# A Translation Method for Belnap Logic

## Imperial College Research Report DoC 98/7

Odinaldo Rodrigues
Department of Computing,
King's College, London
rodrigu@dcs.kcl.ac.uk

Alessandra Russo
Department of Computing,
Imperial College, London,
ar3@doc.ic.ac.uk

September 1998

**Abstract**

In this report we present a translation of Belnap's four-valued logic[1] into classical first-order logic. Soundness and completeness of the translation approach with respect to Belnap's notion of entailment are proved. Examples derivations are also given. These results provide the basis for developing a belief revision approach for Belnap's logic in terms of standard AGM [3] belief revision operators for classical logic[1].

# 1 Introduction

Standard familiar systems such as classical logic, modal logic, intuitionistic logic have in common the principle that contradicting information entails any arbitrary sentence. This principle, known as *ex falsum quod libet*, is,

[1]The use of a classical revision operator for non-classical logics will be investigated in a forthcoming paper.

however, not always appropriate to describe real application deduction processes, where information are often deduced from quite possibly inconsistent databases. Alternative systems have been developed, examples of which include the logic of *first-degree entailment* (also known as system **E**) and the *relevant implication* system (or system **R**), in which deductions between formulae hold only when there is some "connection" between the formulae (e.g. the formulae share some sentential variable). In [2] Belnap provides a semantic characterization of first-degree entailment together with a sound and complete axiomatisation, emphasising its connection with the problem of "how a computer should think" [1].

We provide a translation of Belnap's semantics into a set of first-order logic formulae. Sets of Belnap formulae are translated into a conjunction of atomic predicates. An appropriate classical axiomatisation is defined, which captures the semantic behaviour of Belnap connectives, thus allowing Belnap's notion of entailment to be expressed in terms of classical entailment from the translated theories. This embedding into classical logic has two main advantages. The first one is to provide the basis for analysing belief revision operations for these types of logics. Secondly, theorem provers for four-valued logic can be developed by applying existing classical theorem provers on the classical logic translation of these logics.

In Section 2.1, we illustrate Belnap's semantics, showing some of the features of the deduction process that it formalises and its differences with respect to familiar classically-based deductive systems. In Section 2.2, we define our translation approach and the classical axiomatization of Belnap's four-valued semantics, providing some illustrative derivation examples. We prove the soundness and completeness results of the translation approach, showing that it preserves Belnap's deductive process.

# 2 Translating Belnap's four-valued logic into classical logic

We introduce specific notation as and when necessary throughout the rest of the report. However, the reader might like to bear the following in mind: propositional symbols will usually begin with a lower-case letter, whereas predicate symbols will often begin with an upper–case letter. Greek–letter meta–variables will be used to refer in general to wffs of the Belnap logic (i.e. "object logic"), whereas upper-case meta-variable letters will be used to denote wffs of first-order logic (i.e. "target logic"). Larger entities such as structures, sets, theories and languages will often be symbolised in caligraphic font, $\mathcal{A}, \mathcal{B}, \mathcal{C}, \ldots$.

## 2.1 Belnap's four-valued Logic

As mentioned in [1], Belnap describes its semantic characterisation of four-valued logics as an appropriate logic for expressing practical deductive processes. In database management or question-answer systems, collections of data are proned to include either explicit or hidden inconsistencies. This is due for instance to the fact that information may come from different contradicting sources. The use of a classical deductive process would not be appropriate in this case – since any arbitrary information is classically derivable from an inconsistent collection of data. Explicit inconsistencies may come from different sources equally reliable, whereas hidden inconsistencies are identified only by means of deductive reasoning. The motivation for Belnap's approach is to provide a logic less sensitive to inconsistencies.

**Syntax** Let $\mathcal{L}_B$ be the Belnap propositional language composed of a countable set of propositional letters $\{p, q, r, \ldots\}$ and the connectives $\neg$, $\wedge$ and $\vee$. The set of wffs is given by the standard construction of formulae. For the finite case, a Belnap theory can be seen as a single formula given by the conjunction of a given finite set of wffs. The formula $\neg p \wedge (\neg q \vee r) \wedge \neg r$ is an example of a finite Belnap theory. Because of the soundness and completeness results of Section 2.2, it would not be difficult to extend this logic to deal with infinite theories.

In a proof theoretical terms, Belnap's four-valued logic is characterised by a finite axiomatization. Given two Belnap wffs $\alpha$ and $\beta$, the expression $\alpha \rightarrow \beta$ denotes that $\alpha$ *entails* $\beta$. In this sense, the symbol $\rightarrow$ can be seen as a derivability relation between formulae, or equally between theories and a formula. The expression $\alpha \leftrightarrow \beta$ denotes that $\beta$ can be derived from $\alpha$ ($\alpha \rightarrow \beta$) and vice-versa, or, semantically, that $\alpha$ and $\beta$ are equivalent. The axiomatization given below is known to be sound and complete with respect to the semantics of the logic presented later.

**Definition 1** [Axiomatization] Let $\alpha_1, \ldots, \alpha_n$, $\beta_1, \ldots, \beta_m$ and $\gamma$ be Belnap wffs. A proof theory for Belnap four-valued logic, denoted with $Ax_B$, is the following set of expressions:

1. $\neg\neg\alpha \leftrightarrow \alpha$.
2. $\neg(\alpha \wedge \beta) \leftrightarrow \neg\alpha \vee \neg\beta$.
3. $\neg(\alpha \vee \beta) \leftrightarrow \neg\alpha \wedge \neg\beta$.
4. $\alpha \vee \beta \leftrightarrow \beta \vee \alpha$.
5. $\alpha \vee (\beta \vee \gamma) \leftrightarrow (\alpha \vee \beta) \vee \gamma$.
6. $\alpha \vee (\beta \wedge \gamma) \leftrightarrow (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$.
7. $\alpha \wedge \beta \leftrightarrow \beta \wedge \alpha$.
8. $\alpha \wedge (\beta \wedge \gamma) \leftrightarrow (\alpha \wedge \beta) \wedge \gamma$.
9. $\alpha \wedge (\beta \vee \gamma) \leftrightarrow (\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$.
10. $\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \beta_1 \vee \ldots \vee \beta_m$ provided that $\beta_j = \alpha_i$ for some $i$ and $j$.
11. $\alpha \rightarrow \beta$ and $\beta \rightarrow \gamma$ then $\alpha \rightarrow \gamma$.
12. $\alpha \leftrightarrow \beta$ and $\beta \leftrightarrow \gamma$ then $\alpha \leftrightarrow \gamma$.
13. $\alpha \rightarrow \beta$ if and only if $\neg\beta \rightarrow \neg\alpha$.
14. $(\alpha \vee \beta) \rightarrow \gamma$ if and only if $\alpha \rightarrow \gamma$ and $\beta \rightarrow \gamma$.
15. $\alpha \rightarrow \beta$ if and only if $\beta \leftrightarrow (\alpha \vee \beta)$
16. $\alpha \rightarrow \beta$ if and only if $\alpha \leftrightarrow (\alpha \wedge \beta)$
17. $\alpha \rightarrow (\beta \wedge \gamma)$ if and only if $\alpha \rightarrow \beta$ and $\alpha \rightarrow \gamma$. □

The first nine expressions correspond to standard classical properties of negation, disjunction and conjunction (e.g., commutativity, associativity, De Morgan laws). We will sometimes refer to them as the Belnap axioms. Expressions 10, 11 and 13 capture respectively the reflexivity, transitivity and contrapositive properties of the derivability relation $\rightarrow$, whereas expressions 14-17 correspond to standard classical rules for introduction and elimination of $\vee$ and $\wedge$ respectively. We will sometimes refer to expressions 11-17 as the

Belnap rules. Any Belnap expression of the form $\psi \to \varphi$ can be either an instantiation of one of the axioms 1-10 in definition 1, or obtained using some of the Belnap rules 11-17, together with some axiom instantiations. For any given expression $\psi \to \varphi$, we therefore define the notion of *length* as the "least number" of Belnap rule applications needed to show $\psi \to \varphi$.

The similarity between the above rules and classical rules shows that four-valued logics are indeed very close to standard classical logic. The basic classical rule, which is missing in Belnap logic and which makes this logic *paraconsistent* is the rule $(\alpha \wedge \neg\alpha) \to \beta$, often referred to as *ex falsum quod libel*. This rule allows within a classical framework to derive any arbitrary information from inconsistent assumptions. Belnap logic does not allow so.

**Semantics** The semantics underlying Belnap's logic is four-valued. Let **4** be the set $\{\mathsf{T}, \mathsf{F}, \mathsf{Both}, \mathsf{None}\}$. The elements of this set are the four different truth-values which an atomic sentence can have within a given "state of information". The intuitive meaning of these values is given as follows:

1. $p$ is stated to be true only ($\mathsf{T}$)

2. $p$ is stated to be false only ($\mathsf{F}$)

3. $p$ is stated to be both true and false, for instance, by different sources, or in different points of time ($\mathsf{Both}$), and

4. $p$'s status is unknown. That is, neither true, nor false ($\mathsf{None}$).

The four values form a lattice, called the *approximation lattice* and denoted by **A4** where the ordering relation $\sqsubseteq$ goes "uphill" and respects the monotonicity property, in the sense that information about the truth-value of a formula "grows" from $\mathsf{None}$ to $\mathsf{Both}$. **A4** can be seen in Figure 1.

The truth values of complex formulae are defined based on **A4** and result in the truth tables shown in Figure 2.

The truth tables constitute a lattice, called *logical lattice* and denoted by **L4** (Figure 3). In **L4**, logical conjunction is identified with the meet operation and logical disjunction with the join operation.

The notion of a interpretation of formulae is expressed in Belnap's logic in terms of *set-ups*. A set-up $s$ is a mapping of the atomic formulae into **4**. Using the truth tables given in Figure 2, each set-up can be extended to a mapping of *all formulae* into **4**, in the standard inductive way. We call this extended set-up a **4**-*valuation* and denote it with $v$. Thus, for any

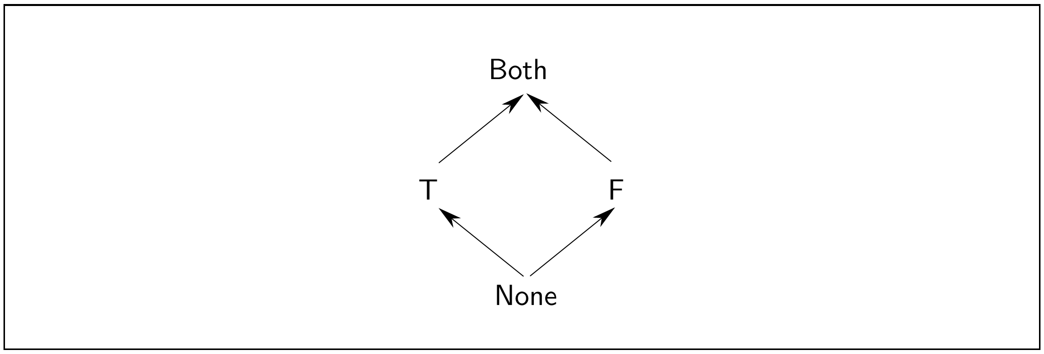


Figure 1: The approximation lattice **A4**.

given Belnap formula $\alpha$ and set-up $s$, the valuation $v(\alpha)$ is always well-defined. This makes Belnap's semantic somewhat different from the classical semantics, because the notion of model, that is, an interpretation that makes a formulae true is non-existent.

The notion of semantic entailment is then expressed in terms of a partial ordering $\preceq$ associated with the logical lattice **L4**. We will denote the semantic entailment relation with $\Rightarrow$ to distinguish it from the proof theoretic notion of entailment $\rightarrow$. The two notions are equivalent, as given by the correspondence 1, and the symbols $\rightarrow$ and $\Rightarrow$ will be often used interchangeably.

**Definition 2** Let $\alpha$ and $\beta$ be two Belnap formulae. We say that $\alpha$ *entails* $\beta$, written $\alpha \Rightarrow \beta$, if for all **4**-valuations $v$, $v(\alpha) \preceq v(\beta)$, where $\preceq$ is the partial ordering associated with the lattice **L4**. Analogously, a *non empty* finite set of formulae $\Gamma$ *entails* $\alpha$, if the conjunction of all formulae in $\Gamma$ entails $\alpha$.

□

(Correspondence)

$$\alpha \rightarrow \beta \text{ iff } \alpha \Rightarrow \beta \tag{1}$$

We now introduce some terminology which will be used throughout this report.

**Definition 3** Let $\alpha$ be a Belnap formula and let $v$ be a **4**-valuation. We say that $\alpha$ is

- *at least true* under $v$ if $v(\alpha) = \mathsf{T}$ or $v(\alpha) = \mathsf{Both}$.
- *at least false* under $v$ if $v(\alpha) = \mathsf{F}$ or $v(\alpha) = \mathsf{Both}$.

| $\wedge$ | None | F | T | Both |
|---|---|---|---|---|
| None | None | F | None | F |
| F | F | F | F | F |
| T | None | F | T | Both |
| Both | F | F | Both | Both |

Truth-table for the connective $\wedge$

| $\vee$ | None | F | T | Both |
|---|---|---|---|---|
| None | None | None | T | T |
| F | None | F | T | Both |
| T | T | T | T | T |
| Both | T | Both | T | Both |

Truth-table for the connective $\vee$

| | None | F | T | Both |
|---|---|---|---|---|
| $\neg$ | None | T | F | Both |

Truth-table for the connective $\neg$

Figure 2: Truth-tables for Belnap's connectives.

- *not true* under $v$ if $v(\alpha) = \mathsf{F}$ or $v(\alpha) = \mathsf{None}$.
- *not false* under $v$ if $v(\alpha) = \mathsf{T}$ or $v(\alpha) = \mathsf{None}$. □

Using the above terminology, the notion of semantic entailment between a theory and a formula given in Definition 2 can be equivalently expressed as follows.

**Definition 4** Let $\Gamma$ be a set of Belnap formulae and $\alpha$ a Belnap formula. $\Gamma$ entails $\alpha$ if and only if for every **4**-valuation $v$,

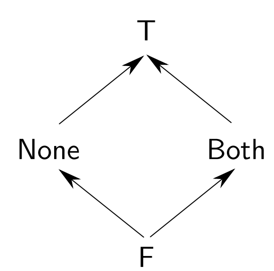


Figure 3: The logical lattice **L4**.

*i)* if all the formulae in $\Gamma$ are at least true under $v$, then $\alpha$ is at least true under $v$;

*ii)* if all the formulae in $\Gamma$ are not false under $v$, then $\alpha$ is not false under $v$. □

This definition will play an important role in the soundness and completeness proofs of the first-order Belnap translation with respect to Belnap semantics.

## 2.2 The translation into classical logic

In this section, we describe a translation approach of Belnap logic into first-order logic and show that it is sound and complete with respect to Belnap's semantic notion of entailment. Let $\mathcal{L}$ be a two sorted first-order language composed of the sort $\mathcal{F}$, called *B-formulae*, and the sort $\mathcal{V}$ called *truth values*.

The set of constants of the sort $\mathcal{F}$ is the set of propositional letters in Belnap's logic, whereas terms of $\mathcal{F}$ are constructed using three main functions $\neg$, $\wedge$, and $\vee$ which correspond to the Belnap connectives. The set of ground terms of $\mathcal{F}$ is therefore equivalent to the set of Belnap wffs. The sort $\mathcal{V}$ is instead composed of two constant symbols $\{\mathsf{tt}, \mathsf{ff}\}$, the basic constants from which Belnap's four-valued semantics can be constructed. $\mathcal{L}$ also contains the two-sorted binary predicate *holds*. *holds* takes as first arguments, $\mathcal{F}$ terms, and as second arguments $\mathcal{V}$ terms. $\mathcal{F}$ variables will be denoted with $x$, $y$, $z$, .... First-order formulae are constructed in the usual way.

Ground atomic formulae can be of two types $holds(\varphi, \mathsf{tt})$ and $holds(\psi, \mathsf{ff})$ for any Belnap wffs $\varphi$ and $\psi$. Atomic formulae of the first type mean that "$\mathsf{tt} \in v(\varphi)$", for some **4**-valuation $v$, which is equivalent to say that for some **4**-valuation, $\varphi$ is at least true. Atomic formulae of the second type state instead that "$\mathsf{ff} \in v(\psi)$", for some **4**-valuation $v$, which is equivalent to say that for some **4**-valuation, $\psi$ is at least false. With these two types of atomic formulae it is possible to express Belnap's full four-valued semantics. In order to simplify the proof, we extend the sort $\mathcal{V}$ with four constant symbols $\mathsf{T}$ , $\mathsf{F}$ , $\mathsf{None}$ and $\mathsf{Both}$, as follows:

**Definition 5** Let $\alpha$ be a Belnap formula. The four truth values that $\alpha$ can assume in Belnap semantics is expressed in the first-order translation by the following additional types of atomic formulae:

$$\begin{array}{lcl} holds(\alpha, \mathsf{T}) & \stackrel{\text{def}}{=} & holds(\alpha, \mathsf{tt}) \wedge \neg holds(\alpha, \mathsf{ff}) \\ holds(\alpha, \mathsf{F}) & \stackrel{\text{def}}{=} & \neg holds(\alpha, \mathsf{tt}) \wedge holds(\alpha, \mathsf{ff}) \\ holds(\alpha, \mathsf{None}) & \stackrel{\text{def}}{=} & \neg holds(\alpha, \mathsf{tt}) \wedge \neg holds(\alpha, \mathsf{ff}) \\ holds(\alpha, \mathsf{Both}) & \stackrel{\text{def}}{=} & holds(\alpha, \mathsf{tt}) \wedge holds(\alpha, \mathsf{ff}) \end{array}$$

□

The atomic formulae on the left-hand side express that under a **4**-valuation $v$, $v(\alpha) = \mathsf{T}$, $v(\alpha) = \mathsf{F}$, $v(\alpha) = \mathsf{None}$ and $v(\alpha) = \mathsf{Both}$ respectively. However, these additional four types of atomic formulae are in reality a short-hand for first-order formulae constructed from the basic language $\mathcal{L}$. We will therefore use throughout the report only the basic atomic formulae of $\mathcal{L}$.

The semantic behavior of Belnap connectives is fully captured by the following first-order axiomatisation.

**Definition 6** Given the two languages $\mathcal{L}_B$ and $\mathcal{L}$, $\mathcal{A}_B$ is the first-order axiomatisation of Belnap four-valued semantics given by the following six axioms:

$$\forall x[holds(x, \mathsf{ff}) \leftrightarrow holds(\neg x, \mathsf{tt})] \qquad \text{(Ax 1)}$$
$$\forall x[holds(x, \mathsf{tt}) \leftrightarrow holds(\neg x, \mathsf{ff})] \qquad \text{(Ax 2)}$$
$$\forall x, y[holds(x \wedge y, \mathsf{tt}) \leftrightarrow (holds(x, \mathsf{tt}) \wedge holds(y, \mathsf{tt}))] \qquad \text{(Ax 3)}$$
$$\forall x, y[holds(x \wedge y, \mathsf{ff}) \leftrightarrow (holds(x, \mathsf{ff}) \vee holds(y, \mathsf{ff}))] \qquad \text{(Ax 4)}$$
$$\forall x, y[holds(x \vee y, \mathsf{tt}) \leftrightarrow (holds(x, \mathsf{tt}) \vee holds(y, \mathsf{tt}))] \qquad \text{(Ax 5)}$$
$$\forall x, y[holds(x \vee y, \mathsf{ff}) \leftrightarrow (holds(x, \mathsf{ff}) \wedge holds(y, \mathsf{ff}))] \qquad \text{(Ax 6)}$$

□

The *translation function* $\tau$ is a mapping from the set of Belnap wffs to the set of ground atomic first-order formulae of the form $holds(\varphi, \mathsf{tt})$. For a given Belnap formula $\varphi$, its first order translation, denoted with $\tau(\varphi)$ or simply $\varphi^\tau$, is the first-order atomic formula $holds(\varphi, \mathsf{tt})$. The translation of a Belnap theory (i.e. finite sets of Belnap formulae) is therefore given by the translation of the conjunction of all Belnap formulae included in the theory. For instance, let $\Gamma = \{\varphi_1, \ldots, \varphi_n\}$ be a Belnap theory, its translation $\tau(\Gamma)$, or $\Gamma^\tau$, is the atomic first-order formula $holds(\varphi_1 \wedge \ldots \wedge \varphi_n, \mathsf{tt})$.

We are now going to prove that the above translation function together with the axiomatisation $\mathcal{A}_B$ is sound and complete with respect to the Belnap's semantic notion of entailment.

**Theorem 1 (Correspondence)** *Let $\psi$ and $\varphi$ be two Belnap formulae.*

$$\psi \to \varphi \quad \text{iff} \quad \mathcal{A}_B, holds(\psi, \mathsf{tt}) \vdash holds(\varphi, \mathsf{tt}) \\ \text{and} \quad \mathcal{A}_B, \neg holds(\psi, \mathsf{ff}) \vdash \neg holds(\varphi, \mathsf{ff}).$$

The proof of the above theorem uses Lemmas 1 and 2. Lemma 1 expresses the completeness of the translation function and the first-order axiomatisation with respect to Belnap's notion of entailment.

The statement captures, in first-order terms, the notion of entailment, given in Definition 4, whenever $\psi$ is of the form $\alpha_1 \wedge \ldots \wedge \alpha_n$, with $\{\alpha_1, \ldots, \alpha_n\}$ being a Belnap theory.

For the first conjunct of the statement, the assumption $holds(\psi, \mathsf{tt})$ is equivalent, by axiom (Ax 3), to $holds(\alpha_1, \mathsf{tt}) \wedge \ldots \wedge holds(\alpha_n, \mathsf{tt})$, which can be read as "all $\alpha_i$, for each $1 \leq i \leq n$, are at least true". The consequence $holds(\varphi, \mathsf{tt})$ can also be read as $\varphi$ is at least true. Analogously, for the second conjunct in the statement, the assumption $\neg holds(\psi, \mathsf{ff})$ is equivalent, by axiom (Ax 4), to $\neg holds(\alpha_1, \mathsf{ff}) \wedge \ldots \wedge \neg holds(\alpha_n, \mathsf{ff})$, where each $\neg holds(\alpha_i, \mathsf{ff})$ can be read as "$\alpha_i$ is not false". Lemma 2 expresses instead the soundness of the translation function and the first-order axiomatisation with respect to belanp's notion of entailment.

**Lemma 1 (Completeness)** *Let $\psi$ and $\varphi$ be two Belnap formulae.*
*If* $\psi \to \varphi$ *then* $\mathcal{A}_B,\ holds(\psi, \mathsf{tt}) \ \vdash \ holds(\varphi, \mathsf{tt})$ *and* $\mathcal{A}_B, \neg holds(\psi, \mathsf{ff}) \ \vdash \neg holds(\varphi, \mathsf{ff})$.

**Proof:** The proof is by induction on the length $n$ of the derivation $\psi \to \varphi$.
Base Case: $n = 0$. Then $\psi \to \varphi$ can only be an instantiation of one of the axioms 1-10 given in Definition 1. The proof is therefore by cases on each of these axioms. Only some of the cases are shown here. The remaining ones are proved following the same type of argument.
Case 1: $\psi \to \varphi$ is an instantiation of $\alpha_1 \wedge \ldots \wedge \alpha_h \to \beta_1 \vee \ldots \vee \beta_k$, for some $h$ and $k$ such that $\alpha_i = \beta_j$ for some $i$ and $j$. We show in Figure 4 that $\mathcal{A}_B, holds(\alpha_1 \wedge \ldots \wedge \alpha_h, \mathsf{tt}) \vdash holds(\beta_1 \vee \ldots \vee \beta_k, \mathsf{tt})$ and in Figure 5 that $\mathcal{A}_B, \neg holds(\alpha_1 \wedge \ldots \wedge \alpha_h, \mathsf{ff}) \vdash \neg holds(\beta_1 \vee \ldots \vee \beta_k, \mathsf{ff})$.
Case 2: $\psi \to \varphi$ is an instantiation of $\alpha \vee (\beta \wedge \gamma) \to (\alpha \vee \beta) \wedge (\alpha \wedge \gamma)$. We show in Figure 7 that $\mathcal{A}_B, holds(\alpha \vee (\beta \wedge \gamma), \mathsf{tt}) \to holds((\alpha \vee \beta) \wedge (\alpha \wedge \gamma), \mathsf{tt})$, and in Figure 6 that $\mathcal{A}_B, \neg holds(\alpha \vee (\beta \wedge \gamma), \mathsf{ff}) \to \neg holds((\alpha \vee \beta) \wedge (\alpha \wedge \gamma), \mathsf{ff})$. Similar argument is applied in the case where $\psi \to \varphi$ is an instantiation of $(\alpha \vee \beta) \wedge (\alpha \wedge \gamma) \to \alpha \vee (\beta \wedge \gamma)$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{A}_B, holds(\alpha_1 \wedge \ldots \wedge \alpha_h, \mathsf{tt})}{holds(\alpha_1, \mathsf{tt}) \wedge \ldots \wedge holds(\alpha_h, \mathsf{tt})}\ (\text{Ax } 3)}{holds(\alpha_i, \mathsf{tt})}\ (\mathcal{E}\wedge)}{holds(\beta_j, \mathsf{tt})}\ (\text{equiv. rewriting})}{holds(\beta_1, \mathsf{tt}) \vee \ldots \vee holds(\beta_k, \mathsf{tt})}\ (\mathcal{I}\vee)}{holds(\beta_1 \vee \ldots \vee \beta_k, \mathsf{tt})}\ (\text{Ax } 5)$$

Figure 4: First-order proof of Belnap axiom 10.

Inductive Step: We assume that there exists a first part of a derivation proving an expression of the form $\alpha \to \beta$ with $n-1$ applications of Belnap rules; and that the n-th application of a Belnap rule gives us the expression $\psi \to \varphi$. We reason by cases on each Belnap rule that could have been applied on this n-th step.

Case 1: We assume that last rule application is the "if-part" of Belnap rule 13 in Definition 1. Therefore, we have that there exists a proof of $\neg\varphi \to \neg\psi$, with $n-1$ rule applications. So by inductive hypothesis we can say that $\mathcal{A}_B, holds(\neg\varphi, \mathsf{tt}) \vdash holds(\neg\psi, \mathsf{tt})$ and that $\mathcal{A}_B, \neg holds(\neg\varphi, \mathsf{ff}) \vdash \neg holds(\neg\psi, \mathsf{ff})$. We want then to show that

$$\mathcal{A}_B, holds(\psi, \mathsf{tt}) \vdash holds(\varphi, \mathsf{tt})$$

and that

$$\mathcal{A}_B, \neg holds(\psi, \mathsf{ff}) \vdash \neg holds(\varphi, \mathsf{ff}).$$

From the inductive hypothesis $\mathcal{A}_B, \neg holds(\neg\varphi, \mathsf{ff}) \vdash \neg holds(\neg\psi, \mathsf{ff})$, we get, by contrapositive of classical logic, that

$$\mathcal{A}_B, holds(\neg\psi, \mathsf{ff}) \vdash holds(\neg\varphi, \mathsf{ff})$$

Hence, using Belnap axiom 2, we get $\mathcal{A}_B, holds(\psi, \mathsf{tt}) \vdash holds(\varphi, \mathsf{tt})$. To show that $\mathcal{A}_B, \neg holds(\psi, \mathsf{ff}) \vdash \neg holds(\varphi, \mathsf{ff})$ we consider the second part of

$$
\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\mathcal{A}_B, \neg holds(\alpha_1 \wedge \ldots \wedge \alpha_h, \mathsf{ff})}{holds(\beta_1 \vee \ldots \vee \beta_k, \mathsf{ff})}\ \text{(assumption)}}{holds(\beta_1, \mathsf{ff}) \wedge \ldots \wedge holds(\beta_k, \mathsf{ff})}\ \text{(Ax 6)}}{holds(\beta_j, \mathsf{ff})}\ (\mathcal{E}\wedge)}{holds(\alpha_i, \mathsf{ff})}\ \text{(equiv.rewriting)}}{holds(\alpha_1, \mathsf{ff}) \vee \ldots \vee holds(\alpha_h, \mathsf{ff})}\ (\mathcal{I}\vee)}{holds(\alpha_1 \wedge \ldots \wedge \alpha_h, \mathsf{ff})}\ \text{(Ax 4)}}{\bot}\ (\mathcal{I}\neg)}{\neg holds(\beta_1 \vee \ldots \vee \beta_k, \mathsf{ff})}
$$

Figure 5: First-order proof of Belnap axiom 10.

the inductive hypothesis. $\mathcal{A}_B, holds(\neg\varphi, \mathsf{tt}) \vdash holds(\neg\psi, \mathsf{tt})$ gives, by contrapositive of classical logic that

$$\mathcal{A}_B, \neg holds(\neg\psi, \mathsf{tt}) \vdash \neg holds(\neg\varphi, \mathsf{tt}).$$

Hence, by Belnap axiom 1, $\mathcal{A}_B, \neg holds(\psi, \mathsf{ff}) \vdash \neg holds(\varphi, \mathsf{ff})$. The case for the "only if-part" of Belnap rule 13 follows the same argument.

Case 2: We assume that last rule application is the "if-part" of Belnap rule 15 in Definition 1. Therefore, we have there exists a proof of $\alpha \to \beta$ with $n-1$ rule applications, where $\psi$ is equal to $\alpha$ and $\varphi$ is equal to $\alpha \vee \beta$. So by inductive hypothesis, $\mathcal{A}_B, holds(\alpha, \mathsf{tt}) \vdash holds(\beta, \mathsf{tt})$ and that $\mathcal{A}_B, \neg holds(\alpha, \mathsf{ff}) \vdash \neg holds(\beta, \mathsf{ff})$. We want to show that

1. $\mathcal{A}_B, holds(\beta, \mathsf{tt}) \vdash holds(\alpha \vee \beta, \mathsf{tt})$ and
   $\mathcal{A}_B, holds(\alpha \vee \beta, \mathsf{tt}) \vdash holds(\beta, \mathsf{tt})$

2. $\mathcal{A}_B, \neg holds(\beta, \mathsf{ff}) \vdash \neg holds(\alpha \vee \beta, \mathsf{ff})$ and
   $\mathcal{A}_B, \neg holds(\alpha \vee \beta, \mathsf{ff}) \vdash \neg holds(\beta, \mathsf{ff})$.

$$\frac{\mathcal{A}_B, \neg holds(\alpha \vee (\beta \wedge \gamma), \mathsf{ff})}{\neg holds(\alpha, \mathsf{ff}) \vee \neg holds(\beta \wedge \gamma, \mathsf{ff})} \ \text{(Ax 6)}$$

$$\frac{\neg holds(\alpha, \mathsf{ff}) \vee \neg holds(\beta \wedge \gamma, \mathsf{ff})}{\neg holds(\alpha, \mathsf{ff}) \vee (\neg holds(\beta, \mathsf{ff}) \wedge \neg holds(\gamma, \mathsf{ff}))} \ \text{(Ax 4)}$$

$$\frac{\neg holds(\alpha, \mathsf{ff}) \vee (\neg holds(\beta, \mathsf{ff}) \wedge \neg holds(\gamma, \mathsf{ff}))}{(\neg holds(\alpha, \mathsf{ff}) \vee \neg holds(\beta, \mathsf{ff})) \wedge (\neg holds(\alpha, \mathsf{ff}) \vee \neg holds(\gamma, \mathsf{ff}))} \ \text{(De Morgan Law)}$$

$$\frac{(\neg holds(\alpha, \mathsf{ff}) \vee \neg holds(\beta, \mathsf{ff})) \wedge (\neg holds(\alpha, \mathsf{ff}) \vee \neg holds(\gamma, \mathsf{ff}))}{\neg holds(\alpha \vee \beta, \mathsf{ff}) \wedge \neg holds(\alpha \vee \gamma, \mathsf{ff})} \ \text{(Ax 6)}$$

$$\frac{\neg holds(\alpha \vee \beta, \mathsf{ff}) \wedge \neg holds(\alpha \vee \gamma, \mathsf{ff})}{\neg holds((\alpha \vee \beta) \wedge (\alpha \vee \gamma), \mathsf{ff})} \ \text{(Ax 4)}$$

Figure 6: First-order proof of left-to-right part of Belnap axiom 6.

The first part of (1) is quite straightforward. We show the second part. Assume $\mathcal{A}_B, holds(\alpha \vee \beta, \mathsf{tt})$. By axiom (Ax 5) and reflexivity of classical logic, $\mathcal{A}_B, holds(\alpha \vee \beta, \mathsf{tt}) \vdash \mathcal{A}_B, holds(\alpha, \mathsf{tt}) \vee holds(\beta, \mathsf{tt})$. By inductive hypothesis, $\mathcal{A}_B, holds(\alpha, \mathsf{tt}) \vdash holds(\beta, \mathsf{tt})$ and by reflexivity of classical logic

$$\mathcal{A}_B, holds(\beta, \mathsf{tt}) \vdash holds(\beta, \mathsf{tt}).$$

Therefore, using classical $\vee$-introduction rule,

$$\mathcal{A}_B, holds(\alpha, \mathsf{tt}) \vee holds(\beta, \mathsf{tt}) \vdash holds(\beta, \mathsf{tt}).$$

Hence, $\mathcal{A}_B, holds(\alpha \vee \beta, \mathsf{tt}) \vdash holds(\beta, \mathsf{tt})$. The proof for (2) follows the same argument.
All the other cases can be easily proved using appropriate properties and rules of classical logic and, if necessary, the Belnap axioms. □

**Lemma 2 (Soundness)** *Let $\psi$ and $\varphi$ be two Belnap formulae.*
*If $\mathcal{A}_B, holds(\psi, \mathsf{tt}) \vdash holds(\varphi, \mathsf{tt})$ and $\mathcal{A}_B, \neg holds(\psi, \mathsf{ff}) \vdash \neg holds(\varphi, \mathsf{ff})$, then $\psi \to \varphi$.*

Some additional propositions and definitions need to be given before proving the above lemma. The soundness of the classical translation is based on the idea that for any given Belnap **4**-valuation it is always possible to construct a classical interpretation $I$ which satisfies the classical axioms $\mathcal{A}_B$ and which preserves Belnap's semantic entailment. We show first how this classical interpretation can be constructed and its properties.

$$\frac{\mathcal{A}_B, holds(\alpha \vee (\beta \wedge \gamma), \mathsf{tt})}{holds(\alpha, \mathsf{tt}) \vee holds(\beta \wedge \gamma, \mathsf{tt})} \text{(Ax 5)}$$

$$\frac{holds(\alpha, \mathsf{tt}) \vee holds(\beta \wedge \gamma, \mathsf{tt})}{holds(\alpha, \mathsf{tt}) \vee (holds(\beta, \mathsf{tt}) \wedge holds(\gamma, \mathsf{tt}))} \text{(Ax 3)}$$

$$\frac{holds(\alpha, \mathsf{tt}) \vee (holds(\beta, \mathsf{tt}) \wedge holds(\gamma, \mathsf{tt}))}{(holds(\alpha, \mathsf{tt}) \vee holds(\beta, \mathsf{tt})) \wedge (holds(\alpha, \mathsf{tt}) \vee holds(\gamma, \mathsf{tt}))} \text{(De Morgan Law)}$$

$$\frac{(holds(\alpha, \mathsf{tt}) \vee holds(\beta, \mathsf{tt})) \wedge (holds(\alpha, \mathsf{tt}) \vee holds(\gamma, \mathsf{tt}))}{holds(\alpha \vee \beta, \mathsf{tt}) \wedge holds(\alpha \vee \gamma, \mathsf{tt})} \text{(Ax 5)}$$

$$\frac{holds(\alpha \vee \beta, \mathsf{tt}) \wedge holds(\alpha \vee \gamma, \mathsf{tt})}{holds((\alpha \vee \beta) \wedge (\alpha \vee \gamma), \mathsf{tt})} \text{(Ax 3)}$$

Figure 7: First-order proof of left-to-right part of Belnap axiom 6.

**Definition 7** Let $v$ be a Belnap **4**-valuation from the set of Belnap wffs to the power set $\wp(\{\mathsf{tt}, \mathsf{ff}\})$. A classical interpretation *associated* with $v$, and denoted with $\mathcal{I}_v$, is a function defined as follows

- $\mathcal{I}_v(\mathsf{tt}) = \mathsf{tt}$ and $\mathcal{I}_v(\mathsf{ff}) = \mathsf{ff}$.

Also, for each ground term $\alpha$ of sort $\mathcal{F}$:

- $\mathcal{I}_v(\alpha) = \alpha$, for each ground term $\alpha$ of sort $\mathcal{F}$.
- $\mathcal{I}_v(holds) = \{\langle\alpha, \mathsf{tt}\rangle \mid \mathsf{tt} \in v(\alpha)\} \cup \{\langle\alpha, \mathsf{ff}\rangle \mid \mathsf{ff} \in v(\alpha))\}$ □

It is easy to show, by definition of $\mathcal{I}_v$, that the following properties hold for any Belnap formula $\alpha$ and **4**-valuation $v$.

- $v(\alpha) = \mathsf{T}$ if and only if $\mathcal{I}_v \models holds(\alpha, \mathsf{tt}) \wedge \neg holds(\alpha, \mathsf{ff})$
- $v(\alpha) = \mathsf{F}$ if and only if $\mathcal{I}_v \models holds(\alpha, \mathsf{ff}) \wedge \neg holds(\alpha, \mathsf{tt})$
- $v(\alpha) = \mathsf{Both}$ if and only if $\mathcal{I}_v \models holds(\alpha, \mathsf{tt}) \wedge holds(\alpha, \mathsf{ff})$
- $v(\alpha) = \mathsf{None}$ if and only if $\mathcal{I}_v \models \neg holds(\alpha, \mathsf{tt}) \wedge \neg holds(\alpha, \mathsf{ff})$

The following proposition shows that a classical interpretation $\mathcal{I}_v$ associated to a given **4**-valuation $v$ is a model of the first-order axioms $\mathcal{A}_B$.

**Proposition 1** *Let $v$ be a **4**-valuation and let $\mathcal{I}_v$ be its associated classical interpretation. Then $\mathcal{I}_v$ is a model of the classical axiomatisation $\mathcal{A}_B$.*

**Proof:** The proof is by cases of each axiom of $\mathcal{A}_B$.

Case 1: (Ax 1). We want to show that $\mathcal{I}_v \models \forall x[holds(x, \mathsf{ff}) \leftrightarrow holds(\neg x, \mathsf{tt})]$. We reason by contradiction. We assume that, for some $x$, $\mathcal{I}_v \models holds(x, \mathsf{ff})$ and $\mathcal{I}_v \not\models holds(\neg x, \mathsf{tt})$. By definition of $\mathcal{I}_v$, $\mathsf{ff} \in v(x)$, which implies by the $\neg$ truth table that $\mathsf{tt} \in v(\neg x)$. Hence $\mathcal{I}_v \models holds(\neg x, \mathsf{tt})$ which contradicts the hypothesis. Similarly for the other case, i.e. $\mathcal{I}_v \not\models holds(x, \mathsf{ff})$ and $\mathcal{I}_v \models holds(\neg x, \mathsf{tt})$.

Case 3: (Ax 3). We want to show that $\mathcal{I}_v \models \forall x, y[holds(x \wedge y, \mathsf{tt}) \leftrightarrow (holds(x, \mathsf{tt}) \wedge holds(y, \mathsf{tt}))]$. We reason by contradiction. Assume that, for some $x$, $\mathcal{I}_v \models holds(x \wedge y, \mathsf{tt})$, and $\mathcal{I}_v \not\models holds(x, \mathsf{tt})$ or $\mathcal{I}_v \not\models holds(y, \mathsf{tt})$. By definition of $\mathcal{I}_v$, $\mathsf{tt} \in v(x \wedge y)$, which implies by the $\wedge$ truth table that $\mathsf{tt} \in v(x)$ and $\mathsf{tt} \in v(y)$. Therefore, $\mathcal{I}_v \models holds(x, \mathsf{tt})$ and $\mathcal{I}_v \models holds(y, \mathsf{tt})$, which is in contradiction with the initial hypothesis. The second case, i.e. assume that, for some $x$, $\mathcal{I}_v \not\models holds(x \wedge y, \mathsf{tt})$, and $\mathcal{I}_v \models holds(x, \mathsf{tt})$ and $\mathcal{I}_v \models holds(y, \mathsf{tt})$, can be proved following the same argument.

Case 5: (Ax 5). We want to show that $\mathcal{I}_v \models \forall x, y[holds(x \vee y, \mathsf{tt}) \leftrightarrow (holds(x, \mathsf{tt}) \vee holds(y, \mathsf{tt}))]$. We reason by contradiction. Assume that, for some $x$, $\mathcal{I}_v \models holds(x \vee y, \mathsf{tt})$, and $\mathcal{I}_v \not\models holds(x, \mathsf{tt})$ and $\mathcal{I}_v \not\models holds(y, \mathsf{tt})$. By definition of $\mathcal{I}_v$, $\mathsf{tt} \in v(x \vee y)$, which implies by the $\vee$ truth table that $\mathsf{tt} \in v(x)$ or $\mathsf{tt} \in v(y)$. Therefore, $\mathcal{I}_v \models holds(x, \mathsf{tt})$ or $\mathcal{I}_v \models holds(y, \mathsf{tt})$, which is in contradiction with the initial hypothesis. The second case, i.e. assume that, for some $x$, $\mathcal{I}_v \not\models holds(x \vee y, \mathsf{tt})$, and $\mathcal{I}_v \models holds(x, \mathsf{tt})$ or $\mathcal{I}_v \models holds(y, \mathsf{tt})$, can be proved following the same argument.

Axioms 2,4 and 6 are proved in an analogous way of the proofs of Axioms, 1,3 and 5, respectively. □

Proof of Lemma 2. We prove the contrapositive statement. We assume that $\psi \not\to \varphi$ and we want to show that either $\mathcal{A}_B, holds(\psi, \mathsf{tt}) \not\vdash holds(\varphi, \mathsf{tt})$ or $\mathcal{A}_B, \neg holds(\psi, \mathsf{ff}) \not\vdash \neg holds(\varphi, \mathsf{ff})$. The hypothesis $\psi \not\to \varphi$ implies different cases or truth values for $\psi$ and $\varphi$ according to the ordering relation $\preceq$ over the logical lattice **L4**. We consider these cases individually. $\psi \not\to \varphi$ implies that for some **4**-valuation $v$, $v(\psi) \not\preceq v(\varphi)$.

Case 1: $v(\psi) = \mathsf{T}$ and $v(\varphi) = \mathsf{Both}$. From $v$, we can construct the associated classical interpretation $\mathcal{I}_v$. By definition, $\mathcal{I}_v \models holds(\psi, \mathsf{tt})$ and $\mathcal{I}_v \models \neg holds(\psi, \mathsf{ff})$. But $\mathcal{I}_v \not\models \neg holds(\varphi, \mathsf{ff})$.

Case 2: $v(\psi) = \mathsf{T}$ and $v(\varphi) = \mathsf{None}$. Then $\mathsf{tt} \notin v(\varphi)$. From $v$, we can construct the associated classical interpretation $\mathcal{I}_v$. By definition, $\mathcal{I}_v \models holds(\psi, \mathsf{tt})$ and $\mathcal{I}_v \models \neg holds(\psi, \mathsf{ff})$. But $\mathcal{I}_v \not\models holds(\varphi, \mathsf{tt})$.

Case 3: $v(\psi) = \mathsf{T}$ and $v(\varphi) = \mathsf{F}$. Then $\mathsf{tt} \notin v(\varphi)$. From $v$, we can construct the associated classical interpretation $\mathcal{I}_v$. By definition, $\mathcal{I}_v \models holds(\psi, \mathsf{tt})$ and $\mathcal{I}_v \models \neg holds(\psi, \mathsf{ff})$. But $\mathcal{I}_v \not\models holds(\varphi, \mathsf{tt})$.

Case 4: $v(\psi) = \mathsf{None}$ and $v(\varphi) = \mathsf{F}$. Then $\mathsf{tt} \notin v(\psi)$, $\mathsf{ff} \notin v(\psi)$ and $\mathsf{tt} \notin v(\varphi)$. From $v$, we can construct the associated classical interpretation $\mathcal{I}_v$. By definition, $\mathcal{I}_v \models \neg holds(\psi, \mathsf{ff})$, but but $\mathcal{I}_v \not\models \neg holds(\varphi, \mathsf{ff})$.

Case 5: $v(\psi) = \mathsf{Both}$ and $v(\varphi) = \mathsf{F}$. Then $\mathsf{tt} \in v(\psi)$, $\mathsf{ff} \in v(\psi)$ and $\mathsf{tt} \notin v(\varphi)$. From $v$, we can construct the associated classical interpretation $\mathcal{I}_v$. By definition, $\mathcal{I}_v \models holds(\psi, \mathsf{tt})$, but but $\mathcal{I}_v \not\models holds(\varphi, \mathsf{tt})$. □

Proof of Theorem 1.
The "if-part" is given by Lemma 2 whereas the "only-if" part is given by Lemma 1. □

# 3 Conclusions and Future work

In this report we have provided a translation method of Belnap's four-valued logic into classical logic. The translation method is sound and complete with respect to Belnap's notion of entailment in the sense that for two Belnap formulae $\psi$ and $\varphi$

$$\begin{aligned} \psi \rightarrow \varphi \quad \text{iff} \quad & \mathcal{A}_B, holds(\psi, \mathsf{tt}) \vdash holds(\varphi, \mathsf{tt}) \\ \text{and} \quad & \mathcal{A}_B, \neg holds(\psi, \mathsf{ff}) \vdash \neg holds(\varphi, \mathsf{ff}). \end{aligned}$$

where $\mathcal{A}_B$ is an axiomatisation in first-order logic describing the properties of Belnap's logic.

This translation is to be used in a forthcoming paper where the use of (classical) belief revision operators for non-classical logics will be investigated.

# Acknowledgements

We would like to thank Prof. Dov Gabbay for useful discussions carried out during the preparation of this report.

# References

[1] N. Belnap. How a computer should think. In G. Ryle, editor, *Contemporary Aspects of Philosophy*, pages 30–56. Oriel Press, 1977.

[2] N. 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.

[3] Peter Gärdenfors. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. A Bradford Book - The MIT Press, Cambridge, Massachusetts - London, England, 1988.