

Implementação Computacional Rigorosa do Princípio de Extensão de Zadeh

Tiago Gonçalves Botelho¹, Onofre Rojas Santos², Sérgio Martins de Souza²

¹ Instituto Federal do Sul de Minas, Campus Muzambinho, Minas Gerais, Brasil

² Universidade Federal de Lavras, Minas Gerais, Brasil

¹ tiago.botelho@cafmuз.gov.br,

² {ors, sergiomartins!@dex.ufla.br;

Abstract. O princípio de extensão de Zadeh é uma ferramenta da lógica fuzzy que permite mapear expressões clássicas em incertas (fuzzy), o que nos permite descrever sistemas incertos de forma natural. As análises via princípio de extensão tradicional, de boa parte das funções oscilantes geram muitos ruídos numéricos de difícil controle. Por esse motivo, desenvolvemos um método computacional que permite aplicar o princípio de extensão para funções não monótonas. Este método se baseia em métodos numéricos de obtenção de raízes de equações, sendo que a aplicação do princípio de extensão é feita de forma “analítica”. Validamos essa metodologia estudando algumas funções não monótonas, com resultados melhores que os encontrados com o formalismo tradicional, baseado nos operadores de máximo e mínimo.

Palavras-chave: Lógica Fuzzy, Princípio de Extensão de Zadeh, Métodos Numéricos.

1 Introdução

Diversos níveis de incerteza e imprecisão estão presentes no cotidiano, e com frequência, o processo de tomada de decisão baseia-se em conceitos vagos, estranhos à lógica clássica, e em parâmetros de natureza subjetiva. Baseada na teoria de conjuntos fuzzy, a lógica fuzzy tem se apresentado como boa alternativa para tratamento de termos incertos, subjetivos e vagos [2], [16].

Uma das ferramentas mais importantes na teoria de conjuntos fuzzy para o tratamento da incerteza é o princípio de extensão de Zadeh. Ele é utilizado para estender operações típicas dos conjuntos clássicos em fuzzy, sendo uma das ideias básicas que promove a extensão de conceitos matemáticos não-fuzzy em fuzzy. Este princípio tem sido estudado e aplicado com sucesso em muitos problemas de aritmética fuzzy [4], [6], [7], sistemas dinâmicos [1], [13] e engenharia [5], [7], [10].

A principal motivação deste trabalho é a utilização de uma técnica que seja capaz de lidar com situações de imprecisão em funções que descrevam oscilações, sem utilizar operações típicas do princípio de extensão, as operações de máximos e mínimos [15]. Na referência [12] foi implementado o princípio de extensão em funções matemáticas gerais, de forma a permitir uma análise dessas funções acrescidas de incertezas em seus parâmetros, e assim verificar a viabilidade do modelo fuzzy em relação aos modelos clássicos já consagrados. No entanto para funções que descrevam oscilações, esta implementação não gera resultados coerentes, em certos sub-intervalos do domínio, que permitam aplicação em uma situação real.

Assim, o objetivo principal do presente trabalho é a implementação computacional do princípio de extensão de Zadeh, utilizando métodos de obtenção de raízes de equações. Desta forma, é possível que se obtenha a imagem de conjuntos fuzzy através de uma função clássica, sem a necessidade de operações de máximos e mínimos, o que produz resultados sem ruídos numéricos nos dados, possibilitando assim uma análise mais realista do processo de fuzzificação.

Para definir se o método desenvolvido está adequado aos objetivos que se destina, é preciso que este método seja validado, a fim de se obter resultados confiáveis que possam ser satisfatoriamente interpretados. A validação foi realizada utilizando funções não monótonas do tipo oscilante.

Este trabalho está organizado da seguinte maneira: Na segunda seção apresenta conceitos sobre o princípio de extensão de Zadeh. A terceira seção é discutido alguns detalhes a respeito da implementação do algoritmo. A seção 4 apresenta resultados da aplicação do algoritmo. Por fim, a seção 5 apresenta as conclusões.

2 Princípio de Extensão de Zadeh

O princípio de extensão é utilizado para obter a imagem de conjuntos fuzzy através de uma função clássica [8], estendendo conceitos da teoria de conjuntos clássicos para a teoria de conjuntos fuzzy. Este princípio surge da necessidade de se aplicar uma função clássica a argumentos imprecisos, sendo indispensável para a estruturação matemática quando se modelam fenômenos envolvendo grande grau de incerteza. A partir de uma função f , é possível aplicar argumentos fuzzy, onde este argumento descreve a distribuição de possibilidade do argumento da função f . Para cada possível valor que a variável da função pode assumir, são aplicados argumentos fuzzy para que se produza a possível imagem, fornecendo também a distribuição de possibilidade dessa imagem. Para determinadas funções pode ocorrer que diferentes valores de entrada sejam mapeados no mesmo valor de saída. Assim, é necessário calcular a possibilidade de cada um dos valores de saída, por meio da combinação dos graus de pertinência para um mesmo valor de saída, isto pode ser feito utilizando o operador *sup*.

Sejam X e Y , dois universos de discurso distintos e seja f uma função de X em Y , $f : X \rightarrow Y$ tal que para cada $x \in X$, $f(x) = y \in Y$. Seja A , um conjunto fuzzy em X , a imagem de A pela função $f(x)$ é um conjunto $B = f(A)$ em Y , cuja função de pertinência é dada por:

$$\mu_B(y) = \sup_{x \in f^{-1}(y)} \mu_A(x). \quad (1)$$

As características descritas a seguir exemplificam alguns conceitos importantes do princípio de extensão:

- A possibilidade de um valor de entrada ser introduzido diretamente para a possibilidade de sua imagem;
- Caso existir múltiplas entradas mapeando um mesmo valor de saída, é necessário realizar a operação *sup* (operação de máxima pertinência) na combinação das possibilidades dos valores de entrada.

A figura 1 apresenta um exemplo de mapeamento da distribuição de possibilidade para uma função não monótona.

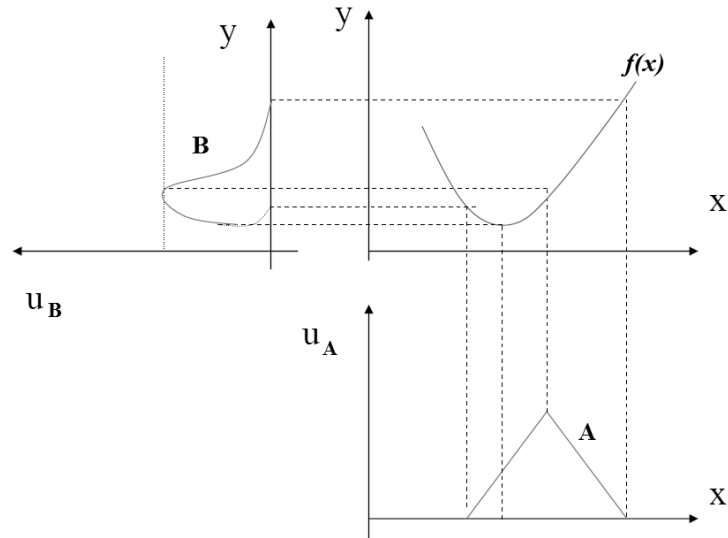


Fig. 1. Exemplo de extensão de uma função não monótona.

Observa-se na figura 1 que para determinados valores de y , existem dois valores de pertinência μ_B e, de acordo com o princípio de extensão, é necessário seleccionar as pertinências máximas de y e descartar as pertinências mínimas. Desta forma teremos de fato executada a operação do princípio de extensão de Zadeh.

A imagem de A por f , representada por B , pode ser deduzida do conhecimento das imagens de x_i por f . Após a obtenção de B , é necessário que o número fuzzy gerado seja transformado em um número crisp. Este procedimento é denominado defuzzificação, e permite interpretar a distribuição de possibilidades da saída de um modelo linguístico fuzzy de forma quantitativa, fornecendo um valor representativo que captura o significado dessa distribuição de possibilidades. Existem muitas técnicas de defuzzificação, a mais utilizada é o centro de gravidade. Este método funciona de forma semelhante a uma média ponderada para uma distribuição de dados, onde retorna a média das áreas que representam os graus de pertinência.

Desta forma $\mu_A(x)$ indica a pertinência para cada valor de x_i de um número fuzzy [14].

$$G(A) = \frac{\sum_{i=0}^n x_i \mu_A(x_i)}{\sum_{i=0}^n \mu_A(x_i)} \quad (2)$$

3 Descrição do algoritmo

Para melhor entendimento do algoritmo, foi adotada a notação algorítmica em forma de diagrama de blocos (figura 2) para ilustração e, posteriormente, a explicação das principais funcionalidades do programa.

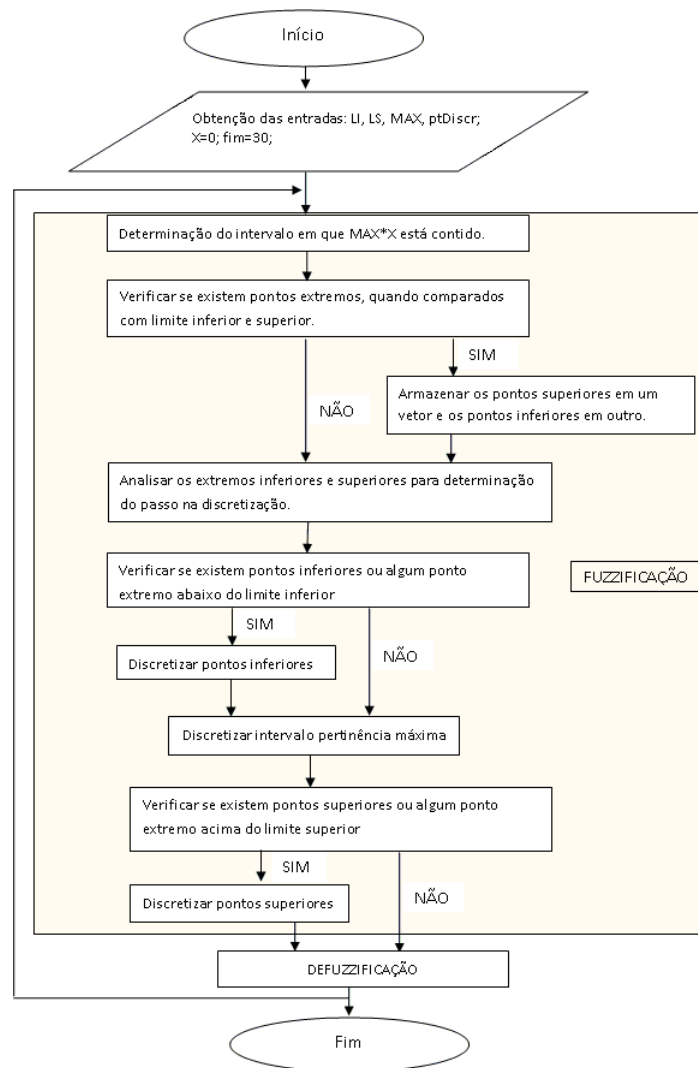


Fig. 2. Figura representativa do algoritmo.

Para realização do trabalho, construímos um algoritmo capaz de aplicar o princípio de extensão de Zadeh, de modo analítico. Na implementação computacional desse algoritmo foi adotada a linguagem de programação C++.

Como exemplo, a figura 3, representando a função $y = \sin(8,5\omega)/(8,5\omega)$ que será utilizado para demonstração do funcionamento do algoritmo para funções oscilantes com amplitude variável. Esta função foi escolhida por ter amplitude diferente, de modo que o algoritmo também pode ser aplicado para funções não monótonas deste tipo.

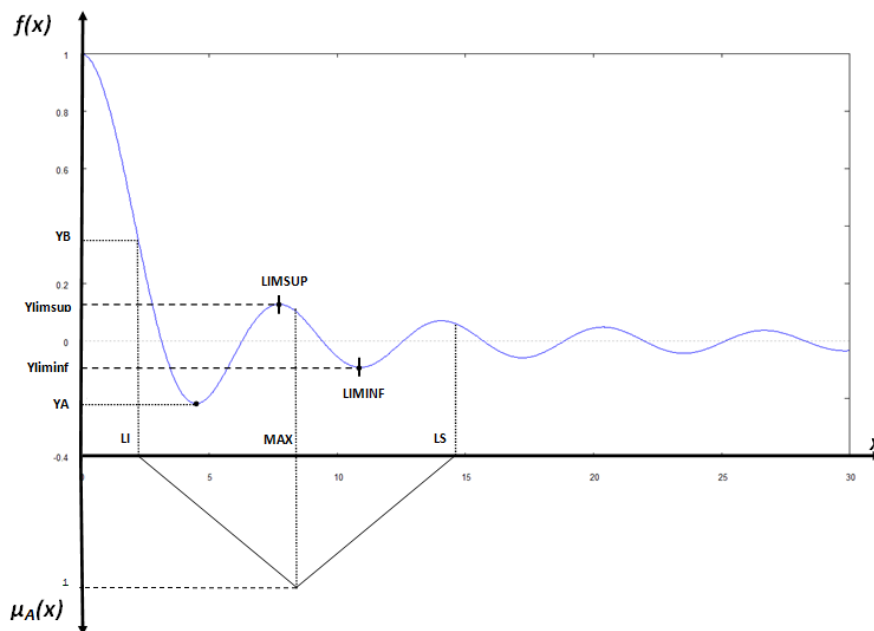


Fig. 3. Representação da função $f(x) = \sin(8,5\omega)/(8,5\omega)$

Etapa 1 - Obtenção dos dados de entrada: Desenvolvemos o programa para manipular números triangulares, entretanto é fácil generalizar para outros tipos de funções não monótonas do tipo oscilante. Nessa fase 1, para implementação da função de pertinência triangular, utilizou-se as seguintes entradas: Limite inferior (LI), Limite superior (LS), pertinência máxima (MAX) e número de pontos a serem discretizados (ptDiscret). Dada a função $f(x)=\sin(x.\omega)/(x.\omega)$, por exemplo, x será a variável a ser fuzzificada, onde será acrescido incerteza a esta variável, aqui representada por ω . A incerteza, inicialmente, abrange o intervalo entre LI, com pertinência 0, MAX com pertinência 1, e LS também com pertinência 0. A variável x irá alterar ao passo de 0,01 entre 0 e 30, estendendo-se assim, ao longo de todo o eixo x . Na figura 3 foi demonstrado um exemplo para um passo, com x valendo 8,5, com $LI=0,27$ e $MAX=1$ e $LS=1,73$ e $ptDiscret=10$. Na etapa seguinte, etapa 2, inicia-se o processo de fuzzificação.

Etapa 2 - Determinação do intervalo em que X.MAX está contido: Nesta etapa do algoritmo, serão analisados os pontos extremos da função, com a finalidade de

encontrar os extremos inferior (LIMINF) e superior (LIMSUP) em que X.MAX está contido, para então mapear este intervalo à imagem função. Para análise de pontos extremos foi aplicado o método iterativo de Newton Raphson, sendo este método considerado mais rápido com relação à convergência [11]. Neste método o valor de x_0 inicial é um valor “próximo” do ponto extremo da função que se espera encontrar. Na obtenção dos resultados, este método utiliza os valores da primeira e segunda derivada da função a ser verificada, e adota-se como critério de parada um valor limite para convergência da função. Por fim, após a obtenção de LIMINF e LIMSUP, é possível calcular os valores para a coordenada $f(x)$, calculando respectivamente, Yliminf e Ylimsup.

$$x_{|n+1|} = x_{|n|} - \frac{f'(x_{|n|})}{f''(x_{|n|})}. \quad (3)$$

Etapa 3 - Verificação se existem pontos extremos, quando comparados com LIMINF e LIMSUP: Determinado o intervalo, para funções de amplitude diferente, é necessário verificar se entre LI e LS existem pontos extremos (de máximo e/ou de mínimo). Além dos limites LIMINF e LIMSUP obtidos na etapa 2, observa-se na figura 3, que existem outros pontos extremos de função além do LIMINF e LIMSUP dentro do intervalo da base da função de pertinência LI e LS. Portanto existe um trecho da função (neste caso, um ponto de mínimo) que também deverá ser mapeado à imagem da função, a variável que representa este ponto de mínimo pode ser identificada na figura 3 como YA. Pontos como este devem ser armazenados em um vetor de pontos extremos inferiores da função, e, caso existam pontos superiores, devem ser armazenados em outro vetor para discretização em etapa posterior.

Etapa 4 - Análise dos extremos inferiores e superiores para determinação do passo na discretização: Nesta etapa analisa-se o eixo $f(x)$, para determinação do passo, aqui denominado Δ , que é o intervalo entre um ponto e seu adjacente, cabe reafirmar que quanto maior o número de pontos a serem discretizados (ptDiscret), menor o Δ e mais preciso será o resultado da fuzzificação. Voltando a figura 3, percebe-se que existe um valor extremo superior, na coordenada $f(x)$, representado por YB. Utiliza-se o número de pontos a serem discretizados para obtenção do Δ , utilizando a fórmula (4):

$$\Delta = \frac{YA - YB}{ptDiscret - 1}. \quad (4)$$

Etapa 5 - Verificação se existem pontos inferiores, ou algum ponto extremo abaixo do LIMINF: Aqui é verificado se existem pontos abaixo do LIMINF, onde será analisado se existe algo no vetor de pontos inferiores. Também pode ocorrer de não se chegar a um ponto extremo, mas estar abaixo do LIMINF ou de algum ponto extremo inferior. Após esta análise, é realizada a discretização com base no Δ . Na figura 3, existe um ponto extremo inferior, o qual a discretização se inicia em YA até um ponto imediato antes de Yliminf no sentido crescente da coordenada x , uma vez que nesta direção a pertinência é maior.

Etapa 6 - Discretização de pontos no intervalo da pertinência máxima (LIMINF e LIMSUP): Esta etapa é responsável pela discretização dentro do intervalo obtido na etapa 4, sempre é executada, podendo ser discretizada ao longo de todo o intervalo, ou não. Caso a verificação da etapa 5 seja verdadeira, o próximo ponto dentro deste intervalo será calculado pela subtração do último ponto discretizado antes de Y_{liminf} pelo Δ . Se a verificação da etapa 5 for falsa, o início da discretização sucederá a partir do primeiro ponto localizado nesta etapa (intervalo de pertinência máxima).

Etapa 7 - Verificação se existem pontos superiores, ou algum ponto extremo acima do LIMSUP: Esta etapa é executada de modo similar a etapa 5, porém, neste caso, os pontos analisados estão acima do LIMSUP. Esta etapa é executada caso existam pontos extremos no vetor de pontos superiores, ou mesmo que não seja ponto extremo superior, mas esteja acima do LIMSUP ou do último ponto extremo do vetor, quando comparado à coordenada $f(x)$. Caso esta análise seja verdadeira, o primeiro ponto dentro deste intervalo será calculado pela subtração do último ponto obtido na etapa 6 pelo Δ . Para a figura 3, a discretização ocorre até o YB. Vale ressaltar que se não existir pontos superiores a Y_{limsup} , é desnecessária a execução desta etapa. Finalizada essa etapa, a etapa de fuzzificação estará concluída. O resultado da fuzzificação pode ser observado na figura 4.

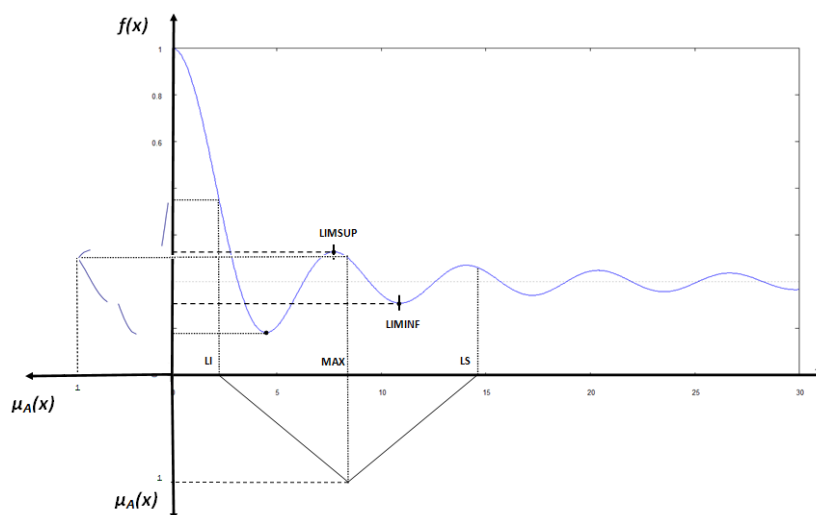


Fig. 4. Extensão da função referente à figura 3.

Etapa 8 - Defuzzificação: Para cada valor de x , existem diferentes valores de $f(x)$, representando assim a função fuzzificada, porém é necessário um valor representativo de $f(x)$. A obtenção desse valor é realizada utilizando um método de defuzzificação. O método de defuzzificação centróide é utilizado para obtenção deste valor representativo, considerando a pertinência e o valor de $f(x)$. A escolha do método de defuzzificação centróide deve-se ao fato de que os resultados obtidos são comparados com outros resultados que utilizaram este método de defuzzificação.

4 Aplicação do método para obtenção do princípio de extensão

Na seção 3 do artigo foi utilizada uma função para fuzzificação com diferentes fases (amplitudes), os resultados de funções deste tipo podem ser vistas na referência [3]. O resultado desta função foi considerado satisfatório devido ao fato que à medida que aumentamos a incerteza, o resultado tende a se afastar do resultado clássico, significando que o processo de fuzzificação ficou bem estruturado. Entretanto não foi encontrada uma implementação que nos forneça condições de comparação com os nossos resultados. Por esse motivo, precisamos de uma função já estudada anteriormente para compararmos os nossos resultados com os resultados obtidos por outros formalismos. Na referência [9] encontramos um conjunto de funções fuzzificadas e, dentre elas, a função $f(x) = \text{sen}(x \cdot \omega)$ pode ser usada para comparação. Vale ressaltar que a intenção deste artigo é mostrar o novo método de obtenção do princípio de extensão, mais exemplos de aplicação deste método são encontrados na referência [3].

Exemplo 4.1: Considere a função, $f(x) = \text{sen}(x \cdot \omega)$. Neste caso, x será a variável a ser fuzzificada e ω será o parâmetro incerto (pertencente ao domínio dos conjuntos fuzzy) obtido a partir das entradas do sistema, em que representará uma função contínua dentro do intervalo entre o limite inferior e superior. Considere na figura 5 que os parâmetros de entrada da função foram: $a = 0,8$; $b = 1$ e $c = 1,2$, resultando em $\alpha_s = 0,2$, valores que representam o parâmetro incerto. Para a variável a ser fuzzificada x vale 15 e 15 pontos foram discretizados. As figuras 5 e 6 são resultantes do método utilizando o princípio de extensão “tradicional”, enquanto a figura 7 é resultante do método “analítico”.

Utiliza-se a estrutura de dados vetor de registros contendo os pontos referentes a x , $f(x)$ e pertinência, os quais são associados aos pares para representar cada ponto nas três funções plotadas na figura 5. Entretanto, para diversos valores de $f(x)$ verifica-se 2 valores de pertinência e isso não está de acordo com o princípio de extensão que afirma que o conjunto imagem resultante do princípio de extensão é outro número fuzzy.

Cada valor de $f(x)$ terá uma única pertinência que corresponde ao valor máximo entre as pertinências de um mesmo $f(x)$. A figura 6 mostra o resultado desta operação, onde são descartadas as pertinências mínimas referentes ao $f(x)$ e mantidas as pertinências máximas.

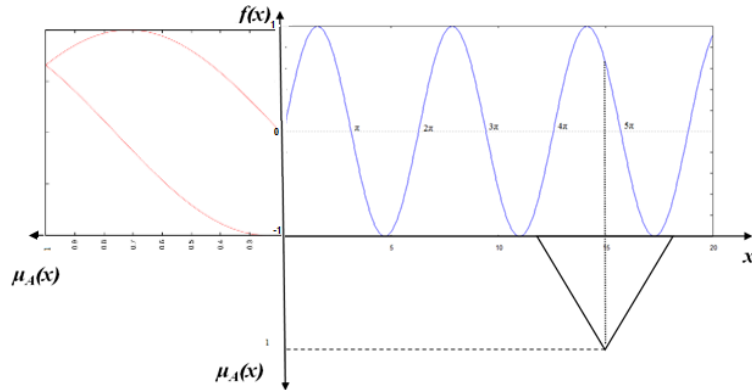


Fig. 5. Processo gráfico da extensão de Zadeh. Obtenção da imagem de $f(x)$ por meio de f para o instante $x=15$.

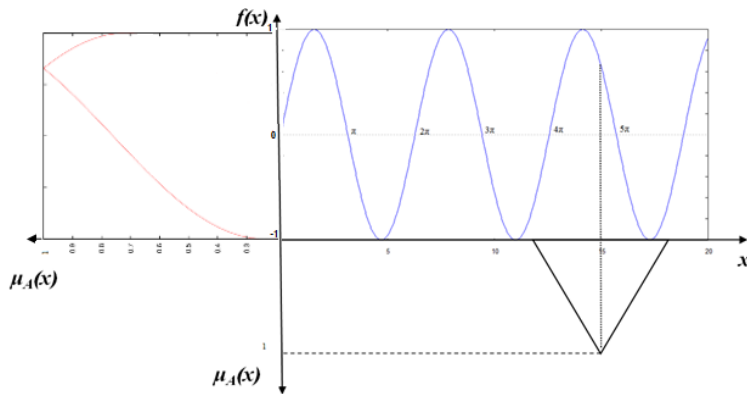


Fig. 6. Número fuzzy $f(x)$ obtido após operação de máxima pertinência.

Para funções deste tipo, é possível também interpretá-las de modo analítico, uma vez que o número fuzzy $f(x)$ obtido nada mais é do que o resultante da delimitação do intervalo entre os picos em máximo e mínimo ou mínimo e máximo em que a pertinência máxima se encontra. Assim, para obtenção da solução utilizando o princípio de extensão de modo analítico, consideramos que a pertinência máxima está em $f(x) = \text{sen}(15 \cdot 1)$, ou seja, está em torno de $f(x) = \text{sen}(15)$ que resultaria $f(x) = 0,6502$.

Fica claro a verificação que em $f(x)$, o valor $0,6502$ está entre intervalos de 1 a -1 ou -1 a 1 . Para obtenção de quais pontos ao longo de x existe neste intervalo, em que $f(x) = 0,6502$ e $x \cdot b = 15$ foi elaborado uma função para obtenção dos mesmos. Esta função retorna um registro denominado “intervalo” contendo os campos limite inferior, limite superior e um contador de quantos intervalos foram procurados, de modo que este contador represente cada π contido na figura 5.

Desta forma obteremos os pares (x_1, y_1) e (x_2, y_2) que representam o intervalo em que o par $(x, b, f(x))$ está contido. O próximo passo é determinar o intervalo entre os pontos com base no número de pontos a serem discretizados que corresponde ao

conjunto suporte $S(A)$ do número fuzzy A . O número de pontos utilizados para este caso é determinado por:

$$\Delta = \frac{y_1 - y_2}{N - 1}. \quad (5)$$

No exemplo 4.1, como usamos 15 pontos na discretização, $y_1 = 1$ e $y_2 = -1$, existe um intervalo de 0,142857 entre cada ponto, partindo do $y_1 = 1$ até $y_2 = -1$. Para determinação dos valores de “ $x.\omega$ ” para os valores de $f(x)$ discretizados, primeiramente verifica-se se o contador de π (*limites.npi*) é par ou ímpar. Para contador de π par, adota-se a seguinte fórmula:

$$x.\omega = \textit{limites.npi} * \pi + \sin^{-1}(y). \quad (6)$$

Para contador ímpar, utiliza-se:

$$x.\omega = \textit{limites.npi} * \pi - \sin^{-1}(y). \quad (7)$$

Assim, para o exemplo 4.1 determinamos que *limites.npi* = 5, $\pi = 3,141593$ e utilizamos o método de inversão de função Newton Raphson, o que gerou a tabela 3 de partições em $f(x)$ baseadas na equação (7):

Tabela 1. Partições de $f(x)$ e obtenção de “ $x.\omega$ ” via inversão de função

| <i>Pontos</i> | $f(x)$ | $x.\omega$ |
|---------------|-----------|------------|
| 1 | 1 | 14,138076 |
| 2 | 0,857143 | 14,678266 |
| 3 | 0,714285 | 14,912360 |
| 4 | 0,571428 | 15,099717 |
| ... | ... | ... |
| 12 | -0,571428 | 16,316208 |
| 13 | -0,714285 | 16,503566 |
| 14 | -0,857143 | 16,737660 |
| 15 | -1 | 17,278020 |

Após determinação dos “ $x.\omega$ ”, calculou-se a pertinência de cada elemento determinado acima fornecendo a função de pertinência os valores “ a ”, “ b ”, “ c ” obtidos como entrada de dados e o valor de “ ω ” de cada elemento, que será determinado pelo “ $x.\omega$ ”/x. Desta forma, a função de pertinência retorna o valor de pertinência que este elemento pertence ao conjunto, onde denominamos $\mu_A(x)$. A figura 7 representa a obtenção do conjunto fuzzy representando a extensão da incerteza.

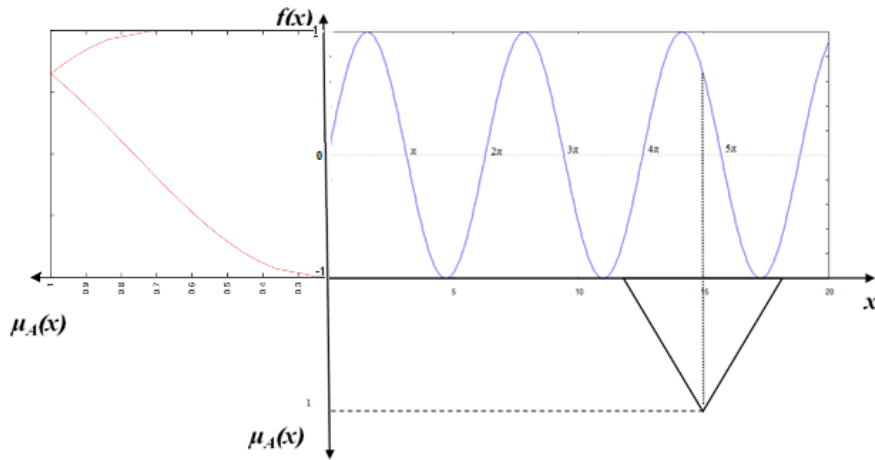


Fig. 7. Obtenção do número fuzzy $f(x)$ obtido via delimitação de intervalo e determinação da função inversa.

A figura 8 é resultado do agrupamento de todos os conjuntos fuzzy obtidos para variável fuzzy $(x.\omega)$ variando de 0 a 30, em intervalos de 0,01, parâmetros citados no exemplo 4.1. Esta figura tem uma escala, onde temos os graus de pertinência representados na escala de cinza variando de 0 a 1. O tom de cor mais escuro representa os valores de $f(x)$, com pertinência máxima igual a 1 em determinado instante " $x.\omega$ ". Esta etapa é denominada fuzziificação.

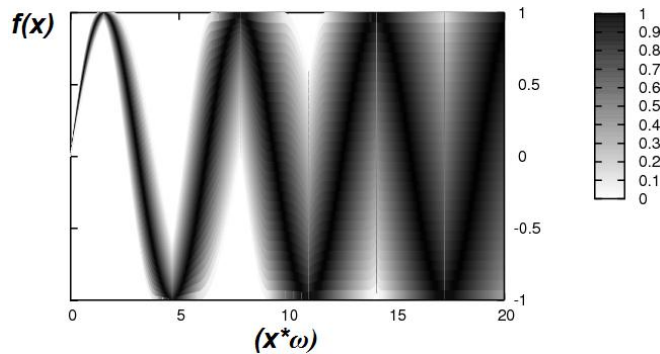


Fig. 8. $f(x) = \text{sen}(x.\omega)$ para $\alpha_s = 0,2$.

Observa-se que, à medida que " $x.\omega$ " aumenta, ocorrem maiores concentrações de incertezas, mostrando assim, que a incerteza propaga de formas diferentes para cada valor da variável fuzziificada x .

Nota-se também que para cada valor de x , existem diferentes valores de $f(x)$, e todos esses valores de $f(x)$ possuem pertinências diferentes, o que representa todos os valores de x sujeitos a incertezas no parâmetro ω .

Para determinação de um único valor de $f(x)$ para cada x , se faz necessário realizar a etapa de defuzzificação, etapa já descrita anteriormente na seção 2. Neste trabalho, foi utilizado o método de defuzzificação centróide, para determinação dos valores de $f(x)$ estimados com parâmetro incerto ω próximo de 1, conforme mostra a figura 9.

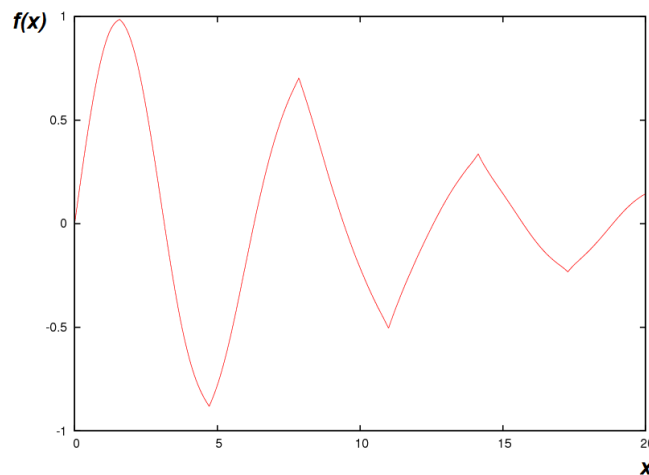


Fig. 9. $f(x) = \text{sen}(x)$ com parâmetro incerto ω próximo de 1 após defuzzificação.

Com este método será obtido a extensão da incerteza do parâmetro “ ω ” a cada valor de “ $f(x)$ ” sem a necessidade de operação de obtenção de máxima pertinência. A ideia deste método, para este caso de função de mesma amplitude e fase, tais como funções seno e cosseno, é que se trabalhe com todos os pontos a serem discretizados. Este método é diferente do método do princípio de extensão tradicional, onde à medida que surge a necessidade de realizar operação de máximos, os pontos são eliminados, observando assim, um intervalo Δ bem maior que este método realiza. Desse modo para um “ $x.\omega$ ” maior, existe “ruído” após a etapa de defuzzificação, e em alguns casos dificulta o trabalho quando se estende essa aplicação para situações reais. Uma comparação entre os dois métodos pode ser vista na figura 10. Esta figura compara o resultado defuzzificado do princípio de extensão tradicional e via método de inversão de função (método desenvolvido neste trabalho).

Na coordenada x , a partir do valor 15 nota-se que se inicia ruído no método utilizando o princípio de extensão tradicional, enquanto o método “analítico” observa-se que em relação às linhas referentes à obtenção da inversa, quanto mais pontos vamos discretizar, mais “amortecida” se torna a função. Isto se deve ao fato da distância entre os pontos discretizados serem menores, o que ocasiona maior precisão e por conseqüência, o amortecimento na função. No entanto esta função chega a uma convergência, onde as funções são plotadas na mesma região. De acordo com os testes realizados, com 60 pontos discretizados a convergência é obtida.

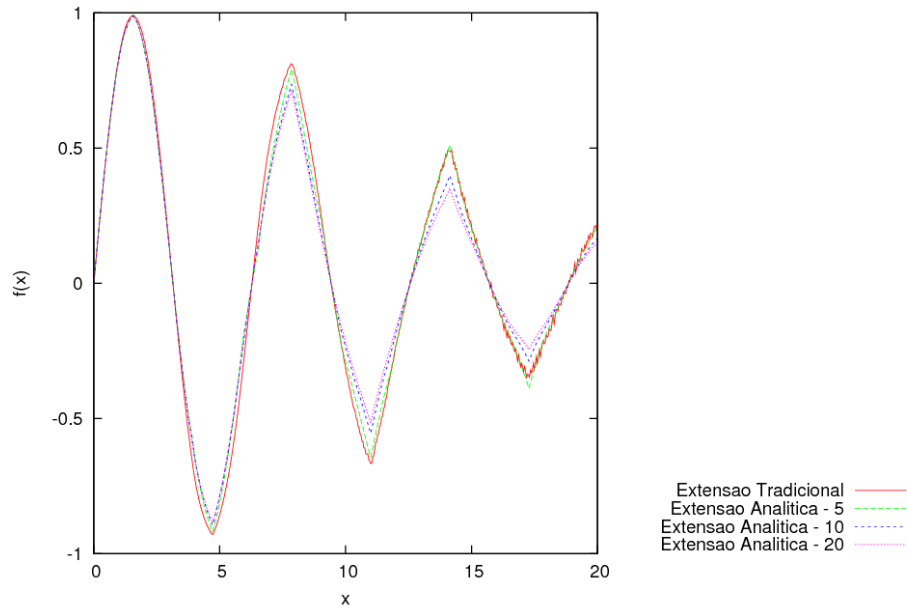


Fig. 10. Comparação método princípio de extensão tradicional e obtenção da extensão por inversa com 5, 10 e 20 números de pontos usados na discretização.

5 Conclusões

Para funções que descrevem oscilações, é possível implementar o princípio de extensão de forma “analítica”, utilizando métodos numéricos de obtenção de raízes de equações. Os resultados obtidos através implementação desta ferramenta, permitem concluir que estes resultados são passíveis de serem aplicados em situações reais. Pode-se concluir também que o processo de fuzzificação ficou bem estruturado, uma vez que para cada valor clássico é encontrado um número fuzzy correspondente.

Como trabalho futuro, pretende-se melhorar a ferramenta, por meio da criação de interface com o usuário e realizar estudo sobre qual método de inversão de função proporcionará melhores convergências para funções não monótonas. Um novo artigo está sendo preparado com resultados da aplicação do princípio de extensão com funções na área de física, difração produzida por fendas, em que se considera a abertura da fenda como parâmetro incerto. Também existem possibilidades de se aplicar esta ferramenta em outros processos físicos com funções que descrevam oscilações, como, por exemplo, o oscilador harmônico.

Referências

1. Barros, L. C., Bassanezi, R. C., Tonelli, P. A.: Fuzzy modelling in population dynamics, *Ecological Modelling* 128, 27--33 (2000)

2. Barros, L. C., Bassanezi, R. C.: *Tópicos de lógica fuzzy e biomatemática*, vol. 5, 354 p, Coleção IMECC – Textos Didáticos, Campinas, Brasil (2006)
3. Botelho, T. G.: Uma nova proposta para implementação computacional do princípio de extensão de Zadeh. Dissertação de Mestrado, Universidade Federal de Lavras, UFLA, Lavras – MG (2012)
4. Castanho, M. J. P., Yamakami, A., Barros L. C., Vendite, L. L.: Modelo matemático fuzzy para descrever o crescimento do câncer de próstata. *Revista de Biomatemática*, vol. 15, n. 15, 41--50 (2005)
5. Fuller, R., Keresztfalvi, T.: On generalization of Nguyen's theorem. *Fuzzy Sets and Systems*, vol. 4, 371--374 (1990)
6. Hanss, M.: *Applied Fuzzy Arithmetic: An Introduction with Engineering Applications*, Springer, Berlin, Germany (2005)
7. Hanss, M.: The transformation method for the simulation and analysis of systems with uncertain parameters. *Fuzzy Sets and Systems*, vol. 130, 277--298 (2002)
8. Jafelice, R. S. M.: Modelagem fuzzy para dinâmica de transferência de soropositivo para HIV em doença plenamente manifesta. Tese de Doutorado, Universidade Estadual de Campinas, UNICAMP Campinas – SP (2004)
9. Lima, F. P.: Ferramenta computacional fuzzy para aplicações do princípio de extensão de Zadeh em funções aritméticas. Dissertação de Mestrado, Universidade Federal de Lavras, UFLA, Lavras – MG (2011)
10. Liu, S.: Solving fuzzy transportation problems based on extension principle. *European Journal of Operational Research*, vol. 153, n. 3, 661--674 (2004)
11. Koh, C. S.: Convergence acceleration of the Newton-Raphson method using successive quadratic function approximation of residual. *IEEE Transactions on Magnetics*, New York, vol. 42, n. 4, 611--614 (2006)
12. Pires, D. M.: Implementação computacional do princípio de extensão de Zadeh. Dissertação de Mestrado, Universidade Federal de Lavras, UFLA, Lavras – MG (2010)
13. Román-Flores, H., Barros, L., Bassanezi, R.: A note on the Zadeh's extensions, *Fuzzy Sets and Systems* vol. 117, 327--331 (2001)
14. Shaw, I., Simões M.: *Controle e modelagem fuzzy*. FAPESP, Editora Edgard. Blucher LTDA, São Paulo, Brasil (1999)
15. Zadeh, L. A.: Generalized theory of uncertainty (GTU)—principal concepts and ideas, *Computational Statistics & Data Analysis*, vol. 51, n. 1, 15--46 (2006)
16. Zadeh, L. A.: *Fuzzy Sets*. *Information and Control*. Berkeley, vol. 8, n. 1, p. 338--353 (1965)