The Value of the Two Values

João Marcos

ABSTRACT. Bilattices have proven again and again to be extremely rich structures from a logical point of view. As a matter of fact, even if one fixes the canonical notion of many-valued entailment and consider the smallest non-trivial bilattice, distinct logics may be defined according to the chosen ontological, epistemological or informational reading of the underlying truth-values. This note will explore the consequence relations of two very natural variants of Belnap's well-known 4-valued logic, and delve into their interrelationship. The strategy will be that of reformulating those logics using only two 'logical values', by way of uniform classic-like semantical and proof-theoretical frameworks, with the help of which such logics may be more easily compared to each other.

1 Introduction

Consider the order-bilattice $(\mathcal{V}, \leq_1, \leq_2)$ where $\mathcal{V} = \{t, \top, \bot, f\}$, the 'truth order' $\leq_1$ has t as its greatest element and f as its least element, as well as intermediate mutually incomparable elements $\top$ and $\bot$, and the 'information order' $\leq_2$ has $\top$ as its greatest element and $\bot$ as its least element, as well as intermediate mutually incomparable elements t and f (see Figure 1). Construing $\mathcal{V}$ as a set of 'truth-values', one may consider the algebraic structures $\mathcal{L}_i = (\mathcal{V}, \wedge_i, \vee_i, \neg_i)$, for $i = 1, 2$, where $\wedge_i$ (resp. $\vee_i$) denotes the meet $\sqcap_i$ (resp. the join $\sqcup_i$) under $\leq_i$, and $\neg_i$ is an order-reversing involution for $\leq_i$ having the intermediate elements, in each case, as fixed-points. It is easy to see that these algebraic structures are 'interlaced', i.e., the operators of $\mathcal{L}_1$ (resp. $\mathcal{L}_2$) are all monotone with respect to $\leq_2$ (resp. $\leq_1$). Even stronger than that, all distributive laws hold between the two meets and the two joins. Morever, $\neg_1$ (called 'negation') and $\neg_2$ (called 'conflation') obviously commute, that is, $\neg_1\neg_2 x = \neg_2\neg_1 x$. Coalescing $\mathcal{L}_1$ and $\mathcal{L}_2$ into a single diagram suggests a strongly symmetric bidimensional structure $\mathcal{B} = (\mathcal{V}, \wedge_1, \wedge_2, \vee_1, \vee_2, \neg_1, \neg_2)$.

Let $\Gamma \cup \Delta$ be a collection of formulas from the term algebra $\mathcal{T}_\mathcal{B}$ freely generated by a denumerable collection of atoms over the connectives (operator symbols) from the structure $\mathcal{B}$, and let Hom denote the set of all homomorphisms, called 'valuations', from $\mathcal{T}_\mathcal{B}$ into $\mathcal{V}$. This is known as a 'truth-functional interpretation', and the meaning of each operator is said to be fixed by a 'truth-table'. There are various notions of entailment that might be associated to the above structure, so that one could talk about an 'inference' that holds, or does not hold, between two given collections of formulas, Γ and Δ. For instance, one could explore the underlying orders once again (they have already been used in defining truth-tables for the connectives), and define, for each dimension i of the above structure its own notion of 'o-entailment' $\models_i^{\mathsf{o}}$, according to which

$\Gamma \models_i^{\mathsf{o}} \Delta$ iff $\bigsqcap_i \mathsf{w}(\Gamma) \leq_i \bigsqcup_i \mathsf{w}(\Delta)$, for every $\mathsf{w} \in \mathsf{Hom}$. A different notion of entailment that is canonically found in the literature on many-valued logics, that will here be called 'p-entailment', assumes some partition of the truth-values $\mathcal{V}_j$ into sets $\mathcal{D}_j$ (called 'designated values') and $\mathcal{U}_j$ (called 'undesignated values'). In that case, the inference $\Gamma \models_j^{\mathsf{p}} \Delta$ is said to hold iff, for every $\mathsf{w} \in \mathsf{Hom}$, either $\mathsf{w}(\Gamma) \cap \mathcal{U}_j \neq \varnothing$ or $\mathsf{w}(\Delta) \cap \mathcal{D}_j \neq \varnothing$. We will almost exclusively be talking about p-entailment, from this point on, and omit accordingly the superscript from $\models^{\mathsf{p}}$ whenever we see no risk of misunderstanding.

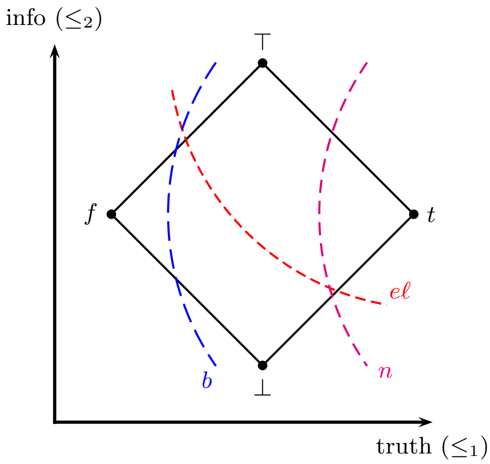

Figure 1. A representative bilattice, sliced in 3 different ways.

It is clear that the canonical many-valued entailment relation is remarkably sensitive to the choice of designated / undesignated values. There are at least three non-trivial such choices that could be made from the viewpoint of the truth order:

[$\mathcal{V}_b$] $\mathcal{D}_b = \{t, \top, \perp\}$ and $\mathcal{U}_b = \{f\}$
[$\mathcal{V}_{e\ell}$] $\mathcal{D}_{e\ell} = \{t, \top\}$ and $\mathcal{U}_{e\ell} = \{\perp, f\}$
[$\mathcal{V}_n$] $\mathcal{D}_n = \{t\}$ and $\mathcal{U}_n = \{\top, \perp, f\}$

Choice [$\mathcal{V}_{e\ell}$] has in fact been intensely investigated in the literature, and the corresponding p-entailment relation, $\models_{e\ell}$, is essentially the same as each of the associated o-entailments. It is known to be adequate for the so-called 'Logic of De Morgan lattices', and some presentations of it are intimately related to a formalization of the so-called 'first-degree entailment'. It is also both paraconsistent and paracomplete. On the other hand, the p-entailment relation $\models_b$, that corresponds to [$\mathcal{V}_b$], is paraconsistent but not paracomplete, and the exact opposite is the case for the p-entailment relation $\models_n$, that corresponds to [$\mathcal{V}_n$]. A reasonable rationale for the choice [$\mathcal{V}_b$], according to the ordinary 'truth-degree interpretation', is that one might be dealing with vague states-of-affairs in which some values should not be ascertained to be 'false', yet they are 'not quite true'. Analogously, for [$\mathcal{V}_n$], there may be other kinds of inexact states-of-affairs in which some values should not be ascertained to be 'true', yet they are 'not quite false' (see Figure 1, again).

The present study will show in more detail what do such entailment relations have in common, and how do they differ from each other. The comparison will be made simpler when the logics involved are recast in terms of semantics and proof-systems that mention only *two* truth-values or *two* syntactic labels, as it happens in classical logic. To enhance the readability of the next sections, comments on the history, the scope and the challenges for our present approach will be left for Section 5. It should be clear at this point, however, that the task that interests us here is the one that is of concern for the logic-designer, much more than for the logic-user, to wit, the task of finding a common *coherent* framework in which it all the above logics can be simultaneously formulated and have their properties contrasted.

2 A Closer Look at the Logical Operators

From the semantical point of view, a logical operator & called *conjunction* is often used to internalize at the object-language level a collection of properties commonly attributed to the metalinguistic 'and', such as:

$[\mathsf{and}_1]$	$\mathsf{w}(\alpha\&\beta) \in \mathcal{D}$	if	$\mathsf{w}(\alpha) \in \mathcal{D}$ and $\mathsf{w}(\beta) \in \mathcal{D}$
$[\mathsf{and}_2]$	$\mathsf{w}(\alpha\&\beta) \in \mathcal{D}$	only if	$\mathsf{w}(\alpha) \in \mathcal{D}$ and $\mathsf{w}(\beta) \in \mathcal{D}$

Similarly, a logical operator || called *disjunction* is often used to internalize properties commonly attributed to the metalinguistic 'or', such as:

$[\mathsf{or}_1]$	$\mathsf{w}(\alpha \mid\mid \beta) \in \mathcal{D}$	if	$\mathsf{w}(\alpha) \in \mathcal{D}$ or $\mathsf{w}(\beta) \in \mathcal{D}$
$[\mathsf{or}_2]$	$\mathsf{w}(\alpha \mid\mid \beta) \in \mathcal{D}$	only if	$\mathsf{w}(\alpha) \in \mathcal{D}$ or $\mathsf{w}(\beta) \in \mathcal{D}$

Obviously, the use of a classical metalanguage, together with the above assumed partition of the truth-values into exactly two classes, allows us to immediately rewrite $[\mathsf{and}_1]$ and $[\mathsf{or}_1]$ as:

$[\mathsf{and}_1]$	$\mathsf{w}(\alpha\&\beta) \in \mathcal{U}$	only if	$\mathsf{w}(\alpha) \in \mathcal{U}$ or $\mathsf{w}(\beta) \in \mathcal{U}$
$[\mathsf{or}_1]$	$\mathsf{w}(\alpha \mid\mid \beta) \in \mathcal{U}$	only if	$\mathsf{w}(\alpha) \in \mathcal{U}$ and $\mathsf{w}(\beta) \in \mathcal{U}$

As it turns out, according to $\models_{e\ell}$, each operator $\wedge_i$ enjoys properties $[\mathsf{and}_1]$ and $[\mathsf{and}_2]$, and each operator $\vee_i$ enjoys properties $[\mathsf{or}_1]$ and $[\mathsf{or}_2]$, for $i = 1, 2$. However, according to either $\models_b$ or $\models_n$, this only holds good for $i = 1$, that is, for the logical operators defined according to the truth-order $\leq_1$. Indeed, for both the latter entailment relations, on what concerns the operators defined according to the information order $\leq_2$, it can easily be checked that $\wedge_2$ enjoys property $[\mathsf{and}_1]$ but fails property $[\mathsf{and}_2]$, while $\vee_2$ enjoys property $[\mathsf{or}_2]$ but fails property $[\mathsf{or}_1]$. One can also say that $\neg_2$ only behaves like a real *negation* according to $\models_{e\ell}$, but not according to $\models_b$ nor according to $\models_n$. Indeed, for the latter entailment relations conflation behaves more like a kind of *identity* relation, in that the formulas φ and $\neg_2\varphi$ are always interderivable (being equivalent, yet not congruent). This much for the similarities between $\models_b$ and $\models_n$. The mixed language of the structure $\mathcal{B}$, where a lot of surprising interactions occur between the original separate algebraic structures, produces other differences and dualities between $\models_b$ and $\models_n$ that go, as we shall see, much beyond the mere contrast between paraconsistent $\times$ paracomplete behavior.

It's not overemphasizing to insist here on the difficulty of the general task of *comparing* two given finite-valued logics just by having a quick look at their truth-tables. It is far from obvious, in fact, how examples of inferences that help in either distinguishing or likening two given logics are even to be found, without exercising some ingenuity. In all such cases, however, the task consists in, given logics $\mathcal{L}_x$ and $\mathcal{L}_y$, finding appropriate collections of formulas, Γ and Δ, such that: (1) $\Gamma \models_x \Delta$ (verified by performing a check ranging over all valuations w_x of $\mathcal{L}_x$) and (2) $\Gamma \not\models_y \Delta$ (verified, in the canonical reading of entailment, by finding a valuation w_y of $\mathcal{L}_y$ such that w_y satisfies all formulas in Γ and simultaneously falsifies all formulas in Δ). Nonetheless, serious difficulties hinder the automation, or even the very accomplishment, of such a task.

Let's assume, for the sake of the argument, that the language of $\mathcal{L}_x$ is at least as rich as the language of $\mathcal{L}_y$ — lest the comparison task is made more difficult by the requirement of some previous massage of the formulas being done by way of previous translations and reinterpretations of the underlying languages. Even based on the same initial language, however, it is not obvious how two distinct given many-valued logics compare to each other. One might think, for instance, of transferring the problem from semantics into proof-theory, with the following reasoning: If both $\mathcal{L}_x$ and $\mathcal{L}_y$ are formulated in terms of rules governing the behavior of their operators, one could in principle use the rules of $\mathcal{L}_x$ to check the derivability of the rules of $\mathcal{L}_y$, or perform an induction on the derivations of $\mathcal{L}_x$ to check whether each rule of $\mathcal{L}_y$ is admissible. But is there a common proof-theoretical language in which all such rules may be expressed? The most well-known automated approaches to finite-valued logics would suggest that this is *not* the case. It is common and straightforward, for instance, to extract signed tableau systems, in each case, from the received truth-tables, by way of a simple trick that transforms the truth-values, or the collections of truth-values, into syntactic signs that are used, say, in front of the formulas. An obvious difficulty for the logic comparison, in that case, appears when $\mathcal{L}_x$ is $\sharp(\mathcal{V}_x)$-valued and $\mathcal{L}_y$ is $\sharp(\mathcal{V}_y)$-valued, for $\sharp(\mathcal{V}_x) \neq \sharp(\mathcal{V}_y)$. In that case, the rules extracted for each logic will be written in different languages, as they will allow in principle for different collections of signs, and will hardly be comparable. At least this specific difficulty would seem to be circumvented, however, in our current case study, where we are concerned about the comparison of different 4-valued logics. One might argue, however, that in this case, where the availability of a common language would not seem to a problem, an even more insidious difficulty slips in. The problem could be described as one that touches upon *coherence* of the whole approach. Indeed, even if we transform the truth-values $\top$ and $\bot$ into signs to be used in expressing the rules of both the logic behind choice $[\mathcal{V}_b]$ and the logic behind choice $[\mathcal{V}_n]$, they could hardly be claimed to have the same significance: While both signs refer to gradations of truth from the viewpoint of $\models_b$, they refer to gradations of falsity from the viewpoint of $\models_n$. Using the same signs, for different logics, as symbols that refer to different entities in their corresponding interpretations, would just confuse the metatheory for the logic-designer and at the same time risk making equivocal the conversation between the users of each logic. It would be as if, for instance, the logic-users employed the same connective symbol to denote, in each case, a different operator — they would certainly experience a lot of difficulty thus in talking to each other about it. That the logic-designer should insist in lifting such misunderstandings to his own high-level framework is likely to make him prone to schizophrenia.

The remainder of the paper will show that there is indeed a strategy that lends itself for a straightforward and fully mechanizable comparison between different many-valued logics, and one that is applicable both to the challenging cases of logics $\mathcal{L}_x$ and $\mathcal{L}_y$ with $\sharp(\mathcal{V}_x) \neq \sharp(\mathcal{V}_y)$ and to the hereby illustrated more subtle cases in which $\sharp(\mathcal{V}_x) = \sharp(\mathcal{V}_y)$ yet $\sharp(\mathcal{D}_x) \neq \sharp(\mathcal{D}_y)$. The next section will show in detail, as an illustration of the application of general methods that will be identified further on, how one of our 4-valued logics may be recast in a classic-like fashion, using only two truth-values. The methods are fully general, and will capitalize in our specific illustrations on the primitive expressivity strength of the language $\mathcal{T}_{\mathcal{B}}$. The new semantics that will originate, as a matter of fact, is to be formulated in such a way that we will be able to show, in the succeeding section, that a classic-like tableau-theoretic presentation is available for that same logic. The chosen formalism will have the property of analyticity, making the associated decision procedures fully automatable.

3 Alternative Bivalent Semantics

The definition of p-entailment that we have been using to characterize the underlying inference relations of our logics in no way depends on the fact that the proposed collections of valuations provide truth-functional interpretations. Indeed, any such inference relation may be determined using only two 'logical values'. Classical Logic, in that case, has the best of both worlds, being truth-functional (and in fact functionally complete) over the collection $\mathcal{V}_2 = \{1, 0\}$, naturally partitioned into 'logical truth' ($\mathcal{D}_2 = \{1\}$) and 'logical falsity' ($\mathcal{U}_2 = \{0\}$). As the following exposition will make clear, however, it is not hard to realize that any other specific p-entailment relation can alternatively be characterized in a similar, classic-like way.

Let's call a 'bivaluation' any mapping of the form $\mathsf{b} : \mathcal{T}_{\mathcal{B}} \longrightarrow \mathcal{V}_2$, and call 'bivalent' any collection Biv of bivaluations. The corresponding notion of p-entailment, that will here be denoted by $\models_{\mathsf{Biv}}$, is defined just as before, but now substituting Biv for Hom. For each $\mathsf{w} \in \mathsf{Hom}$ and associated partition of the truth-values, a 'bivalent counterpart' b_{w} may immediately be defined with the help of the characteristic function $\mathsf{r2} : \mathcal{V} \longrightarrow \mathcal{V}_2$ that takes each designated value into logical truth and each undesignated valued into logical falsity: Just consider the 'bivalent reduction' $\mathsf{b}_{\mathsf{w}} = \mathsf{r2} \circ \mathsf{w}$. There is of course also a bivalent counterpart for the associated entailment relation $\models^{\mathsf{p}}$: Take $\mathsf{Biv} = \{\mathsf{b}_{\mathsf{w}} : \mathsf{w} \in \mathsf{Hom}\}$, and notice that $\Gamma \models_{\mathsf{Biv}} \Delta$ iff $\Gamma \models^{\mathsf{p}} \Delta$. A different question is whether there is anything as convenient, concise and useful as truth-tables that might be used to describe the bivaluation semantics obtained from such a bivalent reduction.

A constructive positive answer to the above question may be provided if the very language of our logic turns out to be expressive enough so as to distinguish between any pair of designated values, and any pair of undesignated values, even after having them 'flattened', in a sense, by the bivalent reduction. In the particular case of $\mathcal{V}_b$, for instance, what we are looking for is a way of distinguishing the three logically true values in $\mathcal{D}_b$ (and similarly, in the case of $\mathcal{V}_n$, for the three logically false values in $\mathcal{U}_n$). This will be possible and easy to describe in case one can find convenient one-variable 'separating formulas' $\circledS_{v_1 v_2}(p)$ to the effect that, given $\mathsf{w}_1, \mathsf{w}_2 \in \mathsf{Hom}$ with $v_1 = \mathsf{w}_1(p) \neq \mathsf{w}_2(p) = v_2$, then $\mathsf{b}_{\mathsf{w}_1}(\circledS_{v_1 v_2}(p)) \neq \mathsf{b}_{\mathsf{w}_2}(\circledS_{v_1 v_2}(p))$. If v_1 and v_2 come from different partitions, one may use p itself as a separating formula. For the case of the logics obtained from choices $[\mathcal{V}_b]$ and $[\mathcal{V}_n]$, in particular, the values t and f may be distinguished from the values $\top$ and $\bot$ through the separating formula $\neg_1 p$. Now, can $\top$ and $\bot$ also be distinguished from each other, using the primitive linguistic resources of $\mathcal{B}$? Yes, they can. Indeed, here is one way of doing it:

p	$\neg_1 p$	$©_n(p) \triangleq p \wedge_1 \neg_2 p$	$©_b(p) \triangleq \neg_1 ©_n(\neg_1 \neg_2(p))$	$\circledS(p) \triangleq ©_b(p \wedge_2 ©_n(p))$
t	f	t	t	t
$\top$	$\top$	f	t	f
$\bot$	$\bot$	f	t	t
f	t	f	f	f

Notice in particular how $©_1$ and $©_2$ might be regarded as characteristic functions of the sets of designated values $\mathcal{D}_n$ and $\mathcal{D}_b$.

Remark 1. *For its expressive power, from this point on we will be working directly with the structure $\mathcal{B}_{\circledS}$, where the definable separating formula $\circledS$ is introduced among the primitive operators. The important thing to bear in mind is that the interpretation of the triple $\langle p, \neg_1 p, \circledS p\rangle$, after the bivalent reduction, gives a unique identification to each $v \in \mathcal{V}_x$, irrespective of the choice $[\mathcal{V}_x]$, so that each truth-value is given a unique 'binary print'.*

It is clear that choice $[\mathcal{V}_{e\ell}]$ is the most symmetric among the three ones we have chosen to distinguish. As a matter of fact, the logic behind this choice may be thought of as a combination of two 'De Morgan lattices', $\langle \mathcal{V}, \wedge_1, \vee_1, \neg_1\rangle$ and $\langle \mathcal{V}, \wedge_2, \vee_2, \neg_2\rangle$, and some classical results about *dualization* of logical operators do easily carry over from the components into the combined structure $\mathcal{B}$. To be more precise, consider the endomorphism ε on $\mathcal{T}_{\mathcal{B}}$ defined by setting:

$$
\begin{array}{rcll}
p^{\varepsilon} & = & p & \text{where } p \text{ is an atom} \\
(\rtimes \alpha)^{\varepsilon} & = & \rtimes(\alpha^{\varepsilon}) & \text{where } \rtimes \in \{\neg_1, \neg_2, \circledS\} \\
(\alpha \oplus \beta)^{\varepsilon} & = & (\alpha^{\varepsilon} \otimes \beta^{\varepsilon}) & \text{where } \langle \oplus, \otimes\rangle \in \{\langle \wedge_i, \vee_i\rangle, \langle \vee_i, \wedge_i\rangle\} \text{ and } i \in \{1,2\}
\end{array}
$$

Then we have the following classic-like result:

Theorem 1. *Given $\Gamma \cup \Delta \subseteq \mathcal{T}_{\mathcal{B}_{\circledS}}$, then $\Gamma \models_{e\ell} \Delta$ iff $\Delta^{\varepsilon} \models_{e\ell} \Gamma^{\varepsilon}$.*

In what follows we will see how such result may easily be proven, and also how it may be extended in order to reveal fascinating connections between our other two distinguished logics.

For all that matters, we will hereupon be concentrating on the choice $[\mathcal{V}_b]$: All our results and considerations will be easily adaptable and dualized for the choice $[\mathcal{V}_n]$. Accordingly, if we use $\neg_1$ and $\circledS$ to express at the object language level the difference between the three designated values behind choice $[\mathcal{V}_b]$, one might arrive to a set of axioms constraining the set of bivalent mappings having domain $\mathcal{T}_{\mathcal{B}_{\circledS}}$ and providing an alternative description of the underlying 4-valued logic, with its original entailment relation. In more precise terms, concerning the collection of bivaluations Biv^b described in the Appendix, the following may be proven:

Theorem 2. Biv^b *provides a sound and complete bivalent semantics for the paraconsistent logic behind choice* $[\mathcal{V}_b]$.

This result is indeed a consequence of the following two lemmas, that can be checked in an entirely constructive fashion. Full details of the corresponding proofs will be exhibited for illustrative cases and subcases.

Lemma 1. *Given* $\mathsf{w} \in \mathsf{Hom}$, *define* $\mathsf{b_w}$ *by:*

$$\mathsf{b_w}(\varphi) = \begin{cases} 0 & \textit{if } \mathsf{w}(\varphi) = f, \\ 1 & \textit{otherwise.} \end{cases}$$

Then, $\mathsf{b_w} \in \mathsf{Biv}^b$.

Proof. One must show that the bivaluation axioms $\mathsf{biv}[xy]\langle v\rangle$ are all respected by this definition. Consider for instance the case in which y is $\neg_2$.

Subcase $\mathsf{biv}[\neg_2]\langle 1\rangle$. Assume $\mathsf{b_w}(\neg_2\varphi) = 1$. By the above definition, this means that $\mathsf{w}(\neg_2\varphi) \neq f$. From the truth-tables of $\mathcal{B}$ one may conclude that $\mathsf{w}(\varphi) \neq f$, and using again the above definition we have that $\mathsf{b_w}(\varphi) = 1$, as desired.

Subcase $\mathsf{biv}[\neg_2]\langle 0\rangle$. Assume $\mathsf{b_w}(\neg_2\varphi) = 0$. By the definition of $\mathsf{b_w}$, $\mathsf{w}(\neg_2\varphi) = f$, and the 4-valued interpretation of $\neg_2$ tells us that $\mathsf{w}(\varphi) = f$. Here, from the definition of $\mathsf{b_w}$ we can say that $\mathsf{b_w}(\varphi) = 0$.

Subcase $\mathsf{biv}[\neg_1\neg_2]\langle 1\rangle$. Assume $\mathsf{b_w}(\neg_1\neg_2\varphi) = 1$. The definition of $\mathsf{b_w}$ gives us $\mathsf{w}(\neg_1\neg_2\varphi) \neq f$. But the truth-tables of $\neg_1$ and of $\neg_2$ inform us that $\mathsf{w}(\neg_1\neg_2\varphi) \neq f$ iff $\mathsf{w}(\neg_1\varphi) \neq f$. So, using the definition of $\mathsf{b_w}$ again, $\mathsf{b_w}(\neg_1\varphi) = 1$.

Subcase $\mathsf{biv}[\neg_1\neg_2]\langle 0\rangle$. Analogous to the previous subcase.

Subcase $\mathsf{biv}[Ⓢ\neg_2]\langle 1\rangle$. Assume $\mathsf{b_w}(Ⓢ\neg_2\varphi) = 1$. From the definition of $\mathsf{b_w}$, we have that $\mathsf{w}(Ⓢ\neg_2\varphi) \neq f$. The truth-table of Ⓢ tells us that $\mathsf{w}(Ⓢ\neg_2\varphi) = t$ and also that it must be the case that $\mathsf{w}(\neg_2\varphi) \in \{t, \bot\}$, and we should check next what the truth-table of $\neg_2$ has to tell us. The first option, $\mathsf{w}(\neg_2\varphi) = t$, is equivalent to writing that (i) $\mathsf{w}(\varphi) = f$; the second option, $\mathsf{w}(\neg_2\varphi) = \bot$, is equivalent to (ii) $\mathsf{w}(\varphi) = \top$. From the interpretation of Ⓢ and (ii) it follows that (iii) $\mathsf{w}(Ⓢ\varphi) = f$. Now, the definition of $\mathsf{b_w}$ allows us to say, in case (i), that (iv) $\mathsf{b_w}(\varphi) = 0$, and to say, in case (ii)+(iii), that (v) $\mathsf{b_w}(\varphi) = 1$ and $\mathsf{b_w}(Ⓢ\varphi) = 0$. Conversely, (i) follows from (iv) and the definition of $\mathsf{b_w}$. Similarly from (v) and the definition of $\mathsf{b_w}$ we may conclude that both $w(\varphi) \neq f$ and $w(Ⓢ\varphi) = f$, and in this case the truth-table of Ⓢ guarantees that we are talking about a situation in which (ii) holds good.

Subcase $\mathsf{biv}[Ⓢ\neg_2]\langle 0\rangle$. Analogous to the previous subcase.

The proofs for the cases of the other connectives follow analogous patterns. The cases in which y is a binary connective, in fact, are quite similar to the latter two subcases. In simplifying the proofs of the corresponding converses it will often be useful to establish and use the following auxiliary facts:

$\mathsf{w}(\varphi) = \top$	iff	$\mathsf{b_w}(\varphi) = 1$ and $\mathsf{b_w}(Ⓢ\varphi) = 0$
$\mathsf{w}(\varphi) = \bot$	iff	$\mathsf{b_w}(\neg_1\varphi) = 1$ and $\mathsf{b_w}(Ⓢ\varphi) = 1$
$\mathsf{w}(\varphi) \in \{\top, \bot\}$	iff	$\mathsf{b_w}(\varphi) = 1$ and $\mathsf{b_w}(\neg_1\varphi) = 1$

Finally, one must also verify the bivaluation axioms $\mathsf{biv}[mn]$. The case in which m is T is guaranteed to hold good, using the above definition, from the fact that each homomorphism w is total. Additionally, the cases in which m is C all reflect, from a bivalent perspective, semantic assignments that are unobtainable from the viewpoint of truth-tables. For $n = 0$, all one needs to guarantee is that each $\mathsf{b_w}$ is a function —and so it must be, as the characteristic mapping of the function w. Moreover, given that it is impossible, for instance, to attribute the values $\mathsf{w}(\varphi) = f$ and $\mathsf{w}(\neg_1\varphi) = f$, given the truth-table of $\neg_1$, the above definition tells us that $\mathsf{b_w}(\varphi) = 0$ and $\mathsf{b_w}(\neg_1\varphi) = 0$ cannot simultaneously obtain, proving thus $\mathsf{biv}[C1]$. Similarly for $n \in \{2, 3\}$. ■

Lemma 2. *Given* $\mathsf{b} \in \mathsf{Biv}^b$*, define* $\mathsf{w_b}$ *by:*

$$\begin{array}{lll} \mathsf{w_b}(\varphi) = t & \textit{if} & \mathsf{b}(\neg_1\varphi) = 0 \\ \mathsf{w_b}(\varphi) = \top & \textit{if} & \mathsf{b}(\varphi) = 1 \textit{ and } \mathsf{b}(Ⓢ\varphi) = 0 \\ \mathsf{w_b}(\varphi) = \bot & \textit{if} & \mathsf{b}(\neg_1\varphi) = 1 \textit{ and } \mathsf{b}(Ⓢ\varphi) = 1 \\ \mathsf{w_b}(\varphi) = f & \textit{if} & \mathsf{b}(\varphi) = 0 \end{array}$$

Then, [A] $\mathsf{w_b}(\varphi) \in \mathcal{D}$ *iff* $\mathsf{b}(\varphi) = 1$*. Moreover,* [B] $\mathsf{w_b} \in \mathsf{Hom}$.

Proof. To check statement [A], the only non-obvious cases are those in which $\mathsf{w_b}(\varphi) \in \{t, \bot\}$, that is, the cases in which either (a) $\mathsf{b}(\neg_1\varphi) = 0$, or (b) both $\mathsf{b}(\neg_1\varphi) = 1$ and $\mathsf{b}(Ⓢ\varphi) = 1$. On what concerns (a), $\mathsf{biv}[C1]$ guarantees that $\mathsf{b}(\varphi) = 0$ cannot be the case. As for (b), $\mathsf{biv}[C2]$ and $\mathsf{b}(Ⓢ\varphi) = 1$ guarantee again that $\mathsf{b}(\varphi) = 0$ is not the case. In both situations we must conclude from $\mathsf{biv}[T0]$ that $\mathsf{b}(\varphi) = 1$.

For statement [B] one has to check that $\mathsf{w_b}$ is well-defined as a 4-valued homomorphism, according to the corresponding truth-tables that interpret each operator from the language of $\mathcal{T}_{\mathcal{B}_Ⓢ}$. Let's consider in detail the particular case in which φ has the form $\alpha \wedge_2 \beta$.

Subcase [$\mathsf{w_b}(\varphi) = t$]. By the above definition of $\mathsf{w_b}$, this is the same as asserting that $\mathsf{b}(\neg_1(\alpha \wedge_2 \beta)) = 0$. But in this case, the bivaluation axiom $\mathsf{biv}[\neg_1\wedge_2]\langle 0\rangle$ guarantees that this is a situation in which either (a) $\mathsf{b}(\neg_1\alpha) = 0$ and $\mathsf{b}(\neg_1\beta) = 0$, or (b) $\mathsf{b}(\neg_1\alpha) = 0$ and $\mathsf{b}(\beta) = 1$ and $\mathsf{b}(Ⓢ\beta) = 0$; or else (c) $\mathsf{b}(\alpha) = 1$ and $\mathsf{b}(Ⓢ\alpha) = 0$ and $\mathsf{b}(\neg_1\beta) = 0$. Using again the definition of $\mathsf{w_b}$, we see that this corresponds to having either (aw) $\mathsf{w_b}(\alpha) = t$ and $\mathsf{w_b}(\beta) = t$, or (bw) $\mathsf{w_b}(\alpha) = t$ and $\mathsf{w_b}(\beta) = \top$; or else (cw) $\mathsf{w_b}(\alpha) = \top$ and $\mathsf{w_b}(\beta) = t$. But this is in accordance to what we desired, as it describes exactly the three pairs of inputs for which the truth-table of $\wedge_2$ outputs the value t.

Subcase [$\mathsf{w_b}(\varphi) = \top$]. This time, by the definition of $\mathsf{w_b}$, we know that both (a) $\mathsf{b}(\alpha \wedge_2 \beta) = 1$ and (b) $\mathsf{b}(Ⓢ(\alpha \wedge_2 \beta)) = 0$. From (a) and $\mathsf{biv}[\wedge_2]\langle 1\rangle$, we are left with three situations to consider. Combining them with what we obtain from (b) and $\mathsf{biv}[Ⓢ\wedge_2]\langle 0\rangle$, and in view of $\mathsf{biv}[T0]$, we are left with but one situation, in which (c) $\mathsf{b}(\alpha) = 1$ and $\mathsf{b}(Ⓢ\alpha) = 0$, and also (d) $\mathsf{b}(\beta) = 1$ and $\mathsf{b}(Ⓢ\beta) = 0$. So, from the definition of $\mathsf{w_b}$, we must conclude that $\mathsf{w_b}(\alpha) = \top$ and $\mathsf{w_b}(\beta) = \top$, again in accordance with the truth-table of $\wedge_2$.

Subcase [$\mathsf{w_b}(\varphi) = \bot$]. Analogous to the previous case, but now using $\mathsf{biv}[\neg_1\wedge_2]\langle 1\rangle$ and $\mathsf{biv}[Ⓢ\wedge_2]\langle 1\rangle$ to conclude that either $\mathsf{w_b}(\alpha) = \bot$, or $\mathsf{w_b}(\beta) = \bot$, or else $\langle \mathsf{w_b}(\alpha), \mathsf{w_b}(\beta)\rangle \in \{\langle t, f\rangle, \langle f, t\rangle\}$.

Subcase [$\mathsf{w_b}(\varphi) = f$]. Use $\mathsf{biv}[\wedge_2]\langle 0\rangle$, and reason as before.

The analysis follows the same pattern for the case of the other operators, using in each case the appropriate bivaluation axioms. One should still check in separate, however, the cases in which φ has the form $\rtimes p$, where $\rtimes p$ represents a separating formula applied to an atom, that is, $\rtimes \in \{\neg_1, Ⓢ\}$.

Consider first the case in which φ has the form $\neg_1 p$.

Subcase [$\mathsf{w_b}(\varphi) = t$]. Using the above definition of $\mathsf{w_b}$, this is to say that $\mathsf{b}(\neg_1\neg_1 p) = 0$. But in this case the bivaluation axiom $\mathsf{biv}[\neg_1\neg_1]\langle 0\rangle$ says that we are exactly in a situation in which $\mathsf{b}(p) = 0$. The definition of $\mathsf{w_b}$ guarantees that such is the case iff $\mathsf{w_b}(p) = f$.

Subcase [$\mathsf{w_b}(\varphi) = \top$]. In that case, the definition of $\mathsf{w_b}$ says that we are in a situation in which both (a) $\mathsf{b}(\neg_1 p) = 1$ and (b) $\mathsf{b}(Ⓢ\neg_1 p) = 0$. However, (b) informs us, given the bivaluation axiom $\mathsf{biv}[Ⓢ\neg_1]\langle 0\rangle$, that either (c) $\mathsf{b}(\neg_1 p) = 0$ or else (d) both $\mathsf{b}(p) = 1$ and $\mathsf{b}(Ⓢp) = 0$. Now, we know that (a) and (c) are jointly untenable, given axiom $\mathsf{biv}[T0]$. The only option left, (d), means, by the definition of $\mathsf{w_b}$, that we have $\mathsf{w_b}(p) = \top$.

Subcase [$\mathsf{w_b}(\varphi) = \bot$]. Follows the same line as the two previous subcases, but now using $\mathsf{biv}[\neg_1\neg_1]\langle 1\rangle$ and $\mathsf{biv}[Ⓢ\neg_1]\langle 1\rangle$.

Subcase [$\mathsf{w_b}(\varphi) = f$]. Immediate from the definition of $\mathsf{w_b}$, as we have $\mathsf{w_b}(\neg_1 p) = f$ iff $\mathsf{b}(\neg_1 p) = 0$ iff $\mathsf{w_b}(p) = t$.

Consider at last the case in which φ has the form $Ⓢp$.

Subcase [$\mathsf{w_b}(\varphi) = t$]. By the definition of $\mathsf{w_b}$, this is exactly the situation in which $\mathsf{b}(\neg_1 Ⓢp) = 0$. Using the bivaluational axiom $\mathsf{biv}[\neg_1 Ⓢp]\langle 0\rangle$, this corresponds to (i) $\mathsf{b}(Ⓢp) = 1$. From the definition of $\mathsf{w_b}$, this already guarantees that (ii) $\mathsf{w_b}(p) \neq \top$. Now, from $\mathsf{biv}[C2]$ and (i) one may also conclude that (iii) $\mathsf{b}(p) \neq 0$, and from this $\mathsf{biv}[T0]$ gives us (iv) $\mathsf{b}(p) = 1$. Recalling statement [A], from (iv) we may conclude that $\mathsf{w_b}(p) \in \{t, \top, \bot\}$. Taking (ii) into account, we're left with $\mathsf{w_b}(p) \in \{t, \bot\}$.

Subcase [$\mathsf{w_b}(\varphi) = \top$]. By the definition of $\mathsf{w_b}$ this would imply that both (i) $\mathsf{b}(Ⓢp) = 1$ and (ii) $\mathsf{b}(ⓈⓈp) = 0$. From (ii) and $\mathsf{biv}[ⓈⓈ]\langle 0\rangle$ we conclude (iii) $\mathsf{b}(Ⓢp) = 0$, contradicting (i) in view of $\mathsf{biv}[C0]$.

Subcase [$\mathsf{w_b}(\varphi) = \bot$]. Again impossible, as the previous subcase. Use $\mathsf{biv}[ⓈⓈ]\langle 1\rangle$.

Subcase [$\mathsf{w_b}(\varphi) = f$]. Here the definition of $\mathsf{w_b}$ puts us in the situation in which (i) $\mathsf{b}(Ⓢp) = 0$ and also tells us, in such situation, that $\mathsf{w_b}(p) \neq \bot$. In one direction, the bivaluation axiom $\mathsf{biv}[C3]$ uses (i) to establish that (ii) $\mathsf{b}(\neg_1 p) \neq 1$, from which $\mathsf{biv}[T0]$ informs us that (iii) $\mathsf{b}(\neg_1 p) = 1$. The latter rules out, from the definition of $\mathsf{w_b}$, the possibility that $\mathsf{w_b}(p) = t$. So, we're left with $\mathsf{w_b}(p) \in \{\top, f\}$. In the other direction, from $\mathsf{biv}[T0]$ we know that either (iv) $\mathsf{b}(p) = 0$ or (v) $\mathsf{b}(p) = 1$. The definition of $\mathsf{w_b}$ uses (iv) by itself to say that $\mathsf{w_b}(p) = f$, and uses (i) and (v) together to say that $\mathsf{w_b}(p) = \top$.

In both the latter cases and all their subcases we see that $\mathsf{w_b}(\rtimes p)$ behaves in accordance to the corresponding truth-tables. ■

While the 4-valued truth-functional presentation of the logic behind choice [$\mathcal{V}_b$] brings along an immediate associated decision procedure in terms of truth-tables, it might not be obvious at first glance at its apparently complicated bivalent presentation that an alternative such procedure is also available in such case. The next section will show though how that can be achieved, once we associate a very simple analytic classic-like proof system to the proposed bivalent semantics.

4 A Uniform Tableau-theoretic Framework

Given the specific format in which the bivaluation axioms for choice [$\mathcal{V}_b$] were presented, it is easy to extract from them a two-signed tableau system that can be used to check the inferences of the corresponding logic. The restrictions on the bivaluations that appear in the Appendix may indeed be converted into tableau rules in the following manner:

(P0) each expression of the form $\mathsf{b}(\varphi) = v$ is rewritten as a signed formula $S_v{:}\varphi$, using one of the two signs $S_1 = T$ or $S_0 = F$;

(P1) a '$\Leftrightarrow$' is read as separating the head of a tableau rule (to the left) from its conclusions (to the right);

(P2) each ',' is understood as separating nodes (signed formulas) from the same branch;

(P3) an '$|$' at the right of a '$\Leftrightarrow$' demarcates bifurcations in the output of a given rule;

(P4) an expression of the form '$h_1, \ldots, h_n \Rightarrow \divideontimes$' denotes a closure rule.

After such a conversion, for each bivaluation axiom $\mathsf{biv}[xy]\langle v\rangle$ there will correspond a tableau rule $\mathsf{tab}[xy]\langle S_v\rangle$, and for each axiom $\mathsf{biv}[Cn]$ there will be, in the tableau system, a corresponding closure rule $\mathsf{tab}[Cn]$. The standard decision procedure that will be presented below shows that the tableau rule that corresponds to $\mathsf{biv}[T0]$ (a kind of dual-cut rule for tableaux), while certainly admissible, does not need to be taken as primitive in the proof system. Now, while the closure rule $\mathsf{tab}[C0]$ is also to be found in two-signed tableaux for classical logic, rules $\mathsf{tab}[C1]$ to $\mathsf{tab}[C3]$ are all distinctive marks of the 4-valued choice $[\mathcal{V}_b]$. We will call Tab^b the tableau system thereby obtained, and use for tableaux here the usual terminology and definitions that are found in the standard literature of the area.

Theorem 3. Tab^b *provides a sound and complete proof system for the logic behind choice* $[\mathcal{V}_b]$*.*

Using the bivalent semantics discussed in the previous section, one direction of this result is obvious, as the tableau rules directly translate bivaluation axioms. To check the converse direction, it will be useful to define a convenient 'complexity measure' $\mathsf{cm} : \mathcal{T}_{\mathcal{B}_{\circledS}} \longrightarrow \mathbb{N}$ over the structure of a formula φ, according to which:

$$\mathsf{cm}(\varphi) = \begin{cases} 0 & \text{if } \varphi \text{ is an atom } p, \\ \mathsf{cm}(\alpha) & \text{if } \varphi \text{ is } \ltimes\alpha \text{ for a separating connective } \ltimes, \\ 1 + \mathsf{cm}(\alpha) & \text{if } \varphi \text{ is } \rtimes\alpha \text{ for any other unary connective } \rtimes, \\ 1 + \mathsf{Max}\,(\mathsf{cm}(\alpha), \mathsf{cm}(\beta)) & \text{if } \varphi \text{ is } \alpha \bowtie \beta \text{ for some binary connective } \bowtie . \end{cases}$$

To extend the complexity measure for signed formulas, just ignore the signs. Now the following lemma suffices to complete the proof of the above result:

Lemma 3. *For any collection of signed formulas it is always possible to produce an exhausted tableau in* Tab^b*.*

Proof. Just notice that every signed formula with non-null complexity, whichever form it has, is the head of a (uniquely determined) tableau rule, and that the body of such rule only contains formulas of lower complexity. ■

As a corollary of the previous result we also immediately obtain a completely standard tableau-theoretic decision procedure for our logic. Indeed, if one wants to check whether $\{\gamma_1, \gamma_2, \ldots, \gamma_m\} \models \{\delta_n, \ldots, \delta_2, \delta_1\}$ is the case, one should construct a tableau for $\{T{:}\gamma_1, T{:}\gamma_2, \ldots, T{:}\gamma_m\} \cup \{F{:}\delta_n, \ldots, F{:}\delta_2, F{:}\delta_1\}$. If it closes at any moment, the inference is valid. Otherwise, when one arrives to an exhausted tableau and it still has open branches, counter-models may be extracted from such branches by collecting all the thereby occurring formulas of complexity 0.

As an illustration, an exhausted tableau to test the validity of the inference $\{\neg_2 p \wedge_1 \neg_2 q)\} \models \{\neg_1(p \vee_2 q)\}$ will use the tableau rules $\mathsf{tab}[\neg_2\vee_2]\langle F\rangle$, $\mathsf{tab}[\wedge_1]\langle T\rangle$, $\mathsf{tab}[\neg_1\neg_2]\langle F\rangle$, $\mathsf{tab}[\neg_2]\langle T\rangle$, $\mathsf{tab}[Ⓢ\neg_2]\langle T\rangle$, $\mathsf{tab}[Ⓢ\neg_2]\langle F\rangle$, plus the closure rules $\mathsf{tab}[C0]$, and $\mathsf{tab}[C3]$, not necessarily in this order, and produce some open branches. A first class of such open branches is the one that contains both the formula $F{:}\neg_1 p$ and the formula $F{:}\neg_1 q$. This is enough to tell us that they represent the 4-valued counter-models in which $\mathsf{w}(p) = t = \mathsf{w}(q)$. A second class of such open branches is the one that contains only one of the above formulas, say $F{:}\neg_1 p$, and on what concerns the other atom, q, it contains both $T{:}q$ and $T{:}Ⓢq$ (*mutatis mutandis* if one exchanges the roles of p and q). All we can say in that case is that the 4-valued counter-models represented by the open tableau for the above inference allow for assignments in which $\mathsf{w}(p) = t$ and $\mathsf{w}(q) \in \{t, \perp\}$ (*mutatis mutandis*, the assignments in which $\mathsf{w}(q) = t$ and $\mathsf{w}(p) \in \{t, \perp\}$ also represent counter-models).

The logic-user might be satisfied with such procedure by itself in testing inferences of the logic underlying choice $[\mathcal{V}_b]$. But in this case there would hardly be a good technical reason to be found for not having stayed with the initial class of 4-valued models, which were even simpler and in many senses more well-behaved. One should insist here, though, that a much more attractive task, from the viewpoint of the logic-designer, is to *compare* different logics. There is a real advantage, in this case, in doing the bivalent reduction, from the semantical perspective, and working with the associated two-signed tableaux. For if that same reduction can be done for other logics, say the logics underlying the choices $[\mathcal{V}_{e\ell}]$ or $[\mathcal{V}_n]$, or, for all that matters, for any other finite-valued logics, for instance, checking whether a primitive or derived rule from the logic $\mathcal{L}_y$ is valid from the perspective of a logic $\mathcal{L}_x$, as long as such logics are written over comparable languages, costs just the same, in principle, as testing an inference of $\mathcal{L}_x$ inside $\mathcal{L}_x$.

For a practical illustration on how the present approach may also be used to produce easy proofs of some important meta-theoretical results concerning the comparison of the logics presented by way of bivalent semantics, consider the following extension of Theorem 1:

Theorem 4. *Given $\Gamma \cup \Delta \subseteq \mathcal{T}_{\mathcal{B}_Ⓢ}$, then:*

$$\Gamma \models_b \Delta \;\; \textit{iff} \;\; \Delta^\varepsilon \models_n \Gamma^\varepsilon \qquad \textit{and} \qquad \Gamma \models_n \Delta \;\; \textit{iff} \;\; \Delta^\varepsilon \models_b \Gamma^\varepsilon.$$

Proof. These may be verified by a quick inspection of the restrictions governing the classes of bivaluations Biv^b and Biv^n, in the Appendix — and in particular instructions **(DA)** and **(DB)**, that make the 'dual' relation between the bivaluation axioms quite explicit. ∎

Taking now the associated adequated tableau systems, Tab^b, $\mathsf{Tab}^{e\ell}$ and Tab^n, into account, an even more inclusive practical result that may immediately be established by 'classic-like dualization' is the following:

Theorem 5. *Let $\overline{T} \triangleq F$ and $\overline{F} \triangleq T$. Given $\{\varphi_1, \varphi_2, \ldots, \varphi_j\} \subseteq \mathcal{T}_{\mathcal{B}_{\circledS}}$ and given signs $S_i \in \{T, F\}$, for $1 \leq i \leq j$, then there is a closed tableau for $\{S_1{:}\varphi_1, S_2{:}\varphi_2, \ldots, S_j{:}\varphi_j\}$ using the rules of Tab^x iff there is a closed tableau for $\{\overline{S_1}{:}\varphi_1^{\varepsilon}, \overline{S_2}{:}\varphi_2^{\varepsilon}, \ldots, \overline{S_j}{:}\varphi_j^{\varepsilon}\}$ using the rules of Tab^y, where $\langle x, y\rangle \in \{\langle b, n\rangle, \langle e\ell, e\ell\rangle, \langle n, b\rangle\}$.*

To sum up with, it is important to mention that general constructive reduction mechanisms such as the ones described above are indeed available, for any finite-valued logic, and the class of logics characterized by bivalent semantics that can be associated to adequate analytic tableaux goes in fact much beyond the class of finite-valued logics. In the next section we will finish by disclosing a bit more about the range of applicability of the above ideas and techniques.

5 Context, considerations, and future work

The algebraic structures now widely known as bilattices (cf. [Ginsberg, 1986]) were introduced in [Belnap, 1976] and frequently investigated in the literature since then (cf. [Belnap, 1977; Ginsberg, 1988; Fitting, 1990; Fitting, 1994; Arieli and Avron, 1998]) and even generalized (cf. [Wansing and Shramko, 2005]) for their potential applications to several areas of computer science. A useful source of information for developments on bilattices is [Arieli and Avron, 2000], a paper to whose title the title of present study explicitly refers. In a sense, there is no essential loss of generality here as we concentrate our efforts over a simple *four-valued* bilattice, as there is a representation theorem that supports it, and its structure stands among other bilattices in a similar way as the canonical structure underlying classical logic stands among other boolean algebras.

Some criticism has recently been raised to the whole idea that such logics could really serve as a foundation for computerized reasoning under inconsistent / incomplete information, arguing that their alleged significance is based on a "confusion between truth-values and information states" (cf. [Dubois, 2008]). This has almost immediately been counteracted by evidence suggesting that the critique is based on a "confusion of information states with belief states" (cf. [Wansing and Belnap, 2010]). The controversy does not affect our work in this paper, nonetheless, as all we do here is largely independent of how the two involved logical dimensions are explicated.

Still and all, we do have seen above how the *logic* represented by the bilattice is very sensible to how its underlying truth-values are grouped and interpreted. Having shown the difficulties that lie in effecting the comparison between the inferences sanctioned by such logics, from the viewpoint of the logic-designer, and the even more difficult task of comparing such logics with arbitrarily other many-valued logics, we have next illustrated a general method that is based in providing a uniform approach to these logics, starting with a bivalent reduction of their semantics. Even though the illustration has been presented in detail only for the choice that led to a paraconsistent logic based on the four-valued structure, the slightly extended language that we have used was chosen to be appropriate also for the application of the same techniques to the alternative paracomplete logic based on a different choice made over the same structure, and in several points we have commented on the adaptations that must be made on the statements and results of the former case so as to deal also with the latter case. In the currently detailed case study, as a matter of fact, what we have is a variant of the output of a general method that allows for the extraction of classic-like semantics for any finite-valued logic (check [Caleiro and Marcos, 2010] for a recent survey). The output of that method, as presented in the Appendix, has indeed been optimized here, by the use of equivalences stated in the classical metalanguage, in much the same way as formulas in DNF (namely, the sentences to the right of the '$\Leftrightarrow$' symbol, representing the body of the axioms / rules) can be manipulated into *reduced* DNF (the simplifications also take the closure rules into account).

Our two-valued formulation of the chosen illustration has also been carefully crafted so as to serve as input to another general method that allowed for the extraction of adequate proof-theoretical counterparts of our logic in terms of *analytic tableaux*. The method, in this case, is even more general than the previous one, and applies not only to logics whose classic-like semantic presentations result from the above mentioned reduction (cf. [Caleiro and Marcos, 2009]), but to many other logics that can be characterized by way of bivaluation axioms of a similar 'dyadic' format (cf. [Caleiro *et al.*, 2005]). In that case, to facilitate the comparison between different logics it might be necessary to fiddle with structural rules in order to guarantee that the obtained proof systems are able to derive all the rules they have as admissible (cf. [Marcos and Mendonça, 2009])). In any case, for each logic obtained through the above methods, fully automated proof tactics are available (cf. [Marcos, 2010]). In particular, it should be noted that analyticity of the proof procedure was guaranteed in the present paper through a simpler strategy as the one used in our previous papers, where the introduction of a separating connective *by abbreviation* required a *proof strategy* to be associated, in general, to secure the *termination* of the construction of an arbitrary tableau proof. Here we simplified matters (and the definition of the *complexity measure*) by introducing the missing separating connective directly as part of the underlying propositional signature.

For sure, on what concerns the beautiful and well-developed theory of bilattices, the present approach provides a novel outlook, yet only starts scratching some of the many issues that are of interest for the logic-designer. For instance, even though we have tried to modify the initial language of the bilattice as little and harmlessly as possible, the introduction of a suitable implication into $\mathcal{B}$ can simplify many of the above tasks, in making the underlying language more expressive in a very helpful way (cf. [Arieli and Avron, 2000]). It rests to be shown, anyway, how much of the deeper interest behind the 4-valued approach can be thoroughly retained within the present classic-like bivalent / two-signed approach.

Finally, for a note on a completely different direction, we have given a few hints along this study on how the above four-valued logic could be seen as a sort of combination of simpler fragments, where interaction axioms have an important role to play. It should be interesting to investigate the underlying technique, for its own interest, as a new mechanism for the combination of truth-functional logics (contrast with [Coniglio and Fernández, 2005]), where 'coalescing' a logic into another would consist in a way of adding a new dimension to a given structure. In that sense, for instance, the inner structure of the four-valued formalism could be seen as a result from a natural combination of classical logic with itself.

Acknowledgments

The author is much obliged to partial funding received from CNPq and CAPES during the periods in which this paper was confected. Criticisms and stimuli received from Chris Fermüller and Carlos Caleiro were also fundamental. Finally, the presentation also owes a lot to the provoking audiences of the 31st Linz Seminar on Fuzzy Set Theory, the SQIG/IT, and the CLE/UNICAMP, where the ideas herein contained were presented at different stages.

BIBLIOGRAPHY

[Arieli and Avron, 1998] Ofer Arieli and Arnon Avron. The value of the four values. *Artificial Intelligence*, 102:97–141, 1998.

[Arieli and Avron, 2000] Ofer Arieli and Arnon Avron. Bilattices and paraconsistency. In D. Batens, C. Mortensen, G. Priest, and J. P. Van Bendegem, editors, *Frontiers of Paraconsistent Logic*, Proceedings of the I World Congress on Paraconsistency, held in Ghent, BE, July 29–August 3, 1997, pages 11–28. Research Studies Press, Baldock, 2000.

[Belnap, 1976] Nuel D. Belnap. How a computer should think. In G. Ryle, editor, *Contemporary Aspects of Philosophy*, pages 30–55. Oriel Press, 1976.

[Belnap, 1977] Nuel D. Belnap. A useful four-valued logic. In J. M. Dunn and G. Epstein, editors, *Modern Uses of Multiple-Valued Logic*, pages 8–37. D. Reidel, 1977.

[Caleiro and Marcos, 2009] Carlos Caleiro and João Marcos. Classic-like analytic tableaux for finite-valued logics. In H. Ono, M. Kanazawa, and R. de Queiroz, editors, *Proceedings of the* XVI *Workshop on Logic, Language, Information and Computation* (WoLLIC 2009), held in Tokyo, JP, June 2009, volume 5514 of *Lecture Notes in Artificial Intelligence*, pages 268–280. Springer, 2009. Preprint available at: http://sqig.math.ist.utl.pt/pub/CaleiroC/09-CM-ClATab4FVL.pdf.

[Caleiro and Marcos, 2010] Carlos Caleiro and João Marcos. Two many values: An algorithmic outlook on Suszko's Thesis. In *Proceedings of the* XL *International Symposium on Multiple-Valued Logic* (ISMVL 2010), pages 93–97. IEEE Computer Society, 2010. Preprint available at: http://sqig.math.ist.utl.pt/pub/CaleiroC/10-CM-ismvl.pdf.

[Caleiro *et al.*, 2005] Carlos Caleiro, Walter Carnielli, Marcelo E. Coniglio, and João Marcos. Two's company: "The humbug of many logical values". In J.-Y. Béziau, editor, *Logica Universalis*, pages 169–189. Birkhäuser Verlag, Basel, Switzerland, 2005. Preprint available at: http://sqig.math.ist.utl.pt/pub/CaleiroC/05-CCCM-dyadic.pdf.

[Coniglio and Fernández, 2005] Marcelo E. Coniglio and Víctor L. Fernández. Plain fibring and direct union of logics with matrix semantics. In *Proceedings of the 2nd Indian International Conference on Artificial Intelligence*, pages 1648–1658. IICAI, 2005.

[Dubois, 2008] Didier Dubois. On ignorance and contradiction considered as truth-values. *Logic Journal of IGPL*, 16(2):195–216, 2008.

[Fitting, 1990] Melvin Fitting. Bilattices in logic programming. In *Proceedings of the* XX *International Symposium on Multiple-Valued Logic* (ISMVL 1990), pages 238–246. IEEE Computer Society, 1990.

[Fitting, 1994] Melvin Fitting. Kleene's three valued logics and their children. *Fundamenta Informaticae*, 20(1–3):113–131, 1994.

[Ginsberg, 1986] Matthew L. Ginsberg. Multi-valued logics. In *AAAI-86 Proceedings*, pages 243–249. AAAI, 1986.

[Ginsberg, 1988] Matthew L. Ginsberg. Multivalued logics: A uniform approach to inference in artificial intelligence. *Computational Intelligence*, 4:265–316, 1988.

[Marcos and Mendonça, 2009] João Marcos and Dalmo Mendonça. Towards fully automated axiom extraction for finite-valued logics. In W. Carnielli, M. E. Coniglio, and I. M. L. D'Ottaviano, editors, *The Many Sides of Logic*, Studies in Logic, pages 425–440. College Publications, London, 2009. Preprint available at: http://sqig.math.ist.utl.pt/pub/MarcosJ/08-MM-towards.pdf.

[Marcos, 2010] João Marcos. Automatic generation of proof tactics for finite-valued logics. *Electronic Proceedings in Theoretical Computer Science*, 21:91–98, 2010. Available at: http://arxiv.org/abs/1003.4802v1.

[Wansing and Belnap, 2010] Heinrich Wansing and Nuel Belnap. Generalized truth values: A reply to Dubois. *Logic Journal of the IGPL*, 18(6):921–935, 2010.

[Wansing and Shramko, 2005] Heinrich Wansing and Yaroslav Shramko. Some useful 16-valued logics: How a computer network should think. *Journal of Philosophical Logic*, 34:121–153, 2005.

Appendix

Here one can find the exhaustive bivalent description of all the above mentioned 4-valued logics.

In the metalinguistic notation below, a ‘,’ replaces an *and*, a ‘|’ replaces an *or*, a ‘$\Rightarrow$’ replaces an *if-then* assertion, a ‘$\Leftrightarrow$’ stands for an *iff* assertion, and a ‘$\divideontimes$’ represents the *absurd*. So, the first bivaluation axiom $\mathsf{biv}[\wedge_2]\langle 1\rangle$ below, for instance, should be read as restricting the class of bivaluations of interest to those in which $\mathsf{b}(\alpha \wedge_2 \beta) = 1$ is the case, for arbitrary formulas α and β of $\mathcal{T}_\mathcal{B}$, if and only if one of three following situations occur: either $\mathsf{b}(Ⓢ\alpha) = 1$ is the case, or $\mathsf{b}(Ⓢ\beta) = 1$ is the case, or else both $\mathsf{b}(\alpha) = 1$ and $\mathsf{b}(\beta) = 1$ are simultaneously the case. Further, an axiom rule such as the first $\mathsf{biv}[C2]$ below should be read as stating that $\mathsf{b}(\alpha) = 0$ and $\mathsf{b}(Ⓢ(\alpha)) = 1$ cannot simultaneously obtain, for any bivaluation b and formula α of $\mathcal{T}_\mathcal{B}$.

The first bivalent semantics described below, Biv^b, corresponds to the logic underlying choice $[\mathcal{V}_b]$, whose completeness proof was presented in detail in the preceding text. It corresponds to the collection of all bivaluations that conform to the restrictions listed in what follows.

$\mathsf{biv}[\wedge_1]\langle 1\rangle$ $\quad \mathsf{b}(\alpha \wedge_1 \beta) = 1 \ \Leftrightarrow$
$(\mathsf{b}(\neg_1\alpha) = 0 \,,\ \mathsf{b}(\beta) = 1)$
$\mid (\mathsf{b}(\alpha) = 1 \,,\ \mathsf{b}(\neg_1\beta) = 0)$
$\mid (\mathsf{b}(Ⓢ\alpha) = 1 \,,\ \mathsf{b}(Ⓢ\beta) = 1)$
$\mid (\mathsf{b}(\alpha) = 1 \,,\ \mathsf{b}(Ⓢ\alpha) = 0 \,,\ \mathsf{b}(\beta) = 1 \,,\ \mathsf{b}(Ⓢ\beta) = 0)$

$\mathsf{biv}[\wedge_1]\langle 0\rangle$ $\quad \mathsf{b}(\alpha \wedge_1 \beta) = 0 \ \Leftrightarrow$
$(\mathsf{b}(\alpha) = 0)$
$\mid (\mathsf{b}(\beta) = 0)$
$\mid (\mathsf{b}(\neg_1\alpha) = 1 \,,\ \mathsf{b}(Ⓢ\alpha) = 1 \,,\ \mathsf{b}(Ⓢ\beta) = 0)$
$\mid (\mathsf{b}(Ⓢ\alpha) = 0 \,,\ \mathsf{b}(\neg_1\beta) = 1 \,,\ \mathsf{b}(Ⓢ\beta) = 1)$

$\mathsf{biv}[\neg_1\wedge_1]\langle 1\rangle$ $\quad \mathsf{b}(\neg_1(\alpha \wedge_1 \beta)) = 1 \ \Leftrightarrow\ \mathsf{b}(\neg_1\alpha) = 1 \mid \mathsf{b}(\neg_1\beta) = 1$

$\mathsf{biv}[\neg_1\wedge_1]\langle 0\rangle$ $\quad \mathsf{b}(\neg_1(\alpha \wedge_1 \beta)) = 0 \ \Leftrightarrow\ \mathsf{b}(\neg_1\alpha) = 0 \,,\ \mathsf{b}(\neg_1\beta) = 0$

$\mathsf{biv}[Ⓢ\wedge_1]\langle 1\rangle$ $\quad \mathsf{b}(Ⓢ(\alpha \wedge_1 \beta)) = 1 \ \Leftrightarrow\ \mathsf{b}(Ⓢ\alpha) = 1 \,,\ \mathsf{b}(Ⓢ\beta) = 1$

$\mathsf{biv}[Ⓢ\wedge_1]\langle 0\rangle$ $\quad \mathsf{b}(Ⓢ(\alpha \wedge_1 \beta)) = 0 \ \Leftrightarrow\ \mathsf{b}(Ⓢ\alpha) = 0 \mid \mathsf{b}(Ⓢ\beta) = 0$

. . . .

$\mathsf{biv}[\wedge_2]\langle 1\rangle$ $\quad \mathsf{b}(\alpha \wedge_2 \beta) = 1 \ \Leftrightarrow$
$(\mathsf{b}(Ⓢ\alpha) = 1)$
$\mid (\mathsf{b}(Ⓢ\beta) = 1)$
$\mid (\mathsf{b}(\alpha) = 1 \,,\ \mathsf{b}(\beta) = 1)$

$\mathsf{biv}[\wedge_2]\langle 0\rangle$ $\quad \mathsf{b}(\alpha \wedge_2 \beta) = 0 \ \Leftrightarrow$
$(\mathsf{b}(\alpha) = 0 \,,\ \mathsf{b}(Ⓢ\beta) = 0)$
$\mid (\mathsf{b}(Ⓢ\alpha) = 0 \,,\ \mathsf{b}(\beta) = 0)$

$$\mathsf{biv}[\neg_1\wedge_2]\langle 1\rangle \qquad \mathsf{b}(\neg_1(\alpha\wedge_2\beta))=1 \iff (\mathsf{b}(\alpha)=0) \mid (\mathsf{b}(\beta)=0) \mid (\mathsf{b}(\neg_1\alpha)=1,\ \mathsf{b}(ⓢ\alpha)=1) \mid (\mathsf{b}(\neg_1\beta)=1,\ \mathsf{b}(ⓢ\beta)=1) \mid (\mathsf{b}(\neg_1\alpha)=1,\ \mathsf{b}(\neg_1\beta)=1)$$

$$\mathsf{biv}[\neg_1\wedge_2]\langle 0\rangle \qquad \mathsf{b}(\neg_1(\alpha\wedge_2\beta))=0 \iff (\mathsf{b}(\neg_1\alpha)=0,\ \mathsf{b}(\neg_1\beta)=0) \mid (\mathsf{b}(\neg_1\alpha)=0,\ \mathsf{b}(\beta)=1,\ \mathsf{b}(ⓢ\beta)=0) \mid (\mathsf{b}(\alpha)=1,\ \mathsf{b}(ⓢ\alpha)=0,\ \mathsf{b}(\neg_1\beta)=0)$$

$$\mathsf{biv}[ⓢ\wedge_2]\langle 1\rangle \qquad \mathsf{b}(ⓢ(\alpha\wedge_2\beta))=1 \iff \mathsf{b}(ⓢ\alpha)=1 \mid \mathsf{b}(ⓢ\beta)=1$$

$$\mathsf{biv}[ⓢ\wedge_2]\langle 0\rangle \qquad \mathsf{b}(ⓢ(\alpha\wedge_2\beta))=0 \iff \mathsf{b}(ⓢ\alpha)=0,\ \mathsf{b}(ⓢ\beta)=0$$

$$\mathsf{biv}[\vee_1]\langle 1\rangle \qquad \mathsf{b}(\alpha\vee_1\beta)=1 \iff \mathsf{b}(\alpha)=1 \mid \mathsf{b}(\beta)=1$$

$$\mathsf{biv}[\vee_1]\langle 0\rangle \qquad \mathsf{b}(\alpha\vee_1\beta)=0 \iff \mathsf{b}(\alpha)=0,\ \mathsf{b}(\beta)=0$$

$$\mathsf{biv}[\neg_1\vee_1]\langle 1\rangle \qquad \mathsf{b}(\neg_1(\alpha\vee_1\beta))=1 \iff (\mathsf{b}(ⓢ\alpha)=0,\ \mathsf{b}(ⓢ\beta)=0) \mid (\mathsf{b}(\alpha)=0,\ \mathsf{b}(\neg_1\beta)=1) \mid (\mathsf{b}(\neg_1\alpha)=1,\ \mathsf{b}(\beta)=0) \mid (\mathsf{b}(\neg_1\alpha)=1,\ \mathsf{b}(ⓢ\alpha)=1,\ \mathsf{b}(\neg_1\beta)=1,\ \mathsf{b}(ⓢ\beta)=1)$$

$$\mathsf{biv}[\neg_1\vee_1]\langle 0\rangle \qquad \mathsf{b}(\neg_1(\alpha\vee_1\beta))=0 \iff (\mathsf{b}(\neg_1\alpha)=0) \mid (\mathsf{b}(\neg_1\beta)=0) \mid (\mathsf{b}(ⓢ\alpha)=1,\ \mathsf{b}(\beta)=1,\ \mathsf{b}(ⓢ\beta)=0) \mid (\mathsf{b}(\alpha)=1,\ \mathsf{b}(ⓢ\alpha)=0,\ \mathsf{b}(ⓢ\beta)=1)$$

$$\mathsf{biv}[ⓢ\vee_1]\langle 1\rangle \qquad \mathsf{b}(ⓢ(\alpha\vee_1\beta))=1 \iff \mathsf{b}(ⓢ\alpha)=1 \mid \mathsf{b}(ⓢ\beta)=1$$

$$\mathsf{biv}[ⓢ\vee_1]\langle 0\rangle \qquad \mathsf{b}(ⓢ(\alpha\vee_1\beta))=0 \iff \mathsf{b}(ⓢ\alpha)=0,\ \mathsf{b}(ⓢ\beta)=0$$

$$\mathsf{biv}[\vee_2]\langle 1\rangle \qquad \mathsf{b}(\alpha\vee_2\beta)=1 \iff (\mathsf{b}(\neg_1\alpha)=0) \mid (\mathsf{b}(\neg_1\beta)=0) \mid (\mathsf{b}(\alpha)=1,\ \mathsf{b}(ⓢ\alpha)=0) \mid (\mathsf{b}(\beta)=1,\ \mathsf{b}(ⓢ\beta)=0) \mid (\mathsf{b}(\alpha)=1,\ \mathsf{b}(\beta)=1)$$

$$\mathsf{biv}[\vee_2]\langle 0\rangle \qquad \mathsf{b}(\alpha\vee_2\beta)=0 \iff (\mathsf{b}(\alpha)=0,\ \mathsf{b}(\beta)=0) \mid (\mathsf{b}(\alpha)=0,\ \mathsf{b}(\neg_1\beta)=1,\ \mathsf{b}(ⓢ\beta)=1) \mid (\mathsf{b}(\neg_1\alpha)=1,\ \mathsf{b}(ⓢ\alpha)=1,\ \mathsf{b}(\beta)=0)$$

$$\mathsf{biv}[\neg_1\vee_2]\langle 1\rangle \qquad \mathsf{b}(\neg_1(\alpha\vee_2\beta))=1 \iff (\mathsf{b}(ⓢ\alpha)=0) \mid (\mathsf{b}(ⓢ\beta)=0) \mid (\mathsf{b}(\neg_1\alpha)=1,\ \mathsf{b}(\neg_1\beta)=1)$$

$$\mathsf{biv}[\neg_1\vee_2]\langle 0\rangle \qquad \mathsf{b}(\neg_1(\alpha\vee_2\beta))=0 \iff (\mathsf{b}(ⓢ\alpha)=1,\ \mathsf{b}(\neg_1\beta)=0) \mid (\mathsf{b}(\neg_1\alpha)=0,\ \mathsf{b}(ⓢ\beta)=1)$$

$$\mathsf{biv}[ⓢ\vee_2]\langle 1\rangle \qquad \mathsf{b}(ⓢ(\alpha\vee_2\beta))=1 \iff \mathsf{b}(ⓢ\alpha)=1,\ \mathsf{b}(ⓢ\beta)=1$$

$$\mathsf{biv}[ⓢ\vee_2]\langle 0\rangle \qquad \mathsf{b}(ⓢ(\alpha\vee_2\beta))=0 \iff \mathsf{b}(ⓢ\alpha)=0 \mid \mathsf{b}(ⓢ\beta)=0$$

$$\mathsf{biv}[\neg_1\neg_1]\langle 1\rangle \qquad \mathsf{b}(\neg_1(\neg_1\alpha))=1 \iff \mathsf{b}(\alpha)=1$$

$$\mathsf{biv}[\neg_1\neg_1]\langle 0\rangle \qquad \mathsf{b}(\neg_1(\neg_1\alpha))=0 \iff \mathsf{b}(\alpha)=0$$

$$\mathsf{biv}[ⓢ\neg_1]\langle 1\rangle \qquad \mathsf{b}(ⓢ(\neg_1\alpha))=1 \iff \mathsf{b}(\alpha)=0 \mid (\mathsf{b}(\neg_1\alpha)=1,\ \mathsf{b}(ⓢ\alpha)=1)$$

$$\mathsf{biv}[ⓢ\neg_1]\langle 0\rangle \qquad \mathsf{b}(ⓢ(\neg_1\alpha))=0 \iff \mathsf{b}(\neg_1\alpha)=0 \mid (\mathsf{b}(\alpha)=1,\ \mathsf{b}(ⓢ\alpha)=0)$$

$$
\begin{array}{llcl}
\mathsf{biv}[\neg_2]\langle 1\rangle & \mathsf{b}(\neg_2\alpha)=1 & \Leftrightarrow & \mathsf{b}(\alpha)=1\\
\mathsf{biv}[\neg_2]\langle 0\rangle & \mathsf{b}(\neg_2\alpha)=0 & \Leftrightarrow & \mathsf{b}(\alpha)=0\\
\mathsf{biv}[\neg_1\neg_2]\langle 1\rangle & \mathsf{b}(\neg_1(\neg_2\alpha))=1 & \Leftrightarrow & \mathsf{b}(\neg_1\alpha)=1\\
\mathsf{biv}[\neg_1\neg_2]\langle 0\rangle & \mathsf{b}(\neg_1(\neg_2\alpha))=0 & \Leftrightarrow & \mathsf{b}(\neg_1\alpha)=0\\
\mathsf{biv}[Ⓢ\neg_2]\langle 1\rangle & \mathsf{b}(Ⓢ(\neg_2\alpha))=1 & \Leftrightarrow & \mathsf{b}(\neg_1\alpha)=0 \mid (\mathsf{b}(\alpha)=1\ ,\ \mathsf{b}(Ⓢ\alpha)=0)\\
\mathsf{biv}[Ⓢ\neg_2]\langle 0\rangle & \mathsf{b}(Ⓢ(\neg_2\alpha))=0 & \Leftrightarrow & \mathsf{b}(\alpha)=0 \mid (\mathsf{b}(\neg_1\alpha)=1\ ,\ \mathsf{b}(Ⓢ\alpha)=1)
\end{array}
$$

$$
\begin{array}{llcl}
\mathsf{biv}[\neg_1Ⓢ]\langle 1\rangle & \mathsf{b}(\neg_1(Ⓢ\alpha))=1 & \Leftrightarrow & \mathsf{b}(Ⓢ\alpha)=0\\
\mathsf{biv}[\neg_1Ⓢ]\langle 0\rangle & \mathsf{b}(\neg_1(Ⓢ\alpha))=0 & \Leftrightarrow & \mathsf{b}(Ⓢ\alpha)=1\\
\mathsf{biv}[ⓈⓈ]\langle 1\rangle & \mathsf{b}(Ⓢ(Ⓢ\alpha))=1 & \Leftrightarrow & \mathsf{b}(Ⓢ\alpha)=1\\
\mathsf{biv}[ⓈⓈ]\langle 0\rangle & \mathsf{b}(Ⓢ(Ⓢ\alpha))=0 & \Leftrightarrow & \mathsf{b}(Ⓢ\alpha)=0
\end{array}
$$

$$
\begin{array}{llcl}
\mathsf{biv}[T0] & & \Rightarrow & \mathsf{b}(\alpha)=0 \mid \mathsf{b}(\alpha)=1\\
\mathsf{biv}[C0] & (\mathsf{b}(\alpha)=0\ ,\ \mathsf{b}(\alpha)=1) & \Rightarrow & ⋇\\
\mathsf{biv}[C1] & (\mathsf{b}(\alpha)=0\ ,\ \mathsf{b}(\neg_1\alpha)=0) & \Rightarrow & ⋇\\
\mathsf{biv}[C2] & (\mathsf{b}(\alpha)=0\ ,\ \mathsf{b}(Ⓢ\alpha)=1) & \Rightarrow & ⋇\\
\mathsf{biv}[C3] & (\mathsf{b}(\neg_1\alpha)=0\ ,\ \mathsf{b}(Ⓢ\alpha)=0) & \Rightarrow & ⋇
\end{array}
$$

Using precisely the same technique illustrated in the preceding text, a bivalent semantics Biv^n corresponding to the logic underlying choice $[\mathcal{V}_n]$ may be produced. Its description is quite simple to present if we take into account the above exhaustive description of Biv^b. Indeed, all we have to do is:

(DA) rewrite each expression of the form $\mathsf{b}(\varphi)=v$ as $\mathsf{b}(\varphi)=1-v$

(DB) exchange each $\wedge$ for a $\vee$, and vice-versa

So, a bivaluation axiom for Biv^b such as:

$$
\begin{array}{ll}
\mathsf{biv}[\neg_1\wedge_2]\langle 0\rangle & \mathsf{b}(\neg_1(\alpha\wedge_2\beta))=0 \Leftrightarrow\\
& \quad (\mathsf{b}(\neg_1\alpha)=0\ ,\ \mathsf{b}(\neg_1\beta)=0)\\
& \mid (\mathsf{b}(\neg_1\alpha)=0\ ,\ \mathsf{b}(\beta)=1\ ,\ \mathsf{b}(Ⓢ\beta)=0)\\
& \mid (\mathsf{b}(\alpha)=1\ ,\ \mathsf{b}(Ⓢ\alpha)=0\ ,\ \mathsf{b}(\neg_1\beta)=0)
\end{array}
$$

becomes in Biv^n the axiom:

$$
\begin{array}{ll}
\mathsf{biv}[\neg_1\vee_2]\langle 1\rangle & \mathsf{b}(\neg_1(\alpha\vee_2\beta))=1 \Leftrightarrow\\
& \quad (\mathsf{b}(\neg_1\alpha)=1\ ,\ \mathsf{b}(\neg_1\beta)=1)\\
& \mid (\mathsf{b}(\neg_1\alpha)=1\ ,\ \mathsf{b}(\beta)=0\ ,\ \mathsf{b}(Ⓢ\beta)=1)\\
& \mid (\mathsf{b}(\alpha)=0\ ,\ \mathsf{b}(Ⓢ\alpha)=1\ ,\ \mathsf{b}(\neg_1\beta)=1)
\end{array}
$$

and a closure rule for Biv^b such as

$$
\mathsf{biv}[C3] \qquad (\mathsf{b}(\neg_1\alpha)=0\ ,\ \mathsf{b}(Ⓢ\alpha)=0) \quad \Rightarrow \quad ⋇
$$

becomes in Biv^n the closure rule:

$$
\mathsf{biv}[C3] \qquad (\mathsf{b}(\neg_1\alpha)=1\ ,\ \mathsf{b}(Ⓢ\alpha)=1) \quad \Rightarrow \quad ⋇
$$

Finally, a bivalent semantics $\mathsf{Biv}^{e\ell}$ corresponding to the logic underlying choice $[\mathcal{V}_{e\ell}]$ is even easier to describe than the previously described semantics, given that there is no real need to use for that effect more than one separating connective — and each one of $\neg_1$, $\neg_2$ and Ⓢ will do the job equally well. As a matter of stipulation, choosing for that effect the latter connective, Ⓢ, the corresponding straightforward bivaluation axioms that we obtain are presented in what follows.

$$
\begin{array}{llcl}
\mathsf{biv}[\neg_1]\langle 0\rangle & \mathsf{b}(\neg_1\alpha)=0 & \Leftrightarrow & \mathsf{b}(\circledS\alpha)=1\\
\mathsf{biv}[\circledS\neg_1]\langle 0\rangle & \mathsf{b}(\circledS\neg_1\alpha)=0 & \Leftrightarrow & \mathsf{b}(\alpha)=1\\[1ex]
\mathsf{biv}[\neg_2]\langle 0\rangle & \mathsf{b}(\neg_2\alpha)=0 & \Leftrightarrow & \mathsf{b}(\circledS\alpha)=0\\
\mathsf{biv}[\circledS\neg_2]\langle 0\rangle & \mathsf{b}(\circledS\neg_2\alpha)=0 & \Leftrightarrow & \mathsf{b}(\alpha)=0\\[1ex]
\mathsf{biv}[\circledS\circledS]\langle 0\rangle & \mathsf{b}(\circledS\circledS\alpha)=0 & \Leftrightarrow & \mathsf{b}(\circledS\alpha)=0\\[1ex]
\mathsf{biv}[\wedge_1]\langle 0\rangle & \mathsf{b}(\alpha\wedge_1\beta)=0 & \Leftrightarrow & \mathsf{b}(\alpha)=0 \ \mid\ \mathsf{b}(\beta)=0\\
\mathsf{biv}[\circledS\wedge_1]\langle 0\rangle & \mathsf{b}(\circledS(\alpha\wedge_1\beta))=0 & \Leftrightarrow & \mathsf{b}(\circledS\alpha)=0 \ \mid\ \mathsf{b}(\circledS\beta)=0\\[1ex]
\mathsf{biv}[\wedge_2]\langle 0\rangle & \mathsf{b}(\alpha\wedge_2\beta)=0 & \Leftrightarrow & \mathsf{b}(\alpha)=0 \ \mid\ \mathsf{b}(\beta)=0\\
\mathsf{biv}[\circledS\wedge_2]\langle 0\rangle & \mathsf{b}(\circledS(\alpha\wedge_2\beta))=0 & \Leftrightarrow & \mathsf{b}(\circledS\alpha)=0\ ,\ \mathsf{b}(\circledS\beta)=0\\[1ex]
\mathsf{biv}[\vee_1]\langle 0\rangle & \mathsf{b}(\alpha\vee_1\beta)=0 & \Leftrightarrow & \mathsf{b}(\alpha)=0\ ,\ \mathsf{b}(\beta)=0\\
\mathsf{biv}[\circledS\vee_1]\langle 0\rangle & \mathsf{b}(\circledS(\alpha\vee_1\beta))=0 & \Leftrightarrow & \mathsf{b}(\circledS\alpha)=0\ ,\ \mathsf{b}(\circledS\beta)=0\\[1ex]
\mathsf{biv}[\vee_2]\langle 0\rangle & \mathsf{b}(\alpha\vee_2\beta)=0 & \Leftrightarrow & \mathsf{b}(\alpha)=0\ ,\ \mathsf{b}(\beta)=0\\
\mathsf{biv}[\circledS\vee_2]\langle 0\rangle & \mathsf{b}(\circledS(\alpha\vee_2\beta))=0 & \Leftrightarrow & \mathsf{b}(\circledS\alpha)=0 \ \mid\ \mathsf{b}(\circledS\beta)=0
\end{array}
$$

The above listed bivaluation axioms for $\mathsf{Biv}^{e\ell}$ clearly tell only half of the story. However, to obtain the other half one simply has to follow instructions **(DA)** and **(DB)**. Axioms $\mathsf{biv}[T0]$ and $\mathsf{biv}[C0]$ must still be added to the above list, but no extra closure axioms are needed for a complete presentation, in the present case.