Senão, a conexão é finalmente admitida. Observe também que, como mostrado no fluxograma, são ajustados o ganho e a potência das conexões logo que a conexão é pré admitida (e caso ela seja rejeitada). Pois, como foi visto na Seção 2, o ganho do EDFA depende da potência de entrada total e essa atualização se faz necessária para se obter resultados coerentes com simulações na rede ótica real.

3.2 RWA Tradicional

Foi implementado também um algoritmo RWA Tradicional, também chamado de RWA Cego (Fonseca 2005). Note que ele é bem mais simples em relação ao IA-RWA e não verifica o QoT da conexões; simplesmente as admite caso encontre um caminho ótico disponível. RWA Cego usa roteamento baseado na menor distância em quilômetros e atribuição de comprimentos de onda usando o algoritmo *First-Fit*.

4. Simulações Numéricas

4.1. Ambiente de Simulação

Através de um ambiente de simulação implementado em C/C++ pelos próprios autores, foi simulado um cenário dinâmico, no qual foram geradas 100000 requisições de conexões. Tais conexões possuem um padrão de tráfego uniforme entre os nós da rede e seguem uma distribuição poissoniana tendo duração com distribuição exponencial (média = 1s). A Rede Ótica utilizada é transparente, ou seja, não possui conversão ótica-elétrica-ótica, e possui 19 nós. Todos os enlaces são bidirecionais e têm comprimento variando entre 240 e 480 km. O comprimento de um *span*, ou seja, a distância entre dois amplificadores óticos é de 80 km. É utilizado um conjunto de W=16 comprimentos de onda em uma rede ótica sem conversão de comprimento de onda. A topologia da rede simulada (NFSNET-2 – modelo real da rede ótica americana, bastante usada na literatura) e detalhe de posicionamento dos amplificadores são ilustrados na Figura 4.

Foram feitas simulações com taxas de transmissão de 10 e 40 Gbps. O algoritmo IA-RWA utiliza $BER_{TH}=10^{-12}$ como critério de QoT para admitir ou rejeitar requisições de conexões na rede. A partir dos parâmetros apresentados na Tabela 1, a quantidade de conexões rejeitadas dentre o número total de pedidos de conexões que chegam na rede ótica é a Probabilidade de Bloqueio da rede.

Tabela 1: Parâmetros da simulação.

Parâmetro	Valor
Forma do bloqueio	Continuidade + QoT
Tipo de roteamento	Fixo
BER_{TH}	10^{-12}
Atenuação da fibra	0,2 dB/km
Potência (P)	0 dBm
Ganho máximo do amplificador (G_{max})	16 dB
Potência de saturação (P_{sat})	10 dBm
Fator de emissão espontânea do amplificador (Nsp)	4
Largura de banda do filtro ótico	50 GHz
Largura de banda do filtro elétrico	Taxa de Bits x 0,8

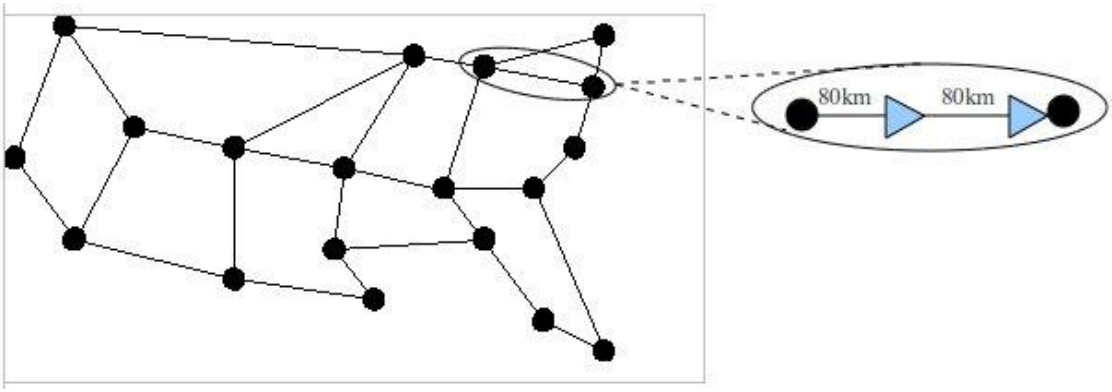


Figura 4: Rede com 19 nós usada nas simulações com detalhe de amplificadores.

4.2. Métricas de desempenho

Neste trabalho forma utilizadas quatro métricas para avaliar o desempenho de uma rede ótica operando com os algoritmos RWA Tradicional e IA-RWA: i) a variação do ganho dos EDFA's; ii) Probabilidade de violação de limiar da QoT (TVP); iii) Probabilidade de violação crítica da QoT (CVP); iv) Probabilidade de bloqueio.

4.2.1 Variação do ganho dos EDFA's

Foi utilizada uma nova métrica que consiste basicamente em computar a variação média do ganho toda vez que uma conexão entra e/ou sai da rede. Pois, como dito anteriormente, a variação do ganho é totalmente dependente da potência de entrada do amplificador ótico e, como as conexões chegam dinamicamente, a potência de entrada

tende a variar intensamente. O ΔG_j é a média da variação do ganho referente à conexão que entra e/ou sai da rede, com ou sem atualizações dos ganhos dos amplificadores, e pode ser obtida por

$$\Delta G_j = \frac{\sum_{i=o}^{A} \Delta G_j^i}{A}, \quad (4)$$

em que, ΔG_j^i é variação do ganho do i-ésimo amplificador, *j* é o número de atualizações dos ganhos dos amplificadores e *A* é o número de amplificadores que sofrem variação de ganho durante a chegada ou saída de uma conexão. Note que *A* pode ser incrementado em mais de um. Para isso, basta que a conexão que entra (ou que sai) cause variação do ganho em mais de um enlace/amplificador. Note ainda que ΔG_0, i.e para j=0, é a variação de ganho sem atualização dos amplificadores; ΔG_1, i.e. para j=1, é a variação de ganho após uma atualização dos ganhos dos amplificadores, e assim por diante.

4.2.2 Probabilidade de violação de limiar

A probabilidade de violação do limiar (TVP – *Threshold Violation Probability*) significa a probabilidade de pelo menos uma conexão ativa em toda a rede ter a sua BER acima do BER_{TH} após a mudança do estado de um caminho ótico na rede, i.e. após a ativação ou encerramento de uma conexão na rede (Fonseca 2005). Esta métrica pode ser expressa como: seja p_X=*Prob*{X=x} a probabilidade de se ter x=0,1,2,… conexões ativas com sua BER acima do máximo permitido depois que uma conexão é estabelecida ou encerrada na rede. O TVP representa a probabilidade de que pelos menos uma conexão ativa ser degradada, i.e. TVP=[1 – p_0], com x=0. O TVP pode ser útil para dar uma representatividade global de QoT na rede, pois provê uma medida de preservação e fornecimento de QoT estatístico. QoT determinístico é atingido à medida que TVP$\rightarrow$0, em outras palavras, o processo de estabelecimento e encerramento de conexões nunca violaria o OSNR de outras conexões ativas na rede inteira (além de satisfazer a sua própria exigência de qualidade).

4.2.3 Probabilidade de violação crítica

Com o objetivo de monitorar degradações excessivas da BER, é utilizada a Probabilidade de Violação Crítica (CVP – *Critical Violation Probability*). CVP é uma métrica similar ao TVP, mas com BER_{TH} igual a 10^{-3} para todos os caminhos óticos e de acordo com métricas já definidas para o SDH por meio do Defeito por Erro Excessivo (dEXC – *Excessive Error Defect*) em ITU-T G.806 (ITU-T 2000). Considera-se que CVP deve ser menor do que 0,01% para que os mecanismos de proteção e restauração da rede não sejam ativados com muita frequência.

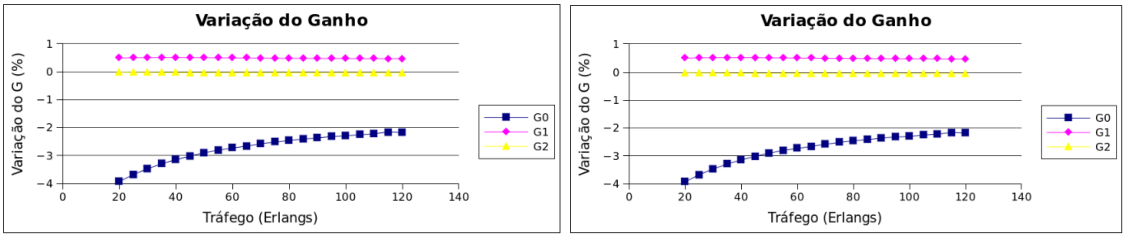
4.2.4 Probabilidade de bloqueio

Uma das métricas que é utilizada para medir o desempenho dos algoritmos propostos é a probabilidade de bloqueio da rede, a qual representa o número de caminhos óticos rejeitados perante o número total de requisições de caminhos óticos que chegam à rede

ótica. A probabilidade de bloqueio é a métrica mais utilizada na literatura para avaliar o desempenho de algoritmos RWA em redes óticas dinâmicas.

4.3. Análise da variação de ganho

As Figuras 5a-5b ilustram o resultado obtido da simulação com relação à variação do ganho inicial (G0) e após duas atualizações dos ganhos dos amplificadores e potência das conexões (G1 e G2). São considerados os algoritmos RWA-Cego e IA-RWA em uma rede operando a uma taxa de transmissão de 40 Gbps. Os resultados para 10 Gbps são semelhantes aos de 40 Gbps, portanto não são apresentados aqui. Como pode ser observado, o G0 se manteve praticamente o mesmo e seu comportamento pode ser capturado através da Equação 3. Quando o tráfego está baixo, a variação é maior e, à medida que o tráfego aumenta, a variação do G0 tende a diminuir de forma inversamente proporcional a potência de entrada no amplificador. Observe que, embora não esteja explícito a potência de entrada nos gráficos, entende-se que, quando o tráfego está intenso, a rede possui muitas conexões que permanecerão mais tempo na rede. Dessa maneira, conclui-se que, se o tráfego está baixo, significa que a potência de entrada nos amplificadores também está baixa; em caso contrário, se o tráfego estiver alto, a potência de entrada dos amplificadores também estará alta.



a) **b)**

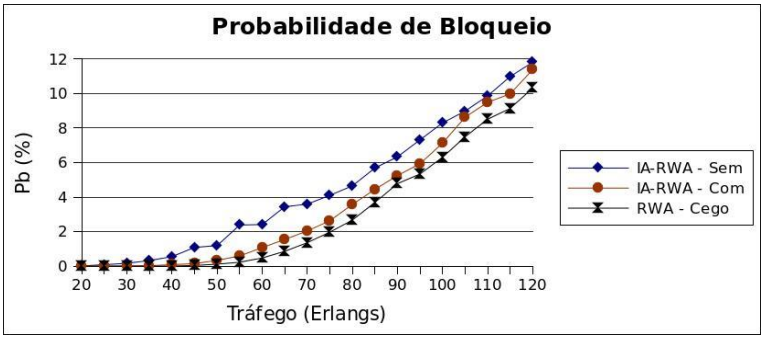
Figura 5: Variação do ganho em simulações sem atualizações (G0) e com uma e duas atualizações de ganho (G1 e G2) para 40 Gbps. a) RWA-Cego; b) IA-RWA.

Mesmo que a variação do ganho diminua quando o tráfego está alto, faz-se necessário atualizar o ganho e a potência de entrada para que todas as conexões presentes na rede entrem em estado de equilíbrio, i.e., passem a "enxergar" o mesmo ganho. Esse comportamento foi capturado pelo G1, primeira atualização, e pelo G2, segunda atualização, ilustrado nas Figuras 5a – 5b. Note que foram necessárias duas atualizações para que a variação do ganho passasse a ser desprezível e garantir que os ganhos dados pelos EDFA's às conexões presentes na rede fossem equilibrados.

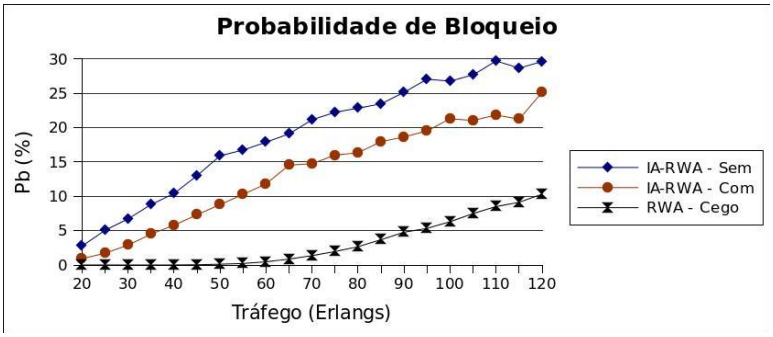
4.4 Benefícios para o desempenho da rede

Para ilustrar os benefícios que o processo de atualização de ganho dos EDFA's pode trazer para o desempenho da rede com relação à probabilidade de bloqueio, foram analisados cenários de rede com e sem atualização de ganho; e operando com os algoritmos IA-RWA e RWA-Cego com taxas de transmissão de 10 e 40 Gbps.

As Figuras 6a – 6b mostram os resultados comparativos do IA-RWA proposto com o RWA-Cego, o qual serve apenas como referencial, já que não bloqueia conexões sem QoT. Ou seja, é possível que quando a rede opera com o algoritmo RWA-Cego,

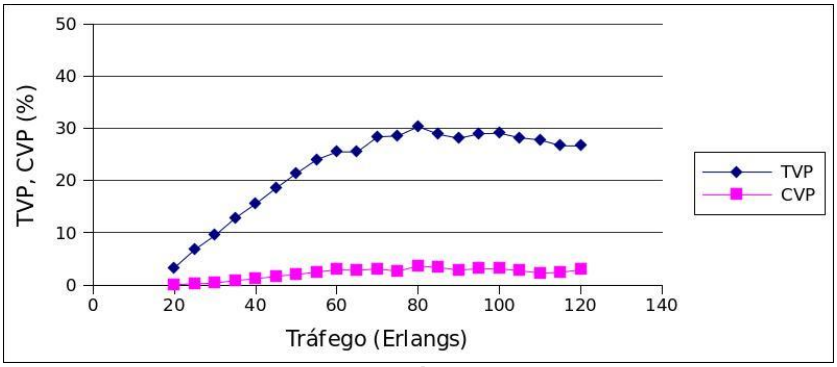


a)

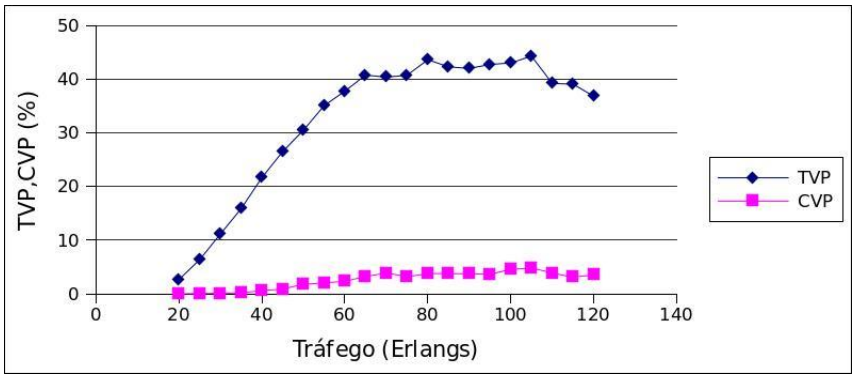


b)

Figura 6: Probabilidade de bloqueio em cenários sem e com atualização de ganho. a) 10 Gbps; b) 40 Gbps.



a)



b)

Figura 7: TVP e CVP para RWA Cego a 40 Gbps. a) sem atualização de ganho; b) com atualização de ganho.

conexões sem QoT, ou seja com BER > 10^{-12}, podem ser admitidas na rede. Perceba que nos cenários nos quais usa-se a atualização dos ganhos, o desempenho da rede melhorou. Por exemplo, comparando as curvas as Figura 6b, verifica-se uma redução de aproximadamente 20% na probabilidade de bloqueio da rede quando o tráfego é alto (e.g. 100 Er), passando de 0,27 para 0,22. Melhora similar no desempenho acontece quando a rede opera a 10 Gbps (Figura 6a). Note que, quanto maior é a taxa de transmissão de bits da rede, maior é a melhoria no seu desempenho.

Outra característica observada nos cenários de rede operando com e sem variação de ganho é que os valores da TVP mudam de um cenário para outro. As Figuras 7a – 7b mostram a TVP e CVP para a rede operando com o algoritmo RWA Tradicional e com uma taxa de transmissão de bits de 40 Gbps. Os resultados para 10 Gbps são similares. Note que no cenário com atualização de ganho, o TVP apresenta resultado bem superior ao cenário sem atualização de ganho. Por exemplo, quando o tráfego na rede é alto (e.g., 100 Er), o TVP para o cenário sem atualização é aproximadamente 29% e 43% para o cenário com atualização de ganho, ou seja, um aumento de 48%. Isto significa que, quando a rede opera com um algoritmo RWA Tradicional, se os ganhos dos EDFA's não são atualizados, a probabilidade de se ter conexões na rede sem QoT pode ser bem maior do que a medida pelo algoritmo RWA Tradicional. Isto pode ter implicações negativas para o serviço oferecido às redes clientes. Um alto valor de TVP significa que existe uma alta probabilidade de que a QoT das conexões presentes na rede estejam violadas, ou seja, suas BER's podem ser altas e comprometer a qualidade da informação transmitida pela rede.

As Figuras 7a – 7b também trazem outra informação interessante. Perceba que a partir de 80 Er, i.e. alto tráfego na rede, os valores de TVP estabilizam em torno de 30% para o caso sem atualização de ganho e 42% para o caso com atualização de ganho. Ou seja, embora o número de conexões esteja aumentando na rede, a probabilidade de se ter conexões sem QoT não é alterada. A explicação para isso está no fato de que quando o tráfego é alto, as variações de ganho dos EDFA's também estabilizam em um certo valor, conforme pode ser visto na Figura 5a e também está previsto pela equação (3). Note na Figura 5a que entre 80 e 120 Er a variação de ganho dos EDFA's praticamente permanece em torno de –2,2%. Esta parece ser uma característica do efeito da saturação de EDFA's, o qual pode ser encarado como um efeito não-linear. A TVP causada na rede devido a outros efeitos não-lineares, como por exemplo, a FWM, aumenta de forma proporcional ao aumento do tráfego na rede, não apresentando pontos de estabilização de valores (Fonseca 2005). Em outras palavras, o crescimento da curva é monotônico.

6. Conclusão

Este artigo apresentou um estudo sobre benefícios que a eliminação das variações do ganho dos EDFA's pode trazer para o desempenho de uma rede ótica dinâmica transparente. Um modelo analítico para estimar as variações de ganho com relação à sua potência de entrada também é apresentado. Para que o ganho dos EDFA's não sofra variações, amplificadores óticos equipados com controle automático de ganho podem ser utilizados na rede (Oliveira 2007). Os resultados das simulações numéricas sugerem que o desempenho da rede pode ser melhorado com a eliminação das variações do ganho dos amplificadores. Isso, porque eles resolveriam a questão da instabilidade

momentânea e, consequentemente, evitariam a perda das informações que estariam trafegando pelo enlace no momento da aquisição/remoção de conexões. Como trabalhos futuros serão investigados outras possibilidades de IA-RWA e a possibilidade de redução no número de amplificadores equipados com controle automático de ganho.

References

Abbade, M. L. F., Marconi, J. D., Cassiolato, R. L., Ishizuca, V., Fonseca, I. E. and Fragnito, H. L. (2009) "Field-Trial Evaluation of Cross-Layer Effect Caused by All-Optical Wavelength Converters on IP Network Applications", In: IEEE/OSA J. Ligthwave Technology.

Ali, M. and Tancevski, L. (2002) "Impact of Polarization-Mode Dispersion on the Design of Wavelength-Routed Networks", In: IEEE Photonics Technology Letters, Vol. 14, p. 720-722.

Almeida Jr., R. C., Campelo, D. R., Cartaxo, A. V. T., Guild, K. M. and Waldman H. (2008) "Efficient Wavelength Assignment Policy for XPM-Impaired WDM Networks", In: IEEE Communication Letters, vol. 12, p. 791 – 793.

Deng, T. and Subramaniam, S. (2004) "Source Power Management in Transparent Wavelength-Routed Mesh Networks", In: Proc. IEEE ICC'04.

Filho, U. S. P., Ribeiro, M. R. N., Maioli, C. P., Freitas, M. and Fonseca, I. E. (2007) "Cost functions for cac/rwa in dynamic optical networks under gvd, spm and xpm", In: Journal of Microwave and Optoelectronics, vol. 6, p. 249–262.

Fonseca, I. E. (2005) "Uma abordagem para Aprovisionamento e Diferenciação de QoS Óptico na Presença de FWM em Redes Ópticas Transparentes", Tese de Doutorado, FEEC/Unicamp.

Fonseca, I. E., Ribeiro, M. R. N., Almeida Jr., R. C. and Waldman, H. (2004) "Preserving Global Optical QoS in FWM Impaired Dynamic Networks", In: IEE Electronics Letters, Vol. 40, p. 191-192.

Fonseca, I. E., Ribeiro, M. R. N., Almeida Jr., R. C. and Waldman, H. (2004) "Meeting Optical QoS in Dynamic Networks with Reduced Complexity", In: Proc. IEEE ECOC'04.

Gomes, A. F., Fernandes, C. E. M., Oliveira, V. A. P., Fonseca, I. E. (2009) "Estratégia de Roteamento em Algoritmos IA-RWA para Redes Ópticas – Uma avaliação da Dispersão de Modo de Polarização," In: Proc. SBrT 2009, Blumenau-SC.

Guidorizzi, H. L. (2008) "Um curso de cálculo", In: LTC, Rio de Janeiro.

Harai, H. (2008) "Optical Packet & Path Integration for Energy Savings toward New Generation Network", In: Proc. IEEE SAINT 2008, p. 389–392.

ITU-T Rec. G.806 (2000) "Characteristics of Transport Equipment - Description Methodology and Generic Functionality".

Lima, M. A. C., Araújo, A.F.R. and César, A.C (2004) "Agregação Dinâmica de Tráfego em Redes Ópticas WDM Utilizando Algoritmo Genético", In: Proc. MOMAG'04.

Martins-Filho, J. F., Bastos-Filho, C. J. A., Arantes, E. A. J., Oliveira, S. C., Coelho, L. D., Oliveira, J. P. G., Dante, R. G., Fontana, E. and Nunes, F. D. (2003) "Novel Routing Algorithm for Transparent Optical Networks Based on Noise Figure and Amplifier", In: Proc. IEEE IMOC2003.

Oliveira, J. C. R. F. (2007) "Amplificadores Ópticos com Controle Automático de Ganho para Aplicação em Redes Ópticas", Tese de Doutorado, FEEC/Unicamp.

Ramamurthy, B. et al., (1999) "Impact of transmission impairments on the teletraffic performance of wavelength-routed optical networks", In: IEEE/OSA J. Ligthwave Technology, vol. 10. p. 1713-1723.

Ramaswami, R. and Sivarajan, K.N. (2010) "Optical networks: a practical perspective", In: Academic Press, 3ª Edition.

Ramaswami, R., (2006) "Optical Networking Technologies: What Worked and What Didn't", In: IEEE Communications Magazine, p. 132 – 139.

Tomkos, I., Azodolmolky, S., Klonidis, D., Aggelou, M. and Margariti, M. (2008) "Dynamic impairment aware networking for transparent mesh optical networks: Activities of EU project DICONET", In: Proc. 10ª IEEE ICTON 2008, p. 6 – 12.

Yuang, M., Chao, I., Bird Lo, Tien, P., Chen, W. J. , Yu-min Lin, Lee, S. S. W. and Ching-yun Chien (2008) "HOPSMAN: An Experimental Testbed System for a 10-Gb/s Optical Packet-Switched WDM Metro Ring Network", In: IEEE Communications Magazine, vol. 47, p. 158 – 166.

Zang, H., Jue, J. P. and Mukherjee, B. (2000) "A Review of Routing and Wavelength Assignment Approaches for Wavelength-Routed Optical *WDM* Networks", In: SPIE Optical Network Magazine, Vol. 1, p. 47-60.

Avaliação de Desempenho do DCCP, CTCP e CUBIC, Usando Tráfegos VoIP

Priscila Doria[1], **Ricardo Salgueiro**[1], **Marco Aurélio Spohn**[2]

[1] Programa de Pós-Graduação em Ciência da Computação
Universidade Federal de Sergipe (UFS)
Departamento de Computação
São Cristovão – SE

[2]Programa de Pós-Graduação em Ciência da Computação
Universidade Federal de Campina Grande – UFCG
Centro de Engenharia Elétrica e Informática
Campina Grande – PB

plgdoria@gmail.com, salgueiro@ufs.br, maspohn@dsc.ufcg.edu.br

Abstract. *DCCP is a prominent transport protocol that has attracted the attention of the scientific community for its rapid progress and good results. Previous works have compared the performance of DCCP with standard transport protocols, assuming a single protocol per link and using the constant bit rate traffic pattern. In this paper, we evaluate the performance of DCCP (CCID2 and CCID3) with modern and representative TCP variants (CTCP and CUBIC). Results show that CCID2 performs better than CCID3 in contention; CUBIC has achieved better throughput, but with higher packet dropping and lower delivery rate; CTCP is outperformed by DCCP CCID2 and CUBIC in most scenarios, outperforming only CCID3.*

Resumo. *O DCCP é um proeminente protocolo de transporte que vem atraindo a atenção da comunidade científica pelos seus rápidos avanços e bons resultados. Trabalhos anteriores comparam o desempenho do DCCP com protocolos clássicos, assumindo um protocolo de cada vez por enlace, sob tráfego CBR. Neste trabalho, o DCCP (CCID2 e CCID3) é comparado a duas variantes modernas e representativas do TCP (CTCP e CUBIC), onde ocorre disputa em pares pelo enlace. Os resultados sugerem que, sob contenção, o desempenho do CCID2 é superior ao do CCID3; o CUBIC obtém melhor vazão, porém com maior descarte de pacotes e menor taxa de entrega; o CTCP é superado pelo CCID2 e pelo CUBIC na maioria dos cenários, superando apenas o CCID3.*

1. Introdução

Os fluxos de multimídia (e.g., fluxos de vídeo, fluxos de áudio) têm uma importância especial nas redes de computadores atuais. Na Internet, deseja-se que os protocolos empregados para transmitir multimídia estejam em harmonia com os demais protocolos. Muitos aplicativos multimídia em tempo real adotam UDP como protocolo de transporte com o intuito de favorecer o desempenho, a despeito da confiabilidade. Entretanto, é típico das transmissões UDP fluir a uma taxa constante, sem observar a largura de banda disponível. O impacto do congestionamento causado pelas transmissões multimídia via UDP motivou

o *Internet Engineering Task Force* (IETF) a propor um novo protocolo padrão da Internet: o Datagram Congestion Control Protocol (DCCP). A principal inovação do DCCP é a priorização do desempenho, como ocorre com o UDP, mas com capacidade de realizar controle de congestionamento, como ocorre com o TCP. O DCCP é um protocolo proeminente, que vem atraindo a atenção da comunidade científica pelo seu rápido desenvolvimento e pelos bons resultados alcançados [Kohler et al. 2006].

O algoritmo de controle de congestionamento padrão do TCP emprega um esquema de aumento aditivo e decréscimo multiplicativo, que usa uma função de crescimento da janela de congestionamento linear e conservadora, juntamente com uma função de decrescimento multiplicativa, em caso de perda de pacotes. Para redes de alta velocidade e com atraso significativo, o TCP padrão pode levar muito tempo para restaurar a taxa de envio após um evento de perda de pacote. Uma maneira simples de contornar esta limitação é modificar as regras de incremento/decremento do TCP no seu mecanismo de controle de congestionamento, de forma que o remetente aumente sua janela de congestionamento mais rapidamente e a diminua mais suavemente, após perdas de pacotes. O comportamento agressivo de tal abordagem pode degradar severamente o desempenho de outros fluxos concorrentes quando a rede está sendo muito utilizada. Entretanto, quando o fluxo de uma variante TCP agressiva de alta velocidade atravessa gargalos de rede juntamente com fluxos de TCP padrão, nesse caso é possível que as variantes agressivas aumentem sua própria fatia de utilização da largura de banda, reduzindo a vazão dos fluxos competidores. Enquanto os fluxos com controle baseado em atraso respondem aos incrementos do RTT, reduzindo a sua taxa de envio, os fluxos com controle baseado em perdas continuam incrementando suas taxas de envio. Portanto, nesses casos, um fluxo baseado em atraso tenderá a obter menos largura de banda do que sua fatia devida em uma divisão equalitária. Para contornar o problema da baixa utilização em redes de grande atraso e alta velocidade, duas novas variantes do TCP foram propostas recentemente: (i) o *Compound* TCP (CTCP) [Tan and Song 2006] e (ii) o CUBIC [Ha et al. 2008].

O CTCP utiliza um algoritmo de controle de congestionamento proposto pela Microsoft™para otimizar o TCP para uso com conexões com grandes janelas de congestionamento. A idéia principal do CTCP é adicionar um componente escalável, baseado em atraso, ao controle de congestionamento do TCP padrão, que é baseado em perdas. A taxa de envio do CTCP é controlada por ambos os componentes, atraso e perda. Ele se tornou um protocolo altamente relevante devido à sua ampla adoção, como parte do *Windows 2008 Server*™e do *Windows Vista*™. O CTCP é uma proposta relativamente nova e, até o limite de esforços realizados neste trabalho, não há avaliações mais detalhadas deste protocolo. O trabalho mais relevante de avaliação da implementação do CTCP foi conduzido pelos seus próprios idealizadores, em bancada de testes.

O CUBIC também é um novo algoritmo de controle de congestionamento do TCP que, em vez de uma função linear, adota uma função cúbica para aumento da janela de congestionamento. O ponto de inflexão desta função cúbica é ajustado pelo valor da janela de congestionamento anterior ao último evento de perda. Os idealizadores do CUBIC defendem que ele proporciona melhores escalabilidade e estabilidade das transmissões em redes velozes e de longa distância. O CUBIC tornou-se um protocolo altamente relevante devido à sua adoção como implementação padrão de TCP no Linux.

Neste trabalho, compara-se o desempenho de duas variantes do DCCP, o CCID2 [Floyd and Kohler 2006] e o CCID3 [Floyd et al. 2006], com o CTCP e o CUBIC, usando o padrão de tráfego VoIP, através de simulações. Considera-se uma topologia onde os protocolos são agrupados em pares e contenções de rede ocorrem à medida que os pacotes de dados trafegam por enlaces de largura de banda limitada, definindo gargalos na rede, de forma que cada par de protocolos deve disputar a largura de banda do enlace. Dessa forma, além de mensurar o desempenho dos diferentes mecanismos de controle de congestionamento, é possível avaliar a propriedade da justiça de cada protocolo ao longo das simulações.

O restante deste trabalho está organizado como segue. A Seção 2 trata dos trabalhos relacionados. A Seção 3 apresenta uma breve descrição dos protocolos que são considerados neste trabalho. A Seção 4 detalha a metodologia usada para conduzir o comparativo de desempenho e apresenta os resultados das simulações. Ao final, na Seção 5, apresentam-se as conclusões.

2. Trabalhos Relacionados

Trabalhos anteriores basearam-se em simulações para comparar o DCCP com outros protocolos de transporte. Trabalhos recentes [Bhatti et al. 2008a, Bhatti and Bateman 2009, Bhatti et al. 2008b] apresentam comparações de desempenho dos protocolos TCP NewReno, BIC, CTCP, CUBIC e DCCP CCID2. Os autores buscaram medir a propriedade da justiça de cada protocolo e concluíram que: (i) o DCCP CCID2 é justo com o TCP NewReno quando o valor do RTT está entre 25 ms e 200 ms e (ii) o BIC, CTCP e o CUBIC são injustos com o DCCP CCID2, quando o RTT tem valores superiores a 25 ms.

Sales [de Sales et al. 2008] comparou os protocolos UDP, DCCP CCID2, DCCP CCID3 e CUBIC em redes *ad hoc* com aplicações multimídia. Nos cenários propostos, o CUBIC e o DCCP puderam partilhar a mesma largura de banda da rede, sem prejuízos. Entretanto, em geral o UDP se mostrou o protocolo mais injusto, consumindo toda a largura de banda disponível.

Takeuchi et al. [Takeuchi et al. 2005] compararam os protocolos UDP, DCCP CCID2, TCP SACK, TCP NewReno e TCP RENO e concluíram que a justiça entre os fluxos do TCP SACK e do DCCP CCID2 depende fortemente do valor utilizado no RTT. Além disso, os autores concluíram que a injustiça mensurada nas simulações, é sobretudo devido à incompatibilidade entre o DCCP e o algoritmo de recuperação rápida do TCP.

Mattsson [Mattsson 2004] comparou os protocolos DCCP CCID2 e TCP SACK. O autor relatou que o módulo do DCCP foi criado para ter um rendimento melhor do que o TCP, mas as simulações mostraram que a taxa de perda e o atraso são semelhantes.

Este trabalho se diferencia dos seus predecessores porque defende que estes adotaram metodologias pouco realistas, pois empregou-se um protocolo de cada vez por cada enlace, evitando assim a disputa entre os protocolos, como também utilizou-se um cenário com padrão de tráfego Constant Bit Rate (CBR) e poucos emissores. Em contrapartida, adota-se neste trabalho um cenário com vários nós transmitindo em tempos aleatórios e usando um padrão de tráfego de voz sobre IP (*Voice over IP*, VoIP). O objetivo é representar uma empresa com filiais e a comunicação telefônica desses estabelecimentos por VoIP.

3. Descrição dos Protocolos

3.1. Datagram Congestion Control Protocol

O *Datagram Congestion Control Protocol* (DCCP) [Kohler et al. 2006] é um protocolo de transporte que provê comunicação *unicast* bidirecional de datagramas não confiáveis com controle de congestionamento. O DCCP é indicado para aplicativos que transferem grandes montantes de dados e podem se beneficiar do controle do equilíbrio entre obsolescência de pacotes e confiabilidade. O DCCP é facilmente extensível com outras formas de controle de congestionamento *unicast*. Três mecanismos de controle de congestionamento já estão especificados: o *TCP-Like Congestion Control* (CCID2) [Floyd and Kohler 2006], *TCP-Friendly Rate Control* (TFRC) (CCID3) [Floyd et al. 2006] e o *TCP-Friendly Rate Control for Small Packets* (TFRC-SP) (CCID4) [Floyd and Kohler 2009]. O CCID4 já está especificado, mas ainda é incipiente e não está contemplado neste trabalho.

O CCID2 envia dados usando uma variante muito próxima do controle de congestionamento do TCP, incorporando confirmações seletivas (SACK). O CCID2 é indicado para remetentes que podem se adaptar a mudanças abruptas da janela de congestionamento, típicas do mecanismo de controle de congestionamento do TCP, com incremento aditivo e decremento multiplicativo. Além disso, é particularmente útil a remetentes que obtém vantagem da largura de banda disponível em um ambiente com variações rápidas de condição.

O CCID3 é um mecanismo de controle de congestionamento baseado no receptor, que provê uma taxa de envio similar à do TCP, porém minimizando as variações abruptas de taxa de envio. A taxa de envio permitida ao remetente é calibrada em resposta à taxa de eventos de perda de pacote, que é reportada ao remetente pelo receptor. Com base nas informações recebidas do receptor, o transmissor determina a taxa de transmissão usando a Equação 1.

$$X = \frac{s}{RTT\sqrt{\frac{2p}{3}} + (4RTT(3\sqrt{\frac{3p}{8}}p(1+32p^2)))} \quad (1)$$

Na equação 1, X representa a taxa de transmissão em $\frac{bytes}{segundo}$, s é o tamanho do pacote em bytes, RTT é o tempo de ida e volta em segundos, e p é a taxa de eventos de perda (i.e., $p \in [0, 1]$).

Para dar suporte à entrega dirigida à obsolescência de pacotes, o DCCP tem uma funcionalidade chamada **escolha tardia de pacotes**, onde o aplicativo pode modificar o que será enviado, de forma tardia no processo, mesmo que o dado do aplicativo já esteja em camadas inferiores da pilha de rede. Esta funcionalidade é vantajosa, dependendo do nível de contenção e tipo de aplicativo, visto que em alguns casos pacotes atrasados podem se tornar irrelevantes após um tempo limítrofe.

3.2. Compound TCP

A *Microsoft*™propôs uma nova variante do TCP chamada Compound TCP (CTCP) [Tan and Song 2006]. O CTCP é parte do *Windows Server 2008*™, do *Windows Vista*™e

do *Windows 7*™, apresentando uma forma alterada do algoritmo de controle de congestionamento baseado em perdas do TCP, incorporando um componente escalável, baseado em atraso [Tan and Song 2006]. Para tanto, uma nova variável é introduzida no bloco de controle do TCP, chamada Janela de Atraso (*Delay Window*), que controla o componente baseado em atraso do CTCP. A janela convencional de controle de congestionamento ($cwnd$) permanece intocada, e controla o componente baseado em perdas de pacotes. Além disso, a janela de envio do CTCP é controlada por ambos os componentes. Especificamente, a janela de envio do TCP (wnd) é calculada como segue:

$$cwnd = cwnd + \frac{1}{cwnd + dwnd} \tag{2}$$

$$wnd = min(cwnd + dwnd, awnd) \tag{3}$$

O CTCP incrementa a janela de congestionamento após receber um ACK, usando a Equação 2. O CTCP usa a Equação 3 para computar a janela de transmissão (wnd) baseando-se na janela de atraso ($dwnd$), na janela de congestionamento ($cwnd$), e na informação de janela do receptor ($awnd$).

3.3. CUBIC

O CUBIC é uma modificação do mecanismo de controle de congestionamento do TCP; em particular, na função de incremento da janela de congestionamento do emissor, de forma a remediar o problema da baixa utilização da largura de banda em redes velozes e de longa distância. Embora a maioria das outras variantes do TCP usem uma função de incremento convexa após um evento de perda de pacote, o CUBIC adota ambos os perfis côncavo e convexo de uma função cúbica para o incremento da janela.

Após uma redução da janela em um evento de perda de pacotes, o CUBIC grava o tamanho da janela de quando ocorreu o evento de perda de pacotes como W_{max} e realiza um decréscimo multiplicativo da janela de congestionamento, seguindo o padrão de recuperação rápida e retransmissao do TCP. Uma vez que seja acionado o controle de congestionamento após uma recuperação rápida, ele inicia o incremento da janela de transmissão usando o perfil côncavo da função cúbica. A função cúbica é calibrada para ter o seu platô em W_{max}, de forma que o perfil de crescimento côncavo da função continue, até que atinja o tamanho W_{max}. Após atingir tal valor, a função cúbica entra em crescimento em perfil convexo. Este estilo de ajuste de janela (côncavo e depois convexo) melhora a estabilidade do protocolo e da rede enquanto mantém uma alta utilização da largura de banda. Isso ocorre porque o tamanho da janela permanece sempre constante, formando um platô próximo do valor W_{max}, onde a utilização da rede foi considerada a maior. Em estado estacionário, muitas amostras do tamanho da janela do CUBIC são próximas de W_{max}; dessa forma promovendo uma alta utilização da rede e boa estabilidade do protocolo.

A janela de envio do TCP (W) é calculada como mostrado na Equação 4, onde C é um parâmetro do CUBIC, t é o tempo decorrido desde o último evento de redução de janela e K é o período que a função levou para incrementar W para W_{max} quando não há outros eventos seguintes de perda, sendo calculado pela Equação 5, onde β é o fator de decréscimo multiplicativo:

$$W(t) = C(t - K)^3 + W_{max} \tag{4}$$

$$K = \sqrt[3]{\frac{W_{max}\beta}{C}} \tag{5}$$

4. Metodologia

Foram realizadas simulações no simulador de redes *Network Simulator 2* [Fall and Varadhan 2007], na versão 2.34, objetivando conduzir um comparativo de desempenho entre os seguintes protocolos de transporte: (i) DCCP CCID2, (ii) DCCP CCID3, (iii) CTCP e (iv) CUBIC, usando padrões de tráfego VoIP. A topologia está detalhada na Seção 4.1. Os protocolos foram combinados em pares distintos para as rodadas de simulação, de forma a ser possível perceber a propriedade da **justiça** de cada protocolo.

4.1. Cenário de Simulação

Considerou-se uma topologia (vide Figura 1) em que quatro redes locais (*Local Area Network*, LAN) estão interconectadas por seus respectivos roteadores, $R = \{R_1, R_2, R_3, R_4\}$, e enlaces adjacentes, $L = \{L_1, L_2, L_3, L_4\}$, formando um cenário típico de uma empresa com um conjunto de filiais. Em cada rede local foram dispostos nós emissores $i \in \mathbb{N} : E_i = \{E_1, E_2, \ldots, E_{48}\}$ e nós destinatários $i \in \mathbb{N} : D_i = \{D_1, D_2, \ldots, D_{48}\}$ em quantidade suficiente e vazões individuais, de tal forma que os enlaces L_1, L_2, L_3, L_4 ficam saturados e em contenção. Assim, para cada rodada de simulação, o respectivo par de protocolos disputa pela largura de banda do enlace compartilhado.

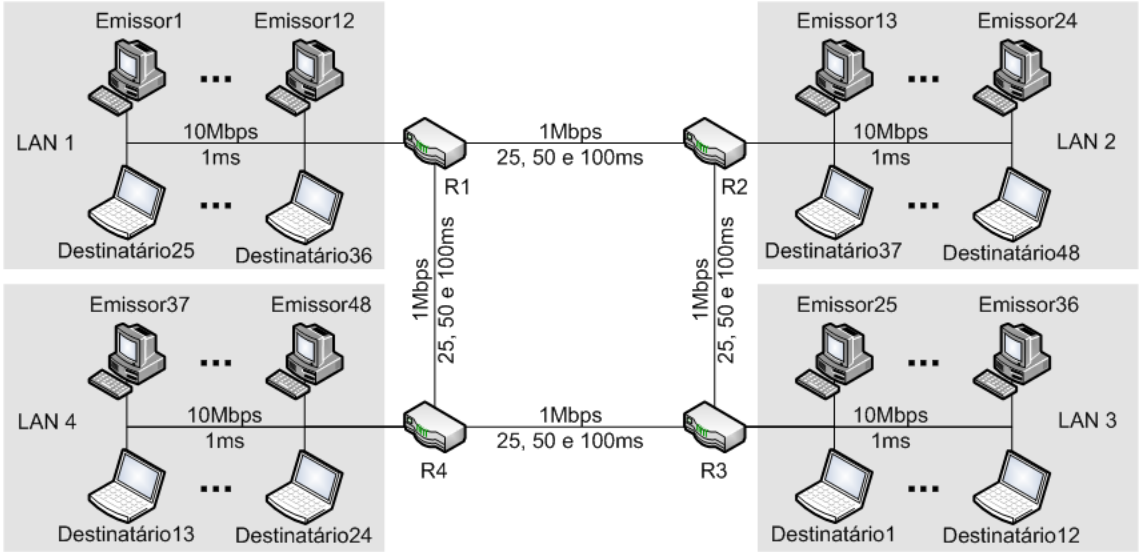


Figura 1. Topologia adotada nas simulações.

As LANs estão configuradas com enlaces internos de largura de banda de 10 Mbps, com latência de 1 ms. L_1, L_2, L_3, L_4 têm largura de banda de 1 Mbps, com latência de 25, 50 e 100 ms. Foram distribuídos 96 nós, uniformemente sobre as quatro LANs,

sendo 12 nós emissores e 12 nós destinatários por LAN. Cada nó emissor em uma LAN tem um nó destinatário correspondente na LAN oposta. Seja E_1 o nó emissor 1 da LAN conectada a R_1, E_1 enviará pacotes na direção de D_1 na LAN conectada a R_3 (i.e., LAN oposta). De forma similar, como E_{48} está disposto na LAN conectada a R_4, D_{48} está disposto na LAN conectada a R_2. Como resultado, os pacotes gerados em uma LAN devem atravessar dois enlaces e um roteador intermediário para alcançar a LAN destino. A fila de pacotes entre os roteadores tiveram os tamanhos ajustados para 300 pacotes e a política de descartes é *DropTail*. Os protocolos foram combinados em pares para cada rodada de simulação, como detalhado a seguir: (a) CTCP *versus* CUBIC, (b) DCCP CCID2 *versus* CUBIC, (c) DCCP CCID2 *versus* CTCP, (d) DCCP CCID2 *versus* DCCP CCID3, (e) DCCP CCID3 *versus* CTCP e (f) DCCP CCID3 *versus* CUBIC.

Considerando que podem ocorrer tempos de inatividade (i.e., tempo sem conversação) significativos em chamadas telefônicas, podendo inclusive superar o tempo de conversação, a razão do tempo de atividade pelo tempo de inatividade é relevante. O comportamento esperado para tal cenário é que nem todos os remetentes estejam falando simultaneamente. Dessa forma, o tráfego VoIP foi simulado pelo aplicativo gerador de tráfego exponencial do NS-2, segundo as recomendações dos trabalhos ITU-T [P.59 1993] e de Bouras et al. [Bouras et al. 2007] para tráfego VoIP simulado. O tempo médio de atividade foi ajustado para $1,004$ s, o tempo médio de inatividade foi ajustado para $1,587$s, a taxa de envio foi ajustada para 80 Kbps, e o tamanho do pacote foi de 160 bytes.

Foram distribuídos 48 remetentes VoIP entre as LANs para que fosse possível medir a propriedade de justiça e o comportamento dos mecanismos de controle de congestionamento de cada protocolo em situações de contenção. O tempo de simulação adotado para as rodadas foi de 900 s.

4.2. Resultados da Simulação

As métricas utilizadas na comparação dos protocolos são: taxa de entrega, pacotes enviados, pacotes descartados, vazão média e atraso médio fim a fim. Os resultados da simulação estão dispostos nas Tabelas 1 a 6, com valores que representam a média de 10 rodadas de simulação, acompanhados dos respectivos intervalos de confiança entre parêntesis, com nível de confiança de 95%.

Conforme dados da Tabela 1, na comparação entre o DCCP CCID2 e o CUBIC é possível perceber que os valores de vazão entre os dois protocolos são próximos quando os valores de RTT são menores. O crescimento linear da janela de transmissão do CCID2 não é muito afetado quando o RTT é baixo, visto que após um evento de descarte, novos pacotes chegam rapidamente, o que provoca um aumento rápido da janela de transmissão do DCCP. É esperado que o CUBIC, com sua função cúbica em fase côncava, retome o valor de janela de transmissão muito rapidamente. Entretanto, como evidenciam os dados das simulações, esse diferencial só resulta em ganho expressivo com RTTs superiores a 50 ms. Com RTT igual 100 ms, é possível perceber que o CUBIC atinge vazão aproximadamente 20% superior ao DCCP CCID2. Tais resultados obtidos neste trabalho opõe-se aos resultados obtidos em trabalhos anteriores [Bhatti and Bateman 2009], que defendem que o CUBIC obteve resultados muito superiores aos do DCCP CCID2, com valores de RTT superiores a 25 ms. Porém, ressalte-se que tais resultados foram produzidos em cenários com padrão de tráfego de fluxo constante.

RTT	Protocolos	Taxa de Entrega (%)	Pacotes Enviados	Pacotes Descartados	Vazão Média (Kbps)	Atraso Médio Fim a Fim(s)
25 ms	DCCP CCID2	98,25 (0,00)	21.435,8 (129,29)	368,10 (5,71)	29,96 (0,18)	0,22 (0,00)
	CUBIC	90,95 (0,00)	21.873,8 (157,10)	1.974,50 (61,89)	28,29 (0,17)	0,21 (0,00)
50 ms	DCCP CCID2	98,43 (0,00)	20.570,6 (59,19)	315,65 (6,17)	28,80 (0,08)	0,27 (0,00)
	CUBIC	92,04 (0,00)	22.282,2 (64,56)	1.765,44 (51,58)	29,17 (0,06)	0,26 (0,00)
100 ms	DCCP CCID2	98,57 (0,00)	18.246,1 (180,10)	249,20 (3,11)	25,58 (0,26)	0,35 (0,00)
	CUBIC	93,26 (0,00)	23.167,6 (193,13)	1.557,43 (43,56)	30,72 (0,23)	0,35 (0,00)

Tabela 1. DCCP CCID2 versus CUBIC

Visto que o CUBIC usa uma função mais agressiva de crescimento da janela de transmissão, é natural que haja descartes de pacotes em situações de contenção. Quando eventos de descarte ocorreram com o CCID2, oportunamente o CUBIC aumentou rapidamente sua janela de transmissão enquanto o CCID2 se recuperava de forma linear. Entretanto, tal comportamento também resultou em um número muito maior de pacotes descartados, provocados por congestionamento. De fato, com RTT de 100 ms, o CUBIC provocou 6 vezes mais pacotes descartados que o DCCP CCID2. O excesso de pacotes descartados resultou em uma taxa de entrega inferior do CUBIC, que foi de 5% a 8% menor que a do CCID2.

Para algumas aplicações multimídia, a perda de uma fração dos pacotes pode ser irrelevante. Porém, alguns aplicativos VoIP requerem uma conexão mais estável, cuja taxa de entrega seja máxima, a despeito da vazão, de forma que a qualidade de áudio possa ser ajustada à vazão disponível. Assim, uma transmissão mais cautelosa como a do DCCP CCID2 seria mais adequada.

Conforme resultados apresentados na Tabela 2, o desempenho do CTCP supera o do CUBIC quando o RTT é igual a 25 ms. Com o RTT=50 ms, a vazão do CTCP e do CUBIC são similares, mas analisando as demais métricas, pode-se concluir que com a taxa de entrega melhor e a baixa quantidade de pacotes descartados, o desempenho do CTCP supera o do CUBIC. Os dados sugerem que o aumento do RTT beneficia mais o desempenho do CUBIC, o que se evidencia com o RTT igual a 100 ms, quando a vazão do CUBIC passa a ser melhor do que a do CTCP. O CTCP possui um componente baseado em atraso em sua função de controle de congestionamento. Por isso, é natural que a vazão do CTCP seja reduzida com valores maiores de RTT, o que não acontece com o CUBIC. Com a redução da vazão do CTCP, o CUBIC oportunamente tentou elevar sua vazão, o que provocou mais congestionamento, reduziu sua taxa de entrega e aumentou sua quantidade de pacotes descartados.

RTT	Protocolos	Taxa de Entrega (%)	Pacotes Enviados	Pacotes Descartados	Vazão Média (Kbps)	Atraso Médio Fim a Fim(s)
25 ms	CTCP	97,17 (0,00)	20.239,0 (95,50)	549,79 (7,13)	28,00 (0,13)	0,21 (0,00)
	CUBIC	91,34 (0,00)	20.315,4 (84,44)	1.754,52 (38,86)	26,39 (0,15)	0,20 (0,00)
50 ms	CTCP	97,29 (0,00)	19.356,5 (107,57)	501,79 (4,75)	26,81 (0,15)	0,26 (0,00)
	CUBIC	91,20 (0,00)	20.739,1 (136,11)	1.818,09 (39,18)	26,90 (0,18)	0,25 (0,00)
100 ms	CTCP	97,42 (0,00)	16.934,4 (123,60)	402,94 (8,04)	23,50 (0,18)	0,34 (0,00)
	CUBIC	89,78 (0,00)	21.939,7 (198,73)	2.257,53 (118,08)	27,98 (0,16)	0,34 (0,00)

Tabela 2. CTCP versus CUBIC

O DCCP CCID2 obteve os melhores resultados quando comparado aos protocolos com controles de congestionamento baseados em atraso, tais como o CTCP e o DCCP CCID3. O CTCP e o DCCP CCID3 usam informação de atraso para controlar a emissão de pacotes e tendem a reduzir a emissão de pacotes quando percebem que a rede está sobrecarregada. Em disputa de enlace, tal procedimento permite que o protocolo oponente utilize uma fatia ociosa da largura de banda da rede durante o recuo do oponente. Como pode ser visto na Tabela 3, na comparação do DCCP CCID2 com o CTCP, o DCCP CCID2 obteve os melhores resultados de taxa de entrega, pacotes descartados e vazão média. Além disso, a variação do RTT não prejudica o desempenho do DCCP CCID2, mas reduz a vazão do CTCP. Fenômeno similar ocorre na Tabela 4, onde o DCCP CCID2 é comparado ao DCCP CCID3.

RTT	Protocolos	Taxa de Entrega (%)	Pacotes Enviados	Pacotes Descartados	Vazão Média (Kbps)	Atraso Médio Fim a Fim(s)
25 ms	DCCP CCID2	98,76 (0,00)	21.736,0 (99,90)	265,30 (4,43)	30,53 (0,14)	0,20 (0,00)
	CTCP	95,34 (0,00)	19.334,2 (83,30)	864,86 (13,00)	26,26 (0,13)	0,20 (0,00)
50 ms	DCCP CCID2	98,96 (0,00)	21.734,3 (158,22)	220,33 (3,45)	30,59 (0,22)	0,25 (0,00)
	CTCP	96,07 (0,00)	19.001,9 (90,36)	709,26 (7,45)	26,01 (0,13)	0,24 (0,00)
100 ms	DCCP CCID2	99,24 (0,00)	21.644,2 (98,17)	155,75 (2,10)	30,55 (0,14)	0,34 (0,00)
	CTCP	97,11 (0,00)	18.588,6 (101,10)	499,63 (9,36)	25,72 (0,14)	0,33 (0,00)

Tabela 3. DCCP CCID2 versus COMPOUND TCP

RTT	Protocolos	Taxa de Entrega (%)	Pacotes Enviados	Pacotes Descartados	Vazão Média (Kbps)	Atraso Médio Fim a Fim(s)
25 ms	DCCP CCID2	99,98 (0,00)	21.917,10 (137,25)	2,53 (0,70)	31,17 (0,20)	0,06 (0,00)
	DCCP CCID3	99,95 (0,00)	7.071,96 (442,54)	0,32 (0,09)	10,06 (0,63)	0,06 (0,00)
50 ms	DCCP CCID2	99,98 (0,00)	21.785,50 (136,75)	1,26 (0,64)	30,98 (0,19)	0,11 (7,01)
	DCCP CCID3	99,93 (0,00)	4.360,65 (114,31)	0,07 (0,04)	6,20 (0,16)	0,11 (7,79)
100 ms	DCCP CCID2	99,97 (0,00)	21.779,20 (132,50)	1,04 (0,67)	30,97 (0,19)	0,21 (2,37)
	DCCP CCID3	99,92 (0,00)	2.325,41 (29,39)	0,01 (0,01)	3,31 (0,04)	0,21 (6,28)

Tabela 4. DCCP CCID2 versus DCCP CCID3

As simulações com o DCCP CCID3 produziram resultados com atraso médio reduzido, poucos pacotes descartados e taxa de entrega de cerca de 99%, independentemente do protocolo oponente. Isso sugere uma alta estabilidade do protocolo. Entretanto, a variação do RTT reduziu o desempenho do DCCP CCID3, o que não ocorreu com os protocolos concorrentes, como pode ser visto nas Tabelas 4, 5 e 6. O comportamento do CCID3 se justifica devido ao seu mecanismo de controle de congestionamento mais preciso, controlado pelos informes do receptor, e que faz uma rigorosa redução da taxa de transmissão para evitar quaisquer congestionamentos nos roteadores. Como resultado, o CCID3 não utiliza eficientemente a largura de banda da rede e seu protocolo oponente tem diversas oportunidades para aumentar sua fatia de ocupação na largura de banda.

5. Conclusão

Neste trabalho foram comparados os desempenhos dos protocolos de transporte DCCP CCID2, DCCP CCID3, CTCP e CUBIC, em cenários onde os protocolos compartilharam o mesmo enlace, dois a dois. Trabalhos anteriores já compararam o desempenho do DCCP com outros protocolos de transporte clássicos. Entretanto, até o limite do nosso

RTT	Protocolos	Taxa de Entrega (%)	Pacotes Enviados	Pacotes Descartados	Vazão Média (Kbps)	Atraso Médio Fim a Fim(s)
25 ms	DCCP CCID3	99,95 (0,00)	6.590,59 (177,90)	0,28 (0,03)	9,37 (0,25)	0,06 (0,00)
	CTCP	99,97 (0,00)	21.774,10 (92,26)	4,83 (0,85)	30,96 (0,13)	0,06 (0,00)
50 ms	DCCP CCID3	99,87 (0,00)	4.116,75 (132,58)	0,11 (0,04)	5,85 (0,19)	0,11 (0,00)
	CTCP	99,97 (0,00)	21.802,60 (108,72)	3,15 (0,63)	31,00 (0,15)	0,11 (0,00)
100 ms	DCCP CCID3	99,76 (0,00)	1.652,28 (84,03)	0,41 (0,03)	2,35 (0,12)	0,21 (0,00)
	CTCP	99,81 (0,00)	21.744,80 (125,30)	35,58 (4,02)	30,87 (0,18)	0,21 (0,00)

Tabela 5. DCCP CCID3 versus CTCP

RTT	Protocolos	Taxa de Entrega (%)	Pacotes Enviados	Pacotes Descartados	Vazão Média (Kbps)	Atraso Médio Fim a Fim(s)
25 ms	DCCP CCID3	99,94 (0,00)	6.006,16 (293,60)	0,40 (0,03)	8,54 (0,42)	0,06 (0,00)
	CUBIC	99,96 (0,00)	21.945,20 (116,57)	7,91 (1,20)	31,20 (0,17)	0,06 (0,00)
50 ms	DCCP CCID3	99,89 (0,00)	3.424,30 (150,92)	0,35 (0,06)	4,87 (0,21)	0,11 (0,00)
	CUBIC	99,93 (0,00)	21.900,70 (125,83)	13,50 (2,82)	31,12 (0,18)	0,11 (0,00)
100 ms	DCCP CCID3	99,62 (0,00)	1.379,63 (44,38)	0,52 (0,01)	1,96 (0,06)	0,22 (0,00)
	CUBIC	99,49 (0,00)	22.000,30 (79,70)	107,33 (5,34)	31,13 (0,11)	0,21 (0,00)

Tabela 6. DCCP CCID3 versus CUBIC

conhecimento, esta é a primeira vez que os protocolos são comparados em um cenário com muitos nós, usando um padrão de tráfego VoIP e disputando o mesmo enlace. Foi possível colher impressões sobre a propriedade da *justiça* dos protocolos comparados.

Os resultados sugeriram que, embora o DCCP CCID3 tenha tido a menor vazão de pacotes dentre os protocolos concorrentes, ambas as variantes do DCCP apresentaram os resultados mais estáveis (i.e., alta taxa de entrega, pouca variação na vazão) e podem ser mais adequadas a aplicações multimídia que são favorecidas pela estabilidade, a despeito da vazão. Além disso, o DCCP CCID2, em oposição aos resultados obtidos com a metodologia de trabalhos anteriores, obteve bons resultados na comparação com o CUBIC e o CTCP. Na maioria dos casos, o CCID2 obteve uma vazão superior à dos protocolos concorrentes, menor quantidade de pacotes descartados e uma taxa de entrega superior a 98%. O CUBIC mostrou-se mais agressivo, apresentando maior quantidade de pacotes descartados e uma taxa de entrega inferior a dos protocolos oponentes. Entretanto, o CUBIC obteve alta vazão média na maioria dos cenários e bons resultados em enlaces com RTT superiores a 100 ms.

Este trabalho aponta novos caminhos de pesquisa. O padrão de tráfego tem demonstrado um impacto significativo no comportamento dos protocolos. Assim, novos padrões de tráfego podem ser testados (e.g., fluxos multimídia de tempo real, vídeo interativo). O impacto de desconexões constantes, como nas redes móveis *ad hoc*, pode ser pesquisado. Finalmente, é possível pesquisar o impacto da infra-estrutura de redes com alto atraso, como as redes de satélites.

Referências

Bhatti, S. and Bateman, M. (2009). Transport protocol throughput fairness. *Journal of Networks*, 4(9):881–894.

Bhatti, S., Bateman, M., and Miras, D. (2008a). A comparative performance evaluation of dccp. In *Performance Evaluation of Computer and Telecommunication Systems, 2008. SPECTS 2008. International Symposium on*, pages 433 –440.

Bhatti, S., Bateman, M., Rehunathan, D., Henderson, T., Bigwood, G., and Miras, D. (2008b). Revisiting inter-flow fairness. In *Proceedings of BROADNETS 2008*, pages 585–592.

Bouras, C., Primpas, D., and Stamos, K. (2007). Enhancing ns-2 with diffserv qos features. In *SpringSim '07: Proceedings of the 2007 spring simulaiton multiconference*, pages 117–124, San Diego, CA, USA. Society for Computer Simulation International.

de Sales, L. M., Almeida, H. O., and Perkusich, A. (2008). On the performance of tcp, udp and dccp over 802.11 g networks. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 2074–2078, New York, NY, USA. ACM.

Fall, K. and Varadhan, K. (2007). The network simulator ns-2: Documentation. `http://www.isi.edu/nsnam/ns/ns-documentation.html`.

Floyd, S. and Kohler, E. (2006). Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 2: TCP-like Congestion Control. RFC 4341 (Proposed Standard).

Floyd, S. and Kohler, E. (2009). Profile for Datagram Congestion Control Protocol (DCCP) Congestion ID 4: TCP-Friendly Rate Control for Small Packets (TFRC-SP). RFC 5622 (Experimental).

Floyd, S., Kohler, E., and Padhye, J. (2006). Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 3: TCP-Friendly Rate Control (TFRC). RFC 4342 (Proposed Standard). Updated by RFC 5348.

Ha, S., Rhee, I., and Xu, L. (2008). Cubic: a new tcp-friendly high-speed tcp variant. *SIGOPS Oper. Syst. Rev.*, 42(5):64–74.

Kohler, E., Handley, M., and Floyd, S. (2006). Datagram Congestion Control Protocol (DCCP). RFC 4340 (Proposed Standard). Updated by RFCs 5595, 5596.

Mattsson, N.-E. (2004). A dccp module for ns-2. Master's thesis, Lulea University of Technology.

P.59, I.-T. R. (1993). Telephone transmission quality objective measuring apparatus: Artificial conversational speech. http://www.itu.int/rec/T-REC-P.59-199303-I.

Takeuchi, S., Koga, H., Iida, K., Kadobayashi, Y., and Yamaguchi, S. (2005). Performance evaluations of dccp for bursty traffic in real-time applications. In *SAINT '05: Proceedings of the The 2005 Symposium on Applications and the Internet*, pages 142–149, Washington, DC, USA. IEEE Computer Society.

Tan, K. and Song, J. (2006). A compound tcp approach for high-speed and long distance networks. In *In Proc. IEEE INFOCOM*.

Gerenciamento de *Handover* Transparente com Suporte Integrado a QoS/QoE em Redes Heterogêneas

Reinaldo Boulhosa[1], TássioCarvalho[1], José Jailton Junior[1], Diego Passos[2], Kelvin Lopes Dias[2], Eduardo Cerqueira[1]

[1]Pós Graduação em Engenharia Elétrica – Universidade Federal do Pará (UFPA)
Caixa Postal 479 – 66075-110 – Belém – PA – Brasil

[2]Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Caixa Postal 7851 – 50670-901 – Recife – PE – Brasil

{rbr,tassio,jjj,cerqueira}@ufpa.br, {kld,dpsilva}@cin.ufpe.br

Abstract.*Theadvances in wireless communication technologies, the demand for new real-timemultimedia applications in with requirements of Quality of Service (QoS) and Quality of Experience (QoE) and the proliferation of mobile devices have motivated the emerging of solutions that allow seamless horizontal and verticalhandovers in heterogeneous networks.This paper proposes ahierarchical technique of transparent handover decision based on cost function in IEEE 802.21 and mapping of applications to service classes to ensure QoS/QoE for users during handovers. The impact and benefits of the proposed solution are evaluated by simulation and by using metrics of QoS and QoE.*

Resumo. *Os avanços nas tecnologias de comunicação sem fio, a demanda por novas aplicações multimídia em tempo real com requisitos de Qualidade de Serviço (QoS) e Qualidade de Experiência (QoE) e a proliferação de equipamentos móveis vem motivando o surgimento de soluções que permitam handovers horizontais e verticais em redes heterogêneas de forma transparente. Este artigo propõe uma técnica hierárquica de decisão de handover transparente baseado em função de custo, no padrão IEEE 802.21 e mapeamento de classes de serviço para garantir QoS/QoE para usuários em períodos de mobilidade. O impacto e os benefícios da proposta são avaliados através de simulação e métricas de QoS e QoE.*

1. Introdução

As redes sem fio de próxima geração serão baseadas em IP, irão suportar milhares de usuários criando, acessando e compartilhando conteúdo multimídia em tempo real e implementarão diferentes tipos de tecnologias de acesso sem fio, como Wi-Fi (*Wireless-Fidelity*, IEEE *(The Institute of Electrical and Electronics Engineers*) 802.11) e WiMAX (*Worldwide Interoperability for Microwave Access*, baseado no IEEE 802.16) [Mahajan and Soans 2010]. Neste contexto é necessário prover a mobilidade transparente de usuários de forma vertical e horizontal com suporte a QoS (*Quality of Service*) e QoE (*Quality of Experience*) em redes heterogêneas.

No que diz respeito à mobilidade e interoperabilidade, o padrão IEEE 802.21 [IEEE P802.21 2009], MIH (*Media Independet Handover*), foi definido e possui protocolos de sinalização e algoritmos que viabilizam o suporte ao *handover* entre redes sem fio com tecnologias de comunicação homogêneas e heterogêneas. Os algoritmos disponibilizam informações das camadas inferiores que ajudam na decisão do *handover* e viabilizam o re-roteamento do cliente MS/SS (*Mobile Station/Subscriber Station*) para o novo ponto de acesso AP (*Access Point*) ou para a uma nova estação base BS (*Base Station*). Desta forma, a mobilidade é feita independe de endereço de rede e tecnologia de comunicação sem fio. Entretanto, a tomada de decisão de *handover* no MIH é realizada somente com base na intensidade do sinal (RSS (*Received Signal Strength*)) de um AP/BS sendo insuficiente para assegurar QoS/QoE das aplicações.

Além do suporte ao *handover*, as redes de próxima geração devem assegurar os requisitos de QoS e QoE das aplicações multimídia em tempo real de forma transparente [Cerqueira et. al. 2008]. Apesar de permitirem a diferenciação de tráfego através de utilização de Classes de Serviços (CoS (*Class of Services*)) e escalonadores, as principais tecnologias de acesso sem fio Wi-Fi (IEEE 802.11e) e WiMAX (IEEE 802.16e) não suportam o mapeamento dinâmico de aplicações em CoSs de forma a assegurar QoS/QoE durante *handovers*.

Este artigo tem como objetivo propor uma solução de gerenciamento de *handover* MIH hierárquico e transparente em redes heterogêneas com suporte a QoS e QoE, denominado MIHQoS/QoE (MIH2Q). O mapeamento de aplicações multimídia em CoSs representa um fator importante para atrair e manter clientes em um mercado extremamente competitivo, ao mesmo tempo que maximiza os lucros dos provedores. O impacto e o benefício da proposta são avaliados através do *Network Simulator version 2* (NS-2) e da ferramenta *Evalvid*, onde se pode notar a melhoria do sistema de mobilidade através análise de parâmetros de QoS e QoE de vídeos reais, como vazão, PSNR (*Peak Signal to Noise Ratio*), SSIM (*Structural Similarity Index*), VQM (*Video Quality Metric*) e MOS (*Mean Opinion Score*).

O restante do artigo está estruturado em: A seção 2 descreverá alguns dos principais trabalhos relacionados utilizados para a elaboração do artigo. A seção 3 apresenta a proposta da função utilizada para viabilizar o *handover* heterogêneo entre as tecnologias levando em consideração o serviço e descreve o funcionamento da solução proposta. A seção 4 faz uma avaliação dos resultados colhidos em simulação apresentando e descrevendo a contribuição destes. As conclusões apresentadas encontram-se na seção 5 onde é descrito os benefícios da proposta. A seção 6 apresenta as referências utilizadas.

2. Trabalhos Relacionados

Como forma de realçar a contribuição e a relevância do artigo, esta seção analisa o estado da arte com foco em gestão de mobilidade, redes heterogêneas e suporte a aplicações multimídia com QoS e QoE.

Existem diversas pesquisas presentes na literatura como [Panaousis and Polyzos 2008] e [Meschke and Tavangarian 2010] que fazem *handover* homogêneo entre as redes através de algoritmos que o antecipam, permitindo assim que os clientes móveis possam manter a conectividade. Esses trabalhos levam em consideração apenas a intensidade do sinal como métrica para decidir em que momento deve-se executar o procedimento de troca de redes (não levando em consideração QoS/QoE e nem a heterogeneidade de tecnologias). A métrica RSS é o fator culminante para tal decisão, principalmente por eles utilizarem o padrão IEEE 802.11k, que define e expõe informações de sinal e de rede para facilitar o gerenciamento de serviços móveis em redes sem fio.

Diferentemente dos trabalhos que levam em consideração apenas a RSS no processo de decisão do *handover* horizontal ou vertical, [He2010] e [Kunarak and Suleesathira 2010] já trazem consigo outras métricas a serem utilizadas como fatoresimportantes para o disparo da decisão do *handover* vertical como taxa de dados, largura de banda disponível, cobertura, distância, velocidade e a preferência do usuário e utilizam respectivamente lógica *fuzzy* e rede neural para analisar e decidir a rede de melhor desempenho e fornecer um *handover* otimizado evitando que o processo seja feito desnecessariamente. Entretanto, o QoS/QoE desejado na nova rede/ponto de acesso não é levado em consideração.

Em [Xie and Wu 2008], os autores apresentam um algoritmo de decisão de *handover* vertical para minimizar o custo das redes através do cálculo da probabilidade de bloqueio e das quedas das chamadas e diferenciando dos demais em sempre utilizar a RSS como padrão em qualquer trabalho que utilize o *handover*. Entretanto, tal como [He2010] e [Kunarak and Suleesathira 2010], os autores não avaliam a qualidade do fluxo multimídia nem a importância do serviço do cliente no processo de decisão.

[Lee *et al.* 2009] apresenta requisitos importantes as redes de próxima geração em suprir a demanda por recursos de tempo real. Eles propõem um algoritmo de decisão de *handover* que, em conjunto com o MIHF (*Media Independent Handover Function*), permite que um cliente móvel decida automaticamente pela melhor rede de acordo com parâmetros de velocidade e RSS. No entanto, eles não levam em consideração métricas de QoS/QoE sobre a importância do serviço no processo mobilidade.

[Putra and Wu 2009] e [Chellappan and Moh 2009] fazem mapeamento de QoS para garantir a qualidade de serviço sobre a largura de banda das redes sem fio. Dependendo das classes de serviço, são aplicados os controles adequados de admissão, conexão e alocação de banda da rede atendendo as necessidades de QoS. Embora possuam o controle e a garantia de qualidade sobre o fluxo da rede, os autores dos trabalhos ficaram vinculados unicamente a rede WiMAX não podendo executar *handovers* verticais e avaliar a QoS em ambientes heterogêneos. No entanto, [Andi et. al. 2010] viabiliza uma avaliação sobre diferentes redes utilizando o MIH e propõe um ambiente com o gerenciamento de múltiplas interfaces com suporte a QoS para terminais móveis sem fio. Diferente dos demais trabalhos, [Andi et. al. 2010] comprovam a eficácia da utilização da QoS na decisão do *handover* vertical. Porém,

nenhum deles faz testes com fluxo multimídia de tempo real nem levam em consideração a qualidade final do vídeo que poderiam ser avaliadas pela QoE.

Após a análise do estado da arte, verificou-se que existem muitos trabalhos publicados que envolvem o gerenciamento de mobilidade em redes homogêneas e heterogêneas, outros que utilizam a QoS como métrica importante para a decisão do *handover*. Outros trabalhos realizam testes com aplicações multimídia voltadas para as redes heterogêneas de próxima geração. No entanto, tais trabalhos não satisfazem todos os requisitos para assegurar a mobilidade transparente em redes heterogêneas para aplicações multimídia em tempo real com suporte a QoS e QoE.

3. Gerênciamento de Mobilidade: MIHQoS/QoE (MIH2Q)

De forma a permitir *handover* transparente de clientes multimídia em redes heterogêneas, esta seção descreve a solução hierárquica e gerenciamento de mobilidade entre redes WiMAX ou WMANs (*Wireless Metropolitan Area Networks*)e as redes Wi-Fi ou WLANs (*Wireless Local Area Networks*) com suporte a QoS e QoE através do padrão IEEE 802.21/MIH. O MIH é utilizado para integrar as sessões IP e gerir a sinalização entre diferentes tecnologias sem fio viabilizando *handovers* verticais.

Para a solução MIH2Q, otimizações no padrão MIH foram realizadas para que a mobilidade seja feita não somente com base na intensidade do sinal/RSS do AP/BS, mas também na velocidade e, principalmente, nos requisitos de QoS/QoE das aplicações e na disponibilidade das classes de serviços nos APs/BSs candidatos (localizados em redes Wi-Fi e/ou WiMAX).

As redes WiMAX definem desde sua concepção cinco tipos de CoS para cinco diferentes tipos de aplicações. São elas: UGS (*Unsolicited Grant Service*), para serviços de voz; rtPS (*Real-Time Pooling Service*), para fluxos de vídeo; nrtPS (*Non-Real-Time Pooling Service*), para a transferência de pacotes de dados através do FTP (*File Transfer Protocol*); BE (*Best Effort*), para fluxo de dados sem exigência de qualidade de serviço como no caso do HTTP (*Hypertext Transfer Protocol*) e o ertPS (*Extension-Real-Time Pooling Service*), para serviços de voz com supressão de ruído e VoIP (*Voice over IP*).

As redes Wi-Fi, no entanto, definem quatro tipos de CoS ou ACs (*Access Categories*) desde a emenda do IEEE 802.11e para diferentes tipos de aplicações. Essas ACs são semelhantes as CoS apresentadas pelo WiMAX e são: AC_VO (*Access Categorie Voice*), para serviços de voz; AC_VI (*Access Categorie Video*), para serviços de vídeo; AC_BK (*Access Categorie Background*), para tráfego de fundo como FTP e AC_BE (*Access Categorie Best Effort*), para o melhor esforço sem exigência de QoS.

Como forma de prover mobilidade transparente com suporte a QoS/QoE, o MIH2Q define uma função de custo baseado em uma hierarquia de decisão da melhor rede que verifica as atribuições baseados nas métricas de velocidade, RSS e parâmetros de QoS e QoE das aplicações suportados pelas CoS.

Para a validação do mesmo, esse artigo realizou seus testes através do simulador de redes NS-2 [Network Simulator 2011] adicionado ao *patch* do [Belghith and Nuaymi

2009] que disponibiliza as classes de serviço (UGS, rtPS e BE) para as redes WiMAX. Nesse *patch*, o autor trabalhou com a utilização de fluxos FTP na classe BE. Posteriormente a isso, este *patch* foi adicionado à função de custo em conjunto à arquitetura do MIH2Q.

3.1 Função de Custo Hierárquica

O processo de decisão de *handover* é executado através de uma função de custo em conjunto com o MIH2Q e acontece por meio da Equação 1 (função γ):

$$\gamma = \frac{\{0{,}25 + [0{,}25 \times CoS] + [0{,}25 \times RSS] + [0{,}25 \times (Velocidade \div 3{,}6)]\}}{3} \qquad (1)$$

Onde:

CoS→ Peso atribuído a classe de serviço do cliente móvel

RSS → Peso correspondente a intensidade do sinal recebida

Velocidade → Velocidade do cliente móvel em km/h

Caso o cliente móvel não possua nenhuma prioridade na disputa de tráfego, então o parâmetro CoS receberá valor igual a 0. Se possuir prioridade UGS/AC_VO, a variável receberá atribuição igual a 1. E se o nó pertencer ao serviço RTPS/AC_VI, o valor atribuído será igual a 2.

Para a variável RSS, também existem três atribuições. Caso a intensidade do sinal seja baixa (entre -127dB e -120dB), o peso atribuído será igual a 0. Se a intensidade do sinalfor mediana (entre -103 e -85dB),o peso atribuído será igual a 1. Caso a intensidade do sinal seja elevada (acima de -85dB),o peso atribuído será igual a 2.A decisão da rede é tomada de acordo com o resultado final calculado pela função de custo γ através dos parâmetros de CoSs, RSS e a velocidade.

Para os parâmetros de CoS e RSS existem três possibilidades correspondentes a pesos (0, 1 e 2). A velocidade é usada na fórmula com seu valor normal descrito na unidade de medida do quilômetro por hora. Tal valor é obtido através da utilização de um GPS (*Global Position System*) [Guet. al. 2009] [Jingxuan et. al. 2009] presente em muitos dos equipamentos modernos de comunicação e da computação atual. Além disso, pode-se captar informações importantes para prever uma possível perda de sinal e assim evitá-la.

Se o resultado final de γ for inferior a 0,7, então o cliente móvel deve realizar o processo de *handover* para uma nova rede (tecnologia diferente); Se for igual ou superior, então o cliente móvel não tem necessidade de mudar entre as redes. Os valores da função de custo tal como o limiar do resultado γ foram obtidos através de experimentos práticos em redes sem fio e justificam as equivalências às métricas utilizadas em redes de múltiplas tecnologias. A função de custo tem como objetivo

inibir que os clientes móveis de alta velocidade associados ao serviço rtPS mudem de rede, e por isso os parâmetros para essa condição recebem os valores mais altos. O fluxograma na Figura 1 exemplifica o seu funcionamento.

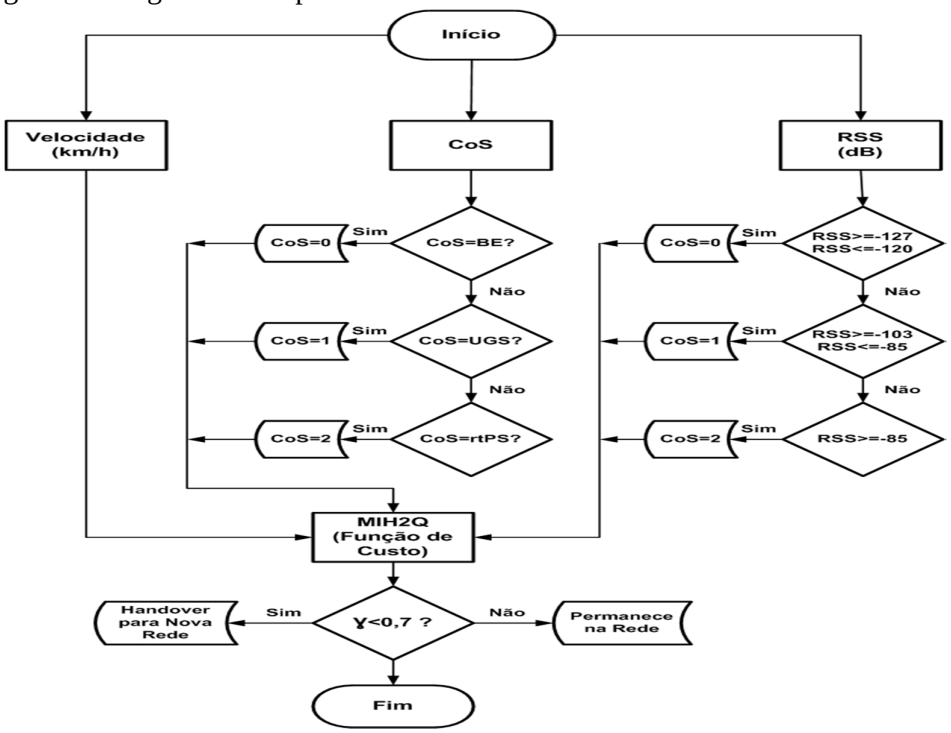


Figura 1. Fluxograma usado para a decisão de mobilidade com o MIH2Q

As redes da próxima geração integram serviços de banda larga (serviço de dados, comunicações multimídia de tempo real, etc.) e serviços de telecomunicações (GPRS (*General Radio Packet Service*), serviços de texto, chamadas em espera, etc.) e, por isso, existe uma necessidade de vazão e conectividades constantes. Com um cenário coberto por diversas redes de diferentes tecnologias sem fio, necessita-se que haja processos que engatilhem a escolha do melhor ponto de acesso dentre as múltiplas interfaces sem fio dos clientes das redes heterogêneas para que estes possam usufruir do melhor serviço, da melhor rede e sempre com a melhor conectividade.

A Figura 2 descreve a arquitetura do MIH2Q para uma rede de próxima geração baseada no *All-IP* exemplificando algumas das possibilidades de clientes móveis se locomoverem horizontalmente ou verticalmente entre as diferentes tecnologias através de equipamentos móveis com múltiplas interfaces (WiMAX e Wi-Fi) com GPS integrado ao MIH que funciona em conjunto a função de custo com suporte a QoS/QoE para a tomada de decisão quanto à necessidade dos *handovers* heterogêneos. A figura exemplifica algumas possibilidades quanto a clientes móveis (SS/MS) em relação ao cálculo da função γ em realizarem ou não o *handover* para uma nova rede quando estas estão disponíveis.

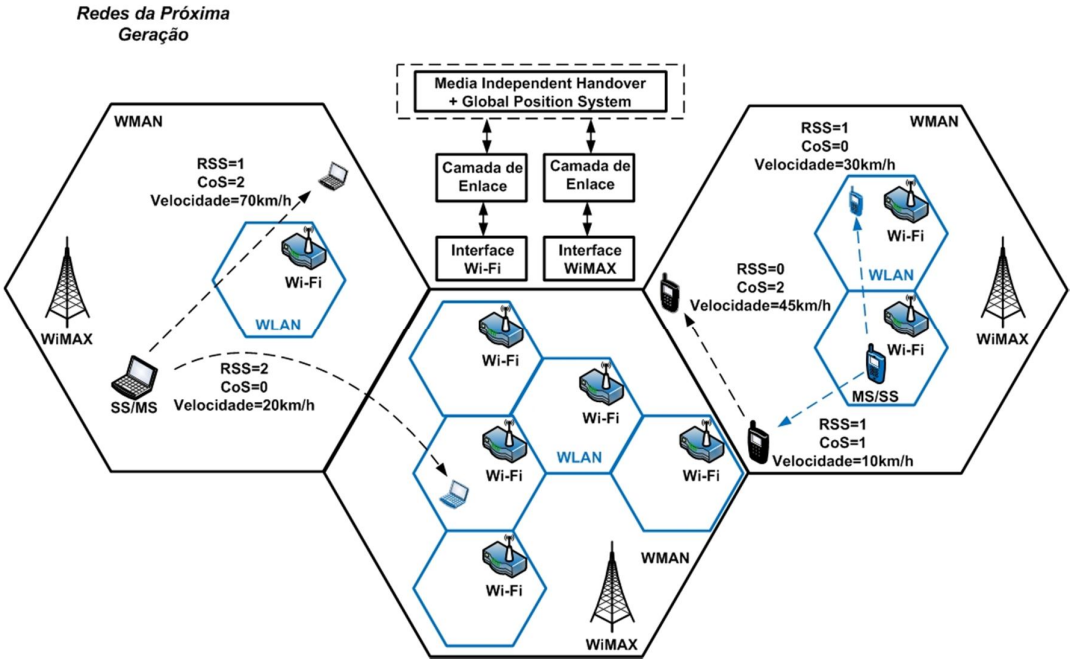


Figura 2. Arquitetura do MIH2Q em Redes Heterogêneas da Próxima Geração

Conforme demonstrado pela figura 2, os clientes podem estar na área de cobertura de uma ou mais redes. De acordo com os valores de RSS, velocidade e a disponibilidade/condições das CoSs (ressaltando que o cliente pode estar recebendo mais de uma aplicação), a solução do MIH2Q pode decidir por uma eventual mudança de rede. Desta forma, o MIH2Q melhora os resultados de vazão e a conectividade impedindo a degradação do enlace da rede e formalizando o ambiente das NGNs com a integração das redes WMAN e WLAN em um ambiente heterogêneo.

4. Avaliação e validação

Conforme mencionado na seção anterior, os resultados deste artigo descrevem a simulação (vinte simulações por experimento) do MIH2Q sobre cenários esperados para as redes sem fio de próxima geração executados no ns-2. O MIH2Q possui uma função de custo que decide o processo de *handover* em uma rede heterogênea WiMAX–Wi-Fi (com 1 km e 100 m de área de cobertura respectivamente) baseado nos parâmetros de velocidade e pesos relacionados a intensidade do sinal e as classes de serviços(rtPS, UGS e BE) utilizadas para o experimento (aplicações com 5 Mbps de vazão). Apesar de não discutido neste artigo, a proposta apresentada pode ser facilmente adaptada para auxiliar no processo de *handover* transparente entre redes com outros tipos de tecnologia de comunicação sem fio como LTE (*Long Term Evolution*).

A aferição da qualidade dos vídeos levando em consideração a perspectiva do usuário foi realizada com o auxílio da ferramenta *Evalvid* que é usada para gerir a transmissão e análise do vídeo transmitido na rede. O processo de decisão de mobilidade é implementado de acordo com o tipo de serviço/qualidade da aplicação, parâmetros de mobilidade e disponibilidade das classes de serviços nas redes/pontos de acesso candidatos. As simulações foram realizadas em três cenários com

comportamentos distintos, onde cada cenário apresenta três clientes móveis recebendo aplicações com diferentes requisitos de qualidade.

As classes de serviços escolhidas para avaliação do artigo foram o UGS, que suporta tráfego com taxa constante (CBR (*Constant Bit Rate*)) ou fluxo como voz sobre IP (VoIP) e que precisa de uma alocação de banda constante. O rtPS que é um serviço para aplicações em tempo real com taxa de transmissão variável (VBR (*Variable Bit Rate*)), como vídeo e o BE que é para tráfego de melhor esforço, onde não existe garantia de QoS como o HTTP e no caso do *patch* utilizado, o FTP, onde são os últimos serviços a receber a banda disponível da rede depois da alocação dos demais.

O processo de avaliação do comportamento da transmissão de vídeo é feita através das métricas de qualidade de experiência objetiva e subjetiva em relação às necessidades do usuário. Este artigo considera as principais métricas de QoE de acordo com o PSNR, SSIM, VQM e MOS..

No PSNR o vídeo é avaliado comparando a qualidade do vídeo original ao vídeo recebido pelo cliente, considerando o aspecto de luminosidade e a sua unidade é em decibel. A Tabela 1 demonstra o mapeamento do PSNR (em decibéis) em MOS.

Tabela 1. Mapeamento de PSNR para MOS

PSNR (dB)	MOS
> 37	Excelente
31 – 37	Bom
25 – 31	Aceitável
20 – 25	Ruim
< 20	Péssimo

No SSIM, o comportamento do vídeo é baseado na medição quadro a quadro comparado ao vídeo original, analisando os seguintes itens: contraste, luminosidade e estrutura, onde é gerado um valor decimal entre 0 e 1, no qual quanto mais próximo do 1 apresenta, melhor a qualidade do vídeo analisado. O VQM analisa os aspectos de ruído, distorção dos frames, cor e do quão embaçado está o vídeo, onde quanto mais próximo o valor se aproximar de 0, melhor será a qualidade dele.

4.1 Experimento 1

O primeiro experimento (Cenário 1) possui três clientes com os serviços de BE, rtPS e UGS e se encontram inicialmente na rede WiMAX por ter maior capacidade, alcance e disponibilidade. Mesmo estando sob o alcance das duas redes, como não existe nenhum tipo de solução incluída, os clientes só sairiam da rede WiMAX se quisessem, caso contrário, como demonstrado pela Figura 3, permanecem na rede na qual estão. Conforme indicam os resultados de simulação da Figura 4, os clientes ficam constantemente competindo pela largura de banda da WiMAX prejudicando uns aos outros em suas respectivas aplicações devido a ausência de qualquer tipo de controle para gerenciar a mobilidade dos clientes móveis, suas CoS e consequentemente a QoS.

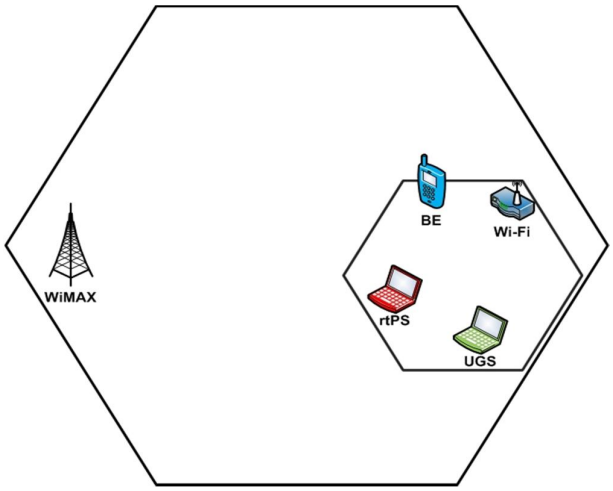


Figura 3. Cenário 1

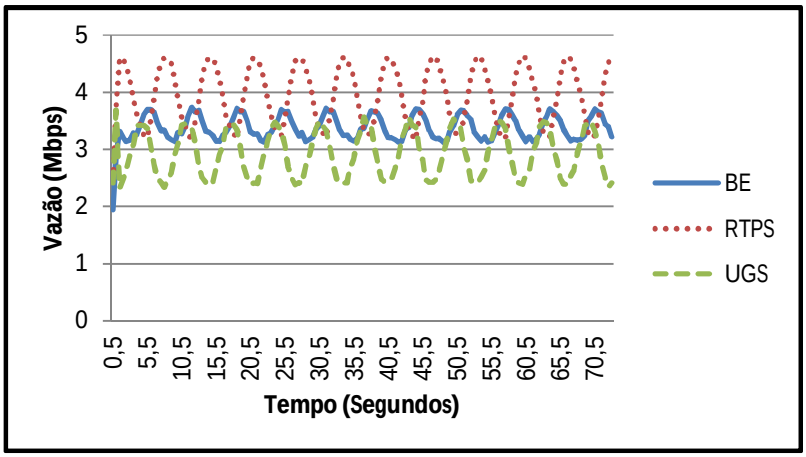


Figura 4. Resultado de Vazão para as CoS no experimento1

A competição pelos recursos de redes prejudica a qualidade das aplicações tanto do ponto de vista da rede (vazão), como do usuário conforme demonstrado na Figura 5 no instante de *frame* 1300.

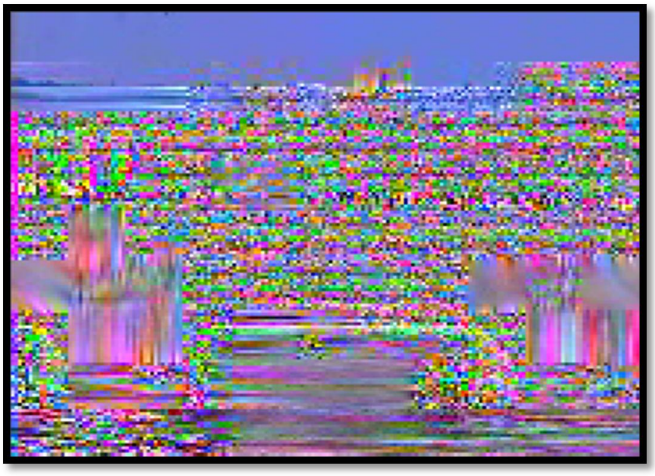


Figura 5. Vídeo de Péssima qualidade no Evalvid do Cenário 1

O valor do PSNR para este vídeo foi de 16.31 e consequentemente o MOS é péssimo. O do SSIM foi 0.31 e o VQM 14.57. Dessa forma, pode-se afirmar que o vídeo deste cenário tem uma péssima qualidade de acordo com a perspectiva do usuário devido a constante concorrência entre os três clientes com suas respectivas aplicações o que degrada a qualidade do vídeo do cliente com o serviço rtPS.

4.2 Experimento2

O segundo experimento (Cenário 2) possui três clientes com os serviços de BE, rtPS e UGS e encontram-se inicialmente em uma rede WiMAX. Com o passar do tempo, o cliente com a aplicação que utiliza o serviço do BE e posteriormente outro cliente com uma aplicação que se utiliza do rtPS realizam o *handover* para uma rede Wi-Fi de menor capacidade conforme indica a Figura 6.

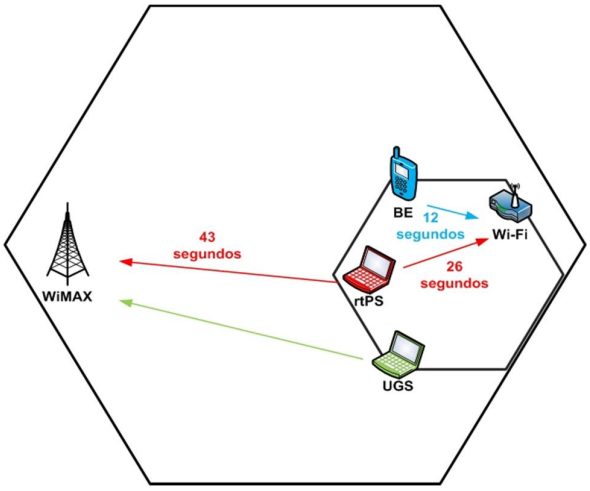


Figura 6. Cenário 2

Durante o instante de tempo de aproximadamente 12 segundos, o cliente com a aplicação que utiliza a CoS do BE faz um *handover* para uma rede Wi-Fi e devido à falta de concorrência consegue temporariamente ter um ganho de vazão. No entanto, aos 25 segundos o cliente com a aplicação mapeada no rtPS faz um *handover* para a limitada rede Wi-Fi e começa a competir pela vazão fazendo com que ambos os clientes percam consideravelmente a mesma e prejudiquem suas aplicações. Assim, o cliente com a aplicação do rtPS tem seu vídeo prejudicado juntamente ao cliente da aplicação do BE, inclusive tendo vazão inferior à vazão enquanto os três clientes competiam na rede WiMAX devido a menor capacidade da rede Wi-Fi atual.

Aos 43 segundos, no entanto, o cliente de vídeo retorna para a rede WiMAX e volta a apresentar vazão suficiente para uma melhor qualidade além de deixar de concorrer pela vazão com o cliente com a aplicação mapeada para o BE conforme demonstram os resultados de vazão apresentados pela Figura 7.

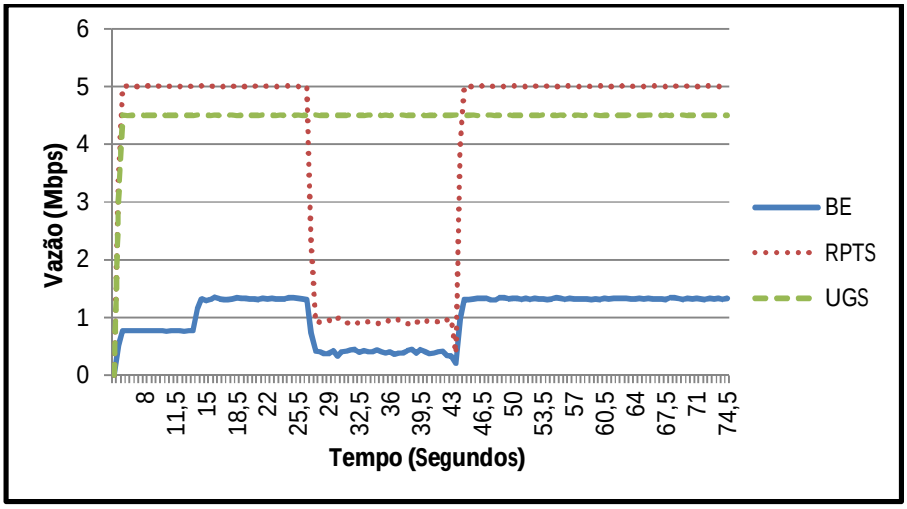


Figura 7. Resultado de Vazão para as CoS no Cenário 2

A Figura 8 comprova a visualização gráfica do vídeo do cliente com o serviço do rtPS no *frame* 1300 durante um h*andover*. Os valores do PSNR, SSIM e VQM foram 25.58, 0.8 e 2.15, respectivamente. Além disso, pode-se dizer que o vídeo neste cenário tem uma qualidade aceitável de acordo com o mapeamento do PSNR para o MOS.

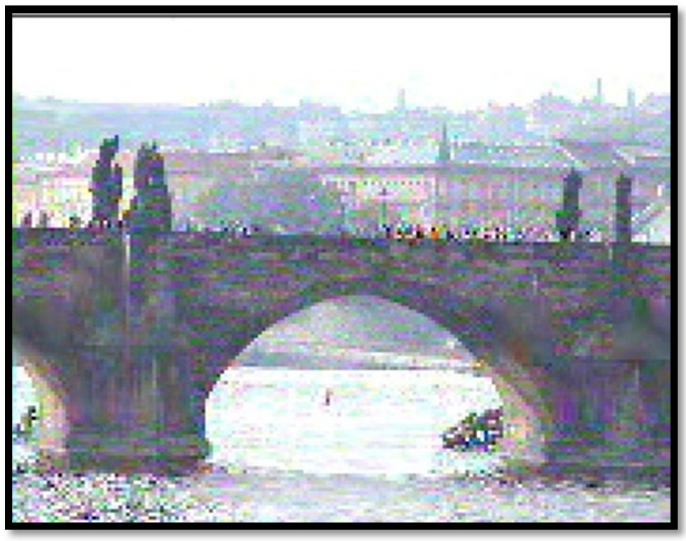


Figura 8. Vídeo de qualidade aceitável no Evalvid do Cenário 2

4.3 Experimento 3

O terceiro experimento (Cenário 3) utiliza o MIH2Q em conjunto a função de custo e possui três clientes recebendo aplicações mapeadas em BE, rtPS e UGS e estão localizados inicialmente em uma rede WiMAX conforme os cenários anteriores. Durante a simulação, a solução proposta faz com que os clientes com aplicações UGS e rtPS que possuem mais prioridade continuem no WiMAX mantendo qualidade de suas aplicações e decide que o cliente com o BE faça um *handover* para a rede Wi-Fi com menor capacidade, mas capaz de suprir sua necessidade já que não apresenta concorrentes para degradar o seu serviço conforme demonstra a Figura 9.

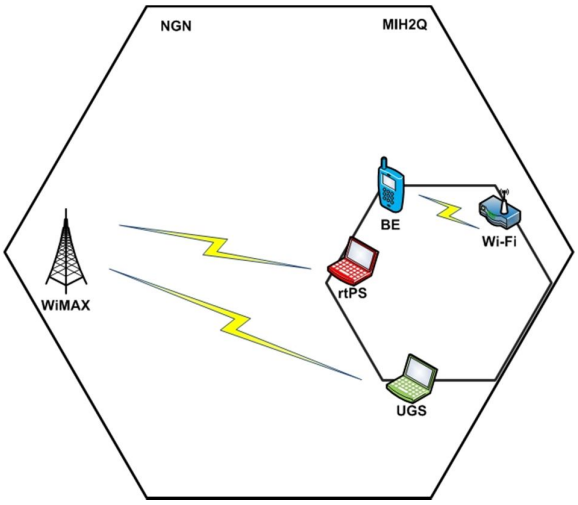


Figura 9. Cenário 3

Conforme o valor de vazão apresentado pela Figura 10 se pode afirmar que a função hierárquica de gerenciamento de mobilidade é eficiente e eficaz, pois faz com que os clientes consigam escolher a melhor rede de acordo com as aplicações e usufruir desta forma da melhor capacidade que as tecnologias podem oferecer para os clientes em determinado momento devido à utilização da proposta do MIH2Q.

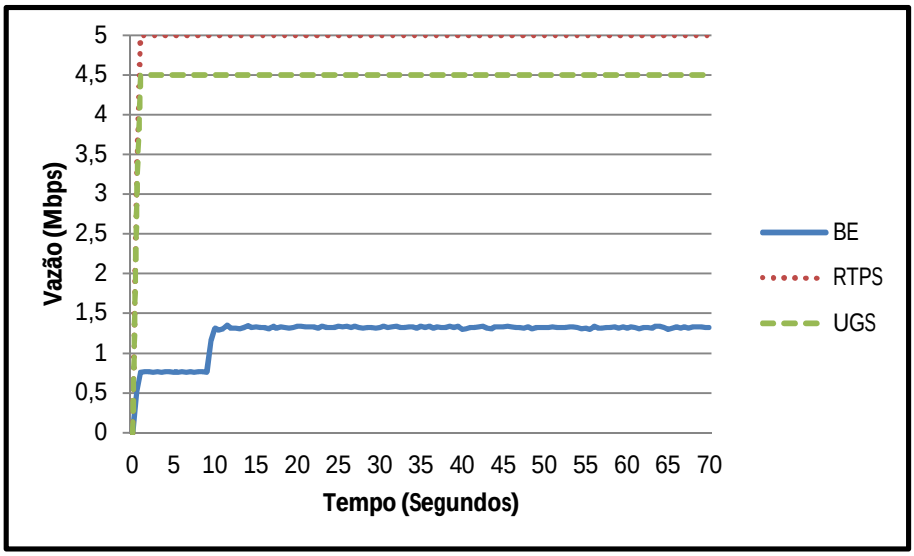


Figura 10. Resultado de Vazão para as CoS no Cenário 3

Analisando o cliente com o serviço rtPS no mesmo *frame* 1300 testado nos cenários anteriores, verifica-se que o PSNR obtém valor igual a 42.16 e o MOS é excelente. Os valores de SSIM e VQM foram 0.98 e 0.51, respectivamente. Dessa forma, pode-se afirmar segundo as métricas de QoE, que o vídeo apresenta uma excelente qualidade conforme pode-se notar e comprovar na Figura 11.

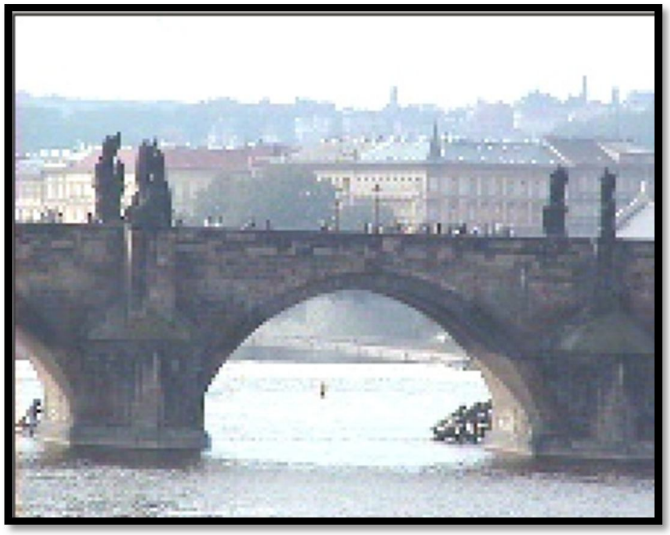


Figura 11. Vídeo de excelente qualidade no Evalvid do Cenário 3

5. Conclusões

Nos cenários das redes da próxima geração, espera-se que existam redes com diferentes tecnologias interligadas horizontalmente e verticalmente e que os clientes móveis com múltiplas interfaces possam mudar entre elas livremente de forma a receber o melhor serviço. Neste contexto, o artigo propôs um gerenciamento de mobilidade vertical baseado em uma função de custo para a decisão da melhor rede nomeada MIH2Q e que não leva em consideração apenas a RSS, mas também a velocidade e o tipo do serviço com suporte à QoS/QoE.

Os resultados de simulação comprovaram a eficácia e a eficiência da solução proposta pelo MIH2Q, mostrando que devido à solução de decisão, clientes de maior prioridade quanto devido à aplicação são impedidos de migrar para redes de menor capacidade e clientes com aplicações de CoS de menor prioridade são alocados em redes menos densas quanto à vazão disponível e a quantidade de clientes concorrentes como as redes Wi-Fi impedindo que diversos clientes de diferentes prioridades possam degradar seus serviços entre si. Com isso, os resultados da CoS BE obtém um ganho de vazão de aproximadamente 72,4% e as aplicações de rtPS e UG mantém vazão máxima e qualidade quanto a QoS e QoE.

Nos trabalhos futuros, espera-se interagir com um número maior de métricas como balanceamento de carga, problemas com propagação de sinal, distância e em outras medidas preponderantes para o processo de decisão da melhor rede.

Agradecimentos

Este trabalho foi financiado pela CAPES, CNPq e FAPESPA.

6. Referências

Andi, W. C., Yeo, C. K., Lee, B. S. (2010).“Environment-Aware QoS framework for multi-interface terminal”.Computer Communications, 33 (2010), 1049-1055.

Belghith, A. and Nuaymi, L. (2009) “Design and Implementation of a QoS- included WiMAX Module for NS-2 Simulator”. Disponível em: http://perso.telecom-bretagne.eu/aymenbelghith/tools/. Acessado em: Janeiro de 2011.

Cerqueira, E. Veloso, L. Curado, M. Monteiro, E. Mendes, P. (2008). “Quality Level Control for Multi-user Sessions in Future Generation Networks”. Global Communications Conference, 2008. IEEE Globecom 2008.

Chellappan, B. Moh, T. Moh, M. (2009) “On Supporting Multiple Quality-of-Services Classes in Mobile WiMAX Handoff”. International Conference on Computing, Engineering and Information

Gu, C. Song, M. Zhang, Y. Man, Y. Hou, C. (2009). “GPS Information assisted handoff mechanism in heterogeneous wireless networks”. Communications Technology and Applications, 2009.ICCTA ‘09. IEEE International Conference on.

He, Q. (2010).“A Fuzzy Logic Based Vertical Handoff Decision Algorithm between WWAN and WLAN”. International Conference on Networking and Digital Society

IEEE 802.16e, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems Amendment 2: Physical and Medium AccessControl Layers for Combined Fixed and Mobile Operation in Licensed Bands, Dec. 2005.

IEEE P802.21 WG (2009). “IEEE Standard for Local and Metropolitan Area Networks: Part 21: Media Independent Handover Services”. IEEE Std. 802.21-2008.

Jingxuan, W. Mingjun, T. Tong, Z. Wei, Y. (2009).“A GPS-Free Wireless Mesh Networks Localization Approach”. Communications and Mobile Computing, 2009.CMC ’09.WRI Internacional Conference on.

Kunarak, S. e Suleesathira, R. (2010).“Predictive RSS with Fuzzy Logic based Vertical Handoff Algorithm in Heterogeneous Wireless Networks”. International Conference on Advanced Technologies for Communications.

Lee, S., Sriram, F., Kim, K.; Kim, Y., e Golmie, N. (2009) “Vertical Handoff Decision Algorithms for Providing Optimized Performance in Heterogeneous Wireless Networks”. IEEE Transactions on Vehicular Technology, Vol. 58, NO. 2.

Mahajan, A. Soans, S. (2010). “Next Generation Mobile Network Concepts Technology and Sample Applications”. Wireless Communications and Sensor Computing. ICWCSC, 2010, Internacional Conference On

Meschke, R. Krohn, M. Daher, R. Gladisch, A. Tavangarian, D. (2010) “Novel Handoff Concepts for Roadside Networks using Mechanisms of IEEE 802.11k & IEEE 802.11 v”.

Network Simulator.(2011). “The Network Simulator version two – NS 2”.Disponível em: http//www.isi.edu/nsnam/ns/. Acessado em: Janeiro de 2011.

Panaousis, E. Frangoudis, P. Ververidis, C. Polyzos, G. (2008) “Optimizing the Channel Load Reporting Process i0,n IEEE 802.11k-enabled WLANs”. Proceedings of the 2008 16th IEEE Workshop on Local and Metropolitan Area Networks.

Putra, D. Ke, K. Wu, H. (2009) “Self-Similar Traffic Assessment on QoS Service Classes of WiMAX Network”

Xie, S.e Wu, M. (2008).“Vertical Handdof Decision Algorithm to Minimize the Network Cost”.

Scheduling Based on Process Behavior Analysis

Paulo H. R. Gabriel[1] **and Rodrigo F. de Mello**[1]

[1]Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação
Av. Trabalhador são-carlense, 400 – São Carlos – SP – Brazil

{phrg,mello}@icmc.usp.br

Abstract. *Process scheduling researches attempt to understand the dynamics of applications in order to improve resource allocation policies. Recent studies in this area have analyzed the behavior of single processes without considering how they interact with each other. This drawback motivated this paper, which proposes a new process scheduling approach based on how applications interact when competing for resources. This approach is based on concepts of dynamical systems theory which state that stabler organizations can be reached by means of perturbations in system components. First, we analyze process occupation variables to obtain behavioral states which represent a system component here. Next, we combine the execution of processes by prioritizing the one with higher estimated CPU load. The execution of processes is then interleaved according to their predicted workloads and the system tends to be stabler. Here, the term system refers to the combination of behavioral states of all processes. To validate this approach, we consider simulated scenarios representing different workloads of a computational environment, obtaining shorter execution times and, therefore, higher performance as result.*

Resumo. *Pesquisas na área de escalonamento têm procurado entender a dinamicidade de aplicações, a fim de melhorar políticas de alocação de recursos. Estudos recentes analisam o comportamento de processos individuais sem considerar como eles interagem entre si. Essa limitação motivou este trabalho, que propõe uma nova abordagem de escalonamento de processos com base na maneira como as aplicações interagem quando competindo por recursos. Essa abordagem é baseada em conceitos de sistemas dinâmicos segundo os quais organizações estáveis podem ser alcançadas por meio de pertubações nos componentes do sistema. Foram analisados processos com diferentes ocupações a fim de obter estados comportamentais que representam um componente do sistema. Em seguida, combinou-se a execução de processos, priorizando os de maior carga estimada de CPU. Intercala-se, portanto, a execução de processos de acordo com suas cargas de trabalhos preditas, o que tende a um sistema mais estável. No contexto deste artigo, o termo sistema se refere à combinação de estados comportamentais de todos os processos. Para validar a abordagem, considerou-se cenários simulados representando diferentes cargas de um ambiente computacional obtendo, como resultado, tempos de execução menores, ou seja, maior desempenho.*

1. Introduction

The interaction of multiple systems is a major research topic for different scientific areas [Kennedy et al. 2001, Alligood et al. 1996, González-Miranda 2004]. In Physics, for example, the synchronization of multiple out-of-phase harmonic oscillators has been studied and formalized. In Astronomy, Kepler equations are part of several studies to analyze the influences of planets' orbits on satellites, comets, etc. In Biology, the relationship of predators and preys can be described by Lotka-Volterra equations. Still in Biology, the behavior of swarms, such as birds migration and organization of ant and bee colonies has been studied in an attempt to understand emergent characteristics.

More recently, different branches of Computer Sciences have looked for inspiration in such studies. For example, bio-inspired meta-heuristics, like ant colony and particle swarm, have been successfully applied to solve several optimization and computational intelligence problems [Kennedy et al. 2001]. Still in this area, process scheduling researches have been attempting to understand the dynamics of applications to improve resource allocation policies [Mello and Yang 2009, Dodonov and Mello 2010]. The main studies in this subject consider the monitoring and analysis of application occupation variables (*e.g.*, CPU load, memory accesses, and hard disk utilization over time) in order to characterize behavioral states and, based on those, improve resource allocation. However, these studies analyze single processes and, consequently, there is no particular study to understand how they interact with each other[1].

The possibility of understanding the behavior of individual processes and also their interactions and influences has motivated this paper, which proposes a new process scheduling approach based on how applications interact when competing for resources. This approach is based on dynamical systems theory [Alligood et al. 1996, González-Miranda 2004] and considers how the behavior of different processes can be rearranged or combined over time in order to obtain stabler dynamical systems[2]. By finding a stabler representation for the combination of multiple processes, we attempt to optimize resource allocation over time. For example, by having two or more processes competing for the same CPU, we analyze occupation variables for every one and, thus, obtain behavioral states associated with each process. Such states represent, for instance, amounts of CPU, memory and hard disk consumed. Afterwards, we attempt to interleave the execution of both processes by prioritizing the one with higher estimated CPU load. This strategy tends to a stabler system, resulting in shorter execution times, meaning higher performance. All these steps are performed by our scheduling policy.

The remainder of this paper is organized as follows: Section 2, presents some related work, focusing on how predictions improve scheduling decisions; Section 3 address concepts of dynamical systems; Section 4 details our approach, presents experimental results and also the analysis; Conclusions and future directions are reported in Section 5.

[1]This interaction is related to how processes compete to obtain resources and modify one each others behavior.

[2]It is important to make clear that 'stabler' is a term related to dynamical systems which means there is less behavioral variation. Such variations could jeopardize system forecasting.

2. Related Work

Several studies have confirmed that application knowledge can improve scheduling decisions. Many of these researches confront features provided by a set of computational resources in order to optimize allocation and minimize the response time of processes. In this venue, [Ferrari and Zhou 1988] performed an experimental study considering some load balancing algorithms proposed in the literature. It starts looking for clues to confirm the costs related to resource allocation and decision making. Using analytical models and simulation, [Sevcik 1989] demonstrated that it is possible to substantially improve scheduling decisions using knowledge about applications.

Following this same research venue, [Devarakonda and Iyer 1989] developed one of the first studies employing a pattern recognition technique to predict the utilization of resources, such as CPU processing, input/output operations and memory accesses. They employed a k-means algorithm to identify states of resource occupation in a monoprocessor UNIX system. The authors, however, did not consider communication and synchronization costs. On the other hand, their study motivated further works focused on the use of historical information.

[Feitelson and Nitzberg 1995], for example, showed that the execution time of parallel applications can be estimated from repeated executions, indicating that historical behaviors of applications can be used to improve future scheduling decisions. The authors also analyzed production data and confirmed that approximately two thirds of applications are executed multiple times. In another study, [Feitelson et al. 1997] noticed that repeated executions of the same application tend to present similar patterns of resource utilization.

Other researchers, like [Gibbons 1997], [Smith et al. 2004] and [Downey 1997] focused their studies on the prediction of response times of parallel applications based on previous executions. [Gibbons 1997] made predictions examining categories of applications while [Smith et al. 2004] used search techniques, such as greedy search and genetic algorithms, to find historical information based on application similarity. They used such information to characterize and predict the behavior of new applications. The same methodology was followed by [Krishnaswamy et al. 2004], who proposed algorithms to estimate the process execution times.

On the other hand, [Downey 1997] estimated process execution times by modeling cumulative distribution functions for every application category. The author employed them to approximate the behavior of future applications. More recently, [Schopf and Berman 1999] proposed the use of stochastic processes to parametrize performance characteristics, such as bandwidth, available CPU, message size and operation accounting. Later, [Lee and Schopf 2003] adopted regression models to establish relationships between performance characteristics and execution times of past applications.

Still in terms of historical information, [Dinda 2001] studied the scheduling problem considering a different point of view: the resource behavior instead of the process one. The author assumed an auto-regressive model with sixteen coefficients to estimate the CPU load of computers. According to his policy, processes are allocated on computers at lower predicted loads. He concluded that this policy improves load balancing, which was also later confirmed by [Chunlin and Layuan 2009].

Looking for a more precise prediction, [Mello and Yang 2009] and [Dodonov and Mello 2010] considered dynamical systems concepts and nonlinear prediction techniques to model and predict process behaviors. [Mello and Yang 2009] monitored processes occupation variables over time and evaluated their similarity and recurrence of behavioral patterns. They calculated the required number of past variable observations to efficiently forecast their future occurrences (*e.g.*, CPU load, memory occupation, etc.). The authors presented a theoretical proof of efficiency for the estimation of the embedded and separation dimensions using real-world application traces. Such estimations are applied to Takens' immersion theorem [Takens 1980] in order to predict future process occupations. The study was extended by [Dodonov and Mello 2010], who proposed a framework to provide an on-line and adaptive behavior prediction mechanism for an efficient application scheduling based on the anticipation of communication events. Results obtained in heterogeneous environments confirm the efficiency of the prediction mechanism, which outperforms conventional scheduling policies.

All these studies have highlighted the relevance of using application knowledge in process scheduling. However, even works that attempt to obtain more precise predictions [Mello and Yang 2009, Dodonov and Mello 2010] are limited to the analysis of individual processes and do not consider the interaction of multiple processes and how it can affect the scheduling. In order to overcome this drawback, we here study and consider adaptations in the interaction of multiple processes to reach a stabler resultant system. Such stability, derived from dynamical systems [Alligood et al. 1996], improves the predictability of behavioral states. The next section describes the dynamical systems concepts considered in our approach.

3. Dynamical Systems

A dynamical system is composed of a set of possible states and a rule that determines its current state in terms of past ones. Mathematically, a dynamical system is described by $x_{n+1} = f(x_n)$, where $n \in B$ ($B \subset \mathbb{R}$) denotes time, $x : B \rightarrow \mathbb{R}$ represents the state of the system and $f : \mathbb{R} \rightarrow \mathbb{R}$ is the rule or evolution law [Alligood et al. 1996]. In this case, the rule defines the next state in function of past ones, therefore this system is characterized as deterministic. Besides deterministic systems, there is also another class, named stochastic, in which rules also involve random terms [Alligood et al. 1996]. In addition, such dynamical systems also rely on initial conditions. These conditions define the input values for the rule which, consequently, affect system outputs.

In order to illustrate a dynamical system, let us consider the Logistic map presented in Equation 1, which is traditionally used to model population growth over time. Let this map start with the following initial conditions $b = 3.8$ and $x_0 = 0.5$, for $t \in [0, 500]$ iterations.

$$x_{t+1} = b \cdot x_t \cdot (1.0 - x_t) \tag{1}$$

Figure 2(a) shows the Logistic map outputs given the presented conditions. By conducting a detailed analysis, one can conclude this function presents low recurrence, chaoticity and behavior instability [Alligood et al. 1996]. Therefore, it is difficult to predict such a system by using statistical methods. However, one can reconstruct this system

in order to observe internal regularities and simplify its understanding. This reconstruction supports the estimation of the rule, which indicates how the system evolves over time. Consequently, by reconstructing a given set of outputs, one may obtain that particular system rule, understand and predict it.

[Whitney 1936] observed the possibility of reconstructing data in multidimensional spaces, applying the concept of differential manifolds. Mathematically, $M \subset \mathbb{R}^k$ is a manifold of dimension m if, for each point $p \in M$, there is a neighborhood $U \subset M$ of p and a homeomorphism $x : U \rightarrow U_0$. U_0 is an open set of $\mathbb{R}^m$, such that the inverse homeomorphism $x^{-1} : U_0 \rightarrow U \subset \mathbb{R}^k$ is an immersion of class C^∞. In summary, for each $u \in U_0$, the derivative $\frac{d(u)}{dx} : \mathbb{R}^m \rightarrow \mathbb{R}^k$ is biunivocal.

Figure 1 shows an example of a plane parametrization $\mathbb{R}^{m-1} \times \mathbb{R}_+$ to $\mathbb{R}^k$. Given a point q', we can, by means of $\phi_q : H_0 \rightarrow H \cap M$, find a corresponding point $q \in M$. This example illustrates the mapping of a point and its neighborhood in a space with a higher number of dimensions.

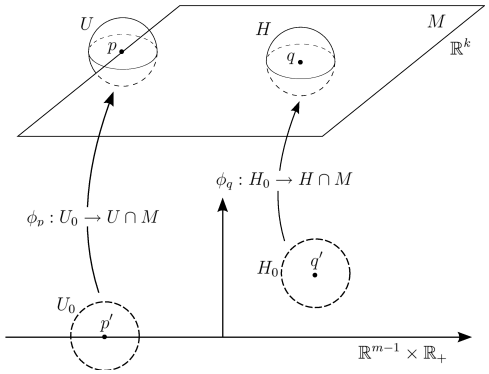


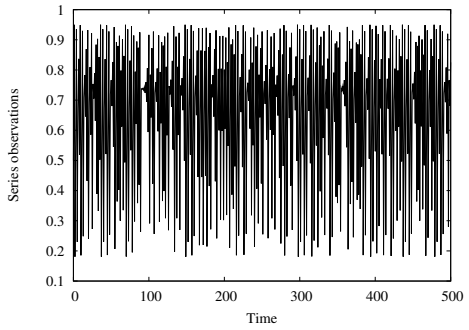
Figure 1. Example of a manifold.

This mapping allow understanding unobservable or underrepresented behaviors which are better described in a higher number of dimensions. From this study, [Whitney 1936] proposed his immersion theorem, whereby every trajectory in m dimensions can be mapped into a space with $2n + 1$ dimensions.

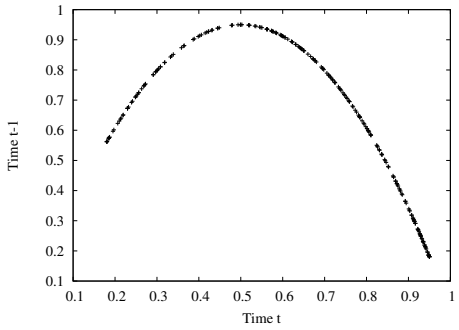
Based on such a theorem, [Takens 1980] proved that, instead of mapping states into a $2n + 1$-dimensional space, one can improve reconstruction considering time offsets. Thus, the outputs of a dynamical system, here seen as an one-dimensional time series $x_0, x_1, \ldots, x_{n-1}$, can be unfolded in a multidimensional (or phase) space in the form $x_n(m, \tau) = (x_n, x_{n+\tau}, \ldots, x_{n+(m-1)\tau})$, where m is the embedded dimension and τ represents the time delay. This theorem has been successfully employed to estimate dynamical system rules, thus, simplifying the behavioral studies and predictions [Alligood et al. 1996]

To illustrate the concepts of embedded dimension and time delay, we consider the output of the Logistic map (Equation 1) unfolded in a 2-dimensional space (*i.e.*, $m = 2$) and with $\tau = 1$, which results in pairs of points (x_t, x_{t+1}) (see Figure 2(b)). After such unfolding, we can observe the rule which generated the outputs and, consequently, study, understand and model other real-world problems by using Takens' immersion theorem. By making a regression of the resulting point pairs, we obtain the dynamical system rule

and can determine future states.



(a) Logistic map outputs.



(b) Logistic map reconstructed with embedded dimension 2 and time delay 1.

Figure 2. Study on Logistic map.

The embedded dimension defines the number of axis of the phase space, which are required to unfold the inherent behavior of systems. In this example, the series demands two axis, but other cases may require more (*e.g.*, the Lorenz map requires 3 axis [Alligood et al. 1996]). Several methods have been proposed to determine the embedded dimension as well as the time delay. The most accepted ones are the False Nearest Neighbors [Kennel et al. 1992] for the embedded dimension and the Auto-Mutual Information [Fraser and Swinney 1986] for the time delay.

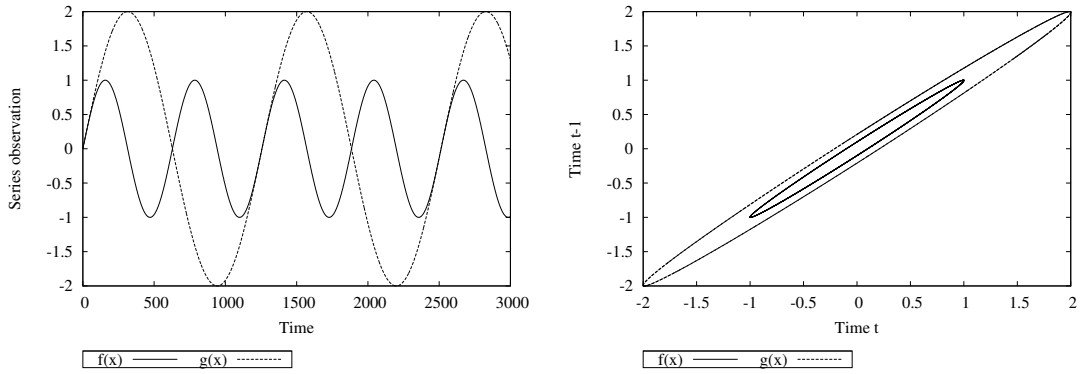
In this paper we employ such methods to estimate embedded dimensions and time delays in order to improve the reconstruction of system outputs in multidimensional spaces. By estimating dynamical system rules, we are able to understand and make predictions for a given system. Systems of interest here are processes and their outputs are composed of occupation variables which change over time. As we are particularly interested in how CPU load varies over time, we monitor that variable for every process and, then, reconstruct each process in a different multidimensional space, obtaining the rule for every process which determines its CPU occupation states. The states of different processes may influence one each other, reducing and/or improving performance. This paper focuses essentially on how to modify such influences in order to make processes run faster.

4. The Proposed Approach

This paper was motivated by influences and interactions of multiple dynamical systems [González-Miranda 2004]. For example, a system A may modify the behavior of another system B as they compete for resources. Understanding such influences, we claim that it is possible to modify the points of interaction among them and, thus, improve performance.

In order to illustrate the main ideas that motivated this study, consider two dynamical systems represented by the following rules: $f(x) = \sin(x)$ and $g(x) = \sin(x/2.0 + y) \cdot 2.0$. The outputs of such systems are shown in Figure 3(a). Also consider that the person who obtained those outputs does not know the rules, thus he/she needs to obtain them by using the procedure described in Section 3. One will reconstruct the outputs by

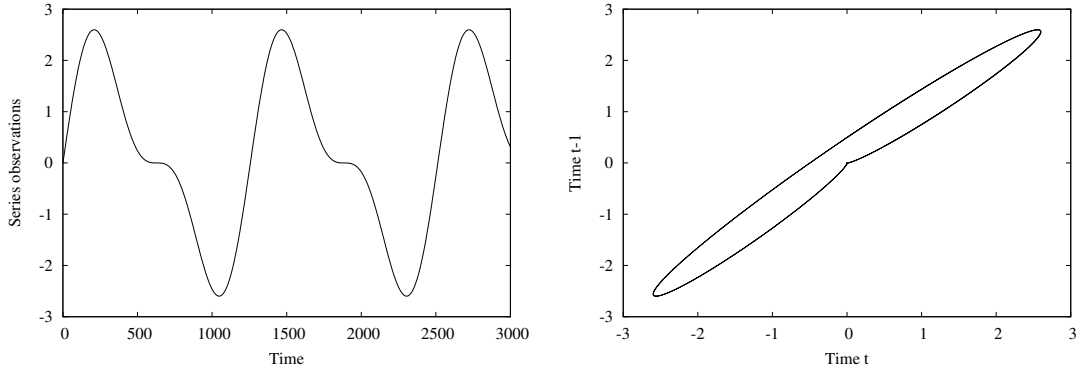
computing the False Nearest Neighbors and the Auto-Mutual Information methods and apply the resultant embedded dimension and time delay to Takens' immersion theorem [Takens 1980], obtaining the reconstructions presented in Figure 3(b).



(a) Functions $f(x)$ and $g(x)$. (b) Attractors of $f(x)$ and $g(x)$ functions.

Figure 3. Functions $f(x)$ and $g(x)$ and their attractors.

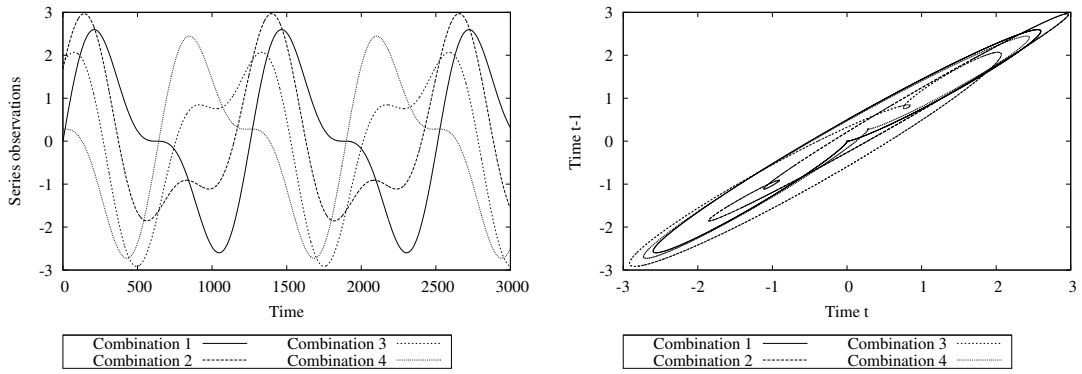
Now consider that both systems are combined through a simple sum of observations. This combination results in a third system which represents the same computer receiving both processes at the same time; the scheduler will then attempt to run both together. This third system is characterized by function $h(x) = \sin(x)+\sin(x/y+2.0)+2.0$, whose outputs are illustrated in Figure 4(a). The reconstructed behavior of this third system (Figure 4(b)) has a region with a behavior different from that previously observed when evaluating single processes. This reconstruction confirms that the combination of systems forces their interactions in terms of resource utilization, modifying the reconstructed behavior, which will also be called dynamical system attractor or simply attractor from now on.



(a) Function $h(x)$. (b) Attractor of $h(x)$ function.

Figure 4. Function $h(x)$ and its attractor.

Additional combinations of functions $f(x)$ and $g(x)$ are illustrated in Figure 5(a). We observe that the attractors (Figure 5(b)) show different behavior for different combinations, motivating the search for situations in which, by combining distinct functions in different ways (*i.e.*, in terms of time displacements), there is an equilibrium between these functions, generating stabler attractors and, therefore, a more uniform behavior.



(a) Offsets of $f(x)$ and $g(x)$ (b) Attractors of the offsets.

Figure 5. Offsets of $f(x)$ and $g(x)$ and their attractors.

To apply this study in the context of process scheduling, we have proposed an analytical model to evaluate combinations (in terms of time displacements) considering the behavior of two processes characterized by the same function: $p(x) = q(x) = 50 \cdot \sin(x) + 50$. Through offsets in x-axis (which represents time), different attractors were obtained, allowing assessing and obtaining stabler situations. Figure 6 shows the response time according to such a model.

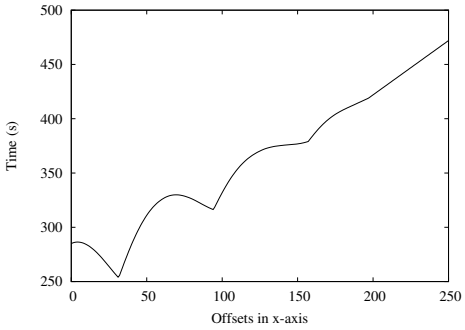
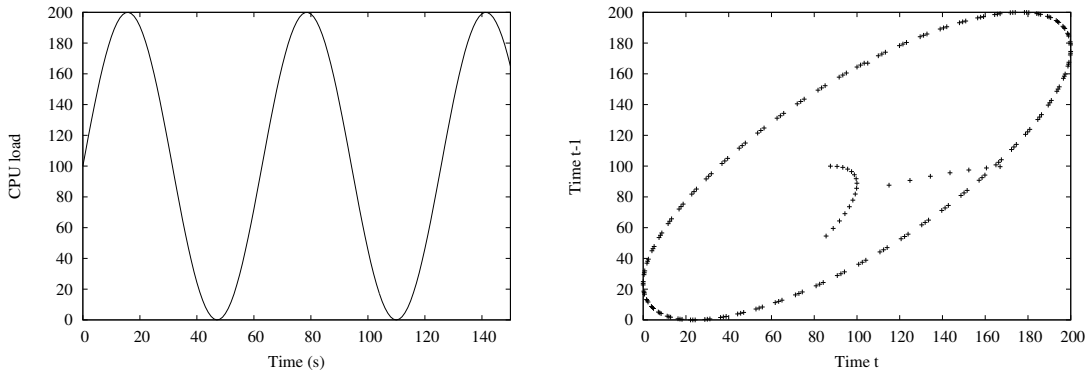


Figure 6. Response times.

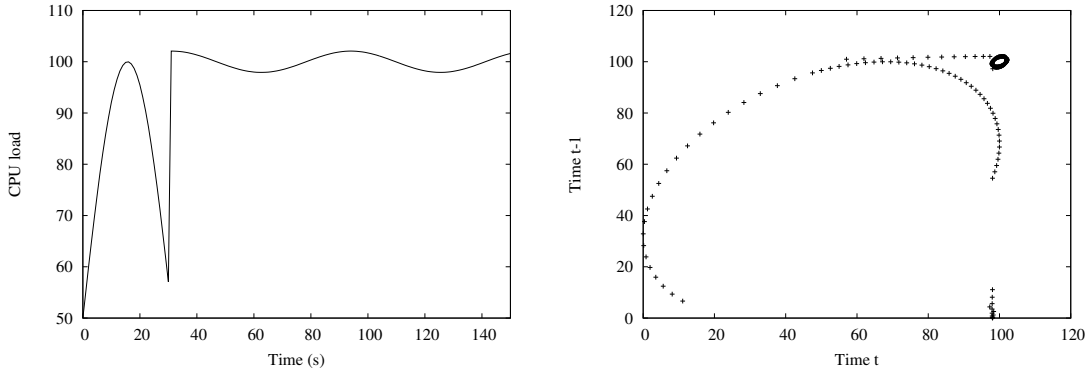
We can observe that, for certain offsets in x, the response time is significantly lower, providing evidence that some combinations of $p(x)$ and $q(x)$ result in a more appropriate scheduling in terms of makespan (makespan represents the total execution time of an application). Figures 7, 8 and 9 illustrate the behavior of combinations of $p(x)$ and $q(x)$ varying offsets in x. The respective attractors are also shown.

By considering this example, we observe that a stabler attractor has been obtained by using offset 31 on x-axis, *i.e.*, the combined behavior of processes tends to a region of higher stability. Such stability is observed, in terms of dynamical systems [Alligood et al. 1996], because there is a slight variation in the resulting attractor formed by the combination of both systems, as seen in Figure 8(b). In different words, the states of the resulting system are more concentrated in a specific region. On the other hand, when such combination results in unstabler systems, we observe novel orbits being formed such as in Figure 9(b) (two well-defined concentric orbits). The presence of more orbits indicates the system has higher probability to vary, making prediction more complex. This



(a) Offset 0. (b) Attractor to offset 0.

Figure 7. Combination of $p(x)$ and $q(x)$ with offset 0.



(a) Offset 31. (b) Attractor to offset 31.

Figure 8. Combination of $p(x)$ and $q(x)$ with offset 31.

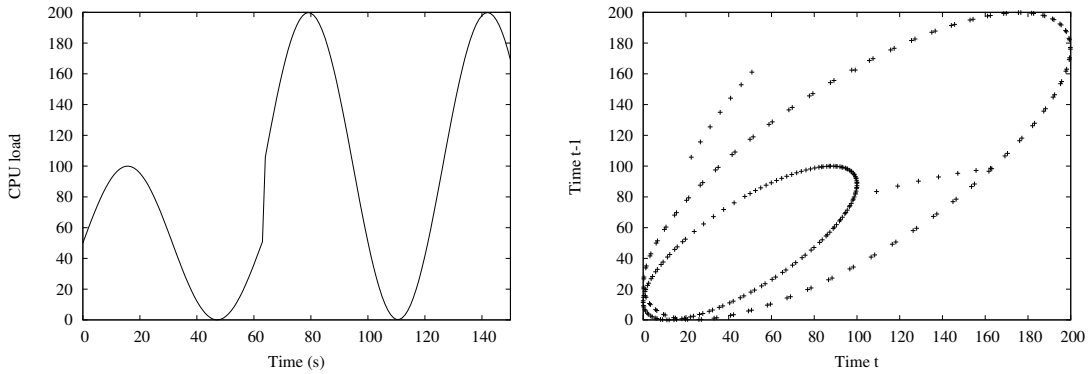
offset corresponds to the point in Figure 6 whose response time is minimal.

Based on these initial observations, we designed and implemented two heuristics in a simulator. By considering series collected during the execution of real applications, they schedule processes according to equilibrium situations.

4.1. Proposed Heuristics and Results

The first heuristic proposed, Heuristic A, considers the temporal behavior of the processes and performs the scheduling based on their predicted CPU usage. This heuristic assumes the prediction of the processes behavior in a time slice (after the current moment). For example, considering a series $S = \{x_0, x_1, \ldots, x_{n-1}\}$ with n observations, we perform the prediction of the next observation x_n based on the approach proposed in [Mello and Yang 2009]. Based on such a prediction, Heuristic A schedules the process with higher CPU utilization at the next observation x_n. Thus, it maximizes the processor utilization and reduces the response time of processes.

By prioritizing processes with higher CPU usage, Heuristic A tends to delay the execution of others with lower demands. To avoid this limitation, we have proposed a variation of the first heuristic which introduces a probabilistic term to select the next process to run. This term is proportional to the next observation x_n for every process and



(a) Offset 64. (b) Attractor to offset 64.

Figure 9. Combination of $p(x)$ and $q(x)$ with offset 64.

is computed as follows. Let P_i be the probability of a process i to be selected, then

$$P_i = \frac{x_n^i}{\sum_{j=1}^{N} x_n^j},$$

where x_n^i is the next observation of process i, that is, the predicted CPU usage. For instance, given a set of three processes running, assuming that the workload of every process in the next time slice is $\{x_n^1 = 13.96, x_n^2 = 6.53, x_n^3 = 2.27\}$, the probability of selecting each one to execute in the next time slice is given by $\{P_1 = 0.61, P_2 = 0.28, P_3 = 0.11\}$. Thus, even processes that demand less resources may also be selected. This new heuristic has been called Heuristic B.

In order to validate both heuristics, we considered five scenarios representing different environment workloads. Such scenarios were designed based on execution traces[3] of the tool `grep` and also on a C-language version of the `Fibonacci` algorithm.The tool `grep` was executed in two different ways:

```
grep -r "?" /,
```

which prints the result in standard output, alternating computation with I/O operations, and

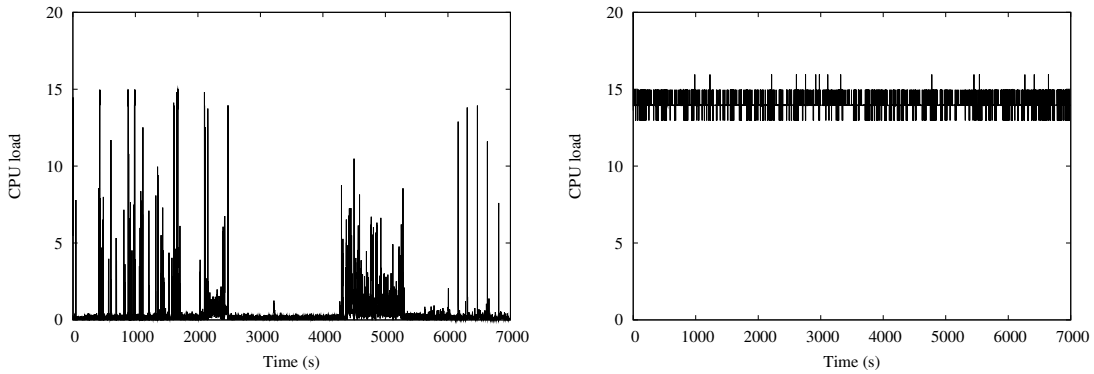
```
grep -r "?" / > /dev/null,
```

which avoids output operations, increasing CPU usage. When using `Fibonacci`, no output was generated.

Figures 10(a) and 10(b) show the CPU load for `grep` and `Fibonacci`, respectively. For `grep`, we observe there is an alternated pattern of CPU utilization and output operations, while `Fibonacci` demands a higher and constant CPU usage.

Based on the obtained traces (shown in Figure 10), the heuristics were simulated and compared against the *Round-Robin* (RR) policy. Only for clarification purposes, RR assigns identical CPU times for each process, which are organized in a circular queue.

[3]We developed a script to run on GNU/Linux to obtain CPU usage at every $150\ ms$, that reads file `/proc/id/stat` – `id` corresponds to the process identifier.



(a) CPU usage for tool `grep`. (b) CPU usage for `Fibonacci` algorithm.

Figure 10. CPU usage.

Table 1 presents the results for five `grep` processes simultaneously started in the computational environment, thus competing for resources. In this case, we assume the execution of `grep` generating outputs. We here compare the total execution time (makespan), the process average running time, the time necessary to switch contexts (CS), that is, the total time the scheduling policy requires to select the next process to run, and, finally, the average CPU utilization.

Table 1. Five identical executions of `grep`.

Policy	Total Time	AVG Time	CS Time	AVG Utilization
RR	33,440.0	6,688.0	13.34	1.90
Heuristic A	34,680.0	6,936.0	1,82	1.83
Performance Gain				-3.57%
Heuristic B	33,769.0	6,753.0	14.00	1.88
Performance Gain				-0.96%

In this case, RR presented better results, since all processes impose the same workload and, thus, a simply execution interleaving tends to better results, in opposition to strategies proposed by both heuristics evaluated. We can also observe that Heuristic B reduced response time and improved resource usage when compared to Heuristic A.

In a second scenario, we considered three identical executions of `Fibonacci` algorithm. In such a circumstance, we obtained a similar behavior for all policies (Table 2), due to the high CPU utilization imposed by every `Fibonacci` process.

Table 2. Three identical executions of `Fibonacci` algorithm.

Policy	Total Time	AVG Time	CS Time	AVG Utilization
RR	21,347.0	7,115.0	4.27	46.73
Heuristic A	21,347.0	7,115.0	4.27	46.73
Performance Gain				0.00%
Heuristic B	21,347.0	7,115.0	4.27	46.73
Performance Gain				0.00%

These two scenarios are composed of processes with identical behaviors; in such a situation our heuristics do not improve performance. However, when a different set of processes is executed, the heuristics indeed affect the final performance. For example,

Table 3 presents four `grep` processes, writing in the standard output, and one execution of `grep` redirecting the output to the null device (as previously shown). We then observe that Heuristic A reduced the response time to approximately 52% when compared to RR and also improved the resource utilization. The performance of Heuristic B is slightly below A, but still better than RR.

Table 3. Five executions of `grep` (four identical and one different).

Policy	Total Time	AVG Time	CS Time	AVG Utilization
RR	18,324.0	3,664.0	6.98	3.07
Heuristic A	8,814.0	1,762.0	3.07	3.75
Performance Gain				+51.91%
Heuristic B	9,493.0	1,898.0	3.19	3.64
Performance Gain				+48.19%

In another scenario, we simulated higher demands for CPU by running three `grep` processes, writing in the standard output, one `grep` execution without outputs and, finally, one `Fibonacci` process. In this case, Heuristics A and B outperformed RR in terms of response time (reducing it in about 31% and 28%, respectively) and CPU utilization (Table 4).

Table 4. Four executions of `grep` (three identical and one different) and one execution of `Fibonacci` algorithm.

Policy	Total Time	AVG Time	CS Time	AVG Utilization
RR	22,382.0	4,476.0	6.98	16.81
Heuristic A	15,448.0	3,089.0	4.15	23.61
Performance Gain				+30.98%
Heuristic B	16,005.0	3,201.0	4.27	22.83
Performance Gain				+28.49%

Finally, in the last scenario (Table 5), we increased CPU demands by adding another `Fibonacci` process in the previous scenario. In this case, performance was also improved by Heuristics A and B, however, such improvement was lower than the previous scenario because, by adding another `Fibonacci` process, the total CPU workload was too high to be treated by a single computer. If we add more processes, there will even be reductions in the resulting improvement, tending, in overloaded situations, to the same performance of RR.

Table 5. Four executions of `grep` (three identical and one different) and two execution of `Fibonacci` algorithm.

Policy	Total Time	AVG Time	CS Time	AVG Utilization
RR	29,364.0	4,894.0	8.38	23.92
Heuristic A	22,564.0	3,760.0	5.58	30.90
Performance Gain				+23.17%
Heuristic B	23,064.0	3,844.0	5.65	30.26
Performance Gain				+21.45%

We have concluded that the proposed heuristics improve both performance and resource usage when heterogeneous processes are submitted to the environment. Another important point is that when processes barely use CPU, our heuristics do not provide improvements (Tables 1 and 3). A second situation occurs when processes impose too

heavy workloads; in such scenario, heuristics tend to present results closer to RR. However, when processes present a different behavior and impose neither too heavy nor too low CPU usage, both heuristics strongly improve performance results as well as resource utilization. This last scenario is typical in desktop computers. Furthermore, we can also employ such heuristics in grid computing environments by avoiding the overload of single computers and making such heuristics responsible for local scheduling.

5. Conclusions

This paper has presented studies on behavioral interactions of processes, motivating the proposal of two heuristics which allocate resources based on how processes demand CPU in future time slices. Both heuristics were implemented in a simulator and compared against the Round-Robin policy.

Results have confirmed that, when processes demand the same or similar CPU usage, both heuristics are slightly inferior to Round-Robin. Still, when processes are similar, but under higher CPU utilization, the heuristics as well as Round-Robin present similar performance. Finally, when processes have different behaviors, both heuristics outperform Round-Robin in terms of reducing the total execution time and improving CPU utilization.

Furthermore, we remember that many computational environments are mostly characterized by processes with different behavior, as desktop systems, confirming the potential usage of such heuristics in real-world scenarios. Even large-scale environments, such as grids, could take advantage of the proposed heuristics by using them as local scheduling policies.

Acknowledgments

This paper is based upon work supported by FAPESP (São Paulo Research Foundation), Brazil, under grant no. 2009/15338-1. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of FAPESP.

References

Alligood, K. T., Sauer, T. D., and Yorke, J. A. (1996). *Chaos: An Introduction to Dynamical Systems*. Springer-Verlag.

Chunlin, L. and Layuan, L. (2009). A system-centric scheduling policy for optimizing objectives of application and resource in grid computing. *Computers and Industrial Engineering*, 57(3):1052–1061.

Devarakonda, M. V. and Iyer, R. K. (1989). Predictability of process resource usage: A measurement-based study on UNIX. *IEEE Transactions on Software Engineering*, 15(12):1579–1586.

Dinda, P. A. (2001). Online prediction of the running time of tasks. In *IEEE International Symposium on High Performance Distributed Computing*, pages 383–382.

Dodonov, E. and Mello, R. F. (2010). A novel approach for distributed application scheduling based on prediction of communication events. *Future Generation Computer Systems*, 26(5):740–752.

Downey, A. B. (1997). Predicting queue times on space-sharing parallel computers. In *International Symposium on Parallel Processing*, pages 209–218.

Feitelson, D. G. and Nitzberg, B. (1995). Job characteristics of a production parallel scientific workload on the NASA ames iPSC/860. In *Job Scheduling Strategies for Parallel Processing*, volume 949 of *LNCS*, pages 337–360. Springer-Verlag.

Feitelson, D. G., Rudolph, L., Schwiegelshohn, U., Sevcik, K. C., and Wong, P. (1997). Theory and practice in parallel job scheduling. In *Job Scheduling Strategies for Parallel Processing*, volume 1291 of *LNCS*, pages 1–34. Springer-Verlag.

Ferrari, D. and Zhou, S. (1988). An empirical investigation of load indices for load balancing applications. In *International Symposium on Computer Performance Modelling, Measurement and Evaluation*, pages 515–528.

Fraser, A. M. and Swinney, H. L. (1986). Independent coordinates for strange attractors from mutual information. *Phys. Rev. A*, 33(2):1134–1140.

Gibbons, R. (1997). A historical application profiler for use by parallel schedulers. In *Job Scheduling Strategies for Parallel Processing*, volume 1291 of *LNCS*, pages 58–77. Springer-Verlag.

González-Miranda, J. M. (2004). *Syncronization and Control of Chaos: An Introduction for Scientists and Engineers*. Imperial College Press.

Kennedy, J., Eberhart, R. C., and Shi, Y. (2001). *Swarm Intelligence*. Morgan Kaufmann Publishers.

Kennel, M. B., Brown, R., and Abarbanel, H. D. I. (1992). Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Phys. Rev. A*, 45(6):3403–3411.

Krishnaswamy, S., Loke, S. W., and Zaslavsky, A. (2004). Estimating computation times of data-intensive applications. *IEEE Distributed Systems Online*, 5(4):1–12.

Lee, B.-D. and Schopf, J. M. (2003). Run-time prediction of parallel applications on shared environments. In *IEEE International Conference on Cluster Computing*, pages 487–491.

Mello, R. F. and Yang, L. (2009). Prediction of dynamical, nonlinear, and unstable process behavior. *The Journal of Supercomputing*, 49(1):22–41.

Schopf, J. M. and Berman, F. (1999). Stochastic scheduling. In *ACM/IEEE Conference on Supercomputing*.

Sevcik, K. C. (1989). Characterizations of parallelism in applications and their use in scheduling. *Performance Evaluation Review*, 17(1):171–180.

Smith, W., Foster, I., and Taylor, V. (2004). Predicting application run times with historical information. *Journal of Parallel and Distributed Computing*, 64(9):1007–1016.

Takens, F. (1980). Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence*, pages 366–381. Springer.

Whitney, H. (1936). Differentiable manifolds. *The Annals of Mathematics*, 37(3):645–680.

Estudo da Fila FB em Sistemas Multiprocessados

Alexandre Magno de Sousa, Cristina Duarte Murta

[1]Departamento de Computação, CEFET/MG
Belo Horizonte - MG

Resumo. *A política de escalonamento* FB *(Foreground-Background) se mostra bastante apropriada para sistemas que experimentam, ao mesmo tempo, cargas com grande variabilidade no tempo de serviço e intensidade de tráfego elevada. Estas características colocam o sistema em um estado crítico, que pode ser tratado pela fila* FB*, sem exigir conhecimento do tamanho da tarefa. Este artigo apresenta um estudo da fila* FB *em ambientes distribuídos, para sistemas fortemente e fracamente acoplados. Os resultados são avaliados quanto ao tempo de execução das tarefas e quanto à justiça do escalonamento. Este artigo contribui para o conhecimento mais amplo das características da fila* FB *em ambientes paralelos e distribuídos.*

Abstract. *The Foreground-Background* (FB) *scheduling policy seems to be quite appropriate for systems that experiences at the same time load with high variability in service times and high traffic intensity. These characteristics can take the system into a critical condition, which can be handled by the* FB *queue, without requiring knowledge of the task size. This paper presents a study of the* FB *queue in strongly and weakly coupled distributed systems. The results are evaluated according to the response time and the scheduling fairness. This paper contributes to improve the knowledge on the characteristics of the* FB *scheduling policy in parallel and distributed environments.*

1. Introdução

O uso crescentemente disseminado dos sistemas computacionais, aliado à evolução de sua arquitetura e à complexidade da carga, torna esses ambientes cada vez mais complexos. Neste contexto, situações críticas, representadas por intensidades de tráfego elevadas e cargas altamente variáveis, são encontradas. Nesses ambientes é desejável que as políticas de escalonamento sejam capazes de tratar estas situações. Este trabalho apresenta um estudo da política de escalonamento FB (*Foreground-Background*), que se mostra bastante apropriada para sistemas que experimentam, ao mesmo tempo, cargas com grande variabilidade no tempo de serviço e intensidade de tráfego elevada.

A política de escalonamento FB trata cada tarefa em função de seu tamanho, sem precisar conhecê-lo. A política FB tem sido estudada recentemente de maneira mais intensa [Nuyens 2004a, Nuyens and Wierman 2008, Wierman 2007b]. No entanto, estes trabalhos abordam o desempenho da política majoritariamente em sistemas com um processador. Em um artigo anterior, apresentamos um estudo da política de escalonamento FB em sistemas mono e multiprocessados [Sousa and Murta 2010]. O presente artigo amplia os conhecimentos alcançados no trabalho anterior, apresentando um estudo de simulação acerca do comportamento da política FB em sistemas multiprocessados fortemente e fracamente acoplados.

Neste trabalho, a política FB é comparada com as políticas tradicionais de escalonamento *Processor-Sharing* (PS) e *First-Come-First-Served* (FCFS) e com uma política ótima de escalonamento, a SRPT (*Shortest-Remaining-Processing-Time*). Estas políticas são simuladas em ambientes multiprocessados, submetidos a cargas cujos tempos de serviço apresentam baixa e alta variabilidade, e intensidades de tráfego variáveis. A principal métrica analisada é o tempo médio de resposta. A análise da justiça no tratamento das tarefas quando o sistema se aproxima de uma situação de instabilidade é também apresentada.

A comparação de desempenho da política FB nos dois modelos de acoplamento de sistemas, e frente a outras políticas, comparadas nas mesmas condições, revela as situações em que ela pode ser aplicada com vantagens frente às demais opções. Devemos observar que, geralmente, o projeto de sistemas de acoplamento forte, também vistos como sistemas paralelos, pode apresentar custos mais altos do que o de sistemas de acoplamento fraco, isto é, sistemas distribuídos.

Este artigo está organizado em cinco seções. A próxima seção trata da fundamentação teórica e da discussão dos trabalhos relacionados. A seção 3 descreve o planejamento dos experimentos. Na seção 4 são apresentados os resultados dos experimentos realizados neste artigo. Por fim, a seção 5 apresenta a conclusão deste trabalho e algumas direções para trabalhos futuros.

2. Fundamentação Teórica e Trabalhos Relacionados

Sistemas multiprocessados podem ser organizados de duas maneiras quanto ao acoplamento: fortemente acoplados e fracamente acoplados [Casavant and Kuhl 1988, Anger et al. 1990, Tanenbaum and Steen 2007]. Nos sistemas fortemente acoplados há uma única fila para todos os servidores, enquanto nos sistemas fracamente acoplados há uma fila para cada servidor. Estes modelos são apresentados nas Figuras 1 e 2.

O sistema fortemente acoplado pode ser visto como um sistema paralelo, em geral modelado pela fila M/G/m. Neste modelo, a utilização do sistema é dada por $\rho = \lambda/(m\mu)$, em que m representa o número de servidores.

Nos sistemas fracamente acoplados há duas decisões de escalonamento: a estratégia de despacho ou roteamento e a política de escalonamento aplicada à fila de cada servidor, ao qual tarefas são assinaladas. A decisão de roteamento é feita no momento da chegada da tarefa. O sistema fracamente acoplado pode ser visto como um sistema distribuído e pode ser modelado por um conjunto de filas M/G/1. Esse sistema consiste de m filas paralelas $Q_1, ..., Q_m$, uma para cada servidor, conforme apresentado na Figura 2. O sistema é homogêneo, isto é, os servidores têm a mesma capacidade de processamento. Neste sistema, a taxa de chegada é definida como $\lambda = \sum_{i=1}^{m} \lambda_i$, e λ_i representa a taxa de chegada da i-ésima fila. A utilização de cada recurso do sistema é dada por $\rho_i = \lambda_i/\mu$ e a utilização do sistema é dada por $P = \sum_{i=1}^{m} \rho_i$.

Nesse trabalho consideramos, para a decisão de roteamento nos sistemas fracamente acoplados, as políticas de despacho RR (*Round-Robin*) e JSQ (*Join the Shortest Queue*). No roteamento RR, a i-ésima tarefa é assinalada ao (i mod m)-ésimo servidor, enquanto no JSQ as tarefas são assinaladas ao servidor com a menor fila de espera.

Numerosas estratégias de roteamento podem ser encontradas na literatura. Além

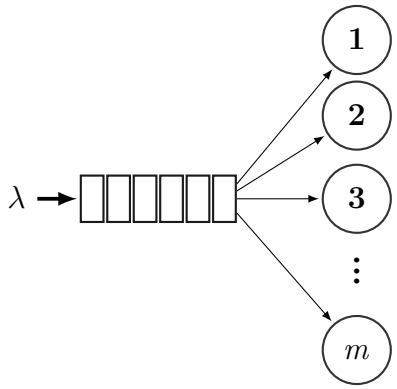


Figura 1. Modelo multiprocessado fortemente acoplado.

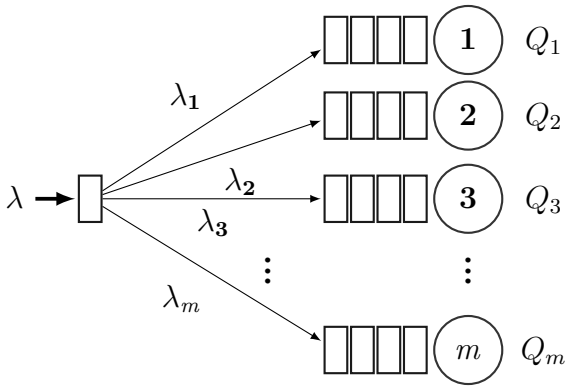


Figura 2. Modelo multiprocessado fracamente acoplado.

das citadas acima, outra estratégia comumente citada é a *Random*, que assinala tarefas aos servidores com uma probabilidade $1/m$, sendo m o número de servidores. *Random* e RR não necessitam de informação dos servidores para fazer a decisão de roteamento. Por sua vez, JSQ requer informação do tamanho da fila de espera de cada servidor.

Há estratégias mais elaboradas, por exemplo, *Size-Based Task Assignment* (SITA-E), em que as tarefas são enviadas aos servidores de acordo com suas faixas de tamanhos [Harchol-Balter et al. 1999, Down and Wu 2006]. A escolha da estratégia de roteamento depende de vários fatores. Se o *overhead* de comunicação é uma medida a ser considerada, sugere-se o uso de *Random* ou RR. No entanto, políticas que utilizam mais informação, tal como JSQ, produzem melhores resultados quanto ao desempenho.

Políticas de serviço ou políticas de escalonamento definem a ordem em que as tarefas são atendidas. Essa ordem pode produzir impacto no desempenho de um sistema que implementa filas de espera para suas solicitações. Exemplos de políticas de escalonamento são *First-Come-First-Served* (FCFS), *Processor-Sharing* (PS), *Shortest-Remaining-Processing-Time* (SRPT), *Shortest-Job-First* (SJF) e *Foreground-Background* (FB).

Políticas de escalonamento que dão prioridade a tarefas menores reduzem o tempo médio de resposta [Conway et al. 2003, Wierman 2007b]. Uma das políticas que prioriza tarefas menores, sendo considerada teoricamente ótima, é a política SRPT [Schrage 1968, Bansal and Harchol-Balter 2001]. No entanto essa política depende do conhecimento a priori do tamanho das tarefas, o que pode tornar difícil ou mesmo impossibilitar sua utilização em ambientes nos quais não é possível conhecer antecipadamente o tamanho das tarefas. As dificuldades observadas na implementação da política SRPT em sistemas reais são discutidas em [Bansal and Harchol-Balter 2001].

Na implementação de políticas de escalonamento preemptivas, tais como SRPT e PS, os custos de preempção devem ser observados. Tais custos são oriundos do sistema operacional, uma vez que a cada preempção ou interrupção de uma tarefa em execução ocorre uma troca de contexto [Tanenbaum and Woodhull 2006]. Os custos de preempção da política SRPT são menores do que os da política PS [Bansal and Harchol-Balter 2001]. Na execução da política SRPT, uma tarefa só sofre preempção quando outra tarefa chega ou quando finaliza. Como cada tarefa chega

e termina exatamente uma vez, então, em média, as tarefas são interrompidas duas vezes. Estudos acerca da influência do tamanho do quantum no desempenho de uma fila M/G/1/RR são apresentados em [Gupta 2008]. Nesses estudos, os valores do quantum são relacionados à distribuição de serviço por meio de uma função matemática.

A política SRPT é ótima, isto é, produz os menores tempos médios de resposta dentre todas as políticas [Schrage and Miller 1966, Schrage 1968]. Todavia, em situações reais, o tamanho das tarefas pode não ser conhecido antecipadamente. Assim, é interessante encontrar alternativas que sejam um indicativo para estimar o tamanho da tarefa. Uma indicação importante do tamanho da tarefa é o número de vezes que ela foi escalonada pelo processador para ser servida. Esse número é conhecido na literatura por *age*. A política FB utiliza essa medida e prioriza tarefas de acordo com o menor valor de *age* [Nuyens 2004b]. Trabalhos recentes têm sido motivados por propostas de projetos de sistemas computacionais que façam o uso da política FB, tais como o escalonamento do fluxo de roteadores e o escalonamento de processos em sistemas operacionais [Nuyens and Wierman 2008, Wierman 2007b].

Resultados mostraram uma relação entre a variabilidade da distribuição de serviço e o desempenho da política FB [Nuyens and Wierman 2008]. Uma forma de expressar a relação de desempenho dessa política frente à variabilidade nas características da carga é por meio do coeficiente de variação [Feng and Misra 2004], que é dado por $C[X] = \sigma/E[X]$, em que σ representa o desvio padrão e $E[X]$ o valor esperado da distribuição de serviço X. De acordo com [Feng and Misra 2004] a fila FB apresenta desempenho ruim para cargas com $C[X] < 1$; comportamento semelhante à política PS para cargas com $C[X] = 1$; e apresenta bom desempenho para cargas com $C[X] > 1$. Quanto maior o $C[X]$, melhor o desempenho dessa política [Nuyens 2004b, Wierman 2007b].

A política FB é classificada como parcialmente justa, pois age com justiça para determinadas cargas de trabalho, e age com injustiça para outras cargas de trabalho [Wierman 2007a]. De acordo com Nuyens e Wierman, apesar da política FB priorizar teoricamente tarefas menores, apenas uma pequena quantidade de tarefas grandes é discriminada [Nuyens and Wierman 2008]. Resultados sobre o comportamento dessa política são encontrados na literatura apenas para sistemas monoprocessados, pois a avaliação analítica dessa fila é complexa e um modelo analítico para sistema multiprocessados ainda não foi desenvolvido [Nuyens 2004b, Wierman 2007b, Nuyens and Wierman 2008].

3. Planejamento dos Experimentos

O objetivo do trabalho é avaliar o desempenho da política FB em ambientes multiprocessados, organizados com acoplamento fraco ou forte, conforme descrito anteriormente. Para avaliarmos a fila FB de maneira comparativa, também foram feitos experimentos com as políticas de escalonamento FCFS e PS, que são comumente utilizadas em sistemas computacionais, e com a política teoricamente ótima, SRPT. As políticas são comparadas quanto ao tempo médio de resposta para a execução das tarefas.

A carga de trabalho foi gerada sinteticamente. Os resultados de desempenho são avaliados em função da variabilidade da carga, dada por $C[X]$. Para os testes foram escolhidas duas cargas de trabalho pouco variáveis (cauda leve), que são exponencial (M) e hiperexponencial (H); e uma carga com variabilidade elevada (cauda pesada), representada pela distribuição de Pareto (P), com parâmetro de variabilidade (*shape*) $\alpha = 1,5$.

As tarefas chegam ao sistema de acordo com processo de Poisson, a taxa de chegada é definida como λ e a taxa de serviço é definida como μ. A utilização do sistema ou a intensidade de tráfego é dada por $\rho = \lambda/(m\mu)$.

Os resultados foram obtidos por simulação baseada em eventos discretos. O simulador utilizado para realização dos experimentos foi implementado em linguagem de programação C. Os resultados foram verificados e validados em um intervalo de confiança de 95%. O número de observações da amostra foi baseada em replicações e é dada por $n = k \times m$, em que m representa o número de execuções, também chamado de tamanho da execução, e k o número de replicações. Para realização dos experimentos, o número de execuções foi definido como $k = 10$ e o número de replicações como $m = 1 \times 10^6$, então $n = 10 \times 10^6$. Conforme MacDougall, o número replicações deve ser relativamente pequeno, entre 1 e 10, e o tamanho das execuções deve ser relativamente grande [MacDougall 1987].

Em simulações que utilizam distribuição de serviço de cauda pesada, o número de amostras para que o sistema alcance o estado estável pode ser muito grande. Isso acontece porque a distribuição apresenta grande variabilidade e a convergência é lenta. Para a distribuição de Pareto, o número de amostras necessárias para alcançar uma precisão de dois dígitos quando o parâmetro de variabilidade é 1,5 é de um milhão [Crovella and Lipsky 1997]. Assim, isso justifica o grande número de execuções e foi definido um valor padrão de $m = 1 \times 10^6$ para todos os tipos de carga nos experimentos. Além disso, um número de execuções suficientemente grande também soluciona o problema de estado transiente (*warmup*) [MacDougall 1987, Jain 1991].

O simulador foi verificado realizando-se uma comparação entre os resultados do simulador e os resultados dos modelos analíticos disponíveis na literatura. Ressalta-se que a grande maioria dos modelos analíticos disponíveis são para sistemas com apenas um servidor. Para sistemas múltiplos servidores, não existem modelos analíticos para as políticas SRPT e FB, portanto, apenas o modelo da fila M/M/m/FCFS foi utilizado [Wierman 2007b, Nuyens and Wierman 2008].

As políticas de escalonamento no simulador funcionam conforme descrito a seguir. Para FCFS, as tarefas são alocadas aos servidores de acordo com a ordem de chegada, se não houver servidor livre, a tarefa é encaminhada à fila. As tarefas são atendidas concorrentemente de acordo com sua ordem de chegada para PS, cada tarefa é executada por um período de tempo (quantum), ao final desse período, se a tarefa não terminou sua execução, ela é encaminhada para fila. Para SRPT, quando uma tarefa chega ao sistema e há um servidor livre, este é alocado à mesma. Caso não haja servidor disponível, o tempo restante de execução das tarefas já alocadas aos servidores é avaliado, se houver tarefas com tempo restante de execução maior do que a nova tarefa, é feito um sorteio entre os servidores dessas tarefas. O servidor sorteado é alocado à nova tarefa e a tarefa que estava alocada à ele vai para fila. Por fim, para FB, existem contadores para verificar quantas vezes a tarefa foi servida. Quando uma tarefa solicita serviço, a prioridade é dada à tarefa com o menor valor desses contadores, a fila é ordenada por ordem crescente dos contadores.

No simulador, a política de despacho ou roteamento RR funciona da seguinte forma: a i-ésima tarefa é assinalada ao (i mod m)-ésimo servidor. A política de rotea-

mento JSQ executa como descrito a seguir: cada tarefa é assinalada ao servidor com o menor número de tarefas na fila. Neste modelo, os custos de comunicação para realizar o roteamento das tarefas não são levados em consideração.

O planejamento de experimentos foi definido como do tipo fatorial e considera os fatores e níveis apresentados na Tabela 1 para a avaliação dos sistemas multiprocessados distribuídos e paralelos.

Fatores	**Níveis**
Política	FB; FCFS; PS; SRPT
Tempo de Serviço	M; H; P
No. de servidores	8
Acoplamento	fraco; forte
Roteamento	RR; JSQ
Utilização do Sistema	0,2 ; 0,4 ; 0,6 ; 0,8 ; 0,9 ; 0,95

Tabela 1. Principais fatores e níveis planejados para os experimentos.

Além dos resultados para o tempo médio de resposta, foram realizados mais dois experimentos para sistemas paralelos. O primeiro experimento teve como meta avaliar a justiça das políticas de escalonamento em relação ao atendimento das tarefas, à medida que a utilização do sistema se aproxima de uma situação de instabilidade. Foram feitos dois experimentos, um para $\rho \approx 0,90$ e outro para $\rho \approx 0,99$. Considerando que as filas FB e SRPT discriminam as tarefas de acordo com seu tamanho, é necessário avaliar a questão da justiça.

O segundo experimento consistiu em analisar o comportamento das políticas FB e PS com relação aos diferentes valores de quantum. Uma vez que essas políticas têm como parâmetro o quantum, é necessário avaliar o efeito de sua variação. A Tabela 2 apresenta os principais fatores e níveis para os experimentos realizados com o quantum.

Fatores	**Níveis**
Política	FB; PS
Quantum	1; 2; 4; 6; 8; 16
Tempo de Serviço	M; H; P
Utilização do Sistema	0,2 ; 0,4 ; 0,6 ; 0,8 ; 0,9 ; 0,95

Tabela 2. Principais fatores e níveis para os testes realizados com o quantum.

4. Resultados

Nesta seção são apresentados os resultados da simulação para os sistemas em teste. Para a geração dos tempos de serviço da carga de trabalho foram escolhidas distribuições de probabilidade que apresentam diferentes coeficientes de variação ($C[X]$). Os valores de $C[X]$ são os seguintes: para a distribuição exponencial, representada por M, $C[X] = 1$; para a distribuição hiperexponencial, representada por H, $C[X] = 4$; e para a distribuição de Pareto com $\alpha = 1,5$, representada por P, $C[X]$ apresentou valores entre 4 e 210 durante a realização dos experimentos.

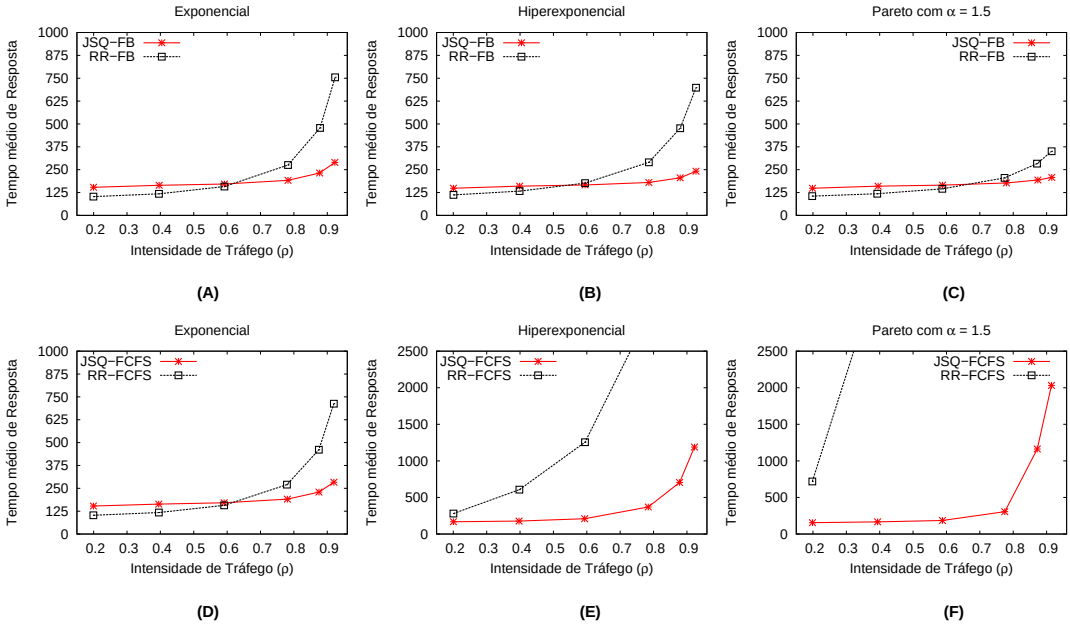


Figura 3. Resultados do sistema distribuído para as filas FB **e** FCFS **com as políticas de roteamento** RR **e** JSQ**.**

4.1. Sistema fracamente acoplado

Os gráficos de **(A)** a **(F)** da Figura 3 apresentam os resultados para as políticas FB e FCFS, frente às políticas de roteamento JSQ e RR, para as cargas M, H e P. Os gráficos **(A)**, **(B)** e **(C)** tratam da fila FB, e os demais apresentam os resultados para a fila FCFS.

Para o caso da fila FB, observamos, para as três cargas, que o roteamento RR produz tempos de resposta ligeiramente menores do que JSQ para cargas leves. No entanto, para cargas elevadas, a política JSQ produz os melhores tempos de resposta. Além disso, a diferença entre as duas estratégias de roteamento diminui à medida que a variabilidade da carga aumenta.

No caso da fila FCFS frente às políticas de roteamento JSQ e RR, observamos que quanto maior a variabilidade da carga, maior a diferença entre as duas estratégias de roteamento. Em resumo, a estratégia de roteamento JSQ, em conjunto com a política de escalonamento FB, produz os melhores resultados em condições de tráfego mediano a intenso e cargas muito variáveis.

Na Figura 4 são apresentados os resultados para as políticas PS e SRPT frente às políticas de roteamento em teste, para as cargas M, H e P. Os resultados indicam comportamentos qualitativamente similares para RR e JSQ, sendo que RR apresenta os menores tempos de resposta para intensidades de tráfego baixas a moderadas, enquanto JSQ produz os melhores resultados para tráfego mais intenso. Lembramos que a política SRPT é ótima, isto é, produz os menores tempos de resposta. A combinação de JSQ com SRPT torna o tempo de resposta praticamente insensível à intensidade de tráfego, para todas as cargas.

Na Figura 5 são apresentados os resultados dos tempos médios de resposta do sistema distribuído, agrupados por distribuição de tempo de serviço. Para cada distribuição

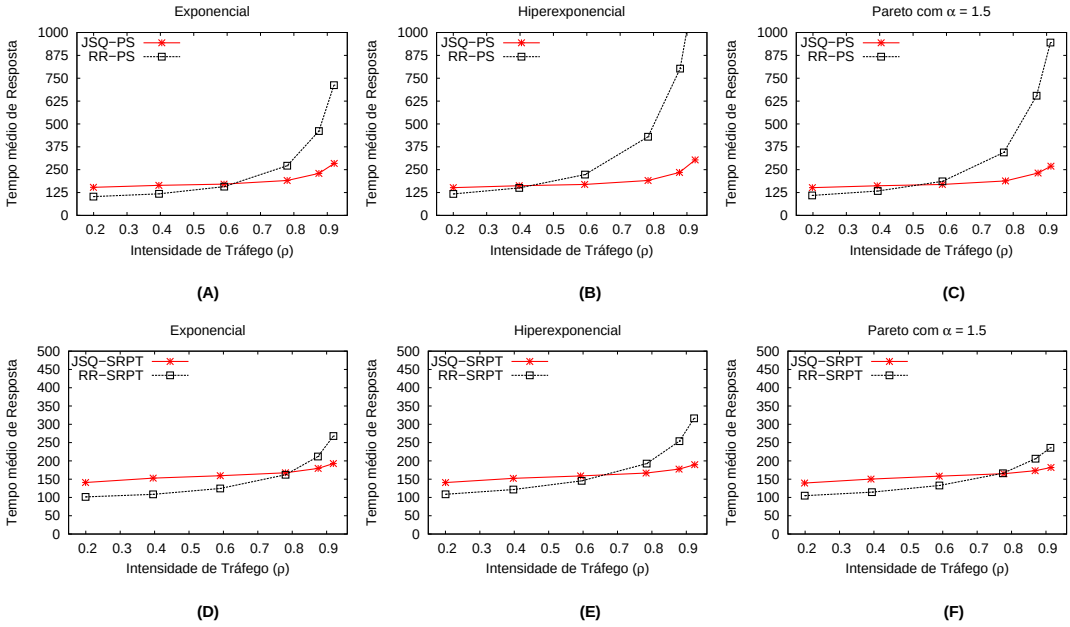


Figura 4. Resultados do sistema distribuído para as filas PS **e** SRPT **com as políticas de roteamento** RR **e** JSQ**.**

e cada estratégia de roteamento podemos comparar as quatro políticas de escalonamento em teste.

Para a distribuição exponencial (gráficos **(A)** e **(D)**) observa-se que as políticas de escalonamento, exceto SRPT, apresentam resultados iguais sob a mesma estratégia de roteamento. Sob RR, os resultados são melhores para pouco tráfego, e sob JSQ os resultados são melhores para tráfego intenso.

A política SRPT apresenta o melhor desempenho em todos os casos, conforme esperado, independente da estratégia de roteamento e da distribuição do tempo de serviço. Para os casos de distribuição hiperexponencial e Pareto, observa-se que a política FB apresenta resultados melhores do que PS e FCFS, para as duas estratégias de roteamento. O resultado de FB é mais próximo de SRPT para carga de Pareto, que é a mais variável.

A fila FCFS obtém os piores resultados para sistemas distribuídos que utilizam RR, resultados que não são apresentados no gráfico **(F)** devido a seus valores extremamente elevados.

4.2. Sistema fortemente acoplado

Os resultados da simulação para o sistema fortemente acoplado são apresentados nas Figuras 6 e 7. Pode-se observar que os tempos de resposta para todas as políticas são similares para intensidades de tráfego pequenas a moderadas. A política SRPT apresenta novamente o melhor desempenho. Para cargas intensas, FB apresenta o segundo melhor resultado, que é tanto mais próximo de SRPT quanto mais variável for a carga. Os resultados de FB são, novamente, melhores do que PS, que é uma das políticas mais utilizadas em sistemas computacionais.

A Figura 7 apresenta os resultados agrupados por política de escalonamento. Esses gráficos permitem avaliar o efeito da distribuição do tempo de serviço em cada tipo de

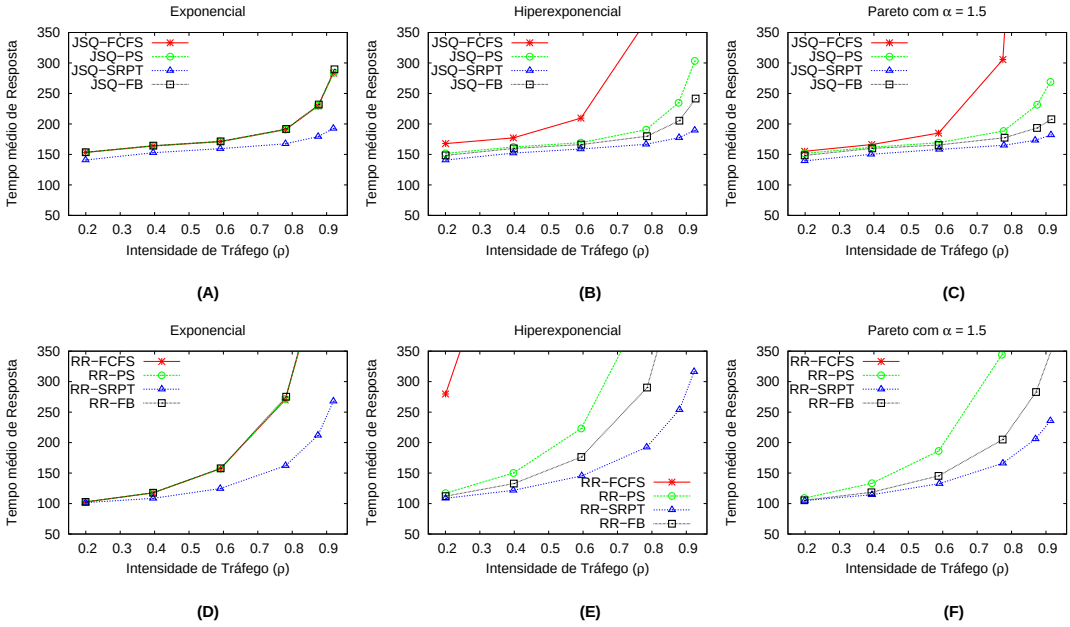


Figura 5. Resultados do sistemas fracamente acoplados agrupados por distribuição de serviço.

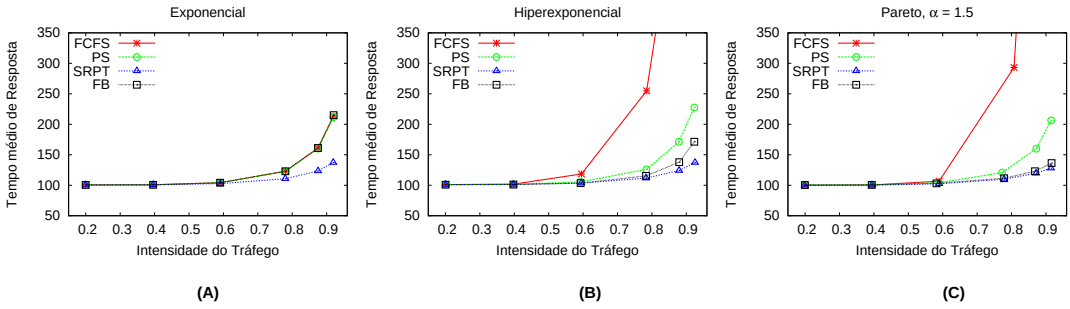


Figura 6. Resultados do sistemas fortemente acoplados de oito servidores para as cargas testadas.

escalonamento. Observa-se que a fila PS é praticamente insensível ao tipo de carga. A fila FB produz tempos menores para cargas mais variáveis.

A Tabela 3 apresenta resultados que permitem a comparação do sistema fortemente acoplado com o sistema fracamente acoplado para a política FB. Os valores apresentados correspondem à razão entre os tempos médios de resposta do sistema fracamente acoplado, representados por RR-FB ou JSQ-FB, pelo resultado da fila FB.

Os resultados indicam que, para todos os tipos de carga, o sistema fortemente acoplado apresenta melhor desempenho do que o sistema fracamente acoplado independente da política de roteamento. Ao comparar as duas estratégias de roteamento, observamos que RR apresenta os melhores resultados do tempo médio de resposta para intensidades de carga baixa e média. No entanto, quando há elevada utilização do sistema, a política de roteamento JSQ apresenta melhor desempenho para os três tipos de carga. A estratégia JSQ provê melhores resultados para cargas mais variáveis.

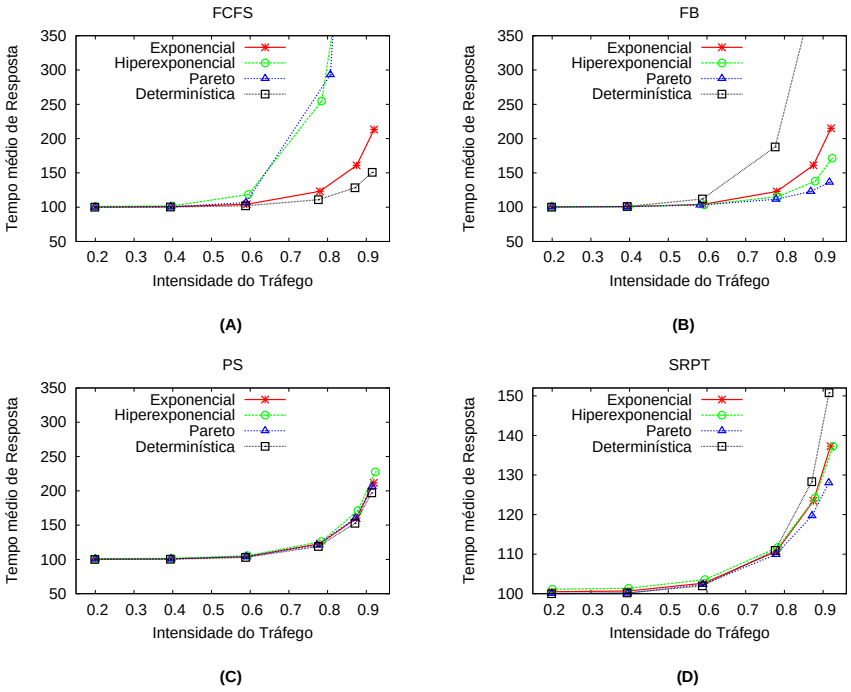


Figura 7. Resultados dos sistemas fortemente acoplados de oito servidores agrupados por política.

	Exponencial		Hiperexponencial		Pareto	
ρ	RR-FB/FB	JSQ-FB/FB	RR-FB/FB	JSQ-FB/FB	RR-FB/FB	JSQ-FB/FB
0,20	1,0206	1,5273	1,1106	1,4710	1,0491	1,4777
0,40	1,1682	1,6323	1,3135	1,5806	1,1771	1,5895
0,60	1,5152	1,6467	1,7052	1,6072	1,4098	1,6046
0,80	2,2397	1,5607	2,5174	1,5596	1,8378	1,5906
0,90	2,9667	1,4389	3,4563	1,4891	2,2997	1,5721
0,95	3,5111	1,3466	4,0737	1,4103	2,5698	1,5224

Tabela 3. Razão dos tempos médios de resposta dos sistemas com acoplamento fraco em relação ao sistema com acoplamento forte para a fila FB**.**

4.3. Avaliação da Justiça quando $\rho \rightarrow 1$

Esta seção apresenta a avaliação da justiça no atendimento das tarefas quando $\rho \rightarrow 1$, isto é, quando o sistema se aproxima de situação de instabilidade. Para essa avaliação, as tarefas foram agrupadas em cinco faixas de tamanhos: Faixa 1: $x = 10^0$; Faixa 2: $10^0 < x \leq 10^1$; Faixa 3: $10^1 < x \leq 10^2$; Faixa 4: $10^2 < x \leq 10^3$; Faixa 5: $10^3 < x \leq 10^4$; Faixa 6: $10^4 < x \leq 10^5$; Faixa 7: $10^5 < x$. As distribuições de serviço simuladas geraram tarefas de diferentes tamanhos, em função de suas características. Para a carga M, as tarefas se limitaram às faixas 1 a 5; para a carga H, as tarefas ficaram entre as faixas 1 e 6; e para distribuição de serviço P, as tarefas geradas ficaram entre as faixas 3 e 7.

O objetivo deste teste é avaliar se tarefas de todas as faixas de tamanho são tratadas de forma equivalente pelas políticas de escalonamento. A justiça é avaliada pela métrica *slowdown*, definida pela equação $S = E[T(x)]/x$, que é a razão entre o tempo de resposta

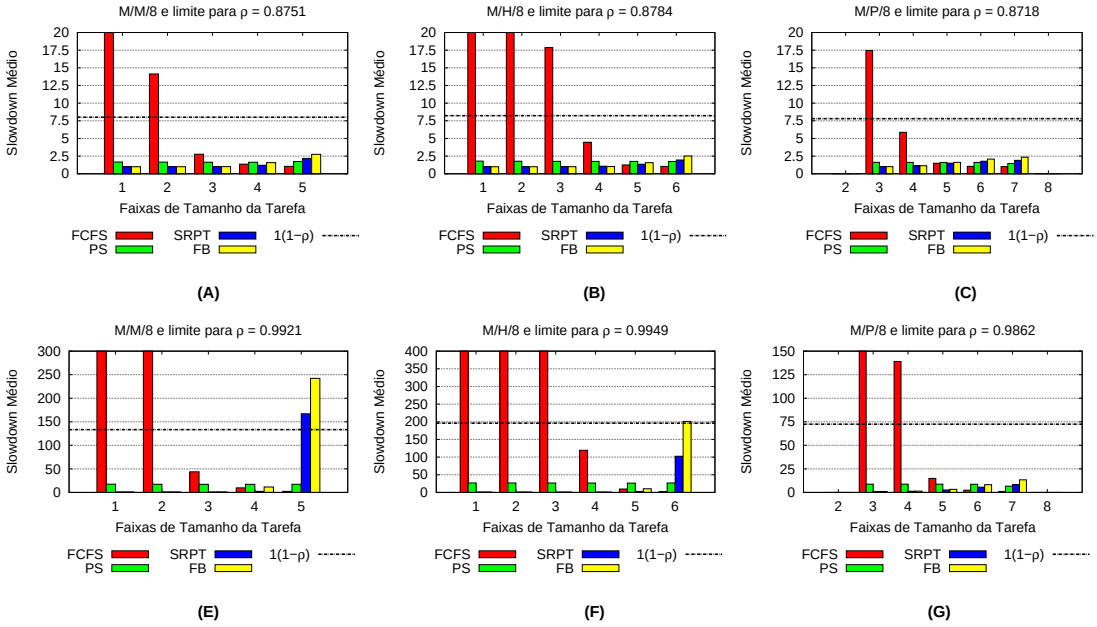


Figura 8. Slowdown médio no sistema fortemente acoplado.

e o tamanho da tarefa. Uma tarefa é tratada com justiça se a condição definida por $S \leq 1/(1-\rho)$ for satisfeita [Nuyens and Wierman 2008]. Esse limite é representado por uma reta paralela ao eixo x nos gráficos da Figura 8.

A Figura 8 apresenta os gráficos[1] dos resultados do *slowdown* médio para o sistema fortemente acoplado, as três distribuições do tempo de serviço, e duas intensidades de tráfego (ρ) por tipo de carga. Os gráficos na primeira linha são para $\rho \approx 0,90$ e os da segunda linha são para $\rho \approx 0,99$. Os gráficos **(A)** e **(E)** apresentam o resultado para a distribuição exponencial (M/M/8) para duas intensidades de carga. Observamos que a fila FCFS trata com injustiça tarefas das classes 1 e 2. As políticas FB e SRPT discriminam apenas as tarefas que se encontram na faixa 5, para a carga mais elevada (gráfico **(E)**). No entanto, para as duas políticas, menos de 0,005% das tarefas são discriminadas. PS trata com justiça tarefas de todas as faixas de tamanho.

Os gráficos **(B)** e **(F)** apresentam o resultado para a distribuição hiperexponencial (M/H/8) para duas intensidades de carga. Neste caso, a fila FCFS trata com justiça apenas as tarefas que se encontram nas faixas 4, 5 e 6. As outras políticas tratam com justiça tarefas em todas as faixas de tamanho. FB se aproxima do limite estabelecido para tarefas da faixa 6. A política SRPT apresenta resultados para *slowdown* melhores do que a política FB para tarefas que se encontram nas faixas 5 e 6.

Finalizando, os gráficos **(C)** e **(G)** apresentam o resultado para a distribuição de Pareto (M/P/8). A fila FCFS continua penalizando tarefas de tamanhos pequenos. As políticas PS, FB e SRPT tratam com justiça tarefas de todos os tamanhos.

Em resumo, observamos que a fila FB se comportou muito bem quanto à justiça para tarefas cujo tamanho apresenta maior variabilidade (hiperexponencial e Pareto) e para intensidades de carga consideráveis, próximas do limite máximo ($\rho \approx 0,99$).

[1]Os gráficos são melhor visualizados na tela devido ao esquema de cores.

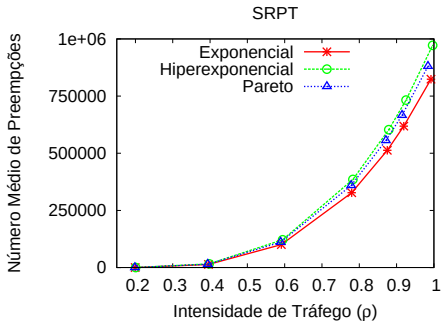


Figura 9. Número de preempções das políticas FB, PS e SRPT.

4.4. Número de Preempções

Para políticas preemptivas, é importante analisar o número de preempções que ocorrem no sistema, uma vez que a preempção gera custos adicionais de tempo para as tarefas de troca de contexto. Foram feitos experimentos para avaliar o número de preempções das políticas FB, PS e SRPT para o sistema fortemente acoplado. Observamos que o número de preempções gerado na execução das políticas FB e PS é semelhante para todos os tipos de carga. O total de tarefas escalonadas em cada caso foi de dez milhões. Para essas tarefas, o número total de preempções para essas duas políticas é de 1×10^8. Esse número não depende da intensidade de tráfego. Assim, conclui-se que o custo de preempção para a política FB é igual ao da política PS.

O gráfico da Figura 9 mostra que o número de preempções da política SRPT. Nesse caso, o número de preempções varia de acordo com a intensidade de tráfego e com os tipos de carga de trabalho. Quanto maior a intensidade de tráfego, maior o número de preempções. O número de preempções foi menor para carga exponencial e maior para carga hiperexponencial. Para Pareto, o número de preempções foi intermediário. Pode-se concluir que, em um problema de escalonamento, o custo de preempções da política SRPT será sempre menor do que o das políticas FB e PS.

4.5. Alterações no Valor do Quantum

O efeito das alterações no quantum é estudado somente para as políticas FB e PS, que definem esse parâmetro. Apesar da política FB priorizar tarefas de acordo com o menor valor de *age*, no momento em que existe um conjunto de tarefas com a mesma prioridade, essa política executa à maneira PS e utiliza um quantum de tempo para execução das tarefas. Para tanto, foram realizados experimentos com diferentes valores de quantum para analisar o impacto no comportamento das políticas FB e PS.

Os testes foram realizados para as cargas exponencial, hiperexponencial e Pareto, com $\alpha = 1,5$, e foram definidos cinco valores de quantum, a saber, 1, 2, 4, 8 e 16 unidades de tempo de acordo com a Tabela 2 do planejamento de experimentos. A alteração nos valores do quantum não causa impacto no tempo médio de resposta, independente do tipo de carga escalonada pela política FB. Da mesma forma, para PS, os diferentes valores de quantum não influenciam o tempo médio de resposta da política PS. Diante dos resultados apresentados, pode-se perceber que os diferentes valores de quantum não influenciam o comportamento das políticas PS e SRPT e não causam nenhum impacto no tempo médio

de resposta. Uma explicação para este resultado é o fato de que, em média, o tamanho das tarefas é pequeno, sendo que a maioria das tarefas é menor do que os valores testados para o quantum, o que torna o quantum sem efeito, uma vez que as tarefas são finalizadas em um quantum.

5. Conclusão

Este artigo apresentou uma análise de desempenho da política de escalonamento FB em ambientes distribuídos com acoplamento fraco e forte. Os resultados de desempenho desta política foram comparados aos resultados das políticas PS, FCFS e SRPT, executadas em condições iguais. As políticas PS e FCFS foram escolhidas por serem bastante conhecidas e estudadas. Assim, elas provêem uma base segura para comparação. Por outro lado, a política SRPT foi escolhida por ser uma política ótima, que provê os melhores resultados em termos da métrica alvo, tempo médio de resposta. Dessa forma, ela oferece um limite inferior, igualmente importante para mostrar o desempenho da fila FB frente ao melhor desempenho possível. No modelo de acoplamento fraco as requisições foram roteadas, no momento da chegada, pelas estratégias de roteamento JSQ e RR.

Os resultados indicam que a fila FB produz melhores tempos médios de resposta do que as filas PS e FCFS e compatíveis com SRPT para cargas com maior variabilidade e em situações de intensidade de tráfego moderada a elevada. Devemos observar que quanto maior a variabilidade da carga e maior a intensidade de tráfego, mais crítica é a situação do sistema, uma vez que o aumento destas condições é prejudicial ao desempenho do sistema. Portanto, FB produz bons resultados em sistemas em estado crítico. Por outro lado, FB não requer conhecimento do tamanho da tarefa, enquanto SRPT, que produz os melhores resultados, requer esse conhecimento. Assim, FB demonstra ser uma política bastante interessante para sistemas em estado crítico que não têm conhecimento a priori do tamanho da tarefa, ou que preferem não utilizar essa informação no escalonamento.

Os trabalhos futuros estão direcionados para a realização de experimentos com sistemas heterogêneos, compostos por recursos de capacidades diferentes. Além disso, experimentos com cargas altamente variáveis, como Pareto com $\alpha = 1{,}2$, são necessários para simular cargas ainda mais variáveis. Tendo em vista o contexto de discriminação de tarefas por classes de prioridades, experimentos com diferentes classes de prioridades de tarefas devem ser realizados.

6. Agradecimentos

Os autores agradecem ao UNIFOR/MG, à FAPEMIG e ao INCT Web pelo apoio.

Referências

Anger, F. D., Hwang, J.-J., and Chow, Y.-C. (1990). Scheduling with Sufficient Loosely Coupled Processors. *Journal of Parallel and Distributed Computing*, 9(1):87–92.

Bansal, N. and Harchol-Balter, M. (2001). Analysis of SRPT Scheduling: Investigating Unfairness. In *Proceedings of ACM SIGMETRICS 2001*.

Casavant, T. and Kuhl, J. (1988). A Taxonomy of Scheduling in General-purpose Distributed Computing Systems. *Software Engineering, IEEE Transactions on*, 14(2).

Conway, R. W., Maxwell, W. L., and Miller, L. W. (2003). *Theory of Scheduling*. Dover Publications, NY, USA.

Crovella, M. E. and Lipsky, L. (1997). Long-lasting Transient Conditions in Simulations With Heavy-tailed Workloads. In *WSC '97: Proceedings of the 29th conference on Winter simulation*, pages 1005–1012, Washington, DC, USA. IEEE Computer Society.

Down, D. G. and Wu, R. (2006). Multi-layered Round robin Routing for Parallel Servers. *Queueing Syst. Theory Appl.*, 53:177–188.

Feng, H. and Misra, V. (2004). On the Relationship between Coefficient of Variation and the Performance of M/G/1 - FB queues. *SIGMETRICS Perform. Eval. Rev.*, 32(2).

Gupta, V. (2008). Finding the Optimal Quantum Size: Sensitivity Analysis of the M/G/1 Round-robin Queue. *ACM SIGMETRICS Performance Evaluation Review*, 36(2).

Harchol-Balter, M., Crovella, M., and Murta, C. (1999). On Choosing a Task Assignment Policy for a Distributed Server System. *IEEE J. of Parallel and Distrib. Computing*.

Jain, R. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation and Modeling*. John Wiley and Sons, New York, NY, USA.

MacDougall, M. H. (1987). *Simulating Computer Systems: Techniques and Tools*. MIT Press, Cambridge, MA, USA.

Nuyens, M. (2004a). *The Foreground-Background Queue*. PhD thesis, Universit van Amsterdam - Met samenvatting in het Nederlands.

Nuyens, M. (2004b). *The Foreground-Background Queue*. PhD thesis, Universit van Amsterdam - Met samenvatting in het Nederlands.

Nuyens, M. and Wierman, A. (2008). The Foreground-Background Queue: A Survey. *Perform. Eval.*, 65(3-4):286–307.

Schrage, L. E. (1968). A Proof of the Optimality of the Shortest Processing Remaining Time Discipline. *Operations Research*, 16:678–690.

Schrage, L. E. and Miller, L. W. (1966). The Queue M/G/1 with the Shortest Remaining Processing Time Discipline. *Operations Research*, 14(4):670–684.

Sousa, A. M. and Murta, C. D. (2010). Simulation of the Foreground-Background Queue in Parallel Systems. *XI WSCAD–SSC 2010*, pages 17–24.

Tanenbaum, A. S. and Steen, M. V. (2007). *Distributed Systems Principles and Paradigms*. Prentice Hall, USA.

Tanenbaum, A. S. and Woodhull, A. S. (2006). *Operating Systems Design and Implementation*. Prentice Hall, USA.

Wierman, A. (2007a). Fairness and Classifications. *SIGMETRICS Performance Evaluation Review*, 34(4):4–12.

Wierman, A. (2007b). *Scheduling for Today's Computer Systems: Bridging Theory and Practice*. PhD thesis, School of Comp. Science, Carnegie Mellon Univ., Pittsburgh.

Auto-Organização na Seleção de Candidatos a Migração em Aplicações Bulk Synchronous Parallel

Lucas Graebin[1], **Diogo Pinheiro de Araújo**[1], **Rodrigo da Rosa Righi**[1]

[1]Programa Interdisciplinar de Pós-Graduação em Computação Aplicada
Universidade do Vale do Rio dos Sinos
CEP: 93.022-000 – São Leopoldo – RS – Brasil

lgraebin@acm.org, {dapinheiro,rrrighi}@unisinos.br

Abstract. *Process rescheduling can be a suitable technique for achieving high performance, mainly when irregular workloads and cluster-of-clusters environments are considered. The decision to move processes to new resources is NP-Hard, and heuristics take place in order to reach good results inside an acceptable time interval. In this way, this paper presents AutoMig – a novel heuristic for BSP (Bulk Synchronous Parallel) applications that self-organizes the selection of candidates for migration on different clusters. Its differential approach consists in a prediction function (pf) that considers processes' computation and communication data as well as their migrations costs. pf is applied over a list of schedules and AutoMig's final step decides whether one of them outperforms the current mapping. The results emphasize gains up to 32% when testing a CPU-bound application in a simulated cluster-of-clusters environment.*

Resumo. *Reescalonamento de processos pode ser uma técnica pertinente para atingir melhor desempenho, principalmente quando cargas irregulares e ambientes com múltiplos clusters são considerados. A decisão de mover processos para novos recursos é NP-Difícil, e heurísticas são usadas com o intuito de atingir bons resultados em um tempo aceitável. Nesse sentido, esse artigo apresenta AutoMig – uma nova heurística para aplicações BSP (Bulk Synchronous Parallel) que auto-organiza a seleção dos candidatos a migração para diferentes clusters. Seu diferencial é uma função de predição (pf) que considera dados de computação e comunicação dos processos, bem como os custos de migração. pf é aplicada sobre uma lista de escalonamentos e AutoMig decide qual é o mais adequado deles para atingir um desempenho melhor que o oferecido pelo mapeamento corrente. Os resultados mostram ganhos de até 32% nos testes de uma aplicação CPU-Bound em um ambiente simulado com múltiplos clusters.*

1. Introdução

Na medida em que a computação distribuída é cada vez mais explorada, recursos de múltiplos *clusters* se tornam os nós de computação dominantes na formação de *grids* [El Kabbany et al. 2011, Qin et al. 2010, Xhafa and Abraham 2010]. Aplicações contendo rotinas para a resolução de sistemas lineares e Transformada Rápida de Fourier (FFT) são exemplos típicos de aplicações paralelas fortemente acopladas que podem tirar proveito de processamento de múltiplos *clusters* [Sanjay and Vadhiyar 2009]. Uma vez que esses *clusters* podem ser heterogêneos entre si e que ligações inter-clusters são

normalmente mais lentas que as usadas internamente, questões como escalonamento e balanceamento de carga devem ser tratados com o objetivo de garantir bom desempenho nessa arquitetura. Uma alternativa para oferecer ambos os tratamentos é o emprego da migração de processos. Assim, é possível remodelar o mapeamento processos-recursos em tempo de execução em resposta ao comportamento dos processos e dos recursos (processadores e rede).

Geralmente, migração de processos é implementada junto com a própria aplicação paralela, resultando numa implementação específica e não extensível para um determinado problema. Além disso, há iniciativas que trabalham com chamadas explícitas dentro do código da aplicação [Bhandarkar et al. 2000] e/ou obrigam execuções extras para conseguir dados de entrada para o escalonador [Silva et al. 2005, Yang and Chou 2009]. Uma abordagem diferente para a migração acontece em nível de *middleware*, onde mudanças na aplicação e conhecimento prévio do sistema normalmente não são requeridos. Nesse sentido, foi desenvolvido um modelo de reescalonamento de processos chamado MigBSP [da Rosa Righi et al. 2010]. Tal modelo foi projetado para operar sobre aplicações baseadas em fases que seguem o estilo BSP (*Bulk Synchronous Parallel*) e atua sobre arquiteturas com múltiplos *clusters*. Além disso, MigBSP faz uso de heurísticas em seus algoritmos porque o problema de encontrar um mapeamento ótimo em sistemas heterogêneos é um problema NP-Difícil [Xhafa and Abraham 2010].

Considerando a escolha dos processos para reescalonamento, o modelo MigBSP cria uma lista de prioridade baseada no maior Potencial de Migração (PM) de cada processo [da Rosa Righi et al. 2010]. PM combina os custos de migração com dados vindos das fases de computação e comunicação para oferecer uma métrica de escalonamento unificada. O processo presente no topo da lista é sempre escolhido para ser inspecionado quanto a migração. Embora essa abordagem acarretasse em bons resultados, concorda-se que uma outra mais otimizada poderia lidar com múltiplos processos quando a chamada de reescalonamento acontece. Uma possibilidade seria selecionar uma percentagem de processos baseado no maior PM. Contudo, uma questão é pertinente: Como é possível achar uma porcentagem otimizada para aplicações dinâmicas e ambientes heterogêneos? A solução poderia envolver o teste de várias instâncias de parâmetros. Certamente, essa ideia consome tempo e novas aplicações e recursos requerem um novo conjunto de testes.

Ao longo do desenvolvimento da versão de MigBSP descrita acima, foi observado o crescimento do uso de sistemas de escalonamento os quais ajustam parâmetros em tempo de execução e escondem complexidades de configuração e decisões de otimização dos usuários [Ding et al. 2009, Nascimento et al. 2007, Sanjay and Vadhiyar 2009]. Nesse contexto, foi desenvolvida uma heurística chamada **AutoMig** que seleciona um ou mais candidatos a migração automaticamente. Ela tira proveito dos conceitos de *List Scheduling* [Duselis et al. 2009] e *Backtracking* [Baritompa et al. 2007] para avaliar, de maneira autônoma, o impacto da migração sobre cada elemento da lista de PM. Em adição, outra característica de AutoMig é a não necessidade de informar parâmetros adicionais em MigBSP para selecionar múltiplos processos para migração. A avaliação do escalonamento usa uma função de predição (*pf*) que considera os custos de migração e trabalha segundo os fundamentos de uma superetapa BSP [Bonorden 2007]. O menor valor obtido indica o melhor plano obtido para o reescalonamento dos processos.

O objetivo desse artigo é descrever AutoMig. Para isso foi construída uma

aplicação que computa a compressão de imagens baseada na técnica de Fractal [Guo et al. 2009]. Uma vez que o programador não modifica sua aplicação para usar AutoMig, considera-se os resultados cujos ganhos médios foram de 7.9% satisfatórios. Tal classificação também se deve ao fato de AutoMig não precisar de dados da aplicação e dos recursos em tempo de projeto ou compilação. Os resultados mostraram situações onde AutoMig passa o desempenho da heurística que escolhe um só processo.

O presente artigo está organizado da seguinte forma. A próxima seção mostra MigBSP e serve como base para as explicações sobre a heurística proposta. A Seção 3 descreve essa heurística. Ambas Seções 4 e 5 apresentam a metodologia e os resultados experimentais, respectivamente. Trabalhos relacionados são discutidos na Seção 6. Por fim, a Seção 7 conclui o artigo e descreve trabalhos futuros.

2. MigBSP: Modelo de Reescalonamento de Processos

MigBSP responde as seguintes questões sobre balanceamento de carga: (i) "Quando" lançar a migração; (ii) "Quais" processos são candidatos a migração; (iii) "Onde" colocar os processos eleitos. As ideias para o tratamento dessas questões estão apresentadas em detalhes no trabalho [da Rosa Righi et al. 2010]. MigBSP trabalha sobre uma arquitetura heterogênea composta por *clusters*, supercomputadores e/ou redes locais. A heterogeneidade trata de processadores com diferentes velocidades e redes com larguras de banda e níveis distintos (Fast e Gigabit Ethernet e ambientes com múltiplos *clusters*, por exemplo). Tal arquitetura é montada com Conjuntos e Gerentes de Conjuntos. Um Conjunto pode ser uma LAN ou um *cluster*. Cada Conjunto possui um Gerente de Conjunto, o qual captura informações sobre seu Conjunto e as troca com os demais Gerentes.

A decisão de remapeamento dos processos é tomada no final de cada superetapa. Com o intuito de gerar menos intrusividade na aplicação, MigBSP atua com duas adaptações que controlam o valor de α ($\alpha \in \mathbb{N}^*$). α é atualizado a cada chamada de reescalonamento e irá indicar o intervalo para o lançamento da próxima. Os objetivos das adaptações são: (i) postergar a chamada de reescalonamento se os processos estão balanceados ou a tornar mais frequente, caso contrário; (ii) atrasar a chamada de reescalonamento caso um padrão sem migrações é observado nas ω últimas chamadas. Uma variável D é usada para denotar a percentagem de quão longe o processo mais lento e o mais rápido podem estar da média para classificar os processos como balanceados. Em suma, quanto maior o valor de α, menor o impacto do modelo na execução da aplicação.

A resposta à questão "Quais" é atingida através da função de decisão chamada Potencial de Migração (PM). Cada processo i computa n funções $PM(i, j)$, onde n é o número de Conjuntos e j significa um Conjunto específico. O diferencial nessa abordagem é a realização de um subconjunto de testes processos-recursos no momento do reescalonamento. $PM(i, j)$ é obtido através da combinação das métricas Computação, Comunicação e Memória (veja Equação 1). A relação entre as métricas está baseada na noção de força da física. Computação e Comunicação são métricas que atuam a favor da migração, enquanto a Memória trabalha em direção oposta. Na medida em que o valor de $PM(i, j)$ aumenta, aumenta também a possibilidade de migração do processo avaliado.

$$PM(i, j) = Comp(i, j) + Comm(i, j) - Mem(i, j) \quad (1)$$

A métrica Computação – $Comp(i, j)$ – considera três informações: (i) predição

do tempo de computação do processo i; (ii) Padrão de Computação do processo i; (iii) grau de desempenho do Conjunto j. A predição é baseada no conceito de Aging [Tanenbaum 2003] e usa dados capturados entre duas chamadas de reescalonamento. O Padrão de Computação mede a estabilidade, ou regularidade, do processo levando em conta a quantidade de instruções realizadas a cada superetapa. Esse valor é perto de 1 caso o processo apresente regularidade e próximo de 0 caso contrário. Assim como a métrica de Computação, a métrica Comunicação – $Comm(i, j)$ – computa o Padrão de Comunicação entre processos e Conjuntos. Esse padrão considera a quantidade de dados recebidos pelo processo i de processos que pertencem ao Conjunto j a cada superetapa. Em adição, essa métrica também usa uma predição do tempo de comunicação entre ativações de rebalanceamento. A métrica Memória – $Mem(i, j)$ – trabalha com os seguintes dados: (i) memória do processo; (ii) tempo de transferência entre o processo i e o Gerente do Conjunto j; (iii) custos de migração. Esse último tópico depende do sistema operacional bem como das ferramentas usadas para a migração de processos.

Os processos calculam $PM(i, j)$ localmente. A cada chamada para migração, eles passam o seu maior $PM(i, j)$ para seus gerentes. Esta entidade troca o PM recebido com os demais gerentes. Como já mencionado, existe uma heurística para escolher o candidato, a qual captura o processo no topo de uma lista decrescente construída com os valores de PM. $PM(i, j)$ está associado a um processo i e a um Conjunto j. O gerente desse conjunto deve selecionar o processador mais propício para receber o processo i. Antes de cada migração, sua viabilidade é verificada através do uso das seguintes informações: (i) carga externa nos processadores origem e destino; (ii) processos BSP que ambos processadores estão executando; (iii) a simulação da execução do processo no processador destino; (iv) o tempo de comunicação entre os processadores; (v) custos de migração. Dessa forma, MigBSP computa dois tempos: t_1 e t_2. t_1 representa a execução local do processo i, enquanto t_2 compreende sua execução em um outro processador e inclui os custos de migração. Para o candidato, um novo recurso é escolhido se $t_1 > t_2$.

3. AutoMig: Proposta para Auto-Selecionar os Candidatos a Migração

AutoMig é uma heurística para a seleção automática de processos para migração a cada chamada de reescalonamento. Ela pode eleger não somente um, mas uma coleção de processos. AutoMig auto-organiza os processos migráveis sem a intervenção do programador em ambos os níveis de aplicação e escalonador. Levando em consideração a função de decisão PM, AutoMig propõe uma solução para a sentença abaixo.

- **Sentença do Problema:** Dados n processos e uma lista do maior PM (Potencial de Migração) de cada um deles no momento do reescalonamento, o desafio consiste na criação e avaliação de no máximo n planos de escalonamento e escolher o mais apto entre aqueles que têm desempenho melhor que o mapeamento corrente.

AutoMig resolve a sentença acima usando os conceitos de *List Scheduling* e *Backtracking*. Primeiramente, é feito o ordenamento da lista de PM de maneira decrescente. Dessa forma, os testes começam pelo processo no topo visto que representa as melhores chances de ganhos com a migração. Num segundo momento, AutoMig processa n tentativas de escalonamento (onde n é o número de processos) através do incremento da movimentação de apenas um processo a cada novo plano. Essa ideia é baseada na técnica do *Backtracking*, onde cada candidato parcial é o pai de candidatos que diferem dele de

apenas um passo de escolha [Baritompa et al. 2007]. O Algoritmo 1 representa a abordagem de AutoMig em detalhes. A Figura 1 expande a explicação do algoritmo em um exemplo com 7 processos. Para cada processo i descrito em $PM(i, j)$, AutoMig emula sua execução aplicando uma predição pf com o intuito de medir o desempenho do novo escalonamento. Esse escalonamento é calculado através do teste de relocação do processo i para o Conjunto j denotado como parâmetro de PM.

Algoritmo 1 Escolhendo os processos com AutoMig no momento do reescalonamento

1: Cada processo computa PM (Potencial de Migração) localmente (veja Equação 1).
2: Cada processo passa o seu maior PM, junto com o número de instruções e um vetor que descreve suas ações de comunicação, para o seu Gerente de Conjunto.
3: Gerentes de Conjunto trocam os valores de PM de seus respectivos processos.
4: Gerentes de Conjunto criam uma lista decrescente baseada nos valores de PM. Tal lista possui n elementos, onde n é o número de processos.
5: Cada Gerente de Conjunto computa pf para os processos sob sua jurisdição. Tais dados serão úteis para mensurar o desempenho do mapeamento corrente.
6: **for** cada elemento partindo de 0 até $n - 1$ que estão na lista de $PM(i, j)$ **do**
7: O elemento considerado é analisado. Gerente do processo i envia dados para o Gerente do Conjunto j. Esse último é responsável por computar os dados para i.
8: O gerente no Conjunto destino escolhe um processador para processo i. A partir de agora, tal processo é considerado como migrado para o Conjunto j para fins de cálculo do algoritmo.
9: Cada Gerente de Conjunto calcula pf para os processos sob sua jurisdição.
10: Gerentes de Conjunto salvam dados de pf em um vetor.
11: **end for**
12: Gerentes de Conjunto trocam os seus dados e computam pf para o escalonamento corrente assim como para cada nível na lista de PM.
13: **if** $Min(pf)$ na lista de $PM < pf$ corrente **then**
14: Considerando a lista de PM, os processos pertencentes ao nível onde pf foi encontrado são selecionados para migração.
15: Gerentes notificam os processos eleitos que estão sob sua jurisdição para migrar
16: **else**
17: Migrações não acontecem.
18: **end if**

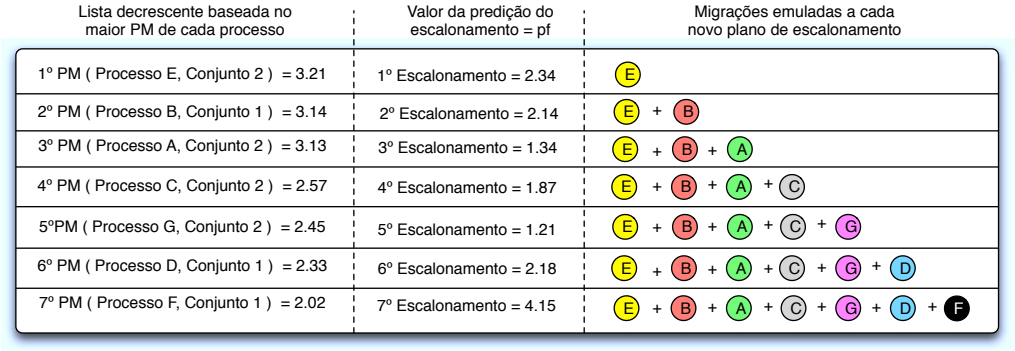


Figura 1. Exemplo da abordagem de AutoMig: Cada elemento na lista de PM produz um novo escalonamento através do incremento de uma migração

O algoritmo começa pela avaliação do escalonamento atual. Na sequência, AutoMig calcula pf para cada um dos níveis da lista de PM. Analisando o exemplo apresentado na Figura 1, o desempenho da predição do processo "A" no terceiro nível de PM leva em consideração a migração deste processo e o fato de que "E" e "B" também foram previamente migrados. A seleção final é obtida através da verificação do menor pf. Os processos pertencentes a essa predição são eleitos para migração caso esse novo mapeamento ultrapasse o desempenho do corrente. A principal parte de AutoMig é a sua função de predição pf. No momento do reescalonamento, cada processo passa as seguintes informações para seu gerente: (i) o maior valor de PM; (ii) um vetor com os custos de migração (métrica Mem) para cada Conjunto; (iii) um vetor o qual contém a quantidade de bytes em ações de comunicação para cada um dos Conjuntos. Esses dois últimos dados são capturados na última superetapa executada. pf considera ambas as partes de computação e comunicação com o intuito de emular o tempo despendido na superetapa. As duas são encontradas através das Equações 2 e 3, respectivamente. Em adição, pf trabalha com os custos de migração de processos para Conjuntos. Inicialmente, pf é usado para computar o desempenho do mapeamento corrente. Cada Gerente o computa para seus processos e salva os resultados localmente (passo 5 no Algoritmo 1).

O próximo passo, procede-se com o cálculo de pf para cada processo na lista de PM. Em cada nível, informações do processo alvo são transferidas para o Conjunto destino. A cargo de exemplo, dados do processo "E" são transferidos para o Conjunto 2 como ilustra a Figura 1. Assim, o gerente no Conjunto destino irá escolher o processador mais apto sob sua jurisdição para o processo e irá calcular ambas as Equações 2 e 3 para ele. Com o intuito de minimizar a comunicação *multicast* entre os gerentes a cada computação de pf, cada Gerente de Conjunto computa $Time_p$ e $Comm_p$ para os processos sob seu comando e salva os dados de um nível específico localmente. Depois de calcular o último pf, os gerentes trocam informações e calculam pf para cada nível da lista, bem como para o escalonamento corrente (passo 12 no Algoritmo 1).

A Equação 2 calcula $Time_p(i)$, onde i representa um processo específico. $Time_p(i)$ usa dados relativos ao poder de computação e dados de carga do processador na qual o processo i executa atualmente ou o processador que está sendo testado para reescalonamento. $cpu_load(i)$ denota a média de carga da CPU nos últimos 15 minutos. Este intervalo de tempo foi adotado baseado nos estudos prévios de Vozmediano e Conde [Moreno-Vozmediano and Alonso-Conde 2005]. A Equação 3 mostra como é atingido o tempo máximo de computação ao considerar o processo i e o Conjunto j. Neste contexto, o Conjunto j pode ser o atual do processo i ou o Conjunto no qual o processo está sendo testado para migração. $T(k, j)$ se refere a taxa de transferência de 1 byte do Gerente do Conjunto j para outro gerente. Nessa mesma ideia, $Bytes(i, k)$ trabalha com o número de bytes transferidos pela rede entre o processo i e os processos que pertencem ao Conjunto k. Por fim, $Mig_Costs(i, j)$ mostra os custos de migração relativos a transferência do processo i para o Conjunto j. A Equação 4 apresenta cada parte da função pf. É importante enfatizar que cada parte de pf pode considerar um processo i e um Conjunto j diferentes. A título de exemplo, um processo específico pode obter o maior tempo de computação, enquanto outro pode despender mais tempo nas ações de comunicação.

$$Time_p(i) = \frac{Instruction(i)}{(1 - cpu_load(i)) \cdot cpu(i)} \quad (2)$$

$$Comm_p(i,j) = Max_k\ (\ \forall\ k \in Sets\ (Bytes(i,k).T(k,j))) \quad (3)$$

$$pf = Max_i\ (Time_p(i)) + Max_{i,j}\ (Comm_p(i,j)) + Max_{i,j}\ (Mig_Costs(i,j)) \quad (4)$$

Depois de concluir o passo 12 do Algoritmo 1, cada Gerente de Conjunto conhece todos os valores de predição. Consequentemente, ele está apto a discernir sobre os processos que irão migrar. Uma ou mais migrações podem ocorrer caso pf para a situação corrente seja maior que aquela encontrada em quaisquer dos níveis da lista de PM. Se isso for o caso, cada Gerente de Conjunto verifica os processos migráveis sobre sua responsabilidade e os avisa sobre a tarefa de relocação. Os principais pontos positivos de AutoMig são: (i) a não necessidade de informar parâmetros adicionais em MigBSP sobre a seleção dos processos; (ii) minimizar comunicação *multicast* entre os gerentes com o intuito de reduzir os custos de escalonamento; (iii) possibilidade de selecionar uma coleção de processos para migração. Além disso, o desempenho final da aplicação irá depender do comportamento dos processos depois do reescalonamento. AutoMig assume que os processos irão reproduzir suas ações de computação e comunicação no computador destino. A questão que trata a regularidade dos processos já é tratada na computação de PM (ver [da Rosa Righi et al. 2010] para mais detalhes).

Kowk e Cheung [Kwok and Cheung 2004] classificam o tópico de balanceamento de carga em quatro classes: (i) política de localização; (ii) política de informação; (iii) política de transferência; (iv) política de seleção. AutoMig responde a última questão usando uma estratégia global [Zaki et al. 1997]. Para tanto, a lista do maior PM de todos os processos BSP é conhecida pelos Gerentes de Conjunto no momento da tentativa de migração. Dessa forma, a principal vantagem de um esquema global compreende uma melhor qualidade nas decisões de balanceamento de carga uma vez que todos os objetos estudados são considerados. Entretanto, a sincronização é a parte mais pesada nessa abordagem [Zaki et al. 1997]. Mas AutoMig tira proveito da organização do modelo BSP que já impõe por si só uma barreira entre os processos. Assim, não é necessário adicionar um custo a mais para implementar a ideia global de balanceamento de carga.

4. Metodologia de Avaliação

A avaliação passa pela simulação do funcionamento de uma aplicação BSP que realiza a compressão de imagens segundo a técnica de fractais. Tal técnica tem gerado interesse na comunidade de compressão de imagens como um possível competidor com formatos já estabelecidos como JPEG e Wavelets [Guo et al. 2009]. Um dos principais problemas na abordagem de fractal é a alta complexidade de compressão. Entretanto, a decodificação acontece num tempo muito menor [Xing 2008]. Tal técnica explora similaridades dentro das imagens, as quais são descritas por uma transformação contrativa da imagem. Essa operação consiste na transformação de blocos os quais aproximam partes menores da imagem por partes mais largas. As partes menores são chamadas de intervalo, enquanto as maiores de domínio. Todos os intervalos formam uma imagem. Os domínios são selecionados livremente dentro da imagem. A modelagem BSP considera a variação nos tamanhos do intervalo e do domínio, bem como no número de processos. Primeiramente, são computadas $\frac{t}{r}$ superetapas, onde $t \times t$ é o tamanho da imagem e r é o tamanho do lado do intervalo. O objetivo é computar um conjunto de intervalos em cada superetapa.

A cada superetapa, um intervalo é testado contra $8((\frac{t}{d})^2.\frac{1}{n})$ domínios, onde d representa o tamanho de um domínio e n o número de processos. Além disso, cada domínio envia $\frac{t}{r}$ intervalos antes de chamar a função de barreira. Tal quantidade deve ser multiplicada por 8 para encontrar o número de bytes transferidos pela rede.

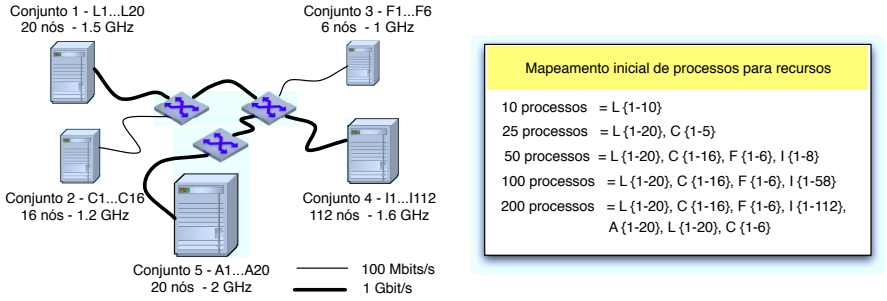


Figura 2. Topologia de testes com múltiplos clusters, descrição dos recursos de processamento e de rede, bem como mapeamento inicial processos-recursos

O principal objetivo da avaliação é observar o desempenho de MigBSP ao usar a heurística AutoMig. Baseado nesse princípio, foi aplicada simulação em três cenários: (i) execução da aplicação simplesmente; (ii) execução da aplicação com MigBSP sem aplicar migrações; (iii) execução da aplicação com MigBSP permitindo as migrações. O cenário ii compreende todos os cálculos de escalonamento mas não faz a transferência de processos. O cenário iii adiciona os custo de migração caso quaisquer processos sejam transferidos para novos recursos. A aplicação e o modelo foram desenvolvidos usando o simulador Simgrid (Módulo MSG) [Casanova et al. 2008]. Ele possui um caráter determinístico, onde os mesmos dados de entrada resultam sempre numa saída igual.

Para avaliar os cenários, foi montada uma infraestrutura com 5 Conjuntos os quais são descritos na Figura 2. Um Conjunto representa um *cluster* onde cada nó tem um único processador. A infraestrutura permite analisar o impacto da heterogeneidade nos algoritmos de AutoMig. Além disso, ela é uma adaptação dos recursos existentes atualmente na Universidade do Vale do Rio dos Sinos. Testes iniciais foram executados usando α igual a 4 e D igual a 0.5 e uma quantidade de processos que varia da seguinte maneira: 10, 25, 50, 100 e 200 processos. Seus mapeamentos iniciais para recursos podem ser vistos na Figura 2. Uma vez que a aplicação desempenha comunicações do processo i para o $i+1$, foi adotada a abordagem contínua na qual cada *cluster* é totalmente preenchido antes de completar o próximo [Pascual et al. 2009]. Além dos parâmetros previamente denotados, usou-se os valores de 40, 20 e 10 para o lado d do domínio e um intervalo obtido por $\frac{d}{2}$. A figura considerada é quadrada com lado 1000. Quanto menor o valor d, maior o número de domínios para serem testados em cada processo. Por fim, os custos de migração estão baseados em execuções com AMPI [Huang et al. 2006] em nossos *clusters*.

5. Analisando o Desempenho e as Decisões de AutoMig

A Tabela 1 apresenta os testes iniciais que ocorrem no tratamento dos valores de 40 e 20 para domínio e intervalo. Esta configuração habilita um número pequeno de domínios para serem computados em cada processo. Sendo assim, cada um requisita poucos recursos de processamento e suas migrações não são viáveis. Os valores de PM nessa situação são negativos devido ao baixo peso das ações de computação e comunicação se comparadas com os custos de migração. Consequentemente, ambos os tempos dos cenários ii e iii

são maiores que o do i. Nesse contexto, uma grande sobrecarga é imposta por MigBSP uma vez que a aplicação executa em aproximadamente 1 segundo.

Tabela 1. Resultados com 40 e 20 para domínio e intervalo (tempo em segundos)

Processos	Cenário i	Cenário ii	Cenário iii
10	1.20	2.17	2.17
25	0.66	1.96	1.96
50	0.57	2.06	2.06
100	0.93	2.44	2.44
200	1.74	3.41	3.41

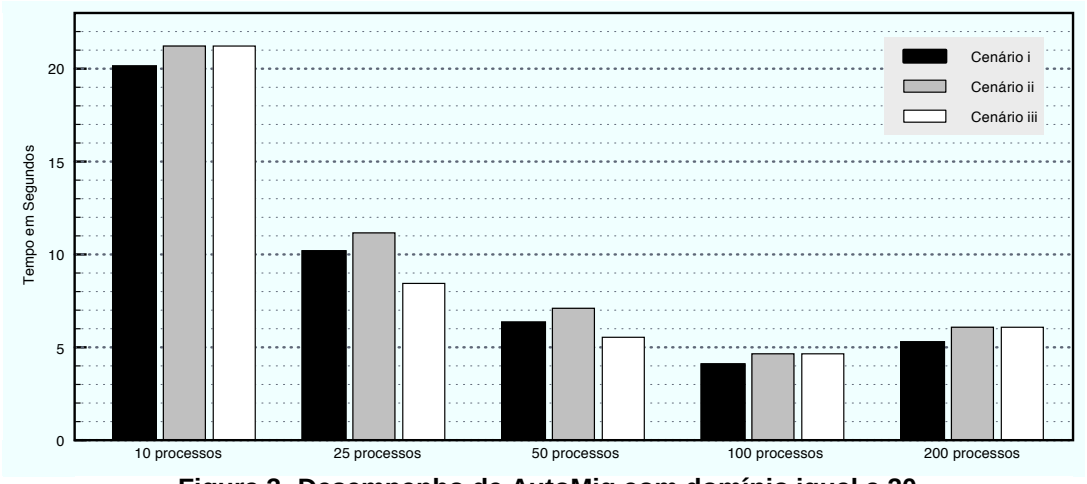


Figura 3. Desempenho de AutoMig com domínio igual a 20

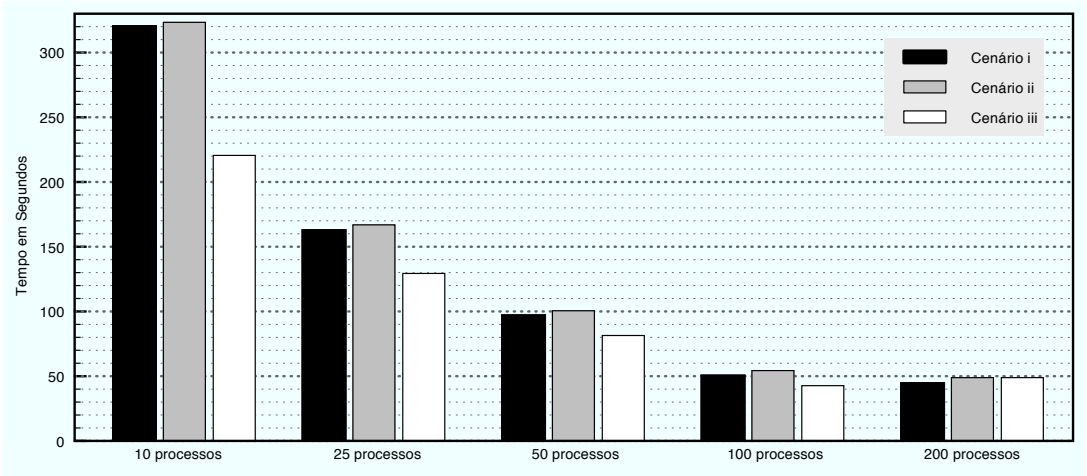


Figura 4. Resultados de AutoMig quando o trabalho para cada processo é aumentado. O gráfico ilustra experimentos com domínio 10 e intervalo 5

O número de domínios é aumentado ao lidar com o número 20 para o lado do domínio (veja gráfico da Figura 3). Como apresentado na execução prévia, migrações não acontecem com 10 processos. Eles estão balanceados e sua reorganização para um *cluster* mais rápido impõe um custo maior que os benefícios. Um valor de pf de 0.21 foi obtido para o mapeamento processos-recursos corrente ao usar 20 para domínio e 10 processos. Todas as predições na lista de PM são maiores de 0.21 e sua média alcança 0.38. Entretanto, essa configuração de domínio lança migrações com 25 e 50 processos. No primeiro caso, 5 processos do *cluster* C são movidos para o *cluster* mais rápido chamado

A. As decisões de AutoMig acarretam um ganho de 17.15% com o reescalonamento de processos nesse contexto. O último *cluster* mencionado recebe todos os processos do *cluster* F quando a aplicação executa com 50 processos. Essa situação mostra ganhos de 20.05% com as migrações. Embora 14 nós do *cluster* mais rápido A estejam livres, AutoMig não seleciona processos para executar neles pois o modelo BSP apresenta uma barreira de sincronização. Por exemplo, ainda que 14 migrações ocorram do *cluster* C para o A, um grupo de processo permanecerá no *cluster* mais lento e irão determinar o tempo da superetapa. Por fim, uma vez que o grão de trabalho diminui na medida que aumenta-se os processos, execuções com 100 e 200 deles não mostraram migrações.

A aplicação BSP demonstrou um bom nível de desempenho com domínio igual a 10, como pode ser visto na Figura 4. O grão de computação aumenta exponencialmente com esta configuração. Isso pode ser observado principalmente na execução de 10 processos, onde todos eles migram para o *cluster* A. Considerando que $8((\frac{t}{d})^2.\frac{1}{10})$ expressa o número de domínios mapeados para cada um dos 10 processos, essa expressão é igual a 500, 2000 e 8000 ao lidar com os valores para domínios iguais a 40, 20 e 10. Usando o valor 10 para domínio e processos, o escalonamento corrente produziu um pf igual a 1.62. Os valores de pf para a lista de PM são vistos a seguir:

- $pf[1..10] = \{1.79, 1.75, 1.78, 1.79, 1.81, 1.76, 1.74, 1.82, 1.78, 1.47\}$.

Considerando o primeiro até o nono pf, pode-se verificar que embora alguns processos rodem mais rápido num *cluster* mais apropriado, existem outros que permanecem no *cluster* com processamento mais lento. Esse último grupo não permite ganhos de desempenho devido a própria modelagem BSP. Essa situação muda no momento de testar o décimo pf. Esse teste leva em questão a migração dos 10 processos para o *cluster* mais rápido e gera um ganho de 31.13% ao comparar os cenários i e iii (ver Figura 5).

Os processos do *cluster* C são movidos para o A nos testes que envolvem 25 processos e um domínio de 10. Nesse caso, os 20 outros processos ficam no *cluster* L porque não existem nós livres suficientes no *cluster* mais rápido. Uma possibilidade é explorar dois processos em um nó do *cluster* A (cada nó possui 2 GHz) mas AutoMig não considera essa ideia pois cada nó do *cluster* L possui 1.2 GHz. Considerando o aumento no número de domínios, as migrações com 100 processos se tornam viáveis e representam uma melhora de 14.95% no desempenho. Por outro lado, o mapeamento inicial de 200 processos ainda permanece na mesma posição e acarreta uma sobrecarga de 7.64% na execução da aplicação. Tal percentagem é obtida ao comparar os cenários i e ii.

Os testes anteriores deixam claro que quanto maior o peso de computação por processo, melhores serão os ganhos com o reescalonamento de processos. Nesse caminho, testou-se AutoMig com domínio e intervalo menores como expressado na Tabela 2. Esta tabela mostra o comportamento para 10 e 25 processos. Ganhos de cerca de 31.6% e 19.81% foram obtidos ao aplicar AutoMig. Em adição, foi verificada uma sobrecarga menor que 1%. Analisando os resultados, quanto maior o tempo da aplicação, menor o impacto de AutoMig sobre ela. Ainda, verificou-se que os benefícios com as migrações se tornam praticamente constantes se forem comparadas com as execuções com domínio 10 e 4. Ao dobrar o número de processos, o tempo da aplicação não cai pela metade. Existe um limite onde a inclusão de novos processos não gera ganhos no tempo de execução. Considerando a questão da escalabilidade, MigBSP (com AutoMig) obteve comportamentos similares se comparado com aqueles capturados do cenário i (ver Figuras 3 e 5).

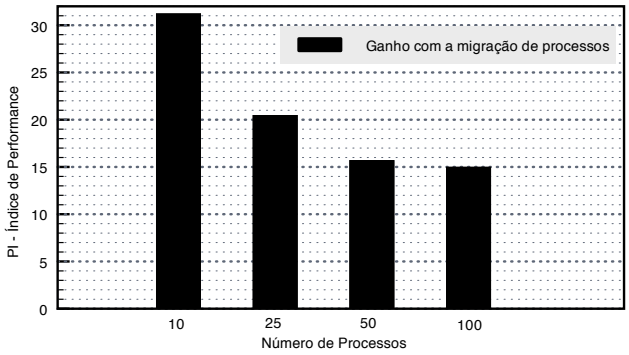


Figura 5. Ganhos com domínio 10. Índice PI = $(\frac{cenario\ i - cenario\ iii}{cenario\ i} * 100)$

Tabela 2. Resultados com 4 e 2 para domínio e intervalo (tempo em segundos)

Processos	Cenário i	Heurística Velha		AutoMig	
		Cenário ii	Cenário iii	Cenário ii	Cenário iii
10	12500.51	12511.87	9191.72	12523.22	8555.29
25	6250.49	6257.18	5311.54	6265.38	5011.77

A Tabela 2 também mostra uma análise comparativa das duas heurísticas de seleção de MigBSP. A título de simplificação, foi nomeada a heurística que seleciona um processo para migração como "Heurística Velha". Embora ambos tenham obtido bons índices de desempenho, AutoMig se sobressai em relação a outra em aproximadamente 8%. Por exemplo, 5 processos são migrados já na primeira tentativa de migração quando 25 processos são testados. Neste caso, todos os processos que estavam executando no *cluster* C são passados para o *cluster* A. Esta reorganização sugerida por AutoMig no começo da aplicação fornece um menor tempo para conclusão da aplicação. Por outro lado, 5 chamadas de reescalonamento são necessárias para atingir a mesma configuração com a Heurística Velha. Por fim, a heurística AutoMig impõe maiores sobrecargas se comparado com a sua antecessora (próximo a 1%). Esta situação foi esperada pois duas comunicações *multicast* entre os Gerentes são realizadas por AutoMig em seu algoritmo.

6. Trabalhos Relacionados

Vadhiyar e Dongarra [Vadhiyar and Dongarra 2005] apresentam suporte a migrações e adaptatividade no sistema GrADS. O custo de migração é considerado como um valor fixo e os ganhos com o reescalonamento são baseados no tempo restante de execução. Assim, GrADS trabalha com aplicações cujas partes sejam conhecidas previamente. Em adição, o mesmo problema é visto em dois trabalhos relacionados. Sanjay e Vadhiyar [Sanjay and Vadhiyar 2009] apresentam um algoritmo de escalonamento que considera um gráfico 3D unindo informações de CPU e largura de banda para a seleção dos recursos com maior carga atual. O segundo trabalho compreende os esforços de Ding [Ding et al. 2009]. Ele criou uma heurística para partição de tarefas que minimiza o tempo de conclusão da carga de trabalho. Sua abordagem conhece o tamanho do problema antes da execução do programa. Alternativamente, AutoMig usa somente dados coletados em tempo de execução para suas tomadas de decisões.

Chen et al. [Chen et al. 2008] propuseram remapeamento de processos com redução do custo. A heurística desenvolvida permite somente movimentos de processos entre a máquina com maior carga e uma outra. Além disso, diferente de AutoMig, a

solução desses autores não considera as ações de comunicação para a seleção. Silva et al. [da Silva e Silva et al. 2010] expuseram o gerenciamento de recursos sobre o *middleware* InteGrade. Eles apresentaram um *grid* como uma coleção de *clusters*, onde cada qual possui o seu próprio gerente (CM - *Cluster Manager*). Análogo a proposta de AutoMig, cada CM é responsável por capturar dados de um *cluster* e trocá-los com os demais CMs.

Quanto a migração em aplicações BSP, destacam-se as bibliotecas PUBWCL e PUB. PUBWCL tira proveito de ciclos ociosos de computadores na Internet [Bonorden et al. 2005]. Seus algoritmos usam dados sobre os recursos de computação, bem como o tempo de processamento dos processos. A PUB [Bonorden 2007], por sua vez, propõe estratégias de balanceamento centralizada e distribuída. Na centralizada, todos os nós enviam dados sobre seu poder de CPU para um mestre. Esse verifica as informações e seleciona um processo para transferência. Na abordagem distribuída, cada nó escolhe c outros nós aleatóriamente e os questiona sobre a sua carga. Um processo é migrado se a carga mínima dos c nós analisados for menor que a carga do nó que realizou o teste. Os pontos negativos de ambas abordagens são: (i) uso isolado de dados de computação; (ii) um processo é migrado por ativação do rebalanceamento.

7. Conclusão

Considerando que o estilo de computação em fases de BSP é uma organização frequente na escrita de programas paralelos de sucesso [Bonorden 2007, De Grande and Boukerche 2011, Hendrickson 2009], AutoMig emerge como uma alternativa para selecionar seus processos para executar em recursos mais apropriados sem a interferência de desenvolvedores. A principal contribuição de AutoMig aparece na sua função de predição pf. Ela é aplicada para o escalonamento corrente bem como para cada nível da lista baseada no PM dos processos. Cada elemento dessa lista informa um novo escalonamento através do incremento de uma reposição de processo. pf considera a carga nos Conjuntos e na rede, estima os processos mais lentos considerando suas atividades de computação e comunicação e adiciona o tempo de transferência do processo testado. O esquema de balanceamento de carga de AutoMig usa uma abordagem global, onde são usados dados de todos os processos. Ao invés de pagar um custo adicional de sincronização para adquirir informações de escalonamento, AutoMig tira proveito do conceito de superetapa do próprio modelo BSP, onde sempre há uma barreira após ações de comunicação entre os processos.

AutoMig e uma aplicação foram desenvolvidos usando o simulador Simgrid. A aplicação trabalha com um parâmetro chamado domínio que torna possível a criação de situações de carga mais facilmente. Uma vez que a aplicação é CPU-Bound, quanto menor o tamanho do domínio, maior o tempo de execução da aplicação e a probabilidade de migração. Os resultados provaram isso, indicando ganhos de até 17.15% e 31.13% para domínios igual a 20 e 10. Particularmente, os resultados revelaram um dos principais pontos positivos de AutoMig na seleção dos processos a migração. Ele pode eleger todos os processos de um *cluster* devagar para executar rapidamente em outro mais apropriado. Entretanto, muitas vezes o *cluster* mais rápido possui poucos nós livres do que o número de candidatos a migração. Assim, AutoMig demonstrou que as migrações são inviáveis nessa situação, devido as regras de execução de uma superetapa BSP. Uma superetapa BSP possui uma barreira que sempre espera pelo processo mais lento (neste caso, os processos que irão permanecer no *cluster* mais lento).

Por fim, trabalhos futuros compreendem o uso de AutoMig em um *middleware* para a computação em nuvem. Este *middleware* irá trabalhar no auto-dimensionamento de recursos para execução de aplicações paralelas. Considerando que cada aplicação tem seu próprio SLA (*Service Level Agreement*), AutoMig surge como a primeira iniciativa para reescalonamento em tempo de execução quando o SLA falha. Caso as migrações não resolverem a questão do desempenho, mais recursos são alocados num segundo momento.

Referências

Baritompa, W., Bulger, D. W., and Wood, G. R. (2007). Generating functions and the performance of backtracking adaptive search. *J. of Global Optimization*, 37:159–175.

Bhandarkar, M. A., Brunner, R., and Kale, L. V. (2000). Run-time support for adaptive load balancing. In *IPDPS '00: Proceedings of the 15 IPDPS 2000 Workshops on Parallel and Distributed Processing*, pages 1152–1159, London, UK. Springer-Verlag.

Bonorden, O. (2007). Load balancing in the bulk-synchronous-parallel setting using process migrations. In *Int. Parallel and Distrib. Processing Symp.*, 2007,pages 1–9.IEEE.

Bonorden, O., Gehweiler, J., and auf der Heide, F. M. (2005). Load balancing strategies in a web computing environment. In *Proceeedings of International Conference on Parallel Processing and Applied Mathematics (PPAM)*, pages 839–846, Poznan, Poland.

Casanova, H., Legrand, A., and Quinson, M. (2008). Simgrid: A generic framework for large-scale distributed experiments. In *Tenth International Conference on Computer Modeling and Simulation (uksim)*, pages 126–131, Los Alamitos, CA, USA. IEEE.

Chen, L., Wang, C.-L., and Lau, F. (2008). Process reassignment with reduced migration cost in grid load rebalancing. *Parallel and Distrib. Processing*, 2008, pages 1–13,IEEE.

da Rosa Righi, R., Pilla, L. L., Carissimi, A., Navaux, P. A., and Heiss, H.-U. (2010). Observing the impact of multiple metrics and runtime adaptations on bsp process rescheduling. *Parallel Processing Letters*, 20(2):123–144.

da Silva e Silva, F. J., Kon, F., Goldman, A., Finger, M., de Camargo, R. Y., Filho, F. C., and Costa, F. M. (2010). Application execution management on the integrade opportunistic grid middleware. *J. Parallel Distrib. Comput.*, 70(5):573–583.

De Grande, R. E. and Boukerche, A. (2011). Dynamic balancing of communication and computation load for hla-based simulations on large-scale distributed systems. *J. Parallel Distrib. Comput.*, 71:40–52.

Ding, D., Luo, S., and Gao, Z. (2009). A dual heuristic scheduling strategy based on task partition in grid environments. In *Int. Joint Conference on Computational Sciences and Optimization*, pages 63–67, IEEE.

Duselis, J., Cauich, E., Wang, R., and Scherson, I. (2009). Resource selection and allocation for dynamic adaptive computing in heterogeneous clusters. In *Cluster Computing and Workshops, 2009. CLUSTER '09. IEEE International Conference on*, pages 1 –9.

El Kabbany, G., Wanas, N., Hegazi, N., and Shaheen, S. (2011). A dynamic load balancing framework for real-time applications in message passing systems. *International Journal of Parallel Programming*, 39:143–182. 10.1007/s10766-010-0134-5.

Guo, Y., Chen, X., Deng, M., Wang, Z., Lv, W., Xu, C., and Wang, T. (2009). The fractal compression coding in mobile video monitoring system. In *WRI International Conference on Comm. and Mobile Computing*, pages 492–495, IEEE.

Hendrickson, B. (2009). Computational science: Emerging opportunities and challenges. *Journal of Physics: Conference Series*, 180(1):012013.

Huang, C., Zheng, G., Kale, L., and Kumar, S. (2006). Performance evaluation of adaptive mpi. In *PPoPP '06: Proceedings of the eleventh ACM SIGPLAN symposium on Principles and practice of parallel programming*, pages 12–21, ACM Press.

Kwok, Y.-K. and Cheung, L.-S. (2004). A new fuzzy-decision based load balancing system for distributed object computing. *J. Parallel Distrib. Comput.*, 64(2):238–253.

Moreno-Vozmediano, R. and Alonso-Conde, A. B. (2005). Influence of grid economic factors on scheduling and migration. In *High Perf. Comp. for Computational Science*, volume 3402 of *Lecture Notes in Comp. Science*, pages 274–287. Springer.

Nascimento, A. P., Sena, A. C., Boeres, C., and Rebello, V. E. F. (2007). Distributed and dynamic self-scheduling of parallel mpi grid applications: Research articles. *Concurr. Comput. : Pract. Exper.*, 19(14):1955–1974.

Pascual, J. A., Navaridas, J., and Miguel-Alonso, J. (2009). Job scheduling strategies for parallel processing. chapter Effects of Topology-Aware Allocation Policies on Scheduling Performance, pages 138–156. Springer-Verlag, Berlin, Heidelberg.

Qin, X., Jiang, H., Manzanares, A., Ruan, X., and Yin, S. (2010). Communication-aware load balancing for parallel applic. on clusters. *IEEE Trans. Comput.*, 59(1):42–52.

Sanjay, H. A. and Vadhiyar, S. S. (2009). A strategy for scheduling tightly coupled parallel applications on clusters. *Concurr. Comput. : Pract. Exper.*, 21(18):2491–2517.

Silva, R. E., Pezzi, G., Maillard, N., and Diverio, T. (2005). Automatic data-flow graph generation of mpi programs. In *SBAC-PAD '05: Proceedings of the 17th International Symposium on Computer Architecture on High Performance Computing*, pages 93–100, Washington, DC, USA. IEEE Computer Society.

Tanenbaum, A. (2003). *Computer Networks*. Prentice Hall, New Jersey, 4th ed.

Vadhiyar, S. S. and Dongarra, J. J. (2005). Self adaptivity in grid computing: Research articles. *Concurr. Comput. : Pract. Exper.*, 17(2-4):235–257.

Xhafa, F. and Abraham, A. (2010). Computational models and heuristic methods for grid scheduling problems. *Future Gener. Comput. Syst.*, 26(4):608–621.

Xing, C. (2008). An adaptive domain pool scheme for fractal image compression. *Education Technology and Training and Geoscience and Remote Sensing*, 2:719–722.

Yang, C.-T. and Chou, K.-Y. (2009). An adaptive job allocation strategy for heterogeneous multiple clusters. In *Int. Conf. on Computer and Information Technology - Volume 02*, pages 209–214, IEEE.

Zaki, M. J., Li, W., and Parthasarathy, S. (1997). Customized dynamic load balancing for a network of workstations. *J. Parallel Distrib. Comput.*, 43(2):156–162.

Avaliação de um Modelo de Consumo Energético Aplicado ao Mapeamento de Processos*

Edson L. Padoin, Eduardo H. M. Cruz, Francieli Z. Boito, Francis B. Moreira, Laércio L. Pilla, Rodrigo V. Kassick, Philippe O. A. Navaux

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{elpadoin, ehmcruz, fzboito, fbmoreira,
llpilla, rvkassick, navaux}@inf.ufrgs.br

Abstract. *Nowadays, energy consumption arises as an essential aspect to the new generations of processor architectures and large-scale HPC systems. In such context, there is high demand for studies on how and where energy is spent on a computer system and which techniques can be used to increase it's efficiency. Energy-consumption models try to ease this analysis by providing theoretical values, but their precision is still not well studied. Therefore, this paper presents a study of the precision of an energy-consumption model for architecture with hierarchical cache and its appliance on the context ao process-mapping techniques on a real system.*

Resumo. *A redução do consumo energético é hoje um dos grandes desafios da computação, essencial para viabilizar as novas gerações de computadores e arquiteturas paralelas de grande escala. Nesse contexto, é fundamental o entendimento de onde e como a energia é aplicada, e quais técnicas podem ser empregadas para aumentar sua eficiência. A modelagem do consumo de energia visa facilitar esta análise, porém sua precisão com relação a arquiteturas reais ainda é pouco explorada. Sendo assim, este artigo visa estudar um modelo de consumo energético para arquiteturas com hierarquia de cache, de forma a aplicá-lo no contexto de mapeamento de processo em uma arquitetura real.*

1. Introdução

Sistemas de computação de alto desempenho são frequentemente avaliados por sua capacidade de computação – quantidades de operações por segundo, velocidade de comunicação, tempo para uma aplicação completar sua execução, dentre outros. No entanto, recentemente, há um interesse crescente no consumo energético destas arquiteturas e quão eficientes elas são nesse aspecto [Hsu and Feng 2005]. Isto se torna ainda mais evidente quando são levadas em consideração as características de novas arquiteturas de clusters que são propostas para os próximos anos, que têm por objetivo atingir o patamar de 10^{18} operações de ponto flutuante por segundo (exa-flop) [Donofrio et al. 2009].

Dadas as previsões de tecnologia disponível [Kogge et al. 2008] e a grande escala destes sistemas, os custos energéticos com processadores, armazenamento, rede de interconexão e refrigeração necessária são inviáveis. Portanto, torna-se essencial o

*Trabalho parcialmente apoiado por CNPq, CAPES, FAPERGS e FINEP

estudo do consumo energético de arquiteturas paralelas sob diferentes cargas de trabalho e, assim, investigar formas de aumentar sua eficiência energética. Nas arquiteturas de processadores *multicore*, por exemplo, o subsistema de memória se apresenta como o provedor de dados aos processadores, sendo um ponto essencial ao desempenho. Por esta razão, ele é constantemente adaptado às necessidades computacionais [Alves et al. 2009] para esconder a diferença de desempenho entre memórias e processadores [Hennessy and Patterson 2007].

A energia consumida pela hierarquia de memória é parte importante do consumo total do sistema. Portanto, uma otimização na hierarquia de memória e interconexão possui um impacto importante. Técnicas simples, como o ajuste de parâmetros das memória cache para uma aplicação específica, podem apresentar uma economia de até 60% do consumo de energia [Herranz and Moreno-Navarro 2003], podendo ser facilmente empregadas em processadores de uso específico. Entretanto, mecanismos mais sofisticados devem ser empregados em processadores de propósito geral, a fim de atingir um equilíbrio entre o desempenho e o custo energético.

O mapeamento de processos é uma técnica que pode ser aplicada de forma a aprimorar a afinidade de memória do sistema, adaptando o mapeamento da aplicação à arquitetura. Um dos parâmetros utilizados para o mapeamento de processos é o compartilhamento de dados entre as tarefas. Com isto, é possível guiar o posicionamento de tarefas para otimizar a utilização de caches compartilhadas e reduzir os sobrecustos das latências de comunicações remotas. Uma vez que o mapeamento melhora o desempenho de uma aplicação, o consumo de energia também sofrerá modificações. Surge então a necessidade de explorar como técnicas de otimização influenciam o consumo energético do sistema.

Porém, medições nem sempre podem ser feitas diretamente em sistemas reais, seja por falta de acesso físico ao recurso ou por este ainda estar em fase de projeto. Nestes casos, modelos de consumo energético podem ser aplicados para estimar o desempenho energético de aplicações, algoritmos e sistemas. Entretanto, estes modelos de consumo energético precisam ser avaliados para confirmar suas precisões e, consequentemente, aplicabilidade.

Neste contexto, este artigo apresenta um estudo de um modelo de consumo energético aplicado ao mapeamento de processos em processadores atuais, de forma a avaliar a precisão do modelo e o impacto do mapeamento de processos sobre o consumo energético da arquitetura. Por ter sido desenvolvido para uma arquitetura muito mais simples que as atuais, o modelo teve que ser adaptado para utilização em uma hierarquia de memória.

O restante do artigo está divido da seguinte maneira: a Seção 2 apresenta o modelo de consumo energético considerado. A Seção 3 introduz conceitos de mapeamento de processos. O método de avaliação utilizado é apresentado na Seção 4 e os resultados obtidos são abordados na Seção 5. Trabalhos relacionados são tratados na Seção 6. Por fim, a Seção 7 apresenta as principais contribuições deste artigo e trabalhos futuros.

2. Modelo de Consumo Energético da Hierarquia de Memória

Modelos especializados para diferentes componentes do sistema são necessários para a manutenção da precisão. Um exemplo disso é o modelo de consumo energético por

memória cache apresentado em [Shiue and Chakrabarti 1999]. Neste modelo, o consumo é diferenciado pela sua fonte, podendo esta ser das células de memória (E_{cell}), do decodificador (E_{dec}) ou da entrada e saída de dados (E_{io}). Assim, o consumo gerado por um dado encontrado na cache (*hit*) depende de E_{cell} e E_{io}, enquanto o de um dado não encontrado (*miss*) depende do custo de um acesso à memória principal.

Um novo modelo foi refinado em [Kim et al. 2000] considerando também os acessos de escrita. Essa nova versão do modelo é utilizada para comparação do consumo energético da hierarquia de memória de uma arquitetura multicore:

$$
\begin{aligned}
Energy &= E_{bus} + E_{cell} + E_{pad} + E_{main} \\
E_{bus} &= E_{add_bus} + E_{data_bus} \\
E_{cell} &= \beta \times (Word_line_size) \times (Bit_line_size + 4,8) \times (N_{hit} + 2 \times N_{miss}) \\
E_{pad} &= E_{add_pad} + E_{data_pad} \\
E_{main} &= Em \times 8L \times N_{miss} \times (1 + dirty_r) \\
E_{add_bus} &= 0,5 \times 10^{-12} \times Pr1 + V^2 \times (N_{hit} + N_{miss}) \times W_{add} \\
E_{data_bus} &= 0,5 \times 10^{-12} \times Pr2 + V^2 \times (N_{hit} + N_{miss}) \times 32 \\
E_{add_pad} &= 20 \times 10^{-12} \times Pr3 + V^2 \times N_{miss} \times W_{add} \\
E_{data_pad} &= 20 \times 10^{-12} \times Pr4 + V^2 \times (1 + dirty_r) \times N_{miss} \times 64 \\
Word_line_size &= m \times (8L + T + St) \\
Bit_line_size &= C/(m \times L)
\end{aligned}
$$

onde C = tamanho da cache; L = tamanho da linha da cache; m = associatividade; T = *tag size* em bits; St = número de bits de *status* por bloco; N_{hit} = número de *hits*; N_{miss} = número de *misses*; W_{add} = largura do barramento de endereço; $dirty_r$ = porcentagem de blocos escritos de volta à memória na substituição; Em = energia consumida por um acesso à memória principal; V = *Voltage level*; *Pr1, Pr2, Pr3 e Pr4* = taxa de troca de bits nos barramentos; e β é uma constante que depende da tecnologia empregada.

Esse modelo não considera a existência de mais de um nível de cache, relacionando o gasto gerado pelas faltas na cache com o acesso à memória principal. Adicionalmente, adaptações são necessárias para a existência de mais de um banco de memória cache para um determinado nível (por exemplo, cada núcleo do processador pode ter uma cache L1 privada e compartilhar a L2 com os demais). Assim, nesse trabalho, é utilizado o seguinte modelo adaptado:

$$
\begin{aligned}
Energy &= Energy'(L_1) \\
Energy'(L_n) &= Energy'(L_{n+1}) + \sum_{i=0}^{i<qt(L_n)} (E_{bus}(L_{ni}) + E_{cell}(L_{ni}) + E_{pad}(L_{ni})) \\
Energy'(L_m) &= \sum_{i=0}^{i<qt(L_{m-1})} E_{main}(L_{m-1,i})
\end{aligned}
$$

Existem $(m+1)$ níveis de cache (L_1, L_2, ..., L_{m-1}); $qt(L_n)$ significa o número de caches no nível n; L_{ni} é a i-ésima memória cache de nível n; e L_m é a memória principal.

3. Mapeamento de Processos

O mapeamento de processos é uma técnica que possibilita o aumento de desempenho em aplicações paralelas através de uma alocação mais eficiente dos recursos. Uma característica importante a ser explorada nos sistemas multiprocessados é o tempo de comunicação entre núcleos – o tempo necessário para que um processador escreva em uma variável compartilhada e a alteração ser propagada para outro núcleo. Esse tempo de comunicação é frequentemente heterogêneo, pois depende da localização dos núcleos envolvidos. Este tempo é também influenciado pela hierarquia de memória cache, já que núcleos podem possuir compartilhamento de diferentes níveis de memória cache ou podem estar localizados em diferentes processadores [Cruz et al. 2011b].

Uma maneira de diminuir o tempo de execução de uma aplicação paralela é mapear threads que compartilham memória em núcleos que compartilham cache [Alves et al. 2009], reduzindo a latência de acesso à dados compartilhados. Além disso, é mais vantajoso manter tarefas comunicantes em núcleos em um mesmo socket do que em processadores distintos. Isso se deve não somente às latências já mencionadas, mas também porque isso reduz a sobrecarga imposta por protocolos de coerência.

A diminuição no tempo de execução das aplicações tende a resultar na diminuição do consumo energético do sistema. Adicionalmente, com um menor número de dados e mensagens de protocolos de coerência trafegando nas interconexões, bem como menos linhas duplicadas em caches distintas, uma diminuição no consumo de potência também é esperada. Isso pode ser obtido através do mapeamento eficiente de processos, tendo em vista que tais componentes são responsáveis por grande parte do consumo energético dos processadores atuais [Kumar et al. 2005].

O mapeamento pode ser feito de maneira dinâmica ou estática. No mapeamento dinâmico [Tam et al. 2007], a cada etapa de processamento as threads devem ser remapeadas para se adequarem ao novo padrão de compartilhamento de dados. Diferentemente, o mapeamento é predefinido no caso estático [Cruz et al. 2010], ou seja, uma vez iniciada a computação, o mapeamento não será mais modificado. Em ambos os casos, a decisão sobre o mapeamento deve ser tomada com base nas informações fornecidas pelo programador, pelo compilador ou outras técnicas que forneçam informações sobre acessos à memória. Neste trabalho, é adotada a política de mapeamento estático baseada em traços de memória capturados em execuções anteriores.

4. Método de Avaliação

Para analisar o consumo energético tanto no modelo apresentado na Seção 2 quanto em um caso real, foram utilizados os benchmarks da família NAS-NPB [Jin et al. 1999] versão 3.3.1 paralelizados com a interface OpenMP. Estes benchmarks foram escolhidos por representarem diversas famílias de aplicações paralelas. Foram executadas as aplicações CG, MG, EP, SP, LU, IS, FT e UA em suas instâncias de tamanho A e B.

Para cada teste, foram avaliadas as estratégias de mapeamento do sistema operacional (**Padrão**) e dois mapeamentos heurísticos baseados em quantidade de acessos a regiões de memória compartilhada (**Quantidade de Acessos**) e em quantidade de memória compartilhada (**Quantidade de Memória**). Para os mapeamentos heurísticos, foram avaliados os melhores e piores mapeamentos.

4.1. Consumo Energético na Arquitetura Real

Para a avaliação em um ambiente real foi utilizado um nó de processamento Dell Poweredge R710 com dois processadores Xeon E5530 de 2.40GHz. Cada processador contém quatro núcleos capazes de execução simultânea de duas threads (HT). A arquitetura desta máquina é apresentada na Figura 1.

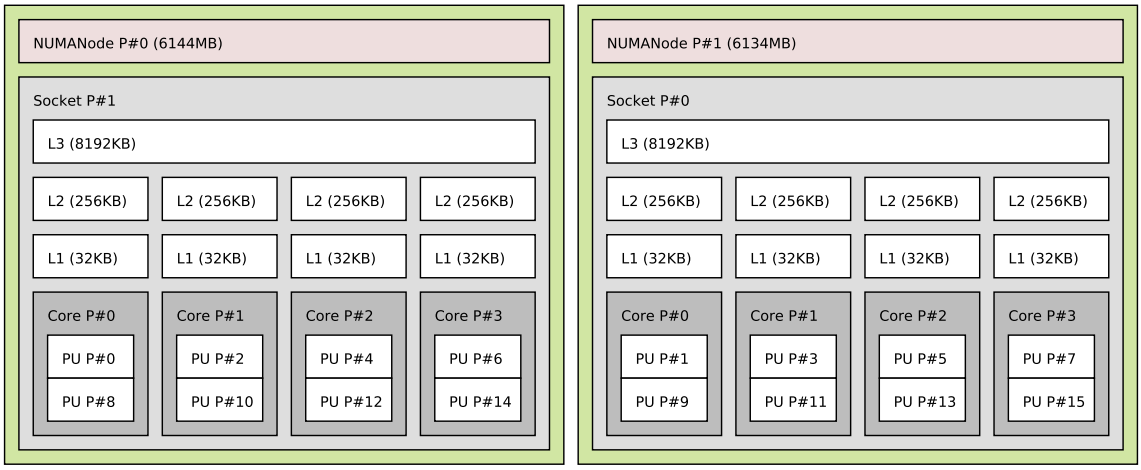


Figura 1. Arquitetura da máquina de teste

A medição de consumo foi feito com a interface IPMI (*Intelligent Platform Management Interface*) através da ferramenta FreeIPMI[1]. Foram feitas quatro medições por segundo durante 10 execuções de cada benchmark.

4.2. Alimentação do Modelo de Consumo Energético

Para obter os dados necessários ao modelo, os benchmarks foram novamente executados no ambiente previamente descrito e foram capturadas informações do processador com o auxílio da ferramenta Vtune[2], extraindo valores para contadores de *hits* e *misses* de cache, entre outros. Os dados restantes foram obtidos através da especificação da arquitetura e da ferramenta CACTI[3] versão 5.3.

Dado que não foi utilizado nenhum simulador para a execução, as taxas de troca de bits (Pr1, Pr2, Pr3 e Pr4 no modelo, como descrito na Seção 2) não puderam ser determinadas. Dessa forma, foi assumido um valor de 25% para elas, de forma semelhante ao feito em [Kim et al. 2000] [Hicks et al. 1997].

A constante β, que depende da tecnologia de integração, também não pode ser determinada. Por isso, seu valor foi escolhido conforme o especificado em [Kim et al. 2000], valendo 1.44×10^{-14}. Portanto, foi assumida uma previsão teórica pessimista em seu valor bruto. Porém, o erro adicionado por isso age de forma uniforme em todas as medidas e não compromete as comparações.

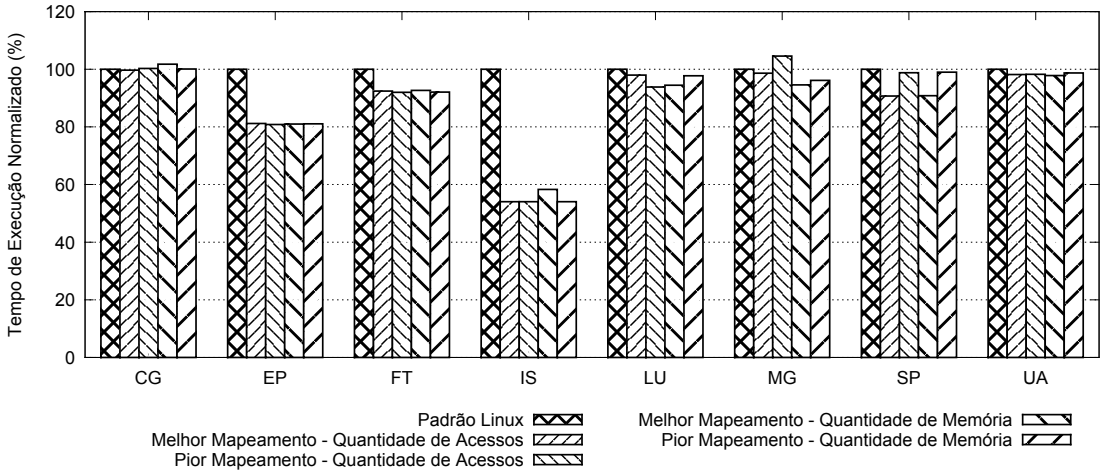
5. Resultados

A Figura 2 contém os tempos de execução normalizados dos benchmarks NPB para os tamanhos de entrada A e B. A normalização dos tempos foi feita em relação ao tempo

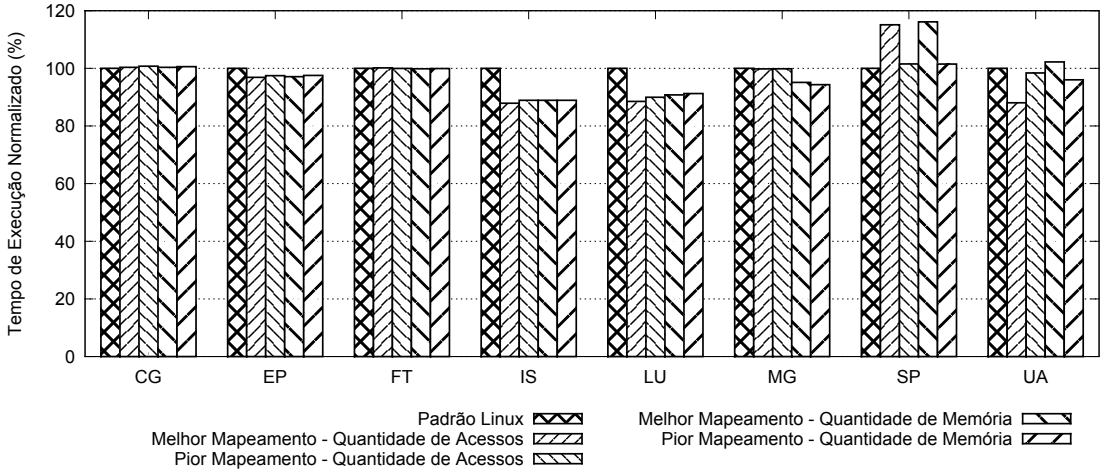
[1]http://www.gnu.org/software/freeipmi/

[2]http://software.intel.com/en-us/articles/intel-vtune-amplifier-xe/

[3]http://quid.hpl.hp.com:9081/cacti/



(a) Tamanho A.



(b) Tamanho B.

Figura 2. Tempos de execução dos benchmarks NPB normalizados.

de execução original de cada benchmark. Duas diferentes heurísticas de mapeamento foram usadas, onde a diferença entre elas está na métrica utilizada na tomada de decisões: uma é baseada na quantidade de acessos realizados à memória compartilhada enquanto a outra é baseada na quantidade de memória compartilhada. Além disso, na Figura 2, são apresentados o melhor e o pior mapeamento estático feito com cada heurística.

Os resultados mostram que o escalonador original do sistemas operacional Linux possui desempenho inferior às políticas de mapeamento experimentadas para todos os benchmarks, com exceção do benchmark SP com tamanho B. Isto acontece porque o escalonador do Linux realiza diversas migrações de tarefas entre os núcleos disponíveis a cada região paralela da aplicação, o que resulta em sobrecustos ligados a uma maior taxa de faltas na cache, já que os dados necessitam ser movimentados entre as caches. O mesmo comportamento pode ser observado em [Cruz et al. 2010].

Entretanto, as diferenças entre os tempos de execução entre o pior e melhor mapeamento apresentam um padrão diferente do esperado. Em várias situações, o pior mapeamento obtém um ganho de desempenho superior ao do melhor mapeamento. A razão para estes resultados divergentes é em função de a máquina utilizada para os experimentos possui apenas um nível na hierarquia de memória cache. Quando mais níveis compartilhados estão presentes, maior é o ganho de desempenho, como pode ser verificado em [Cruz et al. 2011a], onde ganhos de até 75% foram obtidos ao se utilizar técnicas de mapeamento em máquinas NUMA.

	A					B				
	Padrão	**Quantidade de Acessos**		**Quantidade de Memória**		**Padrão**	**Quantidade de Acessos**		**Quantidade de Memória**	
		melhor	**pior**	**melhor**	**pior**		**melhor**	**pior**	**melhor**	**pior**
cg	0,75	0,27	0,20	0,19	0,27	14,07	14,10	14,12	13,95	14,01
ep	1,48	1,50	1,50	1,49	1,50	6,01	6,00	5,94	6,01	5,94
ft	1,02	0,92	0,92	0,93	0,84	11,06	11,04	11,05	11,05	10,97
is	0,32	0,12	0,19	0,19	0,12	0,65	0,65	0,65	0,66	0,65
lu	9,78	6,15	6,23	6,13	6,30	29,08	27,34	27,64	27,62	27,76
mg	0,61	0,56	0,49	0,57	0,49	2,85	1,77	1,76	1,76	1,84
sp	8,74	7,88	7,95	7,89	8,39	48,96	47,65	47,78	49,76	52,28
ua	9,88	6,22	6,37	6,21	6,44	30,90	30,35	30,56	30,27	30,58

Tabela 1. Consumo (Watts-hora) para execução do NPB com mapeamento em máquina real

As Tabelas 1 e 2 apresentam os valores de consumo de energia e de potência média para a execução dos testes cujos resultados foram apresentados na Figura 2. Para os testes com tamanho A, houve em alguns casos uma diminuição no consumo, apesar de os resultados de desempenho apresentarem pouca diferença em tempo de execução.

Nos testes com tamanho B, no entanto, onde a variação de consumo não foi desprezível, há pouca correlação com as diferenças nos tempos de execução. O teste IS, por exemplo, teve seu tempo de execução diminuído para aproximadamente 90% do tempo obtido com o escalonador padrão do SO. Seu consumo de energia, no entanto, manteve-se idêntico para todos os casos. Já o teste SP apresentou diferenças em seu consumo energético em função do maior ou menor tempo de execução.

Estes resultados levam a crer que, apesar da pouca diferença em desempenho ob-

	A					B				
	Padrão	**Quantidade de Acessos**		**Quantidade de Memória**		**Padrão**	**Quantidade de Acessos**		**Quantidade de Memória**	
		melhor	**pior**	**melhor**	**pior**		**melhor**	**pior**	**melhor**	**pior**
cg	226,49	192,03	176,13	170,00	240,69	249,58	252,58	252,88	248,53	252,20
ep	265,92	269,33	269,46	268,03	269,71	267,19	269,79	270,79	270,59	270,59
ft	245,83	253,93	253,56	258,00	253,16	256,84	258,11	258,33	258,30	256,52
is	228,94	141,17	225,87	228,11	211,44	233,78	233,60	235,73	237,49	235,64
lu	255,15	283,83	283,91	282,85	283,59	276,18	283,66	282,66	281,73	282,30
mg	243,26	252,67	250,08	258,06	221,98	244,70	264,98	253,50	254,16	264,33
sp	262,26	267,46	267,57	268,03	264,99	256,95	257,94	257,86	255,93	255,05
ua	254,11	269,75	269,77	269,55	269,45	266,81	268,49	268,32	268,39	268,47

Tabela 2. Potência Média para execução do NPB com mapeamento em máquina real

tida com o mapeamento, o consumo de energia pode ser significativamente afetado por outras variáveis externas ao nó de processamento, sendo necessárias maneiras mais refinadas para avaliar o impacto de otimizações no consumo de energia.

Os valores obtidos com o modelo teórico no tamanho B do benchmark são apresentados na Figura 3. Nota-se que os resultados obtidos possuem escala de grandeza maior do que foi observado com IPMI. Conforme discutido anteriormente, isso era esperado devido aos parâmetros do modelo que foram estimados usando valores adequados à situação tecnológica de mais de 10 anos atrás.

A Figura 4 apresenta os mesmos dados que a 3(b), porém normalizados a partir do valor real de consumo do benchmark SP para facilitar a comparação. O modelo mostra, assim como nas medidas reais, pouca diferença entre os consumos das estratégias de mapeamento.

No entanto, é possível notar que as relações entre o consumo de diferentes benchmarks são quase totalmente mantidas, sendo SP o teste com maior consumo e IS o de menor valor. A única exceção observada foi para o caso do teste MG que, no modelo teórico apresentou consumo mais elevado que o EP quando o inverso foi observado no caso real.

6. Trabalhos Relacionados

O consumo energético de sistemas computacionais tem sido alvo de pesquisa, principalmente em ambientes de alto desempenho. Dentre as diferentes abordagens para o aumento da eficiência energética, estão o controle de voltagem e frequência (*Dynamic Voltage and Frequency Scaling*, DVFS) dos processadores e a aplicação de otimizações nas hierarquias de memória. Para estimar o consumo desses sistemas, usualmente são empregados contadores de hardware [Bellosa 2000, Bircher et al. 2005].

Em [Gordon-Ross et al. 2005] é discutida a redução do espaço de busca para otimização da melhor configuração de cache em relação ao consumo de energia. Já em [Silva-Filho et al. 2006], onde é utilizado o mesmo modelo, é analisado, além do consumo de energia, o desempenho computacional obtido com as novas configurações de memória. Abordagem semelhante é encontrada em [Kim et al. 2000], onde são relacionados trabalhos que apresentam técnicas para diminuição do consumo de energia em

memória cache.

No trabalho de [Hicks et al. 1997] é analisado como o aumento da complexidade das caches, com configurações diferentes de associatividade, tamanho do bloco e tamanho de cache apresentam impacto na energia consumida. A taxa de acerto também é analisada frente a energia consumida. Os autores concluem que as taxas de acerto também desempenham um papel chave na avaliação da eficiência energética. Neste trabalho também é apresentado equações de um modelo de consumo energético que leva em consideração as células de memória, o decodificador e a entrada e saída.

Os autores de [Zhang and Vahid 2003] buscam estimar a energia consumida para uma determinada configuração de cache, envolvendo diferentes tamanhos de blocos e associatividade. Exploram, a partir de equações o consumo de energia, para um conjunto de configurações que impactam no desempenho e tem influência no consumo.

No artigo [Zarandi and Miremadi 2005] novas caches são propostas para melhorar as taxas de sucesso em relação aos regimes convencionais. O mapeamento proposto é simulado tendo como base alguns padrões de rastreamento. Os resultados apontam um aumanto na quantidade de *hits* em relação aos esquemas de mapeamentos atuais, minimizando assim o consumo de energia.

7. Conclusão

Devido às características e tamanho das arquiteturas propostas para os próximos anos, há um crescente interesse na avaliação e controle do consumo energético dos sistemas computacionais. Uma forma de fazê-lo acontece através da modelagem dos fenômenos elétricos que acontecem nestas máquinas. Porém, pouco esforço tem sido colocado na validação destes modelos.

Nesse contexto, este trabalho apresentou os resultados da avaliação do consumo energético da hierarquia de memória de uma arquitetura multicore e sua comparação com um modelo de consumo energético teórico. Para tanto, alguns parâmetros do modelo precisaram ser adaptados para o mesmo ser aplicável às hierarquias de memórias atuais.

Para realizar a avaliação, foi empregado uma técnica de mapeamento de processos utilizando duas métricas para decisão: quantidade de acessos e quantidade de memória. Esta técnica de mapeamento utilizada não apresentou ganhos de desempenho para as aplicações do NPB na arquitetura utilizada. O principal motivo para tal comportamento é que a máquina utilizada nos experimentos possui apenas um nível de hierarquia de memória a ser explorada.

Os resultados obtidos pelas medições de consumo de energia com o IPMI não apresentaram ganhos com o mapeamento. Além do mesmo motivo apontado para o desempenho, outro fator que contribui para tais resultados é o fato de que o IPMI mede o consumo de todo o sistema não permitindo um monitoramento do gasto de energia das memória cache, interconexões e outros componentes de forma isolada.

Adicionalmente, pôde-se observar que o modelo teórico é capaz de captar o comportamento geral de gasto de energia na situação de mapeamento, ainda que a escala não corresponda ao caso real. O erro na escala se deve à constante β do modelo, que foi estimada com o mesmo valor que o utilizado no artigo original. No entanto, o artigo original

se baseia em uma tecnologia de $800nm$, enquanto a tecnologia presente no ambiente de testes é de $45nm$.

Como trabalhos futuros, planeja-se monitorar o consumo de forma isolada de cada componente da arquitetura para melhor compreender o impacto de técnicas de otimizações nos diferentes níveis do processador. O comportamento divergente do benchmark MG também deverá ser estudado para identificar que parâmetros que influenciaram em seu consumo de energia que não foram captados pelo modelo.

Referências

Alves, M. A. Z., Freitas, H. C., and Navaux, P. O. A. (2009). Investigation of shared l2 cache on many-core processors. In *Proceedings Workshop on Many-Core*, pages 21–30, Berlin. VDE Verlag GMBH.

Bellosa, F. (2000). The benefits of event: driven energy accounting in power-sensitive systems. In *Proceedings of the 9th workshop on ACM SIGOPS European workshop: beyond the PC: new challenges for the operating system*, pages 37–42. ACM.

Bircher, W., Valluri, M., Law, J., and John, L. (2005). Runtime identification of microprocessor energy saving opportunities. In *Proceedings of the 2005 international symposium on Low power electronics and design*, pages 275–280. ACM.

Cruz, E. H., Alves, M. A., and Navaux, P. O. (2010). Process mapping based on memory access traces. In *Computing Systems (WSCAD-SCC), 2010 11th Symposium on*, pages 72 –79.

Cruz, E. H., Ribeiro, C. P., Alves, M. A., Carissimi, A., Mehaut, J., and Navaux, P. O. (2011a). Using memory access traces to map threads and data on hierarchical multicore platforms. In *25th IPDPS 13th Workshop on Advances in Parallel and Distributed Computational Models*.

Cruz, E. H. M., Padoin, E. L., Boito, F. Z., and Navaux, P. O. A. (2011b). Avaliando consumo de energia no mapeamento de processos. In *XI Escola Regional de Alto Desempenho (ERAD 2011)*, volume IX, pages 93–94. SBC.

Donofrio, D., Oliker, L., Shalf, J., Wehner, M. F., Rowen, C., Krueger, J., Kamil, S., and Mohiyuddin, M. (2009). Energy-efficient computing for extreme-scale science. *Computer*, 42:62–71.

Gordon-Ross, A., Vahid, F., and Dutt, N. (2005). Fast configurable-cache tuning with a unified second-level cache. In *Proceedings of the 2005 international symposium on Low power electronics and design*, pages 323–326. ACM.

Hennessy, J. L. and Patterson, D. A. (2007). *Computer Architecture: A Quantitative Approach*. Elsevier, USA, 4th edition.

Herranz, A. and Moreno-Navarro, J. (2003). Cache Configuration Exploration on Prototyping Platforms. In *Proceedings of the 14th IEEE International Workshop on Rapid System Prototyping (RSP'03)*, page 164. IEEE Computer Society.

Hicks, P., Walnock, M., and Owens, R. (1997). Analysis of power consumption in memory hierarchies. In *Proceedings of the 1997 international symposium on Low power electronics and design*, pages 239–242. ACM.

Hsu, C.-h. and Feng, W.-c. (2005). A power-aware run-time system for high-performance computing. In *Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, SC '05, pages 1–, Washington, DC, USA. IEEE Computer Society.

Jin, H., Frumkin, M., and Yan, J. (1999). The openmp implementation of nas parallel benchmarks and its performance. In *Technical Report: NAS-99-011*.

Kim, H. S., Narayanan, V., Kandemir, M., and Irwin, M. J. (2000). Multiple access caches: Energy implications. In *Proceedings of the IEEE Computer Society Annual Workshop on VLSI (WVLSI'00)*, WVLSI '00, pages 53–, Washington, DC, USA. IEEE Computer Society.

Kogge, P., Bergman, K., Borkar, S., Campbell, D., Carson, W., Dally, W., Denneau, M., Franzon, P., Harrod, W., Hill, K., and Others (2008). Exascale computing study: Technology challenges in achieving exascale systems. Technical report, University of Notre Dame, CSE Dept.

Kumar, R., Zyuban, V., and Tullsen, D. M. (2005). Interconnections in multi-core architectures: Understanding mechanisms, overheads and scaling. In *Proceedings of the 32nd annual international symposium on Computer Architecture*, ISCA '05, pages 408–419, Washington, DC, USA. IEEE Computer Society.

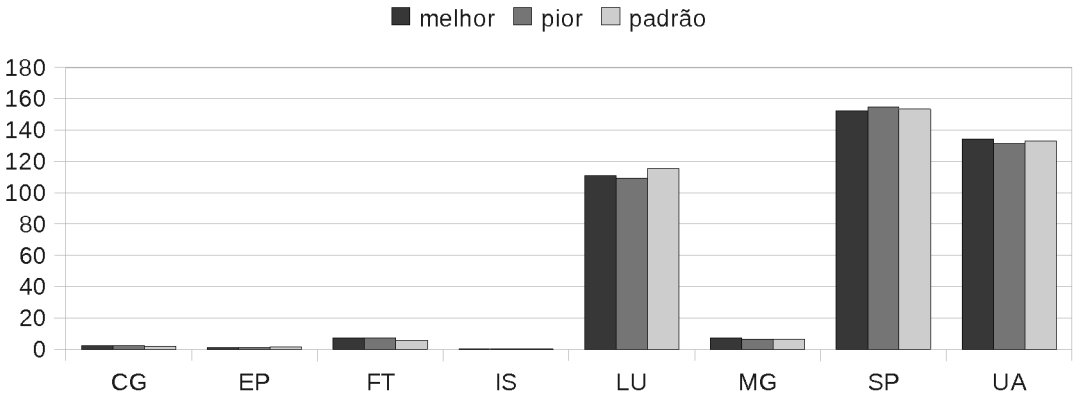
Shiue, W. and Chakrabarti, C. (1999). Memory exploration for low power, embedded systems.

Silva-Filho, A., Cordeiro, F., Sant Anna, R., and Lima, M. (2006). Heuristic for two-level cache hierarchy exploration considering energy consumption and performance. *Lecture notes in computer science*, 4148:75.

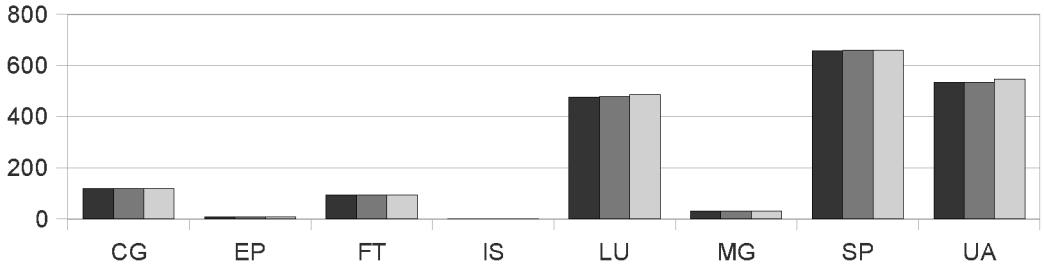
Tam, D., Azimi, R., and Stumm, M. (2007). Thread clustering: sharing-aware scheduling on smp-cmp-smt multiprocessors. *SIGOPS Oper. Syst. Rev.*, 41(3):47–58.

Zarandi, H. and Miremadi, S. (2005). Hierarchical multiple associative mapping in cache memories.

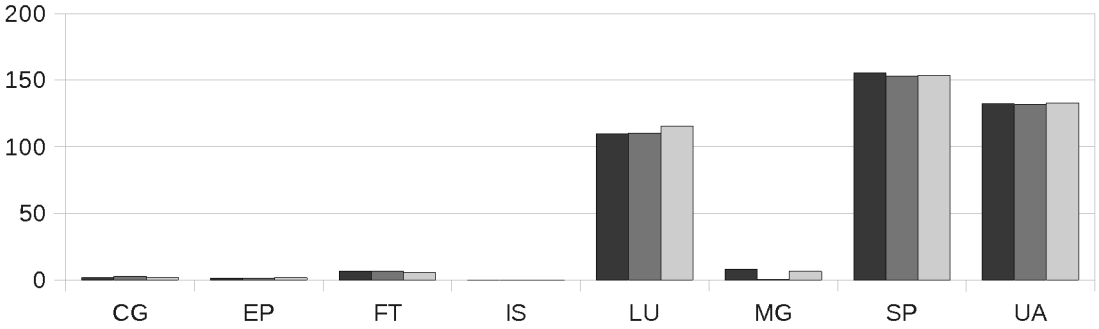
Zhang, C. and Vahid, F. (2003). Cache configuration exploration on prototyping platforms.



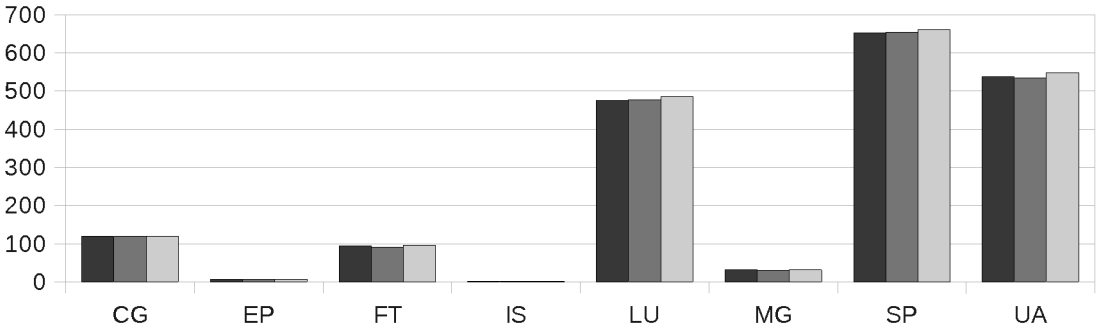
(a) Tamanho A, mapeamento por quantidade de acessos



(b) Tamanho B, mapeamento por quantidade de acessos



(c) Tamanho A, mapeamento por quantidade de memória



(d) Tamanho B, mapeamento por quantidade de memória

Figura 3. Consumo de energia teórico em watts-hora

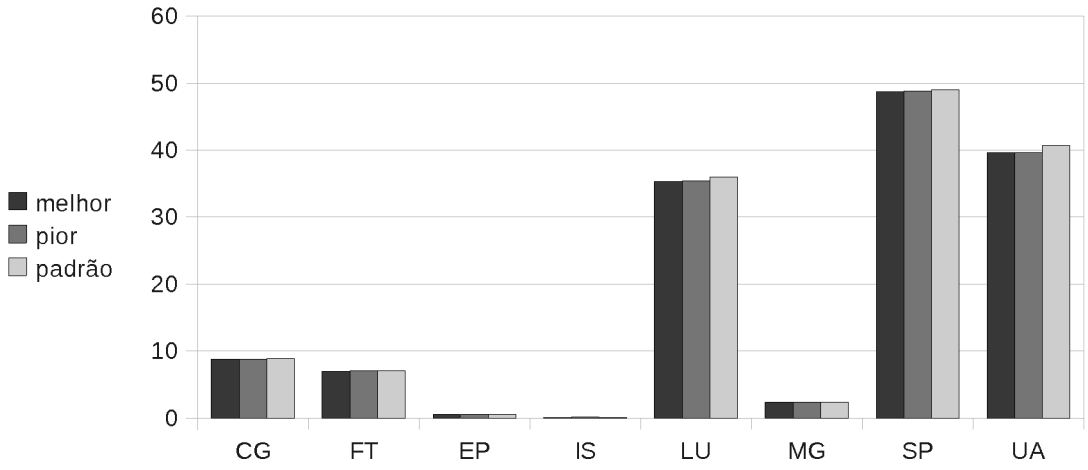


Figura 4. Consumo de Energia (Wh) de acordo com o modelo teórico – Normalizado com o caso real. Tamanho B, Mapeamento por Quantidade de Acessos

Avaliação da Adequação da Plataforma Charm++ para Arquiteturas Multicore com Memória Hierárquica

Laércio L. Pilla[1,2], **Christiane P. Ribeiro**[2],
Philippe O. A. Navaux[1], **Jean-François Méhaut**[2]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Portal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{llpilla,navaux}@inf.ufrgs.br

[2]Laboratoire d'Informatique de Grenoble – INRIA – Universitè de Grenoble
Grenoble – France

{christiane.pousa, jean-francois.mehaut}@imag.fr

Abstract. *Multi-core nodes with Non-Uniform Memory Access (NUMA) are now a common architecture for high performance computing. On such NUMA nodes, the shared memory is physically distributed, making memory access costs vary depending on their distance. Therefore, there is a need to control the memory affinity to provide satisfactory performance. In this context, this paper presents an evaluation of the Charm++ runtime system on these architectures to verify its adequacy to the architecture on communications and load balancing. The results highlight the need of information about the memory hierarchy and machine topology to improve Charm++'s efficiency.*

Resumo. *Máquinas multicore com organização de memória NUMA servem atualmente como base arquitetural para o processamento de alto desempenho. Nestes nós NUMA, a memória compartilhada é fisicamente distribuída, de forma que o custo de acesso à memória pode variar conforme sua distância. Assim, há a necessidade do controle de afinidade de memória para garantir desempenhos satisfatórios. Nesse contexto, este artigo apresenta uma avaliação da plataforma Charm++ nestas arquiteturas para verificar a sua adequação à arquitetura nos quesitos de comunicação e balanceamento de carga. Os resultados ressaltam a necessidade de informações da hierarquia de memória e topologia da máquina para o aumento da eficiência da plataforma Charm++.*

1. Introdução

A importância das arquiteturas com acesso não uniforme à memória (*Non-Uniform Memory Access*, NUMA) vem crescendo como uma solução escalável para reduzir a discrepância de desempenho entre processador e memória (*memory wall* [Lenoski et al. 1993, Wulf and McKee 1995]) e, assim, prover escalabilidade em máquinas multicore. Agregados baseados em processadores AMD Opteron e Intel Nehalem são exemplos de máquinas multicore com organização de memória NUMA. Uma plataforma NUMA é um sistema multiprocessador onde os elementos de processamento compartilham uma memória global que é fisicamente distribuída em bancos de memória. Esses bancos são interconectados através de uma rede especializada. Devido a tal interconexão, o custo de acesso à memória pode variar dependendo da distância (latência)

entre elementos de processamento e bancos de memória, e dependendo do número de elementos de processamento acessando o mesmo banco de memória (largura de banda). Como essas plataformas vêm se tornando cada vez mais presentes em sistemas de Processamento de Alto Desempenho (PAD), é importante reduzir a latência e aumentar a largura de banda disponível para o acesso aos dados. Assim, controlar a afinidade de memória torna-se um elemento crucial para melhorar o desempenho de tais máquinas.

A afinidade de memória é aprimorada quando o mapeamento de threads e dados é feito de forma a reduzir a latência de acesso e contenção de memória no acesso aos dados [Ribeiro et al. 2009]. Essa melhora pode acontecer através de diferentes abordagens, como o uso de mecanismos de alocação eficiente de memória ou através do balanceamento de carga entre os diferentes elementos de processamento. A primeira abordagem foca na distribuição dos dados e em trazê-los para próximo de seus usuários, de forma a reduzir a latência e a contenção de memória. A segunda abordagem trata com uma melhor distribuição de trabalho entre elementos de processamento de forma a evitar pontos de concentração de trabalho e melhorar a comunicação entre threads. A implementação dessas abordagens está usualmente ligada as características do ambiente de programação paralela utilizado. Um exemplo de ambiente é o Charm++ [Kale and Krishnan 1993].

Charm++ é uma plataforma e modelo de programação paralela baseado em C++ desenvolvido com o objetivo de melhorar a produtividade de programação através de uma abstração de alto nível da computação paralela enquanto provendo bom desempenho. Programas em Charm++ são decompostos em objetos comunicantes chamados de *chares*, os quais trocam dados através de chamadas remotas de métodos. Uma vantagem da plataforma Charm++ é que ela captura estatísticas dos objetos durante a execução da aplicação, as quais podem ser utilizadas posteriormente para aprimorar o balanceamento de carga [Bhatele et al. 2009] e, consequentemente, melhorar a afinidade de memória. Porém, apesar da existência de otimizações feitas para ambientes multicore [Mei et al. 2010], elas não levam em consideração as características de uma organização de memória NUMA. Isso pode levar a uma perda de desempenho devido, por exemplo, ao excesso de comunicações entre objetos distantes (em diferentes nós NUMA), os quais poderiam ser aproximados.

Nesse contexto, este artigo apresenta uma avaliação da plataforma Charm++ para arquiteturas multicore com organização NUMA, sendo focados os desempenhos obtidos com diferentes versões de comunicação e através do uso de balanceamento de carga. Dessa forma, torna-se possível avaliar a adequação da plataforma e, consequentemente, desenvolver formas de melhorar a afinidade de memória com a plataforma Charm++.

O restante do artigo é organizado da seguinte forma: a Seção 2 descreve as características das arquiteturas NUMA. A Seção 3 detalha a plataforma Charm++. A Seção 4 apresenta o ambiente e a Seção 5 a avaliação experimental. Trabalhos relacionados são tratados na Seção 6. Por fim, a Seção 7 apresenta as conclusões e trabalhos futuros.

2. Arquiteturas Multicore com Organização NUMA

O uso de arquiteturas multicore segue crescendo para a computação científica, especialmente em PAD. Uma máquina multicore consiste de múltiplos núcleos agrupados em sockets e que compartilham diferentes níveis de hierarquias de cache e memória. Isto é feito de forma a reduzir alguns problemas ligados ao paralelismo no nível de instrução em

um chip e problemas com limites de dissipação de calor [Liu et al. 2009]. Arquiteturas multicore são utilizadas para a construção de poderosas máquinas de memória compartilhada com dezenas a centenas de núcleos. Entretanto, o aumento no número de núcleos requer uma solução de hierarquia de memória eficiente, pois múltiplos núcleos podem utilizar a mesma interconexão para acessar a memória compartilhada, gerando o problema da *memory wall* [Lenoski et al. 1993, Wulf and McKee 1995].

Para dar suporte ao grande número de núcleos e reduzir as limitações de desempenho ligadas à memória, arquiteturas multicore com organização NUMA são utilizadas. Nestas máquinas, múltiplos núcleos acessam a mesma memória compartilhada. Adicionalmente, ela é distribuída em diversos bancos de memória interconectados. Dessa forma, os problemas com o desempenho da memória são reduzidos porque os núcleos podem utilizar diferentes caminhos e bancos de memória para acessar os dados. Entretanto, essa organização gera uma assimetria na latência de acesso aos dados [Ribeiro et al. 2009, Awasthi et al. 2010], levando ao conceito de acessos locais e remotos. Um acesso local é feito quando um núcleo acessa a memória em seu nó, enquanto um acesso remoto ocorre através da requisição de dados alocados em outro nó.

A Figura 1 ilustra uma máquina NUMA possuindo dezesseis núcleos e quatro nós NUMA. A memória compartilhada é distribuída em quatro bancos. Nessa arquitetura, cada quatro núcleos possui seu banco de memória local e os outros bancos são acessados através da rede de interconexão. Adicionalmente, essa máquina possui múltiplos níveis de cache compartilhada para reduzir a latência de acesso aos dados. Nesse caso, cada par de núcleos compartilhada uma cache L2 e dois pares compartilham uma cache L3.

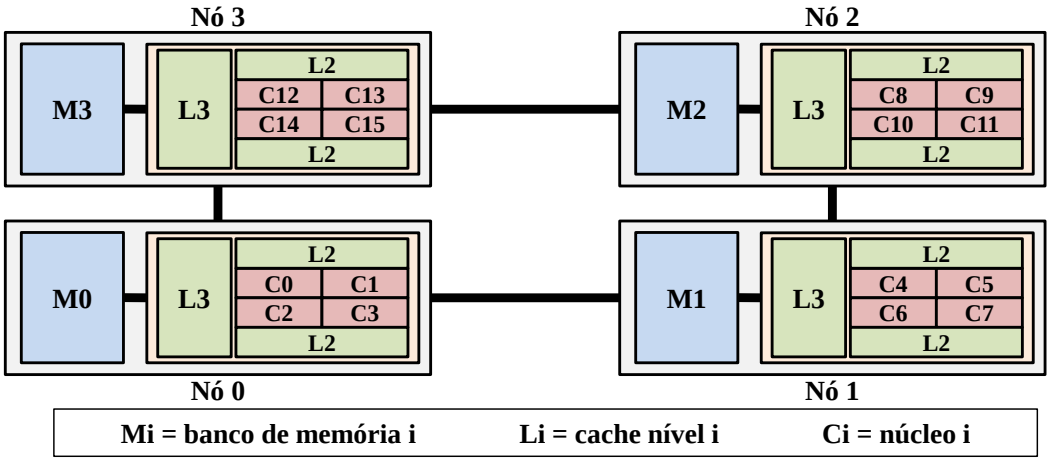


Figura 1. Exemplo de arquitetura multicore com organização NUMA.

Em máquinas NUMA, é particularmente importante garantir um uso eficiente dos bancos de memória para reduzir os custos de acessos remotos nas aplicações. Para tal, mecanismos como escalonamento de threads, alocação de memória e balanceamento de carga podem ser utilizados, dependendo das características da aplicação e da plataforma paralela [Ribeiro et al. 2009, Broquedis et al. 2010, Awasthi et al. 2010].

3. Sistema de Programação Paralela Charm++

Charm++ é uma plataforma paralela que provê uma linguagem de programação paralela orientada a objetos. Ela possui como objetivo prover produtividade ao programa-

dor. Charm++ abstrai características arquiteturais para o desenvolvedor e provê portabilidade sobre arquiteturas baseadas em memória compartilhada e sistemas distribuídos. As aplicações paralelas de Charm++ são escritas em C++ usando uma linguagem de descrição de interface para seus objetos [Kale and Krishnan 1993, Kale et al. 2008].

O processamento nas aplicações em Charm++ é decomposto em objetos chamados *chares*. O programador implementa as computações e comunicações descrevendo como esses objetos vão interagir e o ambiente de Charm++ gerencia as mensagens geradas por essas interações. Os objetos se comunicam através de chamadas remotas de métodos. Ainda, o ambiente é responsável pelo gerenciamento dos recursos arquiteturais. A Figura 2 ilustra a abstração de objetos comunicantes e seu mapeamento físico.

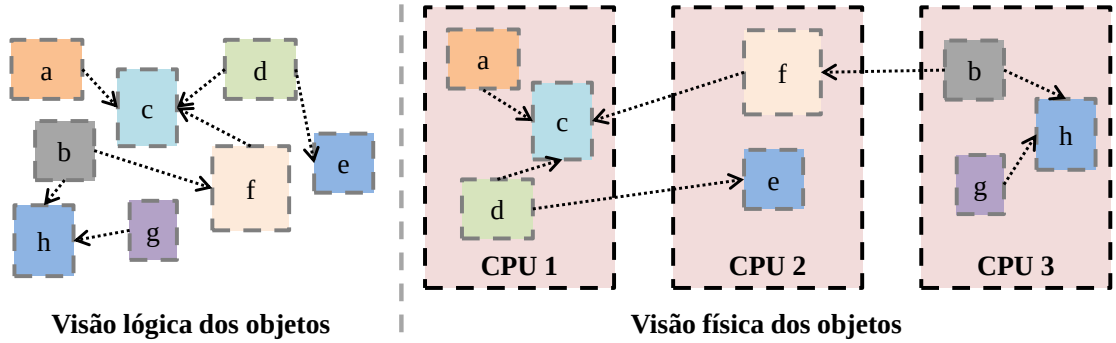


Figura 2. Abstração de aplicação na plataforma Charm++.

Quando utilizada a versão de Charm++ para memória compartilhada, todas as comunicações são feitas através da memória pela troca de ponteiros. Esse mecanismo permite ao ambiente reduzir os sobrecustos com o envio e recebimento de mensagens. Entretanto, no caso de máquinas NUMA, isto pode ser afetado pela assimetria nas latências e largura de banda de memória. Charm++ utiliza o esquema de afinidade de memória do sistemas operacional e não controla explicitamente o posicionamento de dados em memória. Isso pode levar a uma situação onde, por exemplo, mensagens são originalmente geradas e alocadas no nó NUMA 0. Após isto, essas mensagens são enviadas para um chare no nó 1, e encaminhadas sucessivamente até alcançarem o nó NUMA N. Nesta situação, diversos acessos remotos seriam gerados para cada comunicação.

Particularmente em alguns sistemas operacionais como Linux e Windows, a política padrão de gerenciamento de afinidade de memória em máquinas NUMA é a *first-touch*. Essa política posiciona os dados no nó NUMA que primeiro acessá-los [Joseph et al. 2006]. No caso do mecanismo de comunicação de Charm++, uma vez que os dados (por exemplo, uma mensagem) são tocados, essa política de memória não os migrará para aprimorar a afinidade de memória. Isso pode resultar em um posicionamento de dados subótimo em aplicações em Charm++. Dessa forma, torna-se importante avaliar o impacto de tais decisões no desempenho final em máquinas NUMA.

4. Ambiente de Experimentação

Nesta seção são apresentadas as máquinas utilizadas para a avaliação da adequação da plataforma Charm++ a arquiteturas multicore com organização NUMA. Duas máquinas NUMA representativas foram selecionadas para tais experimentos:

- **NUMA16**: baseada em oito processadores AMD Opteron dual-cores. Cada núcleo possui caches L1 e L2 privadas.
- **NUMA32**: baseada em quatro processadores Intel Xeon X7560 com oito núcleos cada. Cada núcleo possui caches L1 e L2 privadas e compartilha uma cache L3 no processador.

A Tabela 1 apresenta as características dessas máquinas, como a largura de banda de memória (obtida através da operação Triad do benchmark Stream [Mccalpin 1995]) e o fator NUMA (razão entre a latências de acesso remoto e local). O fatores NUMA são apresentados em intervalos das penalidades de acesso remoto mínimas e máximas.

Tabela 1. Descrição das arquiteturas multicore com organização NUMA.

Característica	NUMA16	NUMA32
Número de núcleos	16	32
Número de processadores	8	4
Nós NUMA	8	4
Frequência de relógio (GHz)	2.22	2.27
Maior nível de cache (MB)	1 (L2)	24 (L3)
Tamanho da memória (GB)	32	64
Largura de banda da memória (GB/s)	9.77	35.54
Fator NUMA (Min;Max)	[1, 1; 1, 5]	[1, 36; 3, 6]

Para a avaliação experimental foi utilizada a versão $6.2.1$ da plataforma Charm++. Os testes foram baseados na execução de três diferentes benchmarks disponibilizados com Charm++: (i) *kNeighbor*, um aplicação sintética iterativa onde cada chare se comunica com outros k à cada etapa; (ii) *jacobi2D*, uma computação stencil em 5 pontos sobre uma grade bidimensional; e (iii) *lb_test*, um benchmark sintético onde é possível escolher diferentes cargas computacionais e padrões de comunicação. Os resultados apresentados são as médias de desempenho obtidas sobre um mínimo de 20 execuções, com uma confiança estatística de 95% e erro relativo de 10% segundo a Distribuição T de Student.

5. Avaliação Experimental

Esta seção apresenta os experimentos realizados sobre as versões de comunicação e balanceadores de carga de Charm++ em máquinas NUMA.

5.1. Versões de comunicação

Charm++ permite a utilização de diferentes implementações para a troca de mensagens. No caso das máquinas NUMA utilizadas, duas versões de comunicação são de maior interesse: comunicação via troca de datagramas UDP e via troca de ponteiros em memória compartilhada. Apesar da segunda ser favorecida pela arquitetura, é importante verificar os sobrecustos da comunicação via datagramas tendo em vista que máquinas NUMA são utilizadas como nós em clusters de PAD.

A Figura 3 ilustra o desempenho obtido para o benchmark *jacobi2D* executando 100 chares e utilizando as diferentes formas de comunicação nas máquinas NUMA. O eixo vertical representa o tempo em segundos e o eixo horizontal representa diferentes

números de núcleos utilizados na execução do benchmark. As Figuras 3(a) e 3(b) apresentam os tempos obtidos nas máquinas NUMA16 e NUMA32, respectivamente. Como o desempenho do benchmark está mais ligado ao uso intensivo dos recursos de processamento do que pela comunicação, as diferentes versões de comunicação possuem impacto insignificante sobre o desempenho final da aplicação. Além disso, pode-se ver que quanto maior o número de núcleos utilizados, menor o tempo de iteração, demonstrando escalabilidade em ambas máquinas e versões.

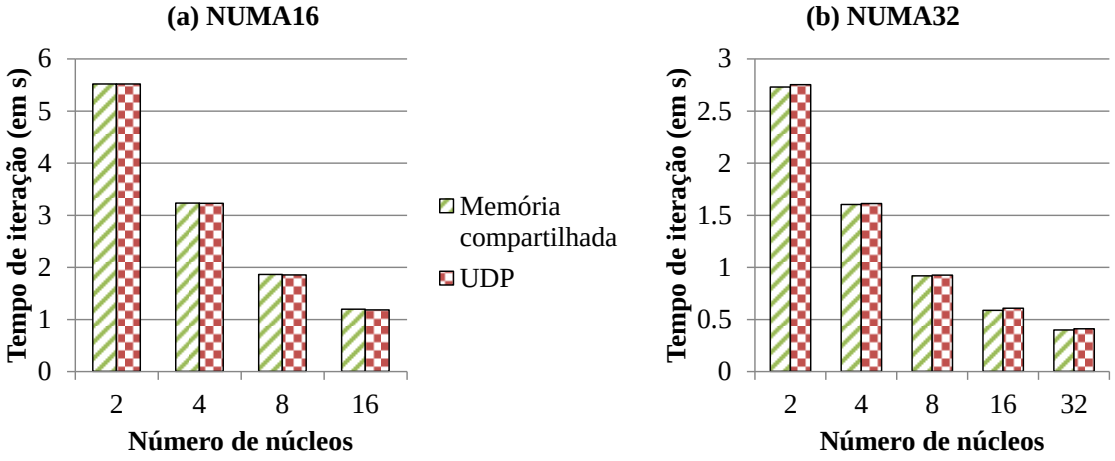


Figura 3. Tempo de iteração médio para as diferentes versões de comunicação com o benchmark *jacobi2D*.

Um caso mais extremo é ilustrado na Figura 4, a qual apresenta o desempenho obtido com o benchmark *kNeighbor* para as diferentes formas de comunicação. Neste experimento, foram utilizados 200 chares e um número de vizinhos comunicantes $k = 3$. Nesta figura, o eixo vertical representa o tempo de execução em milissegundos. Como visto na Figura 4(a), enquanto o tempo de execução é reduzido com o aumento de núcleos para a versão de comunicação via troca de ponteiros, o contrário acontece para a versão via UDP. Essa diferença chega ao seu máximo com 8 núcleos, onde a versão via UDP leva $9\times$ o tempo da versão via memória compartilhada. A redução no tempo de execução quando passando de 8 para 16 núcleos acontece devido a distribuição da comunicação através do aumento de vias de comunicação e distribuição da carga de trabalho. O mesmo não acontece para a máquina NUMA32, como ilustrado na Figura 4, devido à contenção na memória, pois há um maior número de núcleos acessando o mesmo banco de memória nessa máquina – oito núcleos por nó NUMA, enquanto a máquina NUMA16 possui apenas dois núcleos por nó.

Estes resultados mostram que a utilização das diferentes implementações de comunicação possuem um impacto pequeno sobre aplicações com computações intensivas. Entretanto, no caso de aplicações com muitas comunicações em clusters de máquinas NUMA, seria de interesse aplicar um esquema híbrido de comunicação, utilizando a troca de datagramas entre máquinas mas comunicação por memória compartilhada internamente à máquina NUMA. Ainda mais importante, os resultados com o benchmark *kNeighbor* mostram como o problema de contenção em memória pode prejudicar a escalabilidade de aplicações mesmo em arquiteturas com organização NUMA. Dessa forma, torna-se necessário assegurar uma melhor afinidade de memória através de outros meca-

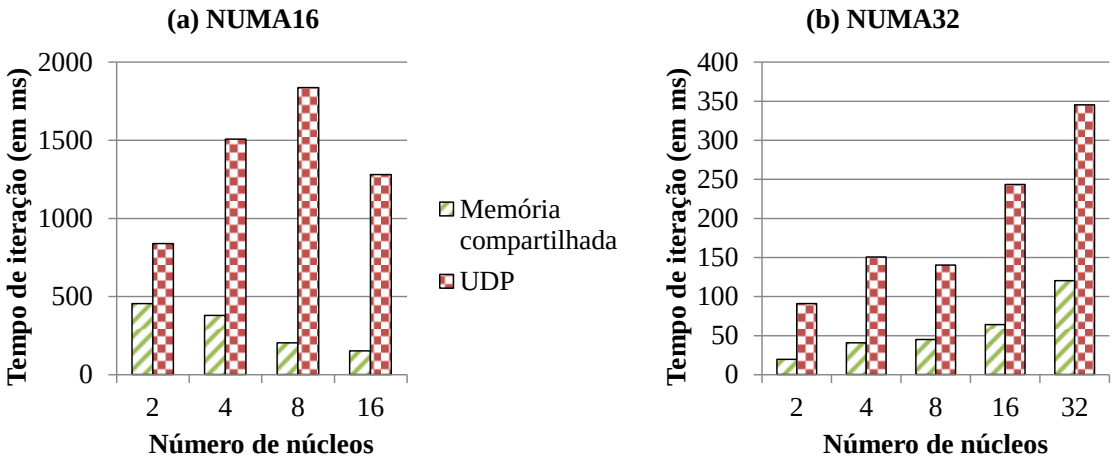


Figura 4. Tempo de iteração médio para as diferentes versões de comunicação com o benchmark *kNeighbor*.

nismos, como o balanceamento de carga.

5.2. Balanceamento de Carga

Através das estatísticas capturadas durante a execução de uma aplicação pela plataforma Charm++ é possível combater o desbalanceamento de carga, contenção de memória e reduzir as latências ligadas à comunicação remota. Para os experimentos com máquinas NUMA, foram considerados três balanceadores de carga disponibilizados com a plataforma: GREEDYLB, METISLB e SCOTCHLB. GREEDYLB reatribui os chares de forma gulosa. O algoritmo iterativamente mapeia o chare com maior tempo de execução para o núcleo com menor carga de trabalho. Sendo assim, o algoritmo não considera as comunicações entre chares. Apesar disso, esta estratégia possui um bom desempenho devido a sua simplicidade e velocidade. Já os balanceadores METISLB e SCOTCHLB são baseados em algoritmos de particionamento de grafos implementados nas bibliotecas METIS [Karypis and Kumar 1995] e SCOTCH [Pellegrini and Roman 1996], respectivamente. Estas estratégias consideram ambos tempos de execução e grafo de comunicação para aprimorar o desempenho das aplicações.

A Figura 5 apresenta os desempenhos obtidos para os diferentes balanceadores de carga aplicados aos benchmarks *lb_test* e *jacobi2D*. Nela, a coluna Base representa o tempo médio das iterações sem o balanceamento de carga. Para o benchmark *lb_test*, foram utilizados 200 chares, com tempos mínimos e máximos de computação de 50 e 200 ms, respectivamente, e um grafo de comunicações aleatório. Como pode ser visto na Figura 5(a), o uso de balanceamento de carga leva a melhoras em ambas máquinas NUMA. Entretanto, em ambos casos levam vantagem os balanceadores que consideram o grafo de comunicação. Por exemplo, na máquina NUMA32, o uso do SCOTCHLB leva a *speedups* de 1.4 sobre o tempo base e de 1.18 sobre o balanceador GREEDYLB.

Resultados similares foram obtidos para o o benchmark *jacobi2D*, como ilustrado na Figura 5(b). Porém, apesar dos *speedups* de até 1.57 sobre o tempo de execução base, os balanceadores de carga obtiveram uma eficiência final por volta de 90% ou menos. Isso acontece porque, apesar dos balanceadores de carga como o SCOTCHLB reduzirem as comunicações externas aos núcleos, eles não possuem conhecimento sobre a organização

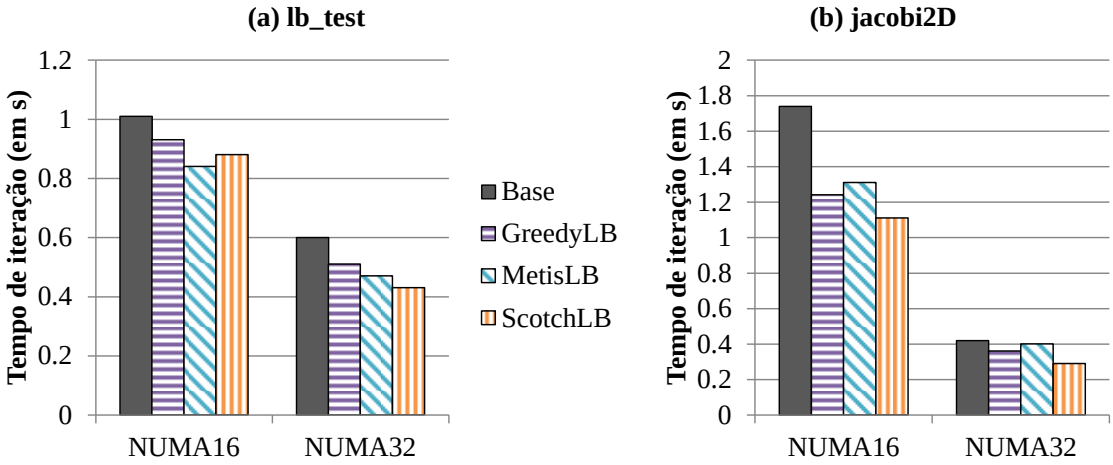


Figura 5. Tempo de iteração médio para diferentes balanceadores de carga e benchmarks.

Tabela 2. Número médio de migrações por chamada de balanceamento de carga.

Benchmark	Máquina	Balanceador de carga		
		GREEDYLB	METISLB	SCOTCHLB
lb_test	NUMA16	188	187	184
	NUMA32	194	194	192
jacobi2D	NUMA16	94	94	93
	NUMA32	97	96	98

e hierarquias de cache e memória. Dessa forma, não são aproveitados os núcleos próximos para reduzir as comunicações remotas de alta latência.

Adicionalmente a estes resultados, a Tabela 2 apresenta as quantidades médias de migrações feitas pelos balanceadores de carga. Como pode ser visto, tais balanceadores de carga não consideram o mapeamento inicial dos chares e, por isso, acabam migrando entre 92% e 98% dos chares. Dessa forma, mostra-se importante a avaliação do impacto que essas migrações afligem no desempenho final da aplicação, tendo em vista que elas resultados em cópias de memória através dos nós NUMA.

A Figura 6(a) apresenta os custos ligados à migração de 20 chares entre diferentes nós NUMA para diferentes tamanhos de chares na máquina NUMA16. Na figura, o eixo vertical representa o tempo em segundos para efetuar tais migrações e é apresentado em escala logarítmica. O eixo horizontal representa o número de bytes de cada chare migrado (também em escala logarítmica). Cada linha representa migrações para nós remotos. A linha intra-socket representa migrações para núcleos dentro do mesmo processador. Como pode ser visto na Figura 6(a), há uma grande diferença entre os custos de migrações para núcleos próximos e distantes para chares pequenos. Entretanto, elas são na ordem de milissegundos e acabam diminuindo com o crescimento dos chares. Ainda assim, há uma diferença de 1.1 segundos entre migrações locais para as mais distantes, o que representa 20% do tempo total de migração. Essa diferença pequena nos tempos de migração encontrados na máquina NUMA16 são explicados pela pequena diferença nas latências

de acesso à memória local e remota (baixo fator NUMA) e pela ausência de uma cache compartilhada entre os núcleos em um mesmo nó NUMA.

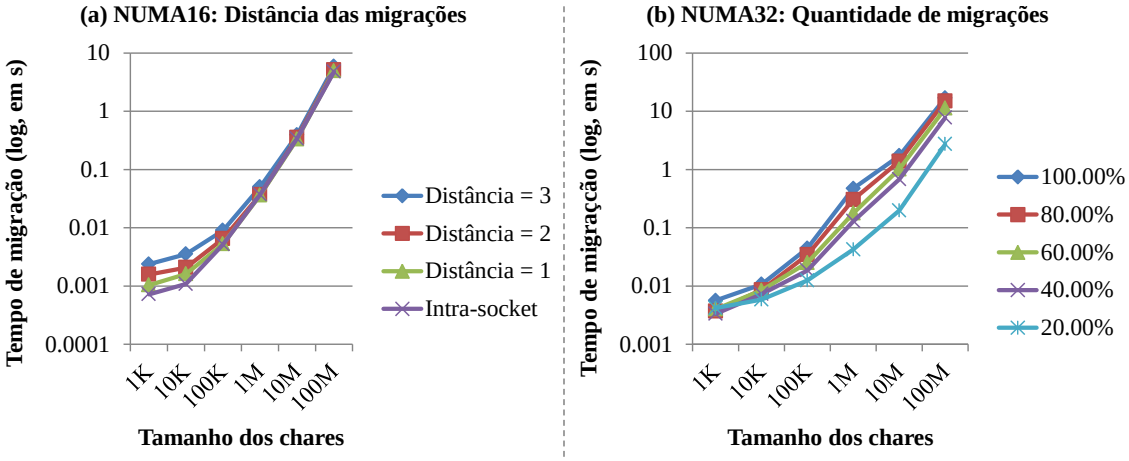


Figura 6. Tempo de migração para diferentes distribuições de chares.

Por fim, a Figura 6(b) ilustra os tempos de migração para diferentes percentuais de migrações aleatórias na máquina NUMA32. Não há praticamente diferença no custo de migração quando utilizando chares de até 10 KB. Porém, tal diferença cresce rapidamente com o tamanho dos chares, chegando ao ponto em que migrar apenas 20% dos chares leva apenas 16% do tempo de migrar todos os chares. Dessa forma, para aumentar o desempenho de uma aplicação em Charm++ em uma máquina multicore com organização NUMA, não basta apenas aprimorar a afinidade de memória, sendo também importante evitar migrações desnecessárias e, consequentemente, cópias remotas de dados.

6. Trabalhos Relacionados

Arquiteturas multicore com organização NUMA tem se tornado uma tendência em arquiteturas paralelas de computação de alto desempenho. Entretanto, poucos sistemas paralelos, linguagens de programação e interfaces oferecem algum suporte para esse tipo de arquitetura. A complexidade da hierarquia de memória dessas arquiteturas traz consigo custos associados aos acessos à esta memória. Nesse contexto, uma adequação dos sistemas paralelos, linguagens de programação e interfaces torna-se necessária.

Uma forma de evitar os custos de acesso à memória em arquiteturas NUMA é usar o mapeamento de threads. O mapeamento pode ser usado para reduzir a distância entre threads que realizam muita comunicação entre si. No caso específico de arquiteturas NUMA, esse mapeamento pode favorecer o uso de memórias caches e reduzir o uso da memória principal. Essa abordagem é utilizada em [Mercier and Clet-Ortega 2009] e [Jeannot and Mercier 2010]. Porém, os autores avaliam apenas a melhora no desempenho das aplicações, e não as características das plataformas e arquiteturas envolvidas.

Em [Mei et al. 2010], os autores apresentam algumas otimizações no sistema Charm++ para máquinas multicore. Uma das otimizações apresentadas é o suporte à afinidade de threads, que permite ao programador mapear threads nos núcleos da máquina e assim, reduzir os custos de comunicação. Resultados apresentados no trabalho mostram ganhos de até 15% quando o suporte para mapeamento de threads foi utilizado. Apesar

dessa otimização não ter sido proposta para arquiteturas NUMA, ela pode ser facilmente aplicada nessas plataformas de forma a reduzir os custos de acesso à memória.

Muitos outros ambientes de programação paralela também sofrem com a assimetria de acessos à memória em máquinas NUMA. Como exemplo, podemos citar ambientes de memória transacional por software (STM) que têm sido estudados nos últimos anos. STM é uma proposta que visa reduzir a complexidade do controle de seções críticas e o custo associado aos *locks*. Apesar de muitos estudos de memória transacional terem considerados máquinas multicore, poucos trabalhos foram propostos para máquinas multicore com organização NUMA. Em [Lu et al. 2010], são propostos mecanismos para STM que consideram os custos não uniformes de acesso à memória. A ideia principal é separar a máquina NUMA em clusters, dependendo da distância entre os componentes da máquina. Assim, a STM poderia reagir as necessidades da aplicação considerando os custos da máquina NUMA. Diferente do nosso trabalho, esse estudo não apresenta uma avaliação de desempenho de STMs em máquinas NUMA e como consequência, ele não trata dos diferentes impactos que uma arquitetura NUMA pode ter sobre STM.

OpenMP é uma interface de programação paralela que surgiu de um consórcio entre universidades e empresas [OpenMP 2010]. A principal proposta do consórcio é a definição de um padrão que simplifique o desenvolvimento de aplicações paralelas. Em OpenMP, aplicações são paralelizadas usando diretivas que informam ao compilador as regiões do código que devem ser paralelizadas. Entretanto, OpenMP não foi desenvolvido para arquiteturas NUMA. Desta forma, não existe nenhum suporte na interface para gerenciar a afinidade de memória. Em [Ribeiro et al. 2008], os autores apresentam uma avaliação de benchmarks e aplicações OpenMP em arquiteturas NUMA. Eles também apresentam o suporte NUMA existente em alguns sistemas operacionais para gerenciar a alocação de dados e o mapeamento de threads para reduzir as penalidades NUMA nas aplicações. Os resultados mostram a importância de gerenciar a afinidade de memória em arquiteturas NUMA e como usuários podem fazer esse controle em aplicações OpenMP usando o suporte do sistema operacional. Porém, os autores não avaliam o suporte de execução OpenMP, focando seus estudos nas aplicações e benchmarks. Ao contrário, a avaliação apresentada deste trabalho foca em compreender e investigar a adequação de um sistema paralelo em arquiteturas NUMA.

7. Conclusões

O aumento na popularidade de arquiteturas multicore com organização NUMA requer a adequação das plataformas de programação paralela para elas, de forma a prover desempenhos satisfatórios e reduzir sobrecustos ligados à comunicações remotas e contenção de memória. Entretanto, esta não é uma tarefa simples, pois a complexidade da hierarquia de memória pode influenciar aplicações de formas diferentes. Assim, faz-se importante estudar como estas plataformas são influenciadas pelos ambientes NUMA.

Os resultados desta avaliação da plataforma Charm++ em máquinas NUMA mostram que as comunicações entre nós NUMA devem ser mantidas através da troca de ponteiros em memória compartilhada, pois outros mecanismos podem acabar com a escalabilidade das aplicações, principalmente no caso de aplicações com comunicação intensiva. Além disso, os resultados com balanceamento de carga apresentam melhoras de desempenho, mas chegam a eficiências limitadas a 90%. Isso acontece devido a ausência de co-

nhecimento sobre a organização NUMA e das hierarquias de memória e cache, o que leva a mapeamentos que não favorecem comunicações através de acessos locais à memória. Por fim, o desenvolvimento de um balanceador de carga com conhecimento sobre a arquitetura deve evitar migrações desnecessárias, devido ao custos de cópias remotas de memória.

Trabalhos futuros incluem o desenvolvimento de um balanceador de carga consciente da organização NUMA e da hierarquia de cache das arquiteturas multicore. Isto requer a medição das diferentes latências de comunicação entre núcleos e nós NUMA. Através da organização destas informações, pretende-se adicioná-las à plataforma paralela Charm++, de forma a aprimorar sua adequação a tais arquiteturas de forma transparente ao programador.

Referências

Awasthi, M., Nellans, D. W., Sudan, K., Balasubramonian, R., and Davis, A. (2010). Handling the problems and opportunities posed by multiple on-chip memory controllers. In *Proceedings of the 19th International Conference on Parallel Architectures and Compilation Techniques (PACT 2010)*, pages 319–330, New York, NY, USA. ACM.

Bhatele, A., Kale, L. V., and Kumar, S. (2009). Dynamic topology aware load balancing algorithms for molecular dynamics applications. In *Proceedings of the 23rd international Conference on Supercomputing (ICS 2009)*, pages 110–116, New York, NY, USA. ACM.

Broquedis, F., Aumage, O., Goglin, B., Thibault, S., Wacrenier, P. A., and Namyst, R. (2010). Structuring the execution of OpenMP applications for multicore architectures. In *Proceedings of the IEEE International Symposium on Parallel & Distributed Processing (IPDPS 2010)*, pages 1–10. IEEE Computer Society.

Jeannot, E. and Mercier, G. (2010). Near-Optimal Placement of MPI Processes on Hierarchical NUMA Architectures. In D'Ambra, P., Guarracino, M., and Talia, D., editors, *Euro-Par 2010 - Parallel Processing*, volume 6272 of *Lecture Notes in Computer Science*, pages 199–210. Springer Berlin / Heidelberg.

Joseph, A., Pete, J., and Alistair, R. (2006). Exploring Thread and Memory Placement on NUMA Architectures: Solaris and Linux, UltraSPARC/FirePlane and Opteron/HyperTransport. In *International Conference on High Performance Computing (HiPC 2006)*, pages 338–352.

Kale, L. V., Bohm, E., Mendes, C. L., Wilmarth, T., and Zheng, G. (2008). Programming Petascale Applications with Charm++ and AMPI. In Bader, D., editor, *Petascale Computing: Algorithms and Applications*, pages 421–441. Chapman & Hall / CRC Press.

Kale, L. V. and Krishnan, S. (1993). Charm++: A portable concurrent object oriented system based on C++. In *Proceedings of the Eighth Annual Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA 1993)*, pages 91–108. ACM.

Karypis, G. and Kumar, V. (1995). METIS: Unstructured graph partitioning and sparse matrix ordering system. *The University of Minnesota*, 2.

Lenoski, D., Laudon, J., Joe, T., Nakahira, D., Stevens, L., Gupta, A., and Hennessy, J. (1993). The dash prototype: Logic overhead and performance. *IEEE Transactions on Parallel and Distributed Systems*, 4(1):41–61.

Liu, M., Ji, W., Wang, Z., and Pu, X. (2009). A memory access scheduling method for multi-core processor. *International Workshop on Computer Science and Engineering (WCSE 2009)*, 1:367–371.

Lu, K., Wang, R., and Lu, X. (2010). Brief announcement: Numa-aware transactional memory. PODC, pages 69–70.

Mccalpin, J. D. (1995). STREAM: Sustainable memory bandwidth in high performance computers. Technical report, University of Virginia.

Mei, C., Zheng, G., Gioachin, F., and Kale, L. V. (2010). Optimizing a parallel runtime system for multicore clusters: a case study. In *Proceedings of the 2010 TeraGrid Conference (TG 2010)*, New York, NY, USA. ACM.

Mercier, G. and Clet-Ortega, J. (2009). Towards an Efficient Process Placement Policy for MPI Applications in Multicore Environments. In Ropo, M., Westerholm, J., and Dongarra, J., editors, *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, volume 5759 of *Lecture Notes in Computer Science*, pages 104–115. Springer Berlin / Heidelberg.

OpenMP (2010). The OpenMP API Specification for Parallel Programming.

Pellegrini, F. and Roman, J. (1996). Scotch: A software package for static mapping by dual recursive bipartitioning of process and architecture graphs. In *International Conference on High-Performance Computing and Networking (HPCN 1996)*, pages 493–498. Springer.

Ribeiro, C. P., Dupros, F., Carissimi, A., Marangozova-Martin, V., Méhaut, J.-F., and de Aguiar, M. S. (2008). Explorando Afinidade de Memória em Arquiteturas NUMA. In *WSCAD '08: Proceedings of the 9th Workshop em Sistemas Computacionais de Alto Desempenho - SBAC-PAD*, Campo Grande, Brazil. SBC.

Ribeiro, C. P., Mehaut, J.-F., Carissimi, A., Castro, M., and Fernandes, L. G. (2009). Memory Affinity for Hierarchical Shared Memory Multiprocessors. In *21st International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD 2009)*, pages 59–66.

Wulf, W. and McKee, S. A. (1995). Hitting the memory wall: implications of the obvious. *SIGARCH Comput. Archit. News*, 23:20–24.

Solução eficiente de modelos estruturados a partir de descritores tensoriais combinados a estruturas arborescentes *

Ricardo M. Czekster, Paulo Fernandes, Afonso Sales, Thais Webber

[1]Pontifícia Universidade Católica do Rio Grande do Sul – Faculdade de Informática
Av. Ipiranga, 6681, Prédio 32 – 90619-900 – Porto Alegre – RS – Brasil

{ricardo.czekster, paulo.fernandes, afonso.sales, thais.webber}@pucrs.br

Abstract. *Structured Markovian formalisms have emerged to facilitate the representation of large state spaces and transitions. However, structuring introduces algorithmic challenges, such as detection of reachable states, as well as specialized numerical solutions. Recent researches have proposed the use of Multi-valued Decision Diagrams (MDD) for the compact storage of reachable states. This paper presents an efficient technique for mapping transitions between states represented by MDD, that can be adapted to the solution via iterative methods, or even direct methods. Preliminary results show significant gains in processing time and memory considering the ratio between state space size and amount of reachable states.*

Resumo. *Formalismos Markovianos estruturados surgiram para facilitar a representação de modelos com inúmeros estados e transições. Entretanto, esta estruturação introduz desafios algorítmicos, tais como detecção de estados válidos, bem como soluções numéricas especializadas. Pesquisas recentes propõem utilizar Diagramas de Decisão Multi-valorados (MDD) para armazenamento compacto deste estados. Neste artigo apresenta-se uma técnica eficiente de mapeamento das transições entre estados, permitindo adequar a solução via métodos iterativos, ou mesmo diretos. Resultados preliminares mostram ganhos significativos em tempo de processamento e memória em função da relação entre o conjunto de estados total e válidos do modelo.*

1. Introdução a Formalismos Estruturados

O uso de Cadeias de Markov é disseminado na modelagem de problemas complexos de diversas áreas, tais como: biologia, economia, linguística, computação no âmbito de avaliação de desempenho, entre outras aplicações. Entretanto este formalismo apresenta o problema conhecido como *explosão do espaço de estados* o que impossibilita a modelagem de sistemas cujo espaço de estados ultrapasse a capacidade de processamento e memória das máquinas disponíveis, logo este problema dificulta o tratamento numérico do modelo [Stewart 1994].

Neste contexto, formalismos com maior nível de abstração, também chamados de formalismos Markovianos estruturados, surgem como uma alternativa interessante para mitigar este problema de modelagem, já que possibilitam utilizar módulos

*Este trabalho é financiado pela Petrobras (Conv. 0050.0048664.09.9). Paulo Fernandes é também financiado pelo CNPq-Brasil (PQ 307284/2010-7) e Afonso Sales recebe financiamento da CAPES-Brasil (PNPD 02388/09-0). A ordem dos autores é meramente alfabética.

ou subsistemas para representar um grande sistema. Alguns formalismos, como é o caso das *Redes de Autômatos Estocásticos* (do inglês, *Stochastic Automata Networks* - SAN) [Plateau 1985] apresentam primitivas diferenciadas para representar comportamentos dependentes nas transições entre os estados. SAN tem como primitivas básicas a possibilidade de definir eventos que habilitam transições de forma local em um subsistema ou autômato, e eventos sincronizantes que habilitam transições simultâneas entre estados em autômatos diferentes. Além disto, taxas para os eventos podem determinar também dependências funcionais entre autômatos na rede, possibilitando a mudança de estado em pelo menos um autômato, ou em mais, dependendo do estado atual de outros [Brenner et al. 2005]. Modelos descritos em SAN podem ser mapeados para uma cadeia de Markov correspondente, que se caracteriza por representar todos os estados possíveis do modelo SAN, considerando as combinações dos estados locais de todos os autômatos da rede [Plateau 1985, Stewart 1994].

Outro fator relevante na solução de modelos estruturados como SAN é que nem sempre todos os estados combinados oriundos de um modelo são considerados possíveis ou válidos [Sales and Plateau 2009]. Em um modelo com vários subsistemas, chamamos de *espaço de estados total* a combinação de todos os estados de todos os seus subsistemas. De acordo com a natureza dos eventos e as transições que estes habilitam, determinados estados do espaço de estados podem ser *inatingíveis*, dando origem ao chamado *espaço de estados atingíveis*. A descoberta e o armazenamento destes estados válidos de uma maneira eficiente pode ser empregada através do uso de estruturas arborescentes chamadas *Diagramas de Decisão Multi-valorados* (do inglês, *Multi-valued Decision Diagrams* - MDD) [Miner and Ciardo 1999b, Sales and Plateau 2009]. Estas estruturas são extremamente compactas na representação de tais estados, entretanto, não apresentam informações sobre as transições do modelo.

Neste contexto, note que tanto as cadeias de Markov, quanto as SAN, podem ser solucionadas via métodos diretos ou iterativos através de multiplicações vetor-matriz, ou mesmo por simulação [Fernandes et al. 1998, Buchholz et al. 2000, Buchholz and Dayar 2004, Fernandes et al. 2008]. Entretanto, no caso dos formalismos estruturados, estes podem tirar proveito de álgebra tensorial para o armazenamento compacto de modelos, consequentemente, apresentando soluções numéricas eficientes, tanto em memória quanto em tempo de processamento.

SAN é um formalismo pioneiro no uso de tensores para representar modelos algebricamente, podendo armazená-lo de forma compacta apesar de aumentar a complexidade dos algoritmos para sua solução. Algoritmos que lidam com tensores oriundos de modelos estruturados são especializados para tratar com matrizes esparsas e operações de multiplicação vetor-matriz não convencionais, a exemplo da multiplicação de vetores por somas e produtos tensoriais [Fernandes et al. 1998]. Na literatura encontram-se maiores detalhes sobre os algoritmos chamados *Shuffle* e *Split*, especializados para solução de modelos em formato tensorial (os modelos também são chamados de descritores Markovianos) [Fernandes et al. 1998, Czekster et al. 2007, Czekster et al. 2010b].

Apesar destes algoritmos lidarem eficientemente com tais descritores, o espaço de estados dos modelos influencia diretamente na complexidade computacional de solução principalmente em termos de memória para armazenamento de vetores. A solução por cadeia de Markov pode demandar menos memória em relação a estes vetores utilizados,

pois a cadeia apenas contém os estados atingíveis do modelo SAN [Buchholz et al. 2000]. Apesar disto, existe uma dificuldade em resolver numericamente modelos em cadeias de Markov suficientemente grandes ao ponto de impossibilitar o armazenamento da matriz de transição, vetores e eventuais estruturas auxiliares. No caso de SAN, modelos com poucos estados válidos em relação ao espaço de estados total que pode ser gigantesco, também podem se tornar impraticáveis de resolver dado a necessidade de armazenar e acessar igualmente gigantescos vetores de probabilidade, apesar do modelo ser armazenado em poucos Kb usando tensores [Czekster 2010, Sales 2009, Webber 2009].

Neste artigo propõe-se uma técnica eficiente para o mapeamento das transições entre os estados representados por MDD, permitindo adequar a solução de modelos via métodos iterativos, ou mesmo diretos. O objetivo é manter a modularidade na modelagem, a estruturação através de álgebra tensorial e o uso de primitivas de modelagem, mas também possibilitar flexibilidade para uma solução eficiente que lide com o espaço de estados atingíveis armazenado de maneira compacta. Resultados preliminares para classes de modelos mostram ganhos significativos em tempo de processamento e memória em função da relação entre o conjunto de estados total e válidos do modelo.

As seções neste artigo estão divididas como segue. Seção 2 retoma conceitos de modelagem através de Redes de Autômatos Estocásticos e discute brevemente sua representação tensorial. Seção 3 ressalta a necessidade da detecção do espaço de estados atingíveis em modelos estruturados, bem como apresenta um exemplo de representação arborescente. Seção 4 apresenta uma visão geral dos algoritmos existentes para solução de descritores e seus problemas relacionados ao uso de memória e ao tempo de processamento. Seção 5 mostra experimentos para classes de modelos, especialmente aqueles com espaço atingível reduzido em relação ao espaço total de estados, onde os resultados numéricos obtidos são promissores, tornando válida a abordagem de solução proposta. Seção 6 aponta para novas direções no contexto de solução de modelos estruturados onde a priorização é manter o poder de modelagem aliado a uma solução otimizada.

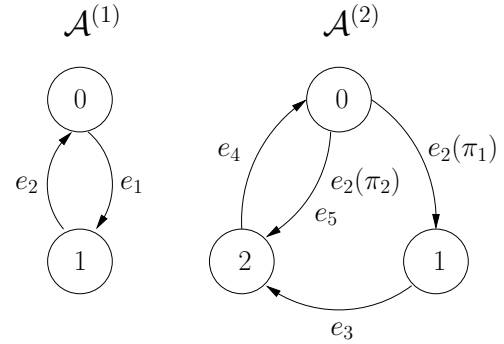
2. Modelagem com Redes de Autômatos Estocásticos e Álgebra Tensorial

Redes de Autômatos Estocásticos (SAN) é um formalismo baseado em Cadeias de Markov [Plateau 1985], que permite a definição de mais de um componente (ou entidade) na forma de autômatos estocásticos. Estas entidades podem interagir entre si através de primitivas de modelagem conhecidas como eventos sincronizantes, bem como podem ter um comportamento independente através de eventos locais. Além disto, eventos são determinados por suas taxas (ou probabilidades) de ocorrência, que no caso das SAN podem também dispor de taxas funcionais para o disparo dos eventos, ou seja, taxas que criam uma certa dependência com os estados correntes de autômatos da rede [Benoit et al. 2004].

A complexidade inerente em modelar sistemas com grande número de estados trouxe a necessidade de ampliação dos formalismos para permitirem uma estruturação mais complexa do que simplesmente uma matriz de transição entre estados, já que componentes podiam ser modelados separadamente mas precisavam ser avaliados em conjunto. Logo, de forma pioneira, as SAN foram mapeadas para os chamados descritores Markovianos [Plateau 1985], que são estruturas obtidas através da representação do modelo em um conjunto de matrizes que operam através de álgebra tensorial. Este avanço possibilitou

grande economia na memória para armazenamento do conjunto de eventos e transições que geram a cadeia de Markov implícita ao modelo. Cabe salientar que as matrizes que compõem um descritor normalmente são de baixa ordem devido à modularização aplicada na modelagem, e também esparsas, visto que os eventos podem ocorrer eventualmente em cada um dos autômatos em uma ou apenas algumas transições. SAN também opera matrizes com elementos funcionais oriundos das taxas de eventos que dependem dos estados ou condições de outras entidades para disparar, para isto introduziu-se também o uso de álgebra tensorial generalizada [Fernandes et al. 1998, Benoit et al. 2004].

A Figura 1 [Sales and Plateau 2009] mostra uma pequena rede de autômatos estocásticos para exemplificar dois autômatos, cinco eventos (4 locais e 1 sincronizante). O evento sincronizante e_2 é apresentado com probabilidades de rotação π_1 e π_2 em diferentes transições. O evento local e_1 é apresentado com uma função associada, esta descrita pela notação provida pelo formalismo através da ferramenta PEPS [Brenner et al. 2007, Czekster et al. 2009]. A interpretação de uma função pode ser vista como avaliação de uma expressão em linguagens de programação, *e.g.*, na linguagem C. Cada comparação é avaliada como *true* (valor 1), ou como *false* (valor 0). O disparo da transição do estado 0 para o estado 1 ocorre com taxa λ se o automato $\mathcal{A}^{(2)}$ esta no estado 0, ou taxa γ se o automato $\mathcal{A}^{(2)}$ esta no estado 2. Se o automato $\mathcal{A}^{(2)}$ esta no estado 1, a transição de 0 para 1 não ocorre, ou seja, taxa igual a zero.



tipo	evento	taxa
loc	e_1	f
sin	e_2	μ
loc	e_3	σ
loc	e_4	δ
loc	e_5	τ

$$f = \left[\left(st\ \mathcal{A}^{(2)} == 0\right) \times \lambda\right] + \left[\left(st\ \mathcal{A}^{(2)} == 2\right) \times \gamma\right]$$

Figura 1. Exemplo de modelo SAN [Sales and Plateau 2009]

A transcrição do modelo para o descritor Markoviano pode ser transparente para o usuário menos experiente, ou pode ser feita diretamente por modeladores mais experientes, o importante é que o modelo será dito estruturado e existem diferentes maneiras de tratar este modelo armazenado de maneira mais eficiente, em matrizes esparsas e operadores tensoriais [Stewart 1994, Czekster 2010].

Dentre os exemplos de aplicações de SAN para modelagem de realidades, tem-se: avaliação de programas paralelos [Baldo et al. 2005], modelagem de sistemas tolerantes a falhas [Bertolini et al. 2004a], modelagem e geração de casos de teste estatísticos [Bertolini et al. 2004b], modelagem de desempenho de escalonamento e balanceamento de carga em sistemas operacionais [Chanin et al. 2006], avaliação de protocolos de redes sem fio [Dotti et al. 2005, Dotti et al. 2011], qualidade de serviço em redes [Czekster et al. 2011, Mokdad et al. 2001], modelagem de equipes em ambientes com desenvolvimento de software distribuído [Czekster et al. 2010a, Fernandes et al. 2011].

3. Detecção do Espaço de Estados Atingíveis em Modelos Estruturados

Uma característica dos formalismos estruturados é a existência de um *espaço de estados total* cujo número de estados é calculado pela combinação dos estados locais de todos os autômatos do modelo, e contido neste, um conjunto de estados válidos, formando o que chamamos de *espaço de estados atingíveis* [Fernandes et al. 1998]. Utilizando álgebra tensorial este fato pouco ou nada afeta o armazenamento das transições a partir de tensores. Entretanto, ao tentar-se resolver o modelo, vetores de probabilidade do tamanho do espaço total são necessários, pois os tensores representam implicitamente uma matriz esparsa da ordem deste espaço de estados total [Buchholz et al. 2000]. Logo, modelagens estruturadas podem introduzir desvantagens no uso de memória para a solução de modelos estocásticos principalmente quando se tratam de modelos com grande diferença entre os espaços de estados total e atingível.

É importante ressaltar que modelos estruturados apresentam uma cadeia de Markov implícita ao seu descritor, oriunda do mapeamento de todos os estados atingíveis do modelo [Fernandes et al. 1998]. Estes podem ser obtidos de forma trivial a partir do disparo de todos os eventos, a partir de todos os estados do modelo, e podem ser armazenados em uma matriz esparsa ou outra estrutura de dados mais complexa para evitar o consumo excessivo de memória. Neste âmbito, o primeiro passo é detectar quais são os estados válidos de um modelo estruturado com grande espaço de estados, onde torna-se difícil realizar manualmente, e armazená-los em alguma estrutura de dados eficiente.

Os estados atingíveis podem ser armazenados através de estruturas arborescentes chamadas de *diagramas de decisão multi-valorados* (do inglês, *Multi-valued Decision Diagrams* - MDD) [Miner and Ciardo 1999b], onde os estados são agrupados de maneira compacta e acessados de forma facilitada através de algoritmos de busca especializados. MDD é um grafo direcionado e acíclico, que apresenta nodos definidos em níveis, sendo estes nodos classificados em não-terminais (níveis mais altos) e terminais (Nível 0).

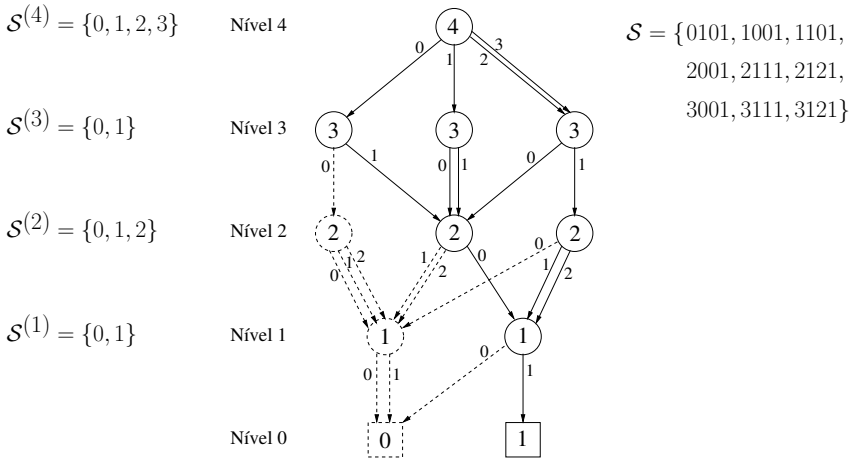


Figura 2. Espaço de estados atingíveis $\mathcal{S}$ representado por um MDD

A Figura 2 mostra a representação do espaço de estados atingíveis $\mathcal{S}$, onde o níveis mais altos representam os autômatos e as transições entre níveis os possíveis estados dos autômatos. O nível mais baixo apresenta dois *nodos terminais*, 0 e 1, representando respectivamente estados inatingíveis e atingíveis. Um elemento de $\mathcal{S}$ é um estado obtido

a partir dos rótulos dos arcos que compõem um caminho partindo do nodo raiz (*i.e.*, nodo do nível mais alto) em direção ao nodo terminal 1. Os nodos e arcos pontilhados formam os elementos que não são representados por $\mathcal{S}$, *i.e.*, são os caminhos que levam ao nodo terminal 0, os quais formam os estados que não são representados por $\mathcal{S}$. Este exemplo representa um modelo com quatro autômatos, sendo cada autômato composto por seus estados individuais $\mathcal{S}^{(i)}$.

Ao tratar-se o espaço atingível armazenando em MDD [Sales and Plateau 2009], torna-se possível a escolha de soluções numéricas adequadas para cada modelo estruturado, considerando características como: tamanho do modelo em termos de espaço total e atingível, bem como memória a ser utilizada pelas estruturas auxiliares.

4. Soluções Numéricas tradicionais para Modelos Estruturados Markovianos

Descritores Markovianos oriundos de modelos estruturados podem ser solucionados através de métodos iterativos [Stewart 1994, Saad 1995] (*e.g.*, método da Potência, Arnoldi, GMRES) que operam a multiplicação de vetores por matrizes na busca de um vetor de probabilidades contendo a solução estacionária (ou transiente) dado uma tolerância para precisão dos resultados [Stewart 1994]. Entretanto, a solução por métodos iterativos é combinada com algoritmos especializados para multiplicação vetor-descritor onde as matrizes operam entre si com álgebra tensorial. Estes algoritmos são implementados de forma a reduzir o número de multiplicações de ponto flutuante na medida que operam apenas sobre elementos não-nulos da estrutura tensorial [Fernandes et al. 1998, Buchholz et al. 2000, Czekster et al. 2007, Czekster et al. 2010b].

Os algoritmos mais utilizados na literatura para multiplicação vetor-descritor são os algoritmos *Shuffle* [Fernandes et al. 1998] e *Split* [Czekster et al. 2010b] que lidam com o descritor de formas distintas acessando os tensores, mas operam igualmente com vetores do tamanho do espaço de estados total, o que se torna um fator limitante nestas soluções dependendo do modelo, principalmente aqueles modelos com espaço atingível conhecido pois este poderia ser mapeado para uma cadeia de Markov bem reduzida em relação ao tamanho do modelo estruturado. Sabe-se que cadeias de Markov, dependendo do número de estados podem ser resolvidas por simples iterações vetor-matriz bem como com outras otimizações relacionadas às propriedades da matriz [Stewart 1994].

Existem avanços recentes realizados no armazenamento de estados válidos em MDD no contexto de Redes de Autômatos Estocásticos [Sales and Plateau 2009]. Entretanto, a solução tradicional ainda precisa ser adaptada para levar em consideração o espaço atingível ao alocar os vetores para a multiplicação vetor-descritor. Mais do que isto, um passo natural para a otimização dos métodos de solução de modelos estruturados pode ser justamente acoplar esta representação MDD para gerar eficientemente a cadeia de Markov adjacente ao modelo, para aqueles casos onde a estruturação favorece a modelagem porém causa gargalos no cálculo dos resultados [Buchholz et al. 2000, Czekster et al. 2007].

5. Solução através do Mapeamento de Transições entre Estados Representados por MDD

O objetivo desta proposta é acelerar o processamento e solução de modelos estruturados através da transcrição do modelo para um descritor Markoviano e respectivo MDD de estados válidos, o que comprovadamente reflete em uma economia de

memória [Sales and Plateau 2009]. Esta estruturação serve para gerar o *descritor de atingibilidade* (do inglês, *Reachability descriptor* - RD) construído a partir deste descritor Markoviano, que combinado ao MDD torna possível aplicar soluções otimizadas para vetor-descritor ou vetor-matriz. O descritor de atingibilidade é uma estrutura também tensorial, criada para facilitar o mapeamento de transições entre os estados na criação do MDD, podendo estabelecer uma interface entre o descritor e o espaço atingível.

Outros trabalhos também utilizam estruturas compactas, *e.g.*, *Diagramas de Matrizes* (do inglês, *Matrix Diagrams* - MxD) para geração e representação das transições do modelo [Miner and Ciardo 1999a, Miner 2002]. Entretanto, na representação por MxD, não são consideradas transições funcionais entre os estados (*i.e.*, transições que podem ou não ocorrer em função do estado de outros subsistemas). Transições funcionais são primitivas de modelagem muito eficientes na descrição de modelos. Sales e Plateau desenvolveram uma representação compacta utilizando MDDs para os estados atingíveis de um modelo SAN permitindo a descrição de transições funcionais [Sales and Plateau 2009].

Os algoritmos implementados neste trabalho priorizam o uso de estruturas de armazenamento esparso, chamadas de *matrizes HBF* (*Harwell-Boeing Format*) [Stewart 1994], onde apenas elementos não-nulos e seus respectivos índices de linha/coluna são considerados. Métodos de acesso e atualização destas estruturas foram implementados para então extrair, a partir da representação estruturada, a cadeia de Markov correspondente à SAN, sem perdas de desempenho quando comparados aos algoritmos que percorrem estado a estado da SAN. A ideia geral para a solução de SAN utilizando esta abordagem pode ser descrita nos seguintes passos:

> **Passo 1:** dado a descrição de um modelo SAN representado por um descritor Markoviano e um estado inicial, calcula-se o espaço de estados atingíveis do modelo representando-o em um MDD [Sales and Plateau 2009]. Para o cálculo destes estados, um descritor de atingibilidade[1] é gerado a partir do descritor Markoviano;
>
> **Passo 2:** a partir deste MDD e utilizando o descritor de atingibilidade, calcula-se para cada estado atingível quais são as transições que levam para os demais estados do modelo e armazenam-se estas transições em uma matriz HBF;
>
> **Passo 3:** utilizando esta matriz de transição em HBF, calcula-se através de um método iterativo (*e.g.*, método da Potência) o vetor de probabilidade resultante do modelo (neste caso, a solução no seu estado estacionário).

É importante ressaltar que o **Passo 1** já foi comprovadamente apresentado como extremamente eficiente em termos de memória e processamento [Sales and Plateau 2009]. Também cabe frisar que se o **Passo 2** fosse feito de maneira ingênua (*i.e.*, analisando quais são as possíveis transições para cada estado possível do modelo) certamente seria o gargalo de todo o processo em termos de tempo. Entretanto, graças ao descobrimento e armazenamento do estados atingíveis do modelo por um MDD, percorremos *apenas* este conjunto de estados para determinarmos as transições do modelo, tornando assim a implementação do algoritmo de obtenção da matriz de transição extremamente eficiente.

[1]Este descritor de atingibilidade é extremamente importante para aceleração na descoberta das transições do modelo. Maiores detalhes sobre o descritor de atingibilidade podem ser encontrados em [Sales and Plateau 2009].

Outro detalhe importante é que além de ser eficiente em memória, a matriz de transição descrita em HBF permite o cálculo de maneira extremamente eficiente da solução do modelo (**Passo 3**).

Finalmente, esta maneira de extrair a solução de modelos torna-se muito flexível, pois reúne eficientes técnicas de armazenamento e representação de modelos com funções, e pode ser usada em ferramentas numéricas para diferentes formalismos estruturados, bastando que os mesmos sejam convertidos para um formato tensorial.

5.1. Experimentos e Análise de Resultados

Os experimentos desta seção foram realizados em uma máquina *Intel Pentium* 3.2 GHz, 4 Gb de RAM e 2 Mb de memória *cache*. O protótipo denominado YAMS (*Yet Another Markovian Solver*) utilizou-se da compilação e geração do descritor Markoviano descrito no ambiente da ferramenta PEPS [Brenner et al. 2007], bem como integra métodos de geração de MDD e do descritor de atingibilidade RD.

As novas rotinas de acesso a estas estruturas foram implementadas de forma a gerar uma matriz de transição HBF com estados válidos a partir de um modelo estruturado, e assim resolvê-la via métodos iterativos clássicos. YAMS foi codificado em C++ e para estes experimentos utilizou-se a versão $4.0.4$ do compilador g++ (GCC – *The GNU Compiler Collection*) com opção de otimização (-O3) e *dynamic linkage*.

Os modelos estruturados em SAN selecionados para os experimentos foram: *Redes ad hoc sem fio* (abreviado como Ad) [Dotti et al. 2005], Primeiro Servidor Disponível - do inglês *First Available Server* (FAS) [Benoit et al. 2004], *Kanban* (abreviado como Kn) [Sales and Plateau 2009], Jantar dos Filósofos - do inglês *Dining Philosophers* (DP) [Sales and Plateau 2009], Compartilhamento de Recursos - do inglês *Resource Sharing* (RS) [Sales and Plateau 2009].

(i) **Redes ad hoc sem fio** (Ad) – este modelo representa um conjunto de nodos móveis em uma rede sem fio sobre o padrão IEEE 802.11 para redes *Ad hoc*, onde um nodo *Fonte* gera pacotes na taxa permitida pelo padrão, com dois estados: *inativo* e *transmitindo*. Os pacotes são repassados na rede pelos nodos do tipo *Retransmissor* com três estados: *inativo*, *recebendo* e *transmitindo*, até o último nodo *Destino*. Este tem dois estados apenas: *inativo* e *recebendo*. O espaço de estados total é dado por $[2^2 \times 3^{N-2}]$ estados. Utiliza-se duas versões deste modelo, com e sem taxas funcionais para os eventos (logo tem-se descritores diferentes mas com a mesma semântica de modelo), variando o número N de nodos na rede.

(ii) **Primeiro Servidor Disponível** (FAS) – modelo que analisa a disponibilidade de servidores considerando uma rede contendo N servidores. Cada servidor tem dois estados: $I^{(i)}$ (inativo) e $B^{(i)}$ (ocupado). Neste modelo, os pacotes chegam e são roteados para os servidores na ordem em que estes estiverem configurados, em sequência, até rotear para um servidor inativo, ou seja, que não está ocupado. O espaço de estados é dado por 2^N estados. Este modelo foi estendido variando o número de servidores N.

(iii) **Kanban** (Kn) – modelo de um sistema de manufatura, com quatro estações (autômatos) semelhantes, onde N *kanbans* (estados que representam rótulos possíveis, do inglês *tags*) controlam o fluxo de peças. O processo apresenta peças que chegam primeiro na estação 1, depois cada peça é separada em duas partes

que entram nas estações 2 e 3, e então, essas duas partes são conectadas em uma única peça novamente que entra na estação 4. Neste ponto, a peça sai do sistema de manufatura. O espaço de estados total é dado por $[(N+1)^{16}]$ e $[(N+1)^{12}]$ estados para as versões com taxas constantes e funcionais respectivamente. Varia-se este modelo pelo número N de kanbans configurado em cada estação.

(iv) **Jantar dos Filósofos** (DP) – modelo do problema clássico de compartilhamento de recursos com exclusão mútua, onde K filósofos sentam em uma mesa circular para comer e pensar, tendo três estados cada: *L* (acessando garfo esquerdo), *R* (acessando garfo direito) ou *T* (pensando). Um filósofo apenas consegue comer se tiver os dois garfos reservados. O espaço de estados total é dado por $[3^K]$ estados. Varia-se este modelo pelo número K de filósofos na mesa.

(v) **Compartilhamento de Recursos** (RS) – modelo de compartilhamento de recursos com diferentes configurações de rede, sendo P o número de processos, com dois estados: *inativo* e *ocupado* - e R como número de recursos compartilhados. O espaço de estados é dado por $[2^P \times (R+1)]$ e $[2^P]$ estados para as versões com taxas constantes e funcionais respectivamente. Este modelo foi estendido para diferentes configurações de P e R.

A Tabela 1 mostra os resultados para os modelos apresentados com diferentes tamanhos de espaço de estados e versões de descritor. A nomenclatura dos modelos é padronizada por siglas, seguida do(s) valor(es) utilizado(s) como variável para estender o modelo, bem como das letras *c* ou *f* para representar um modelo com taxas *constantes* ou *funcional* respectivamente. São mostradas colunas contendo o espaço de estados total (coluna *PSS*) e o número de atingíveis (coluna *RSS*). A seguir, tem-se três colunas com o tamanho das estruturas auxiliares alocadas: memória para o descritor de atingibilidade (coluna *RD*), memória gasta pelo diagrama de decisão multi-valorado (coluna *MDD*) e memória para a matriz HBF (coluna *HBF*).

Analisando os dados numéricos da Tabela 1, nota-se que se as estruturas podem ser alocadas em memória principal, ou seja, se não existe chance de ocorrer *swapping* (na Tabela 2 indicados por "——"), consegue-se obter resultados em tempo satisfatório se comparado ao tempo de execução do PEPS. Por exemplo, para o modelo Ad_20c, com mais de 1,5 bilhão de estados totais e por volta de 12 mil atingíveis foi necessário menos de 840 Kb de memória para armazenar a matriz em HBF, e aproximadamente 75 Kb em estruturas auxiliares (RD e MDD) tanto para o caso constante quanto para o funcional.

A técnica para solução apresentada neste trabalho ressalta que apesar do espaço total (PSS) ser gigantesco, não necessariamente seu espaço de estados atingíveis (RSS) será igualmente massivo. Com a combinação de técnicas de armazenamento e geração automatizada usando estruturas sofisticadas como MDD, observam-se ganhos na solução de modelos estocásticos estruturados de diversas realidades que apresentem um RSS reduzido. Cabe salientar que muitas vezes o modelador não é experiente ao mapear realidades para um modelo. Logo, o espaço de estados total pode vir a ser gigantesco quando comparado com os estados efetivamente atingíveis do modelo.

Um outro caso interessante é o modelo FAS com 23 servidores (FAS_23c e FAS_23f), onde todos os estados são atingíveis. Neste modelo, a matriz HBF possui 1,69 Gb estando dentro de um limite aceitável de uso da RAM, posto que o custo para armazenar as estruturas RD e MDD pouco mais de 50 Kb são necessários.

Tabela 1. Dados de alocação de memória para solução de modelos

modelo	*PSS*	*RSS*	*RD*	*MDD*	*HBF*
Ad_16c	19.131.876	1.766	22,48 Kb	27,32 Kb	105,71 Kb
Ad_16f	19.131.876	1.766	22,16 Kb	32,88 Kb	105,71 Kb
Ad_20c	1.549.681.956	12.102	32,88 Kb	35,18 Kb	839,59 Kb
Ad_20f	1.549.681.956	12.102	32,52 Kb	42,55 Kb	839,59 Kb
FAS_22c	4.194.304	4.194.304	37,07 Kb	15,31 Kb	832 Mb
FAS_22f	4.194.304	4.194.304	27,97 Kb	17,77 Kb	832 Mb
FAS_23c	8.388.608	8.388.608	40,29 Kb	16,36 Kb	1,69 Gb
FAS_23f	8.388.608	8.388.608	30,27 Kb	18,94 Kb	1,69 Gb
Kn_02c	43.046.721	4.600	19,81 Kb	14,34 Kb	511,27 Kb
Kn_02f	531.441	4.600	14,52 Kb	34,60 Kb	511,27 Kb
Kn_03c	4.294.967.296	58.400	20,13 Kb	18,40 Kb	7,70 Mb
Kn_03f	16.777.216	58.400	15,42 Kb	115,16 Kb	7,70 Mb
Kn_04c	152.587.890.625	454.475	20,44 Kb	23,01 Kb	67,66 Mb
Kn_04f	244.140.625	454.475	15,11 Kb	357,81 Kb	67,66 Mb
DP_16c	43.046.721	1.136.689	36,66 Kb	26,29 Kb	194,95 Mb
DP_16f	43.046.721	1.136.689	30,73 Kb	29,97 Kb	194,95 Mb
DP_18c	387.420.489	6.625.109	45,49 Kb	29,72 Kb	1,24 Gb
DP_18f	387.420.489	6.625.109	37,77 Kb	33,92 Kb	1,24 Gb
DP_20c	3.486.784.401	38.613.965	55,27 Kb	33,15 Kb	5,80 Gb
DP_20f	3.486.784.401	38.613.965	45,93 Kb	37,41 Kb	5,80 Gb
RS14_11c	196.608	16.278	34,45 Kb	78,39 Kb	3,71 Mb
RS14_11f	16.384	16.278	12,63 Kb	878,54 Kb	3,71 Mb
RS20_05c	6.291.456	21.700	59,55 Kb	90,58 Kb	3,40 Mb
RS20_05f	1.048.576	21.700	23,03 Kb	10,65 Mb	3,40 Mb

Diferente é a situação dos modelos DP_20c e DP_20f, onde as demandas de memória são grandes demais para que estes sejam resolvidos de maneira eficaz por ambas as ferramentas. Neste exemplo, a matriz HBF sozinha demanda 5.80 Gb, tornando a aplicação da técnica proposta impraticável para o YAMS, assim como já é para o PEPS.

O modelo Kanban (Kn) obteve resultados bastante impressionantes, pois apresenta um grande PSS em relação a um pequeno RSS (vide modelo Kn_04c). Ressalta-se que a matriz HBF ocupa em torno de 68 Mb tornando esta classe de modelo totalmente adequada para aplicação e uso de estruturas de armazenamento otimizadas aliadas a métodos vetor-matriz de solução. Note que para os modelos com PSS superior a 65 milhões de estados (tendo-se 4 Gb de RAM) a solução através da ferramenta PEPS dificilmente é viável, devido à necessidade de alocação de vetores probabilidades de tamanho PSS para realizar a multiplicação vetor-descritor.

A Tabela 2 mostra os tempos gastos na descoberta da solução estacionária das classes de modelo, utilizando o método da Potência, e executando iterações com tolerância igual a $1e$-10. A tabela apresenta os tempos coletados em cada parte do processo de solução, por exemplo, na coluna *RD* mostra-se os tempos para geração do descritor de atingibilidade e a coluna *MDD* mostra o tempo para criar a estrutura MDD em memória. Já a coluna *transições* mostra o tempo necessário para descoberta das transições do modelo (geração da matriz HBF), enquanto que a coluna *matriz* mostra o tempo gasto para salvar a matriz gerada em memória. Por fim, na coluna *YAMS*, temos o tempo para executar o método da Potência até a solução do modelo, e a coluna *PEPS* com o tempo de solução via a ferramenta PEPS para o mesmo número de iterações (coluna *# iterações*)

efetuados no YAMS, para fins comparativos. Todos os tempos estão apresentados em segundos (s), exceto os tempos relativos às execuções na ferramenta PEPS que podem estar apresentados em segundos (s), horas (h) ou dias (d) dependendo do modelo.

Tabela 2. Tempos em segundos (s) coletados para a solução dos modelos

modelo	*RD*	*MDD*	*transições*	*matriz*	*# iterações*	*YAMS*	*PEPS*
Ad_16c	≈0	≈0	0,02	0,02	75.603	2,72	2,87 d
Ad_16f	≈0	≈0	0,03	0,01	75.603	2,74	8,87 d
Ad_20c	≈0	0,01	0,24	0,11	108.918	28,92	—
Ad_20f	≈0	≈0	0,28	0,11	108.918	28,94	—
FAS_22c	≈0	≈0	146,60	75,95	457	181,87	694,80 s
FAS_22f	2,22	≈0	555,31	76,16	457	184,02	1,03 h
FAS_23c	≈0	≈0	220,35	159,08	478	411,92	1.565,40 s
FAS_23f	4,71	≈0	699,86	158,37	478	411,38	2,31 h
Kn_02c	0,49	≈0	0,06	0,04	797	0,13	2,57 h
Kn_02f	≈0	≈0	0,06	0,04	797	0,12	171,60 s
Kn_03c	22,80	≈0	0,95	0,65	1.096	4,31	—
Kn_03f	≈0	≈0	0,97	0,65	1.334	5,25	3,82 h
Kn_04c	438,52	0,01	9,83	5,87	1.448	50,35	—
Kn_04f	0,02	≈0	9,04	5,95	1.448	50,33	—
DP_16c	≈0	≈0	28,18	17,73	1.127	99,10	5,21 h
DP_16f	≈0	≈0	32,96	17,50	1.127	96,80	11,49 h
DP_18c	≈0	≈0	220,71	119,88	1.265	687,65	—
DP_18f	≈0	0,01	253,30	117,12	1.265	704,07	—
DP_20c	≈0	≈0	157,65	92,05	—	—	—
DP_20f	≈0	0,01	182,44	90,81	—	—	—
RS14_11c	≈0	0,01	0,91	0,30	125	0,19	12,37 s
RS14_11f	0,03	0,01	0,72	0,32	125	0,18	11,00 s
RS20_05c	≈0	0,01	3,53	0,29	46	0,07	262,08 s
RS20_05f	0,95	0,02	1,70	0,26	46	0,07	399,46 s

Tabela 2 mostra que, na maioria dos casos, a geração do RD e do MDD foi negligenciável. Dependendo do modelo, gastou-se um tempo considerável determinando as transições existentes, conforme mostra a Tabela 2. Os tempos obtidos são proporcionais ao tamanho do espaço de estados. Para o conjunto de modelos estudados, o tempo de solução mostrou-se bastante reduzido, inclusive para modelos com grande RSS (caso FAS_23c e FAS_23f). No entanto, quando o modelo resulta em um RSS muito grande (DP_20c e DP_20f), tanto a solução do YAMS quanto a do PEPS torna-se impraticável, sendo no YAMS devido ao tamanho do RSS armazenado na matriz HBF. Observa-se também na Tabela 2 que foi gasto um tempo considerável no cálculo das transições do modelo, podendo ser um novo alvo de otimizações em futuras versões do YAMS, onde pode-se empregar outras técnicas de acesso aos tensores e estruturas auxiliares no intuito de acelerar esta operação.

6. Conclusão

O grande desafio dos formalismos propostos para modelagem analítica é justamente proporcionar alto poder de modelagem aliado a um otimizado poder de solução. Uma das grandes vantagens do formalismo SAN por exemplo é prover primitivas de sincronização e dependências funcionais fáceis de serem utilizadas em modelos complexos, pois permitem maior liberdade na composição das transições entre estados de subsis-

temas modelados. Esta estruturação sem dúvida facilita a modelagem de realidades, ainda mais vindo acompanhada da transcrição automática para um descritor Markoviano que se utiliza de álgebra tensorial para armazenamento compacto dos eventos e transições.

A existência de tal mapeamento abriu inúmeras pesquisas no que tange a solução de descritores mantendo tal estrutura tensorial como forma de utilizar menos memória e, em alguns casos, obter bom desempenho em tempo de solução se comparado a uma alternativa tradicional via cadeias de Markov. Neste caso, modelos com grandes espaços de estados atingíveis podem justificar o uso da estruturação como forma de acelerar também a solução. Porém, o que acontece muitas vezes, é que modelos com grandes espaços de estados apresentam espaços atingíveis bem inferiores, o que dificulta o tratamento do modelo estruturado, dado que é preciso operar sobre o espaço total na busca da solução por métodos tradicionais de multiplicação vetor-descritor.

A multiplicação vetor-matriz é simplificada, porém no caso de modelos estruturados, a detecção automática do espaço atingível passa por diversas problemáticas como por exemplo, seu armazenamento e acesso otimizado através de estruturas de dados não-triviais. Ao avançar pesquisas na área, na medida em que foi possível armazenar os estados atingíveis em estruturas econômicas de memória, a solução pode também fazer uso desta vantagem e criar maneiras de gerar a cadeia de Markov implícita de forma eficiente. Pelos experimentos pode-se verificar que na maioria dos casos os custos de gerar estruturas auxiliares como o MDD e outras matrizes foi baixo em relação ao tempo gasto com o processamento tradicional do modelo estruturado na obtenção da solução por métodos iterativos. O tempo de execução reduziu drasticamente por iteração ao operar-se exatamente sobre o espaço atingível e com estruturas de dados otimizadas, comprovando a validade deste trabalho.

Dentre os trabalhos futuros tem-se a proposta de uma ferramenta com tomada de decisão para solucionar modelos estruturados, com amplo espectro de formalismos como entrada, onde seja possível extrair o descritor Markoviano, o MDD de estados atingíveis, o descritor de atingibilidade e o mapeamento para matrizes de transição entre estados atingíveis utilizados na solução. Os métodos de solução não estão restritos aos métodos iterativos, mas também métodos diretos dependendo do número de estados a tratar, ou mesmo simulação para aqueles casos onde a solução numérica não é mais possível via métodos iterativos, mantendo ou não a estruturação proposta inicialmente. Modelos com baixo número de estados atingíveis são o alvo principal deste tipo de solução econômica, pois ao mesmo tempo que modeladores ficam livres para utilizar os recursos do formalismo para diferentes níveis de abstração das realidades, tem-se a possibilidade de resolver modelos grandes em termos de espaço total de estados de forma eficiente e adequada com algoritmos de baixo custo em memória ou com menor impacto computacional.

Referências

Baldo, L., Brenner, L., Fernandes, L. G., Fernandes, P., and Sales, A. (2005). Performance Models for Master/Slave Parallel Programs. *Electronic Notes In Theoretical Computer Science (ENTCS)*, 128(4):101–121.

Benoit, A., Fernandes, P., Plateau, B., and Stewart, W. J. (2004). On the benefits of using functional transitions and Kronecker algebra. *Performance Evaluation*, 58(4):367–390.

Bertolini, C., Brenner, L., Fernandes, P., Sales, A., and Zorzo, A. F. (2004a). Structured Stochastic Modeling of Fault-Tolerant Systems. In *Proceedings of the 12th IEEE/ACM Int. Symposium on Modelling, Analysis and Simulation on Computer and Telecommunication Systems (MASCOTS'04)*, pages 139–146. IEEE Computer Society.

Bertolini, C., Farina, A. G., Fernandes, P., and Oliveira, F. M. (2004b). Test Case Generation using Stochastic Automata Networks: Quantitative Analysis. In *Proc. of the 2nd IEEE Int. Conf. on Software Engineering and Formal Methods*, pages 251–260.

Brenner, L., Fernandes, P., Plateau, B., and Sbeity, I. (2007). PEPS2007 - Stochastic Automata Networks Software Tool. In *Proceedings of the 4th IEEE International Conference on the Quantitative Evaluation of Systems (QEST 2007)*, pages 163–164.

Brenner, L., Fernandes, P., and Sales, A. (2005). The Need for and the Advantages of Generalized Tensor Algebra for Kronecker Structured Representations. *International Journal of Simulation: Systems, Science & Technology (IJSIM)*, 6(3-4):52–60.

Buchholz, P., Ciardo, G., Donatelli, S., and Kemper, P. (2000). Complexity of memory-efficient Kronecker operations with applications to the solution of Markov models. *INFORMS Journal on Computing*, 12(3):203–222.

Buchholz, P. and Dayar, T. (2004). Block SOR for Kronecker structured representations. *Linear Algebra and its Applications*, 386:83–109.

Chanin, R., Corrêa, M., Fernandes, P., Sales, A., Scheer, R., and Zorzo, A. F. (2006). Analytical Modeling for Operating System Schedulers on NUMA Systems. *Electronic Notes in Theoretical Computer Science (ENTCS)*, 151(3):131–149.

Czekster, R. M. (2010). *Solução numérica de descritores Markovianos a partir de reestruturações de termos tensoriais*. PhD thesis, Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS), Brasil.

Czekster, R. M., Fernandes, P., Sales, A., and Webber, T. (2010a). Analytical Modeling of Software Development Teams in Globally Distributed Projects. In *Proc. of the 5th IEEE Int. Conference on Global Software Engineering (ICGSE'10)*, pages 287–296.

Czekster, R. M., Fernandes, P., Sales, A., and Webber, T. (2010b). Restructuring tensor products to enhance the numerical solution of structured Markov chains. In *Proceedings of the 6th International Conference on the Numerical Solution of Markov Chains (NSMC '10)*, pages 36–39, Williamsburg, VA, USA.

Czekster, R. M., Fernandes, P., Sales, A., Webber, T., and Zorzo, A. (2011). Stochastic model for QoS assessment in multi-tier web services. In *International Workshop on Practical Applications of Stochastic Modelling (PASM '11)*, pages 93–109.

Czekster, R. M., Fernandes, P., Vincent, J. M., and Webber, T. (2007). Split: a flexible and efficient algorithm to vector-descriptor product. In *Proc. of the 2nd Int. Conf. on Performance Evaluation, Methodologies and Tools (ValueTools '07)*, Nantes, France.

Czekster, R. M., Fernandes, P., and Webber, T. (2009). GTAexpress: a Software Package to Handle Kronecker Descriptors. In *Proc. of the 6th IEEE Int. Conference on Quantitative Evaluation of Systems (QEST 2009)*, pages 281–282, Budapest, Hungary.

Dotti, F. L., Fernandes, P., and Nunes, C. M. (2011). Structured Markovian Models for Discrete Spatial Mobile Node Distribution. *Journal of the Brazilian Computer Society (JBCS)*, 17(1):31–52.

Dotti, F. L., Fernandes, P., Sales, A., and Santos, O. M. (2005). Modular Analytical Performance Models for Ad Hoc Wireless Networks. In *Proc. of the 3rd IEEE Int. Symp. on Modeling and Optim. in Mobile, Ad Hoc, and Wireless Nets*, pages 164–173.

Fernandes, P., Plateau, B., and Stewart, W. J. (1998). Efficient descriptor-vector multiplication in Stochastic Automata Networks. *Journal of the ACM*, 45(3):381–414.

Fernandes, P., Sales, A., Santos, A. R., and Webber, T. (2011). Performance Evaluation of Software Development Teams: A Practical Case Study. In *International Workshop on Practical Applications of Stochastic Modelling (PASM '11)*, pages 5–21.

Fernandes, P., Vincent, J. M., and Webber, T. (2008). *Perfect Simulation of Stochastic Automata Networks*, volume 5055 of *LNCS*, pages 249–263. Springer-Verlag.

Miner, A. S. (2002). Efficient state space generation of GSPNs using decision diagrams. In *International Conference on Dependable Systems and Networks (DSN 2002)*, pages 637–646, Washington, DC, USA.

Miner, A. S. and Ciardo, G. (1999a). A data structure for the efficient Kronecker solution of GSPNs. In *Proceedings of the 8th International Workshop on Petri Nets and Performance Models*, pages 22–31, Zaragoza, Spain.

Miner, A. S. and Ciardo, G. (1999b). Efficient Reachability Set Generation and Storage Using Decision Diagrams. In *Proc. of the 20th International Conference on Applications and Theory of Petri Nets*, volume 1639 of *LNCS*, pages 6–25. Springer-Verlag.

Mokdad, L., Ben-Othman, J., and Gueroui, A. (2001). Quality of Service of a Rerouting Algorithm Using Stochastic Automata Networks. In *Proceedings of the 6th IEEE Symposium on Computers and Communications*, pages 338–343, Hammamet, Tunisia.

Plateau, B. (1985). On the stochastic structure of parallelism and synchronization models for distributed algorithms. In *Proc. of the 1985 ACM SIGMETRICS Conf. on Measurements and Modeling of Computer Systems*, pages 147–154, Austin, Texas, USA.

Saad, Y. (1995). *Iterative Methods for Sparse Linear Systems*. Boston: PWS Publishing Company.

Sales, A. (2009). *Réseaux d'Automates Stochastiques: Génération de l'espace d'états atteignables et Multiplication vecteur-descripteur pour une sémantique en temps discret*. PhD thesis, Institut National Polytechnique de Grenoble, France. http://tel.archives-ouvertes.fr/tel-00436020/en/.

Sales, A. and Plateau, B. (2009). Reachable State Space Generation for Structured Models which use Functional Transitions. In *Proc. of the 6th IEEE Int. Conference on the Quantitative Evaluation of Systems (QEST 2009)*, pages 269–278, Budapest, Hungary.

Stewart, W. J. (1994). *Introduction to the numerical solution of Markov chains*. Princeton University Press.

Webber, T. (2009). *Reducing the Impact of State Space Explosion in Stochastic Automata Networks*. PhD thesis, Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS), Brasil.