Two, Many, And Differently Many

Diderik Batens*
Centre for Logic and Philosophy of Science
Ghent University, Belgium
Diderik.Batens@UGent.be

JY – November 2, 2014

Abstract

This paper is a modest contribution to a universal logic approach to many-valued semantic systems. The main focus is on the relation between such systems and two-valued ones. The matter is discussed for usual many-valued semantic systems. These turn out to exist for more logics than expected. A new type of many-valued semantics is devised and its use illustrated. The discussion, which involves truth-functionality and the syntactic rendering of truth-values, leads to philosophical conclusions.

Keywords Many-valued semantics, universal logic, paraconsistency.

1 Aim of this paper

The general aim of this paper concerns the relation between two-valued and many-valued semantic characterizations of Tarski logics—reflexive, transitive and monotonic functions that map sets of closed formulas (the premises) to sets of closed formulas (the consequences). Suszko [38] has shown that logics (in this sense of the term) have a two-valued semantics.[1] This casts doubts on the use of "many-valued" as an attribute of logics. Moreover, it suggests that, at least for some logics, there must be an interesting relation between the values of their two-valued semantics and those of their many-valued semantics. We shall see in Section 3 that this relation is usually of a specific kind. In the present article three-valued and four-valued logics will be introduced that are rather unusual. For example, the relation between their values and the values from the two-valued semantic characterization will be very different from the similar relation for usual many-valued logics.

A two-valued semantic characterization of a (so-called) many-valued logic connects the logical symbols to truth-preservation, as Suszko's proof underlines. We shall consider several relations that connect the values of the two-valued characterization to values of a many-valued characterization. Varying the relation brings one from the same two-valued semantics to different many-valued semantics and to different interpretations.

*The author is indebted to Joke Meheus for comments on a former draft.

[1] Throughout this paper, "semantics of a logic **L**" should be read as "characteristic **L**-semantics".

A new such relation will be introduced in this paper. Its use will be illustrated in terms of a specific problem in adaptive logics, viz. the identification of flip-flop logics—Section 6. Readers that have no specific insight in adaptive logics should not worry; the flip-flop problem will be easy to understand. In comparison to **CL** (Classical Logic), some Tarski logics allow for gluts or gaps—see Section 2—with respect to certain logical symbols. Delineating the precise points at which gluts or gaps originate offers important insights for solving the flip-flop problem. As we shall see, the delineation also provides an revealing interpretation of the non-extreme values.

Needless to say, the new type of many-valued logics are not meant to replace the usual ones. Also, they are not superior in any sense of the term. They are helpful, however, to reveal the presuppositions that lurk underneath widespread views on many-valued logics, often confusing technical features with ontological ones. Rather central presuppositions concern truth-functionality.

2 Preliminaries

A logic is defined over a language schema $\mathcal{L}$ of which $\mathcal{F}$ is the set of formulas and $\mathcal{W}$ the set of closed formulas. I shall need some names for sets of schematic letters for non-logical symbols: $\mathcal{S}$ (sentential letters), $\mathcal{P}^r$ (predicates of each rank $r \in \{1, 2, \ldots\}$), $\mathcal{C}$ (individual constants), and $\mathcal{V}$ (individual variables).

To handle quantifiers in the semantics, extend $\mathcal{L}$ to the pseudo-language schema $\mathcal{L}_{\mathcal{O}}$. This is just like $\mathcal{L}$ except that the role of $\mathcal{C}$ is played by $\mathcal{C} \cup \mathcal{O}$, in which $\mathcal{O}$ is a set of pseudo-constants. Strictly speaking, we introduce a pseudo-language schema $\mathcal{O}$ for each model M, requiring that $\mathcal{O}$ has at least the cardinality of the domain of M.[2] $\mathcal{F}_{\mathcal{O}}$ and $\mathcal{W}_{\mathcal{O}}$ are the sets of formulas, respectively closed formulas, of $\mathcal{L}_{\mathcal{O}}$.

A model is a tuple. One of the elements of the tuple is the domain D, which is a non-empty set. Another element is the assignment v, which relates certain linguistic entities to the other elements of the tuple. Next, for each model M, the meaning of the logical symbols of $\mathcal{L}$ is fixed by v_M, the valuation function determined by M.

In the usual **CL**-semantics, v relates the non-logical symbols to the model. More specifically $v: \mathcal{S} \to \{0, 1\}$. In order to turn indeterministic two-valued semantic systems into deterministic ones—see below—I shall generalize this to $v: \mathcal{W}_{\mathcal{O}} \to \{0, 1\}$.[3]

In the case of the **CL**-semantics, the pseudo-language schemas $\mathcal{L}_{\mathcal{O}}$ extend the standard predicative language schema $\mathcal{L}_s$ (which I do not describe here). A **CL**-model $M = \langle D, v \rangle$, in which D is a non-empty set and v an assignment function, is an interpretation of $\mathcal{W}_{\mathcal{O}}$. The assignment function v is defined by:[4]

C1 $\quad v: \mathcal{W}_{\mathcal{O}} \to \{0, 1\}$

[2]The pseudo-language schema $\mathcal{L}_{\mathcal{O}}$ is not a language schema whenever its set of symbols is non-denumerable. The resulting style of semantics—examples follow in the text—offer a means to quantify over non-denumerable sets.

[3]This move is independent of the reference to the pseudo-language schema $\mathcal{L}_{\mathcal{O}}$. To combine the move with a different semantic style, restrict it to $v: \mathcal{W} \to \{0, 1\}$.

[4]The restriction in C2 ensures that $\langle D, v \rangle$ is only a **CL**-model if every element of D is named by a constant or pseudo-constant. In C3, $\wp(D^r)$ is the power set of the r-th Cartesian product of D.

C2 $\quad v\colon \mathcal{C} \cup \mathcal{O} \to D \quad$ (where $D = \{v(\alpha) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$)
C3 $\quad v\colon \mathcal{P}^r \to \wp(D^r)$

The valuation function determined by M, $v_M\colon \mathcal{W}_\mathcal{O} \to \{0,1\}$, is defined as follows:

C$\mathcal{S}$ $\quad$ where $A \in \mathcal{S}$, $v_M(A) = v(A)$
C$\mathcal{P}^r$ $\quad v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1$ iff $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)$
C= $\quad v_M(\alpha = \beta) = 1$ iff $v(\alpha) = v(\beta)$
C¬ $\quad v_M(\neg A) = 1$ iff $v_M(A) = 0$
C⊃ $\quad v_M(A \supset B) = 1$ iff $v_M(A) = 0$ or $v_M(B) = 1$
C∧ $\quad v_M(A \wedge B) = 1$ iff $v_M(A) = 1$ and $v_M(B) = 1$
C∨ $\quad v_M(A \vee B) = 1$ iff $v_M(A) = 1$ or $v_M(B) = 1$
C≡ $\quad v_M(A \equiv B) = 1$ iff $v_M(A) = v_M(B)$
C∀ $\quad v_M(\forall\alpha A(\alpha)) = 1$ iff $\{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\} = \{1\}$
C∃ $\quad v_M(\exists\alpha A(\alpha)) = 1$ iff $1 \in \{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$

$M \Vdash A$ (**CL**-model M verifies A) iff $v_M(A) = 1$. M is a model of Γ iff $M \Vdash A$ for all $A \in \Gamma$. $\Gamma \vDash_{\mathbf{CL}} A$ iff all models M of Γ verify A. $\vDash_{\mathbf{CL}} A$ (A is valid) iff all models verify A.

The metalanguage is classical and will be so in the rest of this article. All identities that occur in the semantics outside the expression $v_M(\alpha = \beta)$ are metalinguistic identities and hence are classical.

I still need to illustrate the use of clause C1. The logic **CLuN** is like **CL** except in that it allows for gl*u*ts with respect to *N*egation—in some **CLuN**-models M there are A such that $v_M(A) = v_M(\neg A) = 1$.[5] A semantics of **CLuN** is obtained from the **CL**-semantics by replacing the equivalence C¬ by an implication

C$\neg^{iu}$ $\quad$ if $v_M(A) = 0$, then $v_M(\neg A) = 1$

The resulting logic is paraconsistent (for example $p, \neg p \nvDash_{\mathbf{CLuN}} q$),[6] invalidates Replacement of Equivalents (for example $\vDash_{\mathbf{CLuN}} p \equiv (p \vee p)$ and $\vDash_{\mathbf{CLuN}} \neg p \supset \neg p$ but $\nvDash_{\mathbf{CLuN}} \neg p \supset \neg(p \vee p)$), invalidates Replacement of Identicals (for example $a = b, \neg Pa \nvDash_{\mathbf{CLuN}} \neg Pb$), and invalidates many other rules validated by **CL** (for example Disjunctive Syllogism, Contraposition, Modus Tollens, . . .).

This **CLuN**-semantics is indeterministic. Indeed, consider a **CLuN**-model $M = \langle D, v\rangle$ in which $v(p) = 1$ and hence $v_M(p) = 1$. In view of C$\neg^{iu}$, both $v_M(\neg p) = 0$ and $v_M(\neg p) = 1$ are possible. To be more precise, the **CLuN**-semantics is bound to contain a copy of this M in which $v_M(\neg p) = 0$ and another copy in which $v_M(\neg p) = 1$. Both copies of M need to belong to the **CLuN**-semantics because a semantics is required to exhaust the logical possibilities—in this case **CLuN**-possibilities.

That a semantics is indeterministic is somewhat annoying. Models are supposed to exhaust the logical possibilities. That variants have to be taken into account—actually a non-denumerable set of variants for each model—introduces

[5] The indeterministic propositional semantics was first formulated in [8]; the deterministic predicative semantics in [10].

[6] Technically speaking, a logic **L** is paraconsistent iff $A, \neg A \vdash_{\mathbf{L}} B$ does not hold generally. Interesting discussions of the underlying philosophical questions are available, for example by Béziau [18, 19].

a complication that is not matched by any advantage.[7] Fortunately it is possible to devise a deterministic semantics for **CLuN** [10] and the result was later generalized to other gluts and gaps—the best survey paper on the matter [15] is only electronically published at this moment. In order to obtain the deterministic **CLuN**-semantics one replaces $\mathrm{C}\neg^{iu}$ by

$\mathrm{C}\neg^{u}$ $\quad v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $v(\neg A) = 1$.

The first disjunct guarantees that Excluded Middle holds, the second disjunct introduces gluts for some A.

This approach is easily generalized, first to gaps with respect to negation and to both gluts and gaps with respect to negation, and next to gluts and gaps with respect to other logical symbols—details are in another paper [15]. Just to give you the flavour, the logic **CLaN** allows for gaps (not gluts) with respect to negation. Its indeterministic semantics requires

$\mathrm{C}\neg^{ia}$ $\quad$ if $v_M(A) = 1$, then $v_M(\neg A) = 0$

and its deterministic semantics is delivered by

$\mathrm{C}\neg^{a}$ $\quad v_M(\neg A) = 1$ iff $v_M(A) = 0$ and $v(\neg A) = 1$.

CLoN allows for both gaps and gluts with respect to negation. Its indeterministic semantics is obtained by dropping the negation clause altogether. Its deterministic semantics is obtained by

$\mathrm{C}\neg^{o}$ $\quad v_M(\neg A) = v(\neg A)$.

Restoring Replacement of Identicals in **CLuN** is easy. Given a **CLuN**-model M, define, for each $A \in \mathcal{W}_{\mathcal{O}}$, an equivalence class $[\![A]\!]$: (i) $A \in [\![A]\!]$ and (ii) if $\alpha, \beta \in \mathcal{C} \cup \mathcal{O}$, A is $B(\alpha)$, and $v(\beta) = v(\alpha)$, then $B(\beta) \in [\![A]\!]$. Note that $[\![A]\!] = \{A\}$ if $A \in \mathcal{S}$. Next, replace $\mathrm{C}\neg^{u}$ by

$\mathrm{C}\neg^{uR}$ $\quad v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $v(\neg B) = 1$ for a $B \in [\![A]\!]$.

and analogously for $\mathrm{C}\neg^{a}$ and $\mathrm{C}\neg^{o}$. Note that the semantics characterizes the same logic if "for a $B \in [\![A]\!]$" is replaced by "for all $B \in [\![A]\!]$"—for every model of the one semantics there is a model of the other semantics that verifies exactly the same members of $\mathcal{W}$—recall that verification depends on v_M and not on v.

The present approach to gluts and gaps leads to rather basic logics. Thus the propositional fragment of **CLuN** was shown [32] to be the intersection of all propositional logics that allow for negation gluts but do not allow for any other gluts nor for any gaps. Obviously some logics extend **CLuN** and are nevertheless paraconsistent, and similarly for other gluts and for gaps. A very popular paraconsistent extension of **CLuN** is a logic that I prefer to call **CLuNs** because its propositional version was first proposed by Kurt Schütte [36].[8] The idea is that de Morgan properties and all similar negation-reducing properties are restored. I consider at once a version that validates Replacement

[7]Still, indeterministic semantic systems, have been around at least since the 1970s and led to interesting studies, for example in work by Arnon Avron and associates [5, 6, 7].

[8]**CLuNs** is apparently the most popular paraconsistent logic. It is known under a mutiplicity of names. Further useful references to studies of **CLuNs** and of its fragments are [2, 3, 4, 8, 16, 22, 24, 25, 26, 27, 28, 29, 33, 34, 37]. Proofs of some results are in [16, 13].

of Identicals. Let $\mathcal{F}^a_{\mathcal{O}}$ be the set of atomic (or primitive) members of $\mathcal{F}_{\mathcal{O}}$—those not containing any logical symbols other than identity—and $\mathcal{W}^a_{\mathcal{O}}$ the set of atomic members of $\mathcal{W}_{\mathcal{O}}$. The deterministic **CLuNs**-semantics is obtained by replacing C$\neg$ in the **CL**-semantics by the following clauses:

C$\neg^{us}$ where $A \in \mathcal{W}^a_{\mathcal{O}}$, $v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $v(\neg B) = 1$ for a $B \in [\![A]\!]$
C$\neg\neg$ $v_M(\neg\neg A) = v_M(A)$
C$\neg\supset$ $v_M(\neg(A \supset B)) = v_M(A \wedge \neg B)$
C$\neg\wedge$ $v_M(\neg(A \wedge B)) = v_M(\neg A \vee \neg B)$
C$\neg\vee$ $v_M(\neg(A \vee B)) = v_M(\neg A \wedge \neg B)$
C$\neg\equiv$ $v_M(\neg(A \equiv B)) = v_M((A \vee B) \wedge (\neg A \vee \neg B))$
C$\neg\forall$ $v_M(\neg\forall\alpha A(\alpha)) = v_M(\exists\alpha\neg A(\alpha))$
C$\neg\exists$ $v_M(\neg\exists\alpha A(\alpha)) = v_M(\forall\alpha\neg A(\alpha))$

The **CLuNs**-semantics enables me to illustrate a method to restore Replacement of Identicals that does not refer to equivalence sets. Let v assign to predicates a couple of extensions rather than a single one: $v\colon \mathcal{P}^r \to \langle\Sigma_1, \Sigma_2\rangle$ with $\Sigma_1 \cup \Sigma_2 = D^r$. Identity is handled as a binary predicate with the special characteristic that $\Sigma_1 = \{\langle o, o\rangle \mid o \in D\}$. For all predicates, including identity, one defines $v^+(\pi^r) = \Sigma_1$ and $v^-(\pi^r) = \Sigma_2$. Finally, one replaces $v(\pi^r)$ by $v^+(\pi^r)$ in C$\mathcal{P}^r$, and one replaces C$\neg^{us}$ by two clauses:

- where $A \in \mathcal{S}$, $v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $v(\neg A) = 1$
- $v_M(\neg\pi^r\alpha_1 \ldots \alpha_r) = 1$ iff $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v^-(\pi^r)$

This may be called the $\pm$-semantics of **CLuNs**.[9]

The words "glut" and "gap" were used rather intuitively until now. Actually, these words were used with several meanings in the literature. For example, Georg Henrik von Wright [40] says that there is an overlap (rather than glut) when a formula is true together with its negation and that there is a gap if a formula is false together with its negation. In the same place, von Wright calls a formula false iff its negation is true. So he also says that there is a glut (or overlap) if a formula is both true and false, and a gap if it is neither.

I shall use the terms differently. I already mentioned that the metalanguage of this paper is fully classical. Unlike von Wright, I shall consider "true" and "false" as exhaustive and exclusive within a classical metalanguage. Finally, "glut" and "gap" will be used in a way that is directly contingent on the clauses of the **CL**-semantics. Any formula in $\mathcal{W}_{\mathcal{O}} - \mathcal{W}^a_{\mathcal{O}}$ has a specific logical form F, determined by its central logical term $*$. The **CL**-semantics contains a specific clause for F. This clause may be seen as the conjunction of two implications, one specifying when $v_M(\mathsf{F}) = 1$ and one specifying when $v_M(\mathsf{F}) = 0$. Consider a model M of a logic **L**. If in M the condition is fulfilled for $v_M(A) = 1$ in **CL**-models but $v_M(A) = 0$, then this situation is said to cause a $*$-gap. If in M the condition is fulfilled for $v_M(A) = 0$ in **CL**-models but $v_M(A) = 1$, then this is said to cause a $*$-glut. Thus that $v_M(p \vee q) = v_M(\neg r) = 1$ and $v_M((p \vee q) \wedge \neg r) = 0$ causes a $\wedge$-gap. Similarly, that $1 \notin \{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$ and $v_M(\exists\alpha A(\alpha)) = 1$ causes a $\exists$-glut.

Let us make this more precise. Define, for each $A \in \mathcal{W}_{\mathcal{O}}$, the set of *direct subformulas* of A, dsub(A), as follows: (i) dsub$(\neg A) = \{A\}$, (ii) where $* \in$

[9] Another version of the approach, requiring only a single clause, is illustrated in a paper under review [39].

$\{\vee, \wedge, \supset, \equiv\}$, $\text{dsub}(A * B) = \{A, B\}$, and (iii) where $\alpha \in \mathcal{V}$ and $* \in \{\forall, \exists\}$, $\text{dsub}(*\alpha A(\alpha)) = \{A(\beta) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$. If $A \notin \mathcal{W}^a_{\mathcal{O}}$, then, in the **CL**-semantics, $v_M(A)$ is a function of the valuation values of the members of $\text{dsub}(A)$. In a semantics of a different logic, $v_M(A)$ will be said to be a glut or a gap if it departs from that function. Consider a **CLuN**-model in which $v_M(p) = 1$, $v_M(\neg p) = 1$ and $v_M(\neg\neg p) = 0$. The former two cause a $\neg$-glut. The combination of the first and third do not cause a gap, notwithstanding the fact that, within the **CL**-semantics, $v_M(\neg\neg A) = 1$ whenever $v_M(A) = 1$.

This seems the best point to mention a few simple technicalities. The set of *subformulas* of $A \in \mathcal{W}_{\mathcal{O}}$, $\text{sub}(A)$ is the smallest set such that (i) $\text{dsub}(A) \subseteq \text{sub}(A)$ and (ii) if $B \in \text{sub}(A)$, then $\text{sub}(B) \subseteq \text{sub}(A)$. Note that $A \notin \text{sub}(A)$.

We shall need the set of *first subformulas* of $A \in \mathcal{W}_{\mathcal{O}}$, $\text{fsub}(A)$. This is the smallest set such that (i) $A \in \text{fsub}(\neg A)$, (ii) where $* \in \{\vee, \wedge, \supset, \equiv\}$, $A \in \text{fsub}(A * B)$, (iii) where $\alpha \in \mathcal{V}$ and $* \in \{\forall, \exists\}$, $A(a) \in \text{fsub}(*\alpha A(\alpha))$, and (iv) if $A \in \text{fsub}(B)$, then $\text{fsub}(A) \subseteq \text{fsub}(B)$.[10] The crucial distinction with the set $\text{sub}(A)$ is in (ii) and (iii). We actually need $\text{fsub}(A)$ to define the set of *first superformulas* of A, viz. $\text{fsup}(A) = \{B \in \mathcal{W}_{\mathcal{O}} \mid A \in \text{fsub}(B)\}$.

Another concept we shall need is that of a (finite) *pseudo-partition*. $\{\Sigma_1, \ldots, \Sigma_n\}$ is a pseudo-partition of Σ iff (i) $\Sigma_i \cap \Sigma_j = \emptyset$ for all different $i, j \in \{1, \ldots, n\}$ and (ii) $\bigcup\{\Sigma_1, \ldots, \Sigma_n\} = \Sigma$ (but it is not required that the members of the pseudo-partition are non-empty).

3 The Usual Many-Valued Approach

The **CL**-semantics is deterministic: the valuation value of every formula is determined by the model; in the presence of $\mathcal{L}_{\mathcal{O}}$ it is determined by the assignment values of the non-logical symbols that occur in the formula and its subformulas. The **CL**-semantics is also recursive: there is a complexity function such that, for every non-atomic formula $A \in \mathcal{W}_{\mathcal{O}}$, $v_M(A)$ depends only on valuation values of formulas that are less complex than A. The **CL**-semantics is also truth-functional: there is a function that connects the valuation value of every non-atomic formula $A \in \mathcal{W}_{\mathcal{O}}$ to the valuation values of subformulas of A; only the value of atomic formulas is directly determined by assignment values.

A presupposition of this type of semantics is that a distinction can be made between two things. On the one hand, there is the model itself: $M = \langle D, v\rangle$. This represents a state of the world. All non-logical symbols receive their meaning here—the assignment assures that they do. On the other hand, there is the realm of the logical symbols. These are required to formulate statements about the world. Indeed, by merely concatenating atomic statements one cannot express that an atomic statement is false or that one of two atomic statements is true but not necessarily both. Still, truth-functionality makes complex statements parasitic on atomic statements in that the valuation values of the complex statements are fully determined by the valuation values of atomic statements.

To be sure, a semantics is defined with respect to a language schema. In this sense, its models represent at best structural states of the world. In order to transform the models into representations of actual states of the world, one needs to replace the language schema by a language that is covered by the schema. So

[10] In (iii), a is the alphabetically first individual constant, which is used here as a metalinguistic name of itself.

a semantics involves a hypothesis about the structure of the language in which the world, or some parts or aspects of it, may be adequately described. Needless to say, adequacy comes in degrees and the estimated adequacy may be poor due to the present state of our knowledge.[11]

When many-valued logics came around, new valuation values were added next to "true" and "false". There were two intuitions behind the new values. One idea was that some sentences do not have a truth-value, but are indeterminate in one of several senses. According to the other idea there are further truth-values, which are sometimes seen as expressing degrees of partial truth. The logical symbols of those logics were still truth-functions, viz. with respect to the extended set of valuation values. Apparently the architects of many-valued logics first had the idea of additional 'truth-values' and next devised truth-functional operators in terms of them. They apparently did not imagine, and possibly could not imagine, that a logical symbol would not be truth-functional.[12]

Within a many-valued semantics, semantic consequence is defined in terms of designated and non-designated values. This shows the way from the many-valued semantics to the two-valued one. It is instructive to consider also the opposite road. This road was explored a long time ago for the propositional case, among others by me [9], and the generalisation to the predicative level is obvious. The idea is that bivalent values of several formulas are 'melted together' into a many-valued value of a single formula. The approach works fine for some paraconsistent logics, for example for **CLuNs**. Consider, for any $A \in \mathcal{W}_{\mathcal{O}}$, the couple $\langle v_M(A), v_M(\neg A)\rangle$ in the above two-valued semantics. The possible couples are $\langle 1, 0\rangle$, $\langle 1, 1\rangle$, and $\langle 0, 1\rangle$. These may be handled as three valuation values and, if they are so handled, it is convenient to rename them to T, I, and F, which correspond to "consistently true", "inconsistent" and "consistently false" respectively. The resulting three-valued semantics is truth-functional, as I show below. The matter is utterly simple for the propositional case. For predicative models, the easiest approach requires that the assignment is redefined, for example as in the next paragraph.

In a three-valued **CLuNs**-model $M = \langle D, V\rangle$, defined over the language $\mathcal{L}_{\mathcal{O}}$, the domain D is a set and the assignment V has the following four properties. (i) $V\colon \mathcal{S} \to \{T, I, F\}$.[13] (ii) $V\colon \mathcal{C} \cup \mathcal{O} \to D$ (where $D = \{V(\alpha) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$). (iii) $V\colon \mathcal{P}^r \to \langle \Sigma_1, \Sigma_2, \Sigma_3\rangle$ such that $\{\Sigma_1, \Sigma_2, \Sigma_3\}$ is a pseudo-partition of $\wp(D^r)$. To simplify the notation, consider V as composed in this case of the three functions V^T, V^I, and V^F, with $V^T(\pi^r) = \Sigma_1$, $V^I(\pi^r) = \Sigma_2$, and $V^F(\pi^r) = \Sigma_3$.[14] (iv) Identity is handled as a binary predicate with the special characteristic that $V^T(=) \cup V^I(=) = \{\langle o, o\rangle \mid o \in D\}$.

The valuation function $V_M\colon \mathcal{W}_{\mathcal{O}} \to \{T, I, F\}$ is defined as follows:

C$\mathcal{S}$ where $A \in \mathcal{S}$, $V_M(A) = V(A)$

[11] Carnap [20, 21] clearly saw the linguistic relativity of the semantic enterprise. Apparently, many have forgotten his insight and seem to presume that they can talk about states of the world in an absolute way.

[12] I do not intend to refer, for example, to a worlds semantics but rather to a non-truth-functional semantics such as the ones from Section 2, the indeterministic as well as the deterministic ones.

[13] There is no need to assign a three-valued assignment value to all members of $\mathcal{W}_{\mathcal{O}}$.

[14] The three functions determine for which r-tuples the predicate is true, inconsistent, and false respectively.

C$\mathcal{P}^r$ for $X \in \{T, I, F\}$, $V_M(\pi^r\alpha_1 \ldots \alpha_r) = X$ iff $\langle V(\alpha_1), \ldots, V(\alpha_r)\rangle \in V^X(\pi^r)$

Ccon

	$\neg$
T	F
I	I
F	T

$\supset$	T	I	F
T	T	I	F
I	T	I	F
F	T	T	T

$\wedge$	T	I	F
T	T	I	F
I	I	I	F
F	F	F	F

C$\forall$ $V_M((\forall\alpha)A(\alpha)) = T$ iff $V_M(A(\beta)) = T$ for all $\beta \in \mathcal{C} \cup \mathcal{O}$
$V_M((\forall\alpha)A(\alpha)) = F$ iff $V_M(A(\beta)) = F$ for at least one $\beta \in \mathcal{C} \cup \mathcal{O}$
$V_M((\forall\alpha)A(\alpha)) = I$ otherwise

Define $M \Vdash A$ (a three-valued **CLuNs**-model M verifies A) iff $V_M(A) \in \{T, I\}$; and so on.

The other logical symbols are defined explicitly: $A \vee B =_{df} \neg(\neg A \wedge \neg B)$, $A \equiv B =_{df} (A \supset B) \wedge (B \supset A)$, and $(\exists\alpha)A(\alpha) =_{df} \neg(\forall\alpha)\neg A(\alpha)$.

This three-valued **CLuNs**-semantics is equivalent to the two-valued **CLuNs**-semantics from Section 2 in that their semantic consequence relations coincide—this is easily shown by slightly modifying the proof of Theorem 1 in [16].

So this is the usual approach to many-valued logics. There is a n-tuple of functions $\langle f_1, \ldots, f_n\rangle$ such that $f_i \colon \mathcal{W}_{\mathcal{O}} \to \mathcal{W}_{\mathcal{O}}$ for each i; the n-tuple of bivalent values $\langle v_M(f_1(A)), \ldots, v_M(f_n(A))\rangle$ functions as the many-valued value $V_M(A)$. Expressed somewhat crudely, the (bivalent) values of formulas containing A are pushed into the (many-valued) value of A.

The attractiveness of the approach seems related to the fact that the many-valued valuation values appear to be a kind of truth-values that are more sophisticated than the bivalent valuation values and that are introduced for sound philosophical reasons. The view on logical symbols is simply the traditional view: they are truth-functions. They differ from the classical logical symbols as a result of the modified set of valuation values.

Let us proceed more carefully. We already knew that a logic that has a many-valued semantics also has a two-valued semantics. It seems obvious that any many-valued semantics can be described as obtained by pushing the bivalent values of a tuple of formulas into the many-valued value of an atomic formula. From a technical point of view, the two semantics are on a par. Still, there is the philosophical question which semantics is ontologically correct. Are there many truth-values or are these merely tuples of binary truth-values? Consider again **CLuNs**. Is the truth of $\neg A$ a consequence of the fact that A has the truth-value I or is saying that A has the value I merely a statement summarizing that A and $\neg A$ are both true?

The truth-values of the bivalent **CL**-semantics may be seen as 'expressed' within the standard **CL**-language by A and $\neg A$ respectively. Similarly, one may (explicitly) define n logical symbols $\mathsf{V}_1, \ldots, \mathsf{V}_n$ within the language of Łukasiewicz's n-valued logics $\mathbf{Ł}_n$ such that, for all $i \in \{1, \ldots, n\}$, $M \Vdash \mathsf{V}_i A$ iff $v_M(A) = i$—see [35, 39]. It is not possible to do so for **CLuNs**; a definable symbol does so correspond to the semantic value I, but no definable logical symbol so corresponds to T or to F.

Some will see this as an argument to consider the truth-values of the two-valued **CL**-semantics and those of the n-valued $\mathbf{Ł}_n$-semantics as real truth-values, at least with respect to the presuppositions of those logics, but will not consider the valuation values of the three-valued **CLuNs**-semantics as real truth-values.[15] They might argue that, if statements may have three different truth-values, then the logical symbols of your language should enable you to express, for each of the truth-values, that it pertains to a statement. Whether you may get to *know* the actual truth-value of a statement is altogether a different matter.[16]

Others will be less demanding and consider the fact that a logic has an adequate many-valued semantics in which all its logical symbols are truth-functions as a sufficient reason to consider those values as truth-values. The view apparently presupposes that some logics do not have such a semantics, but is that correct?

It seems unlikely a priori that **CLuN** has an adequate semantics in which all logical symbols are truth-functions. Apart from some transparent exceptions, such as $(p\wedge\neg p)\wedge\neg(p\wedge\neg p) \vdash_{\mathbf{CLuN}} p\wedge\neg p$, inconsistencies seems to be independent of each other within **CLuN**. Consider for example the set $\Delta = \{\neg p, \neg(p \wedge p), \neg(p \wedge (p \wedge p)), \neg(p \wedge (p \wedge (p \wedge p))), \ldots\}$. Each member of Δ is **CL**-equivalent to $\neg p$. However, for every $\Delta' \subset \Delta$, there is a **CLuN**-model M such that $M \Vdash A$ for all $A \in \{p\} \cup \Delta'$ whereas $M \nVdash A$ for all $A \in \Delta - \Delta'$. And there is more. Let $\mathcal{W}_p$ comprise the formulas in which occurs no other non-logical term than p. For every $\Delta' \subseteq \Delta$, infinitely many members of $\mathcal{W}_p$ are verified by some **CLuN**-models of $\{p\} \cup \Delta'$ and falsified by other **CLuN**-models of $\{p\} \cup \Delta'$. Here are some examples of members of $\mathcal{W}_p$ for which this holds: $\neg B$ for any $B \in \Delta'$; $\neg((p \wedge p) \wedge p)$ and all similar results of commuting two different conjuncts in a conjunctive subformula of a $B \in \Delta'$; all $\neg(p \wedge B) \in \mathcal{W}_p - \Delta'$ such that the considered models verify B; and so on. Notwithstanding all this, all logical symbols are truth-functions in an infinite-valued **CLuN**-semantics. This is shown in Section 4.

If even the negation of **CLuN** is a truth-function in a many-valued semantics, one wonders whether being a truth-functional logic (with respect to some many-valued semantics) is a distinctive feature and, if it were distinctive, whether there is anything interesting about it. According to a truth-functional logic, the truth-value of every formula is fully determined by the truth-value of its atomic subformulas. This hardly means anything if it is realized by coding the binary valuation value of every complex formula into the many-valued valuation value of an atomic formula.

Independent of philosophical worries concerning truth-functionality, one may wonder whether many-valued logics may not be cast in a way that is different from the usual approach. I shall be looking for an unusual mould in Section 5.

4 The Case of the Logic CLuN

In this section it is shown that even **CLuN** is a many-valued logic on the usual many-valued approach. It seems instructive to describe the way in which this result was obtained, both in order to make the argument transparent and in order to illustrate the way in which the result may be extended to other logics.

The usual approach requires that all information required to fix the valuation

[15] Similarly for the valuation values of the three-valued **LP**-semantics [34] and for the valuation values of four-valued semantics for relevant logics [1].

[16] As was already pointed out by Viktor Kraft [31], nothing warrants that the syntactically atomic sentences of a language are also epistemologically atomic.

value of a formula A is contained in the valuation value of the atomic subformulas of A. It follows that infinitely many bits of information have to be contained in a single valuation value. Recall indeed that the following holds for the two-valued **CLuN**-semantics: if $v_M(p) = 1$, then $v_M(\neg p)$ may be 1 or 0, depending on the value of $v(\neg p)$; if $v_M(p) = v_M(\neg p) = 1$, then the same applies to $v_M(\neg\neg p)$; and so on. So the information contained in $v(p), v(\neg p), v(\neg\neg p), \ldots$ in a two-valued **CLuN**-model needs to be compressed in the valuation value of p within the corresponding many-valued **CLuN**-model—I shall write this valuation value as $V_M(p)$.[17] The situation is even more complex. If $v_M(p \wedge q) = 1$, then $v_M(\neg(p \wedge q))$ may be 1 as well as 0, depending on the value of $v(\neg(p \wedge q))$, and this information too must be contained in $V_M(p)$ or in $V_M(q)$. Note also that, in the considered case, $v(\neg(p \wedge q))$ and $v(\neg(p \wedge r))$ are not only independent of each other, but also of $v(\neg p)$, $v(\neg q)$, and $v(\neg r)$.

So, if the approach can be applied to **CLuN**, then the valuation value $V_M(A)$ contains all information contained in an infinite list $\langle v_M(A), v(B^1), v(B^2), \ldots\rangle$ in which $B^1, B^2, \ldots$ are formulas of which A is a subformula. In view of this, it seems fitting to identify $V_M(A)$ with an infinite sequence of zeros and ones. In order for the approach to be viable, several difficulties have to be resolved.

Consider $V_M(\neg(p \wedge q))$. This should be a truth-function of $V_M(p)$ and of $V_M(q)$. So, in terms of the two-valued semantics, the information on $v(\neg(p \wedge q))$ needs to be contained in $V_M(p)$ or $V_M(q)$. For every binary logical term $*$, I shall store the information on $v(\neg(A * B))$ in $V_M(A)$. This is obviously a conventional matter and there are several alternatives. The information on the two-valued assignment value $v(\neg\forall x Px)$ should also be contained within the many-valued valuation value of an atomic formula. I shall store it in $V_M(Pa)$. In view of these conventions I defined (at the end of Section 2), for every $A \in \mathcal{W}_\mathcal{O}$, the set $\mathrm{fsub}(A)$ of 'first subformulas' of A and the set $\mathrm{fsup}(A)$ of 'first superformulas' of A. For every $B \in \mathrm{fsup}(A)$, $v(B)$ (from the two-valued model) will be stored in $V_M(A)$ (from the many-valued model).[18]

Fact 1 *For all* $A \in \mathcal{W}_\mathcal{O}$*,* $\mathrm{fsub}(A)$ *is finite and decidable.*

Fact 2 *For all* $A \in \mathcal{W}_\mathcal{O}$*, there is exactly one* $B \in \mathrm{fsub}(A) \cap \mathcal{W}^a_\mathcal{O}$*.*

Fact 3 *If* $A, B \in \mathcal{W}^a_\mathcal{O}$ *are different, then* $\mathrm{fsup}(A) \cap \mathrm{fsup}(B) = \emptyset$*.*

Fact 4 *If* $A \in \mathrm{fsub}(B)$*, then* $\mathrm{fsup}(B) \subseteq \mathrm{fsup}(A)$*.*

Fact 5 *If* $A \notin \mathrm{fsub}(B)$ *and* $B \notin \mathrm{fsub}(A)$*, then* $\mathrm{fsup}(A) \cap \mathrm{fsup}(B) = \emptyset$*.*

So the present state of our plot is to identify a many-valued valuation value with $\langle v_M(A), v(B^1), v(B^2), \ldots\rangle$, where $\langle B_1, B_2, \ldots\rangle$ is an ordering of $\mathrm{fsup}(A)$. However, notwithstanding Fact 3, $\mathrm{fsup}(A)$ is uncountable if $\mathcal{L}_\mathcal{O}$ is uncountable and this is always the case if model M is uncountable. In that case, however,

[17] In the text I use the name M for both models although they are not only different but even different in kind. Where it matters, I shall obviously introduce different names.

[18] If B does not have the form $\neg C$, then $v(B)$ does not play any role within the **CLuN**-semantics. So one might just as well decide not to store the value of $v(B)$ in $V_M(A)$ for such $B \in \mathrm{fsup}(A)$. While the disadvantage of the approach followed in the text is that some digits of $V_M(A)$ are irrelevant, the advantage is that the approach is more general, as is the case for the assignment function of the two-valued semantics itself. That the advantage outweighs the disadvantage will show in Section 5.

there is no list $\langle v_M(A), v(B^1), v(B^2), \ldots\rangle$. Fortunately the Löwenheim-Skolem Theorem enables one to restrict a semantics to its countable models. So let us do this and consider only countable pseudo-language schemas $\mathcal{L}_\mathcal{O}$ in the sequel of the present section.[19]

Fact 6 *For all $A \in \mathcal{W}_\mathcal{O}$, $\mathrm{fsup}(A)$ is infinite, enumerable, and decidable.*

Consider a Gödel numbering and let $G(A)$ be the Gödel number of A. Let $L_A = \langle B_1, B_2, \ldots\rangle$ be such that (i) $\{B_1, B_2, \ldots\} = \mathrm{fsup}(A)$ and (ii) $i \leq j$ iff $G(B_i) \leq G(B_j)$. Where $B \in \mathrm{fsup}(A)$, let $\#_A(B)$ be the place of B in L_A—so if $L_A = \langle B_1, B_2, \ldots\rangle$, then $\#_A(B_i) = i$.

Fact 7 *For all $A \in \mathcal{W}_\mathcal{O}$, L_A is a recursive list.*

Fact 8 *If $B \in \mathrm{fsup}(A)$, then all members of L_B occur in the same order in L_A.*

Fact 9 *If $B \in \mathrm{fsup}(A)$, then there is a computable function f such that $L_B = f(L_A)$.*

Let $\mathfrak{S}$, the set of valuation values, be the set of infinite sequences of 0s and 1s; let $\mathfrak{s}$ be a variable for members of $\mathfrak{S}$; let $\mathfrak{S}_D = \{\langle d_0, d_1, \ldots\rangle \in \mathfrak{S} \mid d_0 = 1\}$ (the set of designated values). Where $V_M(A) = \langle d_0, d_1, \ldots\rangle$, define $V_M(A)[A] = d_0$ and define, for all $B \in \mathrm{fsup}(A)$, $V_M(A)[B] = d_{\#_A(B)}$.[20]

In a $\mathfrak{S}$-valued **CLuN**-model $M = \langle D, V\rangle$ (defined over the countable pseudo-language schema $\mathcal{L}_\mathcal{O}$), the domain D is a countable set and the assignment V has the following four properties. (i) $V\colon \mathcal{S} \to \mathfrak{S}$. (ii) $V\colon \mathcal{C} \cup \mathcal{O} \to D$ (where $D = \{V(\alpha) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$). (iii) Where $\mathfrak{S} = \{\mathfrak{s}^1, \mathfrak{s}^2, \ldots\}$, $V\colon \mathcal{P}^r \to \langle \Sigma_{\mathfrak{s}^1}, \Sigma_{\mathfrak{s}^2}, \ldots\rangle$ such that (a) $\{\Sigma_{\mathfrak{s}^1}, \Sigma_{\mathfrak{s}^2}, \ldots\}$ is a pseudo-partition of $\wp((\mathcal{C} \cup \mathcal{O})^r)$ and (b) if $V(\beta) = V(\alpha_i)$ $(1 \leq i \leq r)$, $\langle \alpha_1, \ldots, \alpha_{i-1}, \alpha_i, \alpha_{i+1}, \ldots, \alpha_r\rangle \in \Sigma_{\mathfrak{s}^j}$, and $\langle \alpha_1, \ldots, \alpha_{i-1}, \beta, \alpha_{i+1}, \ldots, \alpha_r\rangle \in \Sigma_{\mathfrak{s}^k}$, then $\mathfrak{s}^j, \mathfrak{s}^k \in \mathfrak{S}_D$ or $\mathfrak{s}^j, \mathfrak{s}^k \in \mathfrak{S} - \mathfrak{S}_D$. To simplify the notation, consider V as composed in this case of uncountably many functions $V^{\mathfrak{s}^1}$, $V^{\mathfrak{s}^2}$, …, with $V^{\mathfrak{s}^1}(\pi^r) = \Sigma_{\mathfrak{s}^1}$, $V^{\mathfrak{s}^2}(\pi^r) = \Sigma_{\mathfrak{s}^2}$, and so on. (iv) Identity is handled as a binary predicate with the special characteristic that $\bigcup\{V^{\mathfrak{s}}(=) \mid \mathfrak{s} \in \mathfrak{S}_D\} = \{\langle \alpha, \beta\rangle \mid \alpha, \beta \in \mathcal{C} \cup \mathcal{O}; V(\alpha) = V(\beta)\}$. The valuation function $V_M\colon \mathcal{W}_\mathcal{O} \to \mathfrak{S}$ is defined as follows:

C$\mathcal{S}$ where $A \in \mathcal{S}$, $V_M(A) = V(A)$

C$\mathcal{P}^r$ $V_M(\pi^r \alpha_1 \ldots \alpha_r) = \mathfrak{s}$ iff $\langle \alpha_1, \ldots, \alpha_r\rangle \in V^{\mathfrak{s}}(\pi^r)$

C$\neg$ Where $L_{\neg A} = \langle C^1, C^2, \ldots\rangle$, $V_M(\neg A) = \langle \max(1 - V_M(A)[A], V_M(A)[\neg A]), V_M(A)[C^1], V_M(A)[C^2], \ldots\rangle$.

C$\wedge$ Where $L_{A\wedge B} = \langle C^1, C^2, \ldots\rangle$, $V_M(A \wedge B) = \langle \min(V_M(A)[A], V_M(B)[B]), V_M(A)[C^1], V_M(A)[C^2)], \ldots\rangle$.

C$\vee$ Where $L_{A\vee B} = \langle C^1, C^2, \ldots\rangle$, $V_M(A \vee B) = \langle \max(V_M(A)[A], V_M(B)[B]), V_M(A)[C^1], V_M(A)[C^2)], \ldots\rangle$.

C$\supset$ Where $L_{A\supset B} = \langle C^1, C^2, \ldots\rangle$, $V_M(A \supset B) = \langle \max(1 - V_M(A)[A], V_M(B)[B]), V_M(A)[C^1], V_M(A)[C^2)], \ldots\rangle$.

[19] These are actually language schemas. Still $\mathcal{L}_\mathcal{O}$ need to be different from $\mathcal{L}$ in order to allow for models that are not ω-complete.

[20] So, if $L_A = \langle B_1, B_2, \ldots\rangle$ and $V_M(A) = \langle 1011 \ldots\rangle$, then $V_M(A)$ contains the information that in the corresponding two-valued model M' holds: $v_{M'}(A) = 1$, $v(B_1) = 0$, $v(B_2) = 1$, $v(B_3) = 1$, and so on.

C$\equiv$ Where $L_{A\equiv B} = \langle C^1, C^2, \ldots\rangle$, $V_M(A \equiv B) = \langle \min(\max(1 - V_M(A)[A], V_M(B)[B]), \max(1 - V_M(B)[B], V_M(A)[A])), V_M(A)[C^1], V_M(A)[C^2]], \ldots\rangle$.

C$\forall$ Where $L_{\forall\alpha A(\alpha)} = \langle C^1, C^2, \ldots\rangle$, $V_M(\forall\alpha A(\alpha)) = \langle \min\{V_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}, V_M(A(a))[C^1], V_M(A(a))[C^2], \ldots\rangle$.

C$\exists$ Where $L_{\exists\alpha A(\alpha)} = \langle C^1, C^2, \ldots\rangle$, $V_M(\exists\alpha A(\alpha)) = \langle \max\{V_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}, V_M(A(a))[C^1], V_M(A(a))[C^2], \ldots\rangle$.

Define $M \Vdash A$ (a $\mathfrak{S}$-valued **CLuN**-model M verifies A) iff $V_M(A) \in \mathfrak{S}_D$; and so on. To avoid confusion, let $\Gamma \vDash^{\mathfrak{S}}_{\mathbf{CLuN}} A$ denote that A is a **CLuN**-semantic consequence of Γ on the $\mathfrak{S}$-valued semantics.

Fact 10 *The $\mathfrak{S}$-valued* **CLuN***-semantics is recursive.*

Fact 11 *All logical symbols are truth-functions in the $\mathfrak{S}$-valued* **CLuN***-semantics.*

Lemma 1 *If $M = \langle D, v\rangle$ is a two-valued* **CLuN***-model, then there is a $\mathfrak{S}$-valued* **CLuN***-model $M' = \langle D, V\rangle$ such that, for all $A \in \mathcal{W}_{\mathcal{O}}$, $V_{M'}(A)[A] = v_M(A)$ and, where $L_A = \langle B_1, B_2, \ldots\rangle$, $V_{M'}(A)[B_i] = v(B_i)$ for all $i \in \{1, 2, \ldots\}$.*

Proof. From M construct M' as follows. For all $\alpha \in \mathcal{C} \cup \mathcal{O}$, $V(\alpha) = v(\alpha)$. If $A \in \mathcal{S}$ and $L_A = \langle B_1, B_2, \ldots\rangle$ then $V(A) = \langle v_M(A), v(B_1), v(B_2), \ldots\rangle$. For all $\alpha_1, \ldots, \alpha_r \in \mathcal{C} \cup \mathcal{O}$, $\langle \alpha_1, \ldots, \alpha_r\rangle \in V^{\mathfrak{s}}(\pi^r)$ iff, where $L_{\pi^r\alpha_1\ldots\alpha_r} = \langle B_1, B_2, \ldots\rangle$, $\mathfrak{s} = \langle v_M(\pi^r\alpha_1 \ldots \alpha_r), v(B_1), v(B_2), \ldots\rangle$.

From this one establishes the lemma for $A \in \mathcal{W}^a_{\mathcal{O}}$. With that as a basis, one establishes the lemma by an obvious induction. ■

Lemma 2 *If $M' = \langle D, V\rangle$ is a $\mathfrak{S}$-valued* **CLuN***-model, then there is a two-valued* **CLuN***-model $M = \langle D, v\rangle$ such that, for all $A \in \mathcal{W}_{\mathcal{O}}$, $v_M(A) = V_{M'}(A)[A]$ and, where $L_A = \langle B_1, B_2, \ldots\rangle$, $v(B_i) = V_{M'}(A)[B_i]$ for all $i \in \{1, 2, \ldots\}$.*

Proof. Obvious in view of the converse of the transformation described in the proof of Lemma 1. ■

Theorem 1 $\Gamma \vDash_{\mathbf{CLuN}} A$ *iff* $\Gamma \vDash^{\mathfrak{S}}_{\mathbf{CLuN}} A$.

Proof. $\Rightarrow$ Suppose that $\Gamma \nvDash^{\mathfrak{S}}_{\mathbf{CLuN}} A$. So there is a $\mathfrak{S}$-valued **CLuN**-model M that verifies all members of Γ and falsifies A. By the transformation described in the proof of Lemma 1, there is a two-valued **CLuN**-model M' that verifies exactly the same formulas as M.

$\Leftarrow$ Suppose that $\Gamma \nvDash_{\mathbf{CLuN}} A$. In view of the relevant Löwenheim-Skolem Theorem, a countable two-valued **CLuN**-model M verifies all members of Γ and falsifies A. By the transformation described in the proof of Lemma 2, there is a $\mathfrak{S}$-valued **CLuN**-model M' that verifies exactly the same formulas as M. ■

Let $\mathfrak{T}$ be the set of all finite sequences of 0s and 1s. The logic **CLuN** is compact. This enables us to replace the $\mathfrak{S}$-valued semantics by a semantics that takes its values from $\mathfrak{T}$.

As a first step we represent some infinite sequences of 0s and 1s by finite sequences. This is easy: remove trailing ones from the sequences, reducing for example the infinite sequence $\langle 110100011\ldots\rangle$, in which the ellipsis represents ones only, by the finite sequence $\langle 1101000\rangle$. Of course the first member of a sequence is always retained. So the finite sequences are $\langle 0\rangle, \langle 1\rangle, \langle 00\rangle, \langle 10\rangle, \langle 000\rangle, \langle 010\rangle, \langle 100\rangle,$

$\langle 110\rangle, \ldots$, all but one of which end with a 0. The remaining infinite sequences are those in which there is no last 0. We simply remove them.

The set $\mathfrak{T}$ of valuation values is the set of sequences $\langle d_0, \ldots, d_n\rangle$ such that (i) $n \geq 0$, (ii) $d_0, \ldots, d_n \in \{0, 1\}$, and (iii) $d_n = 0$ if $n > 0$. Let $\mathfrak{t}$ be a variable for members of $\mathfrak{T}$. Let $\mathfrak{T}_D = \{\langle d_0, \ldots, d_n\rangle \in \mathfrak{T} \mid d_0 = 1\}$ (the set of designated values). Where $V_M(A) = \langle d_0, \ldots, d_n\rangle$, define $V_M(A)[A] = d_0$ and define, for all $B \in \text{fsup}(A)$, $V_M(A)[B] = d_{\#_A(B)}$ if $\#_A(B) \leq n$ and $V_M(A)[B] = 1$ otherwise.[21] Where $\langle d_0, \ldots, d_n\rangle$ is a sequence of 0s and 1s, let $\langle d_0, \ldots, d_n \rhd\rangle$ be the result of removing trailing 1s as long as the sequence counts more than one digit. So $\langle d_0, \ldots, d_n \rhd\rangle \in \mathfrak{T}$.

Fact 12 *$\mathfrak{T}$ is denumerable (infinite and enumerable) and decidable.*

After these preliminaries, let us turn to the semantics. In a $\mathfrak{T}$-valued **CLuN**-model $M = \langle D, V\rangle$ (defined over the countable pseudo-language schema $\mathcal{L}_\mathcal{O}$), the domain D is a countable set and the assignment V has the following four properties. (i) $V\colon \mathcal{S} \to \mathfrak{T}$. (ii) $V\colon \mathcal{C} \cup \mathcal{O} \to D$ (where $D = \{V(\alpha) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$). (iii) Where $\mathfrak{T} = \{\mathfrak{t}^1, \mathfrak{t}^2, \ldots\}$, $V\colon \mathcal{P}^r \to \langle \Sigma_{\mathfrak{t}^1}, \Sigma_{\mathfrak{t}^2}, \ldots\rangle$ such that (a) $\{\Sigma_{\mathfrak{t}^1}, \Sigma_{\mathfrak{t}^2}, \ldots\}$ is a pseudo-partition of $\wp(D^r)$ and (b) if $V(\beta) = V(\alpha_i)$ $(1 \leq i \leq r)$, $\langle \alpha_1, \ldots, \alpha_{i-1}, \alpha_i, \alpha_{i+1}, \ldots, \alpha_r\rangle \in \Sigma_{\mathfrak{t}^j}$, and $\langle \alpha_1, \ldots, \alpha_{i-1}, \beta, \alpha_{i+1}, \ldots, \alpha_r\rangle \in \Sigma_{\mathfrak{t}^k}$, then $\mathfrak{t}^j, \mathfrak{t}^k \in \mathfrak{T}_D$ or $\mathfrak{t}^j, \mathfrak{t}^k \in \mathfrak{T} - \mathfrak{T}_D$. To simplify the notation, consider V as composed in this case of the denumerably many functions $V^{\mathfrak{t}^1}, V^{\mathfrak{t}^2}, \ldots$, with $V^{\mathfrak{t}^1}(\pi^r) = \Sigma_{\mathfrak{t}^1}$, $V^{\mathfrak{t}^2}(\pi^r) = \Sigma_{\mathfrak{t}^2}$, and so on. (iv) Identity is handled as a binary predicate with the special characteristic that $\bigcup\{V^{\mathfrak{t}}(=) \mid \mathfrak{t} \in \mathfrak{T}_D\} = \{\langle \alpha, \beta\rangle \mid \alpha, \beta \in \mathcal{C} \cup \mathcal{O}; V(\alpha) = V(\beta)\}$.

In the clauses below, let $V_M(A)$ count n_A members and let $V_M(A(a))$ count $n_{A(a)}$ members. The valuation function $V_M\colon \mathcal{W}_\mathcal{O} \to \mathfrak{T}$ is defined as follows:

C$\mathcal{S}$ where $A \in \mathcal{S}$, $V_M(A) = V(A)$

C$\mathcal{P}^r$ $V_M(\pi^r\alpha_1 \ldots \alpha_r) = \mathfrak{t}$ iff $\langle \alpha_1, \ldots, \alpha_r\rangle \in V^{\mathfrak{t}}(\pi^r)$

C$\neg$ Where $L_{\neg A} = \langle C^1, C^2, \ldots\rangle$, $V_M(\neg A) = \langle \max(1 - V_M(A)[A], V_M(A)[\neg A]), V_M(A)[C^1], \ldots, V_M(A)[C^{n_A}] \rhd\rangle$.

C$\wedge$ Where $L_{A\wedge B} = \langle C^1, C^2, \ldots\rangle$, $V_M(A \wedge B) = \langle \min(V_M(A)[A], V_M(B)[B]), V_M(A)[C^1], \ldots, V_M(A)[C^{n_A})] \rhd\rangle$.

C$\vee$ Where $L_{A\vee B} = \langle C^1, C^2, \ldots\rangle$, $V_M(A \vee B) = \langle \max(V_M(A)[A], V_M(B)[B]), V_M(A)[C^1], \ldots, V_M(A)[C^{n_A})] \rhd\rangle$.

C$\supset$ Where $L_{A\supset B} = \langle C^1, C^2, \ldots\rangle$, $V_M(A \supset B) = \langle \max(1 - V_M(A)[A], V_M(B)[B]), V_M(A)[C^1], \ldots, V_M(A)[C^{n_A})] \rhd\rangle$.

C$\equiv$ Where $L_{A\equiv B} = \langle C^1, C^2, \ldots\rangle$, $V_M(A \equiv B) = \langle \min(\max(1 - V_M(A)[A], V_M(B)[B]), \max(1 - V_M(B)[B], V_M(A)[A])), V_M(A)[C^1], \ldots, V_M(A)[C^{n_A})] \rhd\rangle$.

C$\forall$ Where $L_{\forall\alpha A(\alpha)} = \langle C^1, C^2, \ldots\rangle$, $V_M(\forall\alpha A(\alpha)) = \langle \min\{V_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}, V_M(A(a))[C^1], \ldots, V_M(A(a))[C^{n_{A(a)}}] \rhd\rangle$.

C$\exists$ Where $L_{\exists\alpha A(\alpha)} = \langle C^1, C^2, \ldots\rangle$, $V_M(\exists\alpha A(\alpha)) = \langle \max\{V_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}, V_M(A(a))[C^1], \ldots, V_M(A(a))[C^{n_{A(a)}}] \rhd\rangle$.

[21] So, if $L_A = \langle B_1, B_2, \ldots\rangle$ and $V_M(A) = \langle 1110\rangle$, then $V_M(A)$ contains the information that in the corresponding two-valued model M' holds: $v_{M'}(A) = 1$, $v(B_1) = 1$, $v(B_2) = 1$, $v(B_3) = 0$, and $v(B_i) = 1$ whenever $i > 3$; similarly, $V_M(A) = \langle 0\rangle$ then contains the information that in the corresponding two-valued model M' holds: $v_{M'}(A) = 0$ and $v(B_i) = 1$ for all $i \in \{1, 2, \ldots\}$.

A $M \Vdash A$ (a $\mathfrak{T}$-valued **CLuN**-model M verifies A) iff $V_M(A) \in \mathfrak{T}_D$; M is a model of $\{A_1, \ldots, A_n\}$ iff $M \Vdash A_1$, ..., and $M \Vdash A_n$. $B_1, \ldots, B_n \vDash^{\mathfrak{T}}_{\mathbf{CLuN}} A$ iff every $\mathfrak{T}$-valued **CLuN**-model of $\{B_1, \ldots, B_n\}$ verifies A; $\Gamma \vDash^{\mathfrak{T}}_{\mathbf{CLuN}} A$ iff $B_1, \ldots, B_n \vDash^{\mathfrak{T}}_{\mathbf{CLuN}} A$ for some $B_1, \ldots, B_n \in \Gamma$.

The only 'inconvenience' with this semantics is that, for some Γ and A, $\Gamma \nvDash^{\mathfrak{T}}_{\mathbf{CLuN}} A$ while no $\mathfrak{T}$-valued **CLuN**-model of Γ falsifies A. An example is $\{\neg^{3n}p, \neg^{3n+1}p, \neg^{3n+2}p \supset q \mid n \in \mathbb{N}\} \nvDash^{\mathfrak{T}}_{\mathbf{CLuN}} q$ in which $\neg^i$ abbreviates a sequence of i occurrences of $\neg$. Every two-valued **CLuN**-model M of $\{\neg^{3n}p, \neg^{3n+1}p, \neg^{3n+2}p \supset q \mid n \in \mathbb{N}\}$ that falsifies q has, for all $n \in \mathbb{N}$, $v(\neg^{3n+1}p) = 1$ and $v(\neg^{3n+2}p) = 0$. There is a $\mathfrak{S}$-valued **CLuN**-model that corresponds to M but obviously not a $\mathfrak{T}$-valued one. Indeed, as $V_{M'}(p) = V(p)$ counts finitely many digits, say m, there is bound to be a n such that $\#_p(\neg^{3n+2}p) > m$, whence $V_{M'}(\neg^{3n+2}p) \in \mathfrak{T}_D$; but then $V_{M'}(\neg^{3n+2}p \supset q) \notin \mathfrak{T}_D$ or $V_{M'}(q) \in \mathfrak{T}_D$; so if M' is a model of $\{\neg^{3n}p, \neg^{3n+1}p, \neg^{3n+2}p \supset q \mid n \in \mathbb{N}\}$, it falsifies q. Of course the inconvenience has no effect on the semantic consequence relation in view of the special way in which it is defined.

Fact 13 *The $\mathfrak{T}$-valued* **CLuN***-semantics is recursive.*

Fact 14 *All logical symbols are truth-functions in the $\mathfrak{T}$-valued* **CLuN***-semantics.*

Lemma 3 *If $M = \langle D, v\rangle$ is a two-valued* **CLuN***-model, $M \Vdash A$, $M \nVdash B$, and $M' = \langle D, v'\rangle$ is obtained from M by letting $v(\neg C) = 1$ whenever $\neg C \notin \{A, B\} \cup \mathrm{sub}(A) \cup \mathrm{sub}(B)$, then M' is a two-valued* **CLuN***-model, $M' \Vdash A$ and $M' \nVdash B$.*

Proof. By an obvious induction on the length of A or of B, whichever is longer. ■

Lemma 4 *If $M = \langle D, v\rangle$ is a two-valued* **CLuN***-model, and, for every $A \in \mathcal{W}^a_{\mathcal{O}}$, $v(B) = 0$ for at most finitely many $B \in \mathrm{fsup}(A)$, then there is a $\mathfrak{T}$-valued* **CLuN***-model $M' = \langle D, V'\rangle$ such that, for all $A \in \mathcal{W}_{\mathcal{O}}$, $V_{M'}(A)[A] = v_M(A)$ and, where $L_A = \langle B_1, B_2, \ldots\rangle$, $V_{M'}(A)[B_i] = v(B_i)$ for all $i \in \{1, 2, \ldots\}$.*

Proof. Suppose that the antecedent is true. In view of Lemma 1, there is a $\mathfrak{S}$-valued **CLuN**-model $M'' = \langle D, V''\rangle$ with the required property. The $\mathfrak{S}$-valued **CLuN**-model M'' is transformed to the required $\mathfrak{T}$-valued **CLuN**-model $M' = \langle D, V'\rangle$ by the following steps. Consider an $A \in \mathcal{S}$ and let $V_{M''}(A) = \mathfrak{s}$. As, for every $A \in \mathcal{W}^a_{\mathcal{O}}$, $v(B) = 0$ for at most finitely many $B \in \mathrm{fsup}(A)$, there is bound to be a last 0 in the sequence $\mathfrak{s}$. So the result of removing trailing 1s from $\mathfrak{s}$ results in a member of $\mathfrak{T}$. The reasoning for formulas $\pi^r\alpha_1 \ldots \alpha_r$ proceeds similarly. Next one invokes an obvious induction on the length of $A \in \mathcal{W}_{\mathcal{O}}$ as in Lemma 1. ■

Lemma 5 *If $M' = \langle D, V\rangle$ is a $\mathfrak{T}$-valued* **CLuN***-model, then there is a two-valued* **CLuN***-model $M = \langle D, v\rangle$ such that, for all $A \in \mathcal{W}_{\mathcal{O}}$, $v_M(A) = V_{M'}(A)[A]$ and, where $L_A = \langle B_1, B_2, \ldots\rangle$, $v(B_i) = V_{M'}(A)[B_i]$ for all $i \in \{1, 2, \ldots\}$.*

Proof. Obvious. ■

Lemma 6 *If M is a two-valued* **CLuN***-model, M' is a $\mathfrak{T}$-valued* **CLuN***-model, and M and M' correspond in the sense of Lemmas 4 and 5, then $M \Vdash A$ iff $M' \Vdash A$ for all $A \in \mathcal{W}_{\mathcal{O}}$.*

Proof. A proof by cases gives one the result for $A \in \mathcal{W}^a_{\mathcal{O}}$. This provides the basis for the obvious induction on the complexity of $A \in \mathcal{W}_{\mathcal{O}}$. ■

Theorem 2 $\Gamma \vDash^{\mathfrak{T}}_{\mathbf{CLuN}} A$ *iff* $\Gamma \vDash_{\mathbf{CLuN}} A$.

Proof. $\Rightarrow$ Suppose that $\Gamma \nvDash_{\mathbf{CLuN}} A$. Consider any $B_1, \ldots, B_n \in \Gamma$. As **CLuN** is compact, $B_1, \ldots, B_n \nvDash_{\mathbf{CLuN}} A$. Note that $B_1 \wedge \ldots \wedge B_n \nvDash_{\mathbf{CLuN}} A$. In view of the relevant Löwenheim-Skolem Theorem and of Lemma 3, there is a countable binary **CLuN**-model $M = \langle D, v\rangle$ such that $M \Vdash B_1 \wedge \ldots \wedge B_n$, and $M \nVdash A$ and $v(C) = 0$ for at most finitely many formulas $C \notin \mathcal{W}^a_{\mathcal{O}}$. In view of Lemmas 4 and 6, it follows that there is a $\mathfrak{T}$-valued **CLuN**-model M' such that $M' \Vdash B_1 \wedge \ldots \wedge B_n$ and $M' \nVdash A$.

$\Leftarrow$ Suppose that $\Gamma \nvDash^{\mathfrak{T}}_{\mathbf{CLuN}} A$. The definition of $\vDash^{\mathfrak{T}}_{\mathbf{CLuN}}$ entails, for arbitrary $B_1, \ldots, B_n \in \Gamma$, that $B_1 \wedge \ldots \wedge B_n \nvDash^{\mathfrak{T}}_{\mathbf{CLuN}} A$. So a $\mathfrak{T}$-valued **CLuN**-model verifies $B_1 \wedge \ldots \wedge B_n$ and falsifies A. By Lemmas 5 and 6, there is a two-valued **CLuN**-model M' such that $M' \Vdash B_1 \wedge \ldots \wedge B_n$ and $M' \nVdash A$. ■

The members of $\mathfrak{T}$ are obtained from members of $\mathfrak{S}$ by removing trailing 1s. The reader may find it more convenient to consider finite sequences of 0s and 1s that are obtained by removing trailing 0s. Another alternative is finite sequences from which trailing 1s are removed except for the first one as well as finite sequences from which trailing 0s are removed except for the first one. In both cases the result corresponds to the one presented above and the inconvenience is exactly the same.

5 A Different Many-Valued Approach

Take another look at the two-valued **CLuNs**-semantics from Section 2, comparing it to the three-valued **CLuNs**-semantics from Section 3. That a two-valued model verifies $p \wedge \neg p$ is the result of $v(p) = 1$ and $v(\neg p) = 1$ and these are separate and independent 'facts'. If a two-valued model verifies $A \wedge \neg A$ for complex A, then again $M \Vdash A$ depends on one set of 'facts' whereas $M \Vdash \neg A$ depends on a separate set of 'facts'. The idea behind the three-valued semantics is completely different. That a three-valued model verifies $p \wedge \neg p$ depends on the sole 'fact' that $V(p) = I$. Moreover, if $M \Vdash A \wedge \neg A$ for complex A, then, as a little inspection readily reveals, there are always inconsistent 'facts' on which both $M \Vdash A$ and $M \Vdash \neg A$ depend.

I now set out to construct a very different kind of many-valued semantics, called tuaf semantics. For a start, the assignment of the tuaf semantics will be the same as the assignment of the two-valued semantics. I shall retain the convention that valuation values are determined starting from the least complex formulas. If $v(p) = 1$, then p will obtain the valuation value t (for *true*), independent of the valuation value of $\neg p$. If $v_M(p) = t$ and $v(\neg p) = 0$, then $v_M(\neg p) = f$ (for *false*). However, if $v_M(p) = t$ and $v(\neg p) = 1$, then $v_M(\neg p) = u$ (for *glut*). Similarly, if $v_M(p) = f$ and $v(\neg p) = 0$, then $v_M(\neg p) = a$ (for *gap*).[22] The idea is that a formula receives the valuation value u, respectively a, iff it has the wrong truth-value with respect to its subformulas. This idea is clearly different from the usual one, described in Section 3. However, as we

[22]Some values will be absent for some logics; **CLuNs**, for example, does not allow for gaps.

shall soon see, some choices have still to be made in order to turn the approach into something workable.

By way of preparation, we start with an alternative formulation of the **CL**-semantics. The semantics from Section 1 will be called the *clausal semantics*. Let us turn it into a *tabular semantics* by leaving the assignment function unchanged, replacing the ten clauses specifying the valuation function by the following ten tables—the last two are amalgamated.

Where $A \in \mathcal{S}$:

$v(A)$	A
1	1
0	0

Where $\alpha_1, \ldots, \alpha_n \in \mathcal{C} \cup \mathcal{O}$ and $\pi \in \mathcal{P}^n$:

$\langle v(\alpha_1), \ldots, v(\alpha_n)\rangle, v(\pi)$	$\pi\alpha_1 \ldots \alpha_n$
$\in$	1
$\notin$	0

Where $\alpha, \beta \in \mathcal{C} \cup \mathcal{O}$:

$v(\alpha), v(\beta)$	$\alpha = \beta$
$=$	1
$\neq$	0

	$\neg$
1	0
0	1

$\wedge$	1	0
1	1	0
0	0	0

$\vee$	1	0
1	1	1
0	1	0

$\supset$	1	0
1	1	0
0	1	1

$\equiv$	1	0
1	1	0
0	0	1

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\{1\}$	1	1
$\{0, 1\}$	0	1
$\{0\}$	0	0

We have seen that some logics display gluts or gaps or both. For them, I shall articulate a semantics in which the valuation has the form $v_M : \mathcal{W}_\mathcal{O} \rightarrow \{t, u, a, f\}$, in which the values intuitively stand for true, glut, gap, and false—hence the name *tuaf semantics*. The designated values are t and u. So $M \Vdash A$ iff $v_M(A) \in \{t, u\}$. This settles at once the semantic consequence relation $\Gamma \vDash A$.

Some logics do not allow for gluts or for gaps. So only two or three of the values will be used in their tuaf semantics. For example, the tuaf semantics for **CL** is boringly isomorphic to **CL**'s tabular semantics: for the valuation, every 1 is replaced by t and every 0 by f—the point of the replacement will soon become clear. I spell out this semantics for future reference.[23]

Where $A \in \mathcal{S}$:

$v(A)$	A
1	t
0	f

Where $\alpha_1, \ldots, \alpha_n \in \mathcal{C} \cup \mathcal{O}$ and $\pi \in \mathcal{P}^n$:

$\langle v(\alpha_1), \ldots, v(\alpha_n)\rangle, v(\pi)$	$\pi\alpha_1 \ldots \alpha_n$
$\in$	t
$\notin$	f

Where $\alpha, \beta \in \mathcal{C} \cup \mathcal{O}$:

$v(\alpha), v(\beta)$	$\alpha = \beta$
$=$	t
$\neq$	f

[23] I use the same notation, $v_M(A)$, for the valuation function in all three kinds of semantics and I shall do so for all logics. The matter is always disambiguated by the context.

These three first tables, which concern the atomic formulas, are identical for all subsequent logics. They will not be repeated.

	$\neg$
t	f
f	t

$\wedge$	t	f
t	t	f
f	f	f

$\vee$	t	f
t	t	t
f	t	f

$\supset$	t	f
t	t	f
f	t	t

$\equiv$	t	f
t	t	f
f	f	t

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\{t\}$	t	t
$\{f, t\}$	f	t
$\{f\}$	f	f

The matter gets interesting when we move to logics that tolerate gluts or gaps. Let us start with **CLuN**. Its tabular semantics is identical to that for **CL**, except that the table for negation is replaced.

A	$v(\neg A)$	$\neg A$
1	0	0
1	1	1
0	(any)	1

The table describes $v_M(\neg A)$ as a function of $v_M(A)$ and of $v(\neg A)$. The "(any)" indicates that the value of $v(\neg A)$ has no effect at this point, viz. where $v_M(A) = 0$.

Let us turn to the tuaf semantics of **CLuN**. Its assignment is as for all two-valued semantics in this paper and the three valuation tables for atomic formulas is as for **CL**. The rest of the valuation function is determined by the following tables—some explanation follows.

A	$v(\neg A)$	$\neg A$
t	0	f
t	1	u
u	0	f
u	1	u
f	(any)	t

$\wedge$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	f

$\vee$	t	u	f
t	t	t	t
u	t	t	t
f	t	t	f

$\supset$	t	u	f
t	t	t	f
u	t	t	f
f	t	t	t

$\equiv$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	t

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\subseteq \{t, u\}$	t	t
$= \{f\}$	f	f
(other)	f	t

As the value u is *introduced* by the table for negation—if $v_M(A) \in \{t, u\}$ and $v(\neg A) = 1$, then $v_M(\neg A) = u$—the value u has to occur in all tables in which the input entries are valuation values. The "(any)" has the same meaning as in the tabular semantics. The "(other)" obviously means that the set $\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$ contains at least one f and at least one t or u.

Until now, the design of the tuaf semantics proceeded on somewhat insecure grounds. There apparently is a clear design behind it, but the design is not made fully explicit. So let us see where precisely the tuaf semantics assigns the value u? This question may be answered in several ways and, depending on the answer, the tuaf semantics of other logics will vary. That the question may be answered in at least two different ways is caused by the fact that **CLuN** has the following remarkable property: for any formula A, an adequate **CLuN**-semantics has models M and M' such that (i) $M \Vdash A$ and $M \Vdash \neg A$, (ii) $M' \Vdash A$ and $M' \nVdash \neg A$, and (iii) M and M' verify exactly the same subformulas of A. So for no true A, however complex, does the truth of its negation result from its proper subformulas. **CLuN** has also a different, actually more general, property. Consider a complexity function that assigns to a formula A a complexity $c(A)$ that is higher than the complexity it assigns to any proper subformula of A. For any formula A, there are two-valued **CLuN**-models M and M' such that the aforementioned (i) and (ii) obtain whereas M and M' verify exactly the same subset of $\{B \mid c(B) < c(A)\}$.

A first view on the tuaf semantics of **CLuN** may be called the *agreement view*. In the *tabular* semantics for **CL**, every table defines, for a non-atomic form A, $v_M(\mathsf{A})$ as a function of the valuation value of subformulas of A. So it is easy to check whether the valuation function of **CLuN**, or of any other logic **L** allowing for gluts or gaps, agrees with **CL** *at a specific point*. If a two-valued **L**-model M has $v_M(A) = v_M(B) = v_M(A \wedge B) = 1$, then the tuaf **L**-model M' has $v_{M'}(A \wedge B) = t$ because $v_M(A \wedge B) = 1$ agrees with all **CL**-models M that have $v_M(A) = v_M(B) = 1$. If a two-valued **L**-model M has $v_M(A) = 0$ and $v_M(A \wedge B) = 1$, then the tuaf **L**-model M' has $v_{M'}(A \wedge B) = u$ because **CL**-models have $v_M(A \wedge B) = 0$ whenever they have $v_M(A) = 0$. By a similar reasoning, if the **L**-model M has $v_M(A) = v_M(B) = 1$ and $v_M(A \wedge B) = 0$, then the tuaf **L**-model M' has $v_{M'}(A \wedge B) = a$. It is left to the reader to check that the occurrence of output entries u in the tuaf semantics of **CLuN** is in line with the agreement view.

Next there is what I shall call the *interference view*. The two output entries u in the tuaf semantics of **CLuN** depend on the assignment. For both $v(\neg A) = 1$. If this is modified to $v(\neg A) = 0$, the value of $v_M(\neg A)$ is modified to f. So, on the interference view, a complex formula A obtains the valuation value u, respectively a, iff the two-valued valuation value depends on the assignment value $v(A)$, and not only on valuation values of subformulas of A. So this view incorporates the agreement view and moreover takes into account whether the valuation value of a *specific* complex formula is a function of the valuation values of its subformulas. Put differently, the values u and a are assigned at points where gluts or gaps *originate*.

An example clarifies this even further. In the two-valued **CLuNs**-semantics, $v_M(\neg\neg A) = v_M(A)$. Let $v_M(p) = 1 = v(\neg p)$ in a two-valued **CLuNs**-model M, whence $v_M(\neg p) = v_M(\neg\neg p) = 1$. So the corresponding tuaf model M has $v_M(\neg p) = u$ because if $v(\neg p)$ were 0, then $v_M(\neg p)$ would be 0 in the two-valued semantics. However, $v_M(\neg\neg p) = t$ on the interference view. Indeed, although $v_M(\neg p) = 1$ and $v_M(\neg\neg p) = 1$ in the two-valued semantics, the latter value does not depend on $v(\neg\neg p)$ but is a direct result of $v_M(p) = 1$.[24] Note that the

[24]The counterfactual and causal phraseology can obviously be rephrased extensionally (in terms of all models that have certain properties).

agreement view leads to a different result at this point for **CLuNs**; it leads to $v_M(\neg\neg p) = u$ in the considered example.

It is again left to the reader to check that the occurrence of output entries u in the tuaf semantics of **CLuN** is in line with the interference view. Both views explain the absence of output entries u outside the negation table. They also clarify in general why $\neg A$ may have the valuation value u whereas A will never have that valuation value, unless of course in case A itself has the form $\neg B$. In the sequel of this paper, I shall restrict attention to the interference view.

The tabular semantics as well as the tuaf semantics of **CLuNs** require the equivalence classes defined in Section 2. The tabular semantics is just like that for **CL**, except that the table for negation is replaced by the following tables.

Where $A \in \mathcal{W}^a_{\mathcal{O}}$:

A	$\{v(\neg B) \mid B \in [\![A]\!]\}$	$\neg A$
1	$= \{0\}$	0
1	$\neq \{0\}$	1
0	(any)	1

A	$\neg\neg A$
$A \wedge \neg B$	$\neg(A \supset B)$
$\neg A \vee \neg B$	$\neg(A \wedge B)$
$\neg A \wedge \neg B$	$\neg(A \vee B)$
$(A \vee B) \wedge (\neg A \vee \neg B)$	$\neg(A \equiv B)$
$\exists\alpha\neg A(\alpha)$	$\neg\forall\alpha A(\alpha)$
$\forall\alpha\neg A(\alpha)$	$\neg\exists\alpha\neg A(\alpha)$
1	1
0	0

The lower table is obviously a summary of seven tables, each stating that the formula in the right column has the same value as the formula in the left column. Of course, the fascinating bit is the tuaf semantics. Again, the assignment is as for **CL** and so are the three tables for the atomic formulas.

Where $A \in \mathcal{W}^a_{\mathcal{O}}$:

A	$\{v(\neg B) \mid B \in [\![A]\!]\}$	$\neg A$
t	$= \{0\}$	f
t	$\neq \{0\}$	u
f	(any)	t

A	$\neg\neg A$
$A \wedge \neg B$	$\neg(A \supset B)$
$\neg A \vee \neg B$	$\neg(A \wedge B)$
$\neg A \wedge \neg B$	$\neg(A \vee B)$
$(A \vee B) \wedge (\neg A \vee \neg B)$	$\neg(A \equiv B)$
$\exists\alpha\neg A(\alpha)$	$\neg\forall\alpha A(\alpha)$
$\forall\alpha\neg A(\alpha)$	$\neg\exists\alpha\neg A(\alpha)$
t	t
u	t
f	f

$\wedge$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	f

$\vee$	t	u	f
t	t	t	t
u	t	t	t
f	t	t	f

$\supset$	t	u	f
t	t	t	f
u	t	t	f
f	t	t	t

$\equiv$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	t

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\subseteq \{t, u\}$	t	t
$= \{f\}$	f	f
(other)	f	t

There is only one output entry u in all these tables. Atomic formulas never receive the value u. Formulas of which the central symbol is not a negation cannot receive a u because their valuation value (does not depend on their assignment value but) is fully determined by the valuation value of less complex formulas. In the table for the negation of complex formulas, there is a u among the input entries. Please note that the only formulas in the left column that may have a valuation value u are those of the form A that moreover have the form $\neg C$. But even if this formula has the value u, the formula of the corresponding form $\neg\neg A$ needs the value t because the assignment does not interfere. Indeed, it holds within the two-valued **CLuNs**-semantics that $v_M(\neg\neg A) = 1$ if $v_M(A) = 1$, whatever $v(\neg\neg A)$.

I mentioned before that the tuaf semantics introduces values u and a where the gluts or gaps *originate*. Please check this. If $v_M(Pa) = v_M(\neg Pa) = 1$ in the clausal or tabular **CLuNs**-semantics, the tuaf semantics settles for $v_M(\neg Pa) = u$. It holds within the two-valued **CLuNs**-semantics that if $v_M(Pa) = v_M(\neg Pa) = v_M(Qb) = 1$, then $v_M(Pa \wedge Qb) = v_M(\neg Pa \vee \neg Qb) = 1$, and hence also $v_M(Pa \wedge Qb) = v_M(\neg(Pa \wedge Qb)) = 1$. The tuaf semantics settles for $v_M(\neg(Pa \wedge Qb)) = t$. This is precisely as we want it: the glut does not originate with $\neg(Pa \wedge Qb)$; it originates with $\neg Pa$.

Consider the tuaf semantics of a very weak extension of **CLuN**, viz. with $A \supset \neg\neg A$, for which I shall use the rather arbitrary name $\mathbf{CLuN}^{NN}$. The tuaf semantics is identical to that of **CLuN**, except for the tables for negation—there are two of them. Let $\mathcal{W}^n_{\mathcal{O}}$ be the set of formulas that do not have $\neg$ as their first symbol.

Where $A \in \mathcal{W}^n_{\mathcal{O}}$:

A	$v(\neg A)$	$\neg A$
t	0	f
t	1	u
f	(any)	t

A	$v(\neg\neg A)$	$\neg\neg A$
t	(any)	t
u	(any)	t
f	0	f
f	1	u

There are no input entries u in the first table because only formulas of the form $\neg B$ can have the value u. The output value of the second line of the second table is t because the assignment does not interfere. If $v_M(A) = f$, then $v_M(\neg A) = t$, whence $v_M(\neg\neg A) \in \{u, f\}$.

There are two output entries u in this semantics. So negation gluts originate at two kinds of points in $\mathbf{CLuN}^{NN}$-models, first where the negation of a non-negated verified formula is itself verified and next where the double negation of a falsified formula is verified—the negation of the formula is then verified and, by the interference of the assignment, also its double negation.

Let us, as a non-paraconsistent illustration, consider the tuaf semantics for the logic with the beautiful name **CLuCoDaM**, in words, the logic that leaves room for conjunction gluts, for both disjunction gluts and disjunction gaps, and for implication gaps. Let us consider the version in which Replacement of Identicals is not added. So we do not need the equivalence classes from the **CLuNs**-semantics. Moreover, I skip the tabular semantics. The reader may very easily construct it in case the tuaf semantics would not be obvious at once. The assignment and the valuation tables for atomic formulas are as for **CL**.

	$\neg$
t	f
u	f
a	t
f	t

$\equiv$	t	u	a	f
t	t	t	f	f
u	t	t	f	f
a	f	f	t	t
f	f	f	t	t

$v(A \wedge B) = 1:$

$\wedge$	t	u	a	f
t	t	t	u	u
u	t	t	u	u
a	u	u	u	u
f	u	u	u	u

$v(A \vee B) = 1:$

$\vee$	t	u	a	f
t	t	t	t	t
u	t	t	t	t
a	t	t	u	u
f	t	t	u	u

$v(A \supset B) = 1:$

$\supset$	t	u	a	f
t	t	t	f	f
u	t	t	f	f
a	t	t	t	t
f	t	t	t	t

$v(A \wedge B) = 0:$

$\wedge$	t	u	a	f
t	t	t	f	f
u	t	t	f	f
a	f	f	f	f
f	f	f	f	f

$v(A \vee B) = 0:$

$\vee$	t	u	a	f
t	a	a	a	a
u	a	a	a	a
a	a	a	f	f
f	a	a	f	f

$v(A \supset B) = 0:$

$\supset$	t	u	a	f
t	a	a	f	f
u	a	a	f	f
a	a	a	a	a
f	a	a	a	a

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\subseteq \{t, u\}$	t	t
$\subseteq \{a, f\}$	f	f
(other)	f	t

This semantics illustrates a variety of cases. As there are no gluts or gaps with respect to negation, equivalence, and the quantifiers, the output entries are all t and f in the tables for those logical symbols. For conjunction there are only gluts. So if $v(A \wedge B) = 0$, one obtains the normal table; if $v(A \wedge B) = 1$ every f in the normal table is replaced by a u. Implication and disjunction illustrate the other cases.

The tuaf semantics of da Costa's $\mathbf{C}_n$ logics [23] illustrates a possible complication. Let $A^{(1)}$ abbreviate $\neg(C \wedge \neg C)$ and let $A \equiv^c B$ denote that A and B are congruent in the sense of Kleene *or* that one formula results from the other by deleting vacuous quantifiers—Kleene [30, p. 153] summarizes his definition as follows: "two formulas are congruent, if they differ only in their bound variables, and corresponding bound variables are bound by corresponding quantifiers." The congruence requirement may be handled by first defining a pre-valuation, which looks just like a tuaf semantics itself, and next defining a valuation from the pre-valuation. The tuaf semantics of $\mathbf{C}_1$ clarifies the matter.

The assignment function is again the general one, as in the **CL**-semantics from the beginning of this section. For atomic formulas, the pre-valuation $v_M : \mathcal{W}_\mathcal{O} \to \{t, u, f\}$ has the same tables as the tuaf valuation of **CLuN**—these tables are not repeated.

Where $A \in \mathcal{W}^a_{\mathcal{O}}$:

$v_M(A)$	$v(\neg A)$	$v_M(\neg A)$
t	0	f
t	1	u
f	(any)	t

Where $\dagger \in \{\vee, \wedge, \supset\}$ and $A \dagger B$ has not the form $C \wedge \neg C$:

$v_M(A \dagger B)$	$v_M(A^{(1)})$	$v_M(B^{(1)})$	$v(\neg(A \dagger B))$	$v_M(\neg(A \dagger B))$
t	t	t	(any)	f
t	(other)		0	f
t	(other)		1	u
f	(any)		(any)	t

Where $\mathsf{Q} \in \{\forall, \exists\}$:

$v_M(\mathsf{Q}\alpha A(\alpha))$	$\{v_M(A(\beta)^{(1)}) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$	$v(\neg\mathsf{Q}\alpha A(\alpha))$	$v_M(\neg\mathsf{Q}\alpha A(\alpha))$
t	$= \{t\}$	(any)	f
t	$\neq \{t\}$	0	f
t	$\neq \{t\}$	1	u
f	(any)	(any)	t

The other tables apply to all members of $\mathcal{W}_{\mathcal{O}}$:

$v_M(\neg A)$	$v(\neg\neg A)$	$v_M(\neg\neg A)$
t	(any)	f
u	0	f
u	1	u
f	(any)	t

$\neg A$	$A^{(1)}$
t	t
u	f
f	t

$\wedge$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	f

$\vee$	t	u	f
t	t	t	t
u	t	t	t
f	t	t	f

$\supset$	t	u	f
t	t	t	f
u	t	t	f
f	t	t	t

$\equiv$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	t

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha A(\alpha)$	$\exists\alpha A(\alpha)$
$\in \wp\{t, u\}$	t	t
$= \{f\}$	f	f
(other)	f	t

Let $f(A)$ be the obtained by first deleting all vacuous quantifiers in A and then systematically replacing all variables in the result by the first variables of the alphabet in alphabetical order. Next, define the valuation values V_M in terms of the pre-valuation values v_M by $V_M(A) = v_M(f(A))$.

Alternatively, a pre-valuation v_M is called a valuation iff $v_M(A) = v_M(B)$ whenever $A \equiv^c B$.

Transforming the above semantics to a $\mathbf{C}_n$ logic (for $1 < n < \omega$) is an easy exercise left to the reader—the formulation of the tables for $\mathbf{C}_1$ and the plot described in the previous paragraph indicate the road. For $\mathbf{C}_\omega$, one replaces the tables for negation by the left and middle table below; for $\mathbf{C}_{\overline{\omega}}$ (which is $\mathbf{C}_\omega$ extended with classical negation, $\sim$) one adds the table to the right below.

$v_M(A)$	$v(\neg A)$	$v_M(\neg A)$
t	0	f
t	1	u
f	(any)	t

$v_M(\neg A)$	$v(\neg\neg A)$	$v_M(\neg\neg A)$
t	(any)	f
u	0	f
u	1	u
f	(any)	t

A	$\sim A$
t	f
u	f
f	t

Incidentally, an indeterministic tuaf semantics is often more transparent than its deterministic counterpart. As an indeterministic tuaf semantics does not refer to the valuation, it has less clutter in the heads of the tables. So let me display the relevant tables, viz. negation tables, for $\mathbf{C}_1$.

Where $A \in \mathcal{W}^a_{\mathcal{O}}$:

A	$\neg A$
t	$[f, u]$
f	t

Where $\dagger \in \{\vee, \wedge, \supset\}$ and $A \dagger B$ has not the form $C \wedge \neg C$:

$A * B$	$A^{(1)}$	$B^{(1)}$	$\neg(A * B)$
t	t	t	f
t	(other)		$[f, u]$
f	(any)		t

Where $\mathsf{Q} \in \{\forall, \exists\}$:

$\mathsf{Q}\alpha A(\alpha)$	$\{v_M(A(\beta)^{(1)}) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$	$\neg\mathsf{Q}\alpha A(\alpha)$
t	$\{t\}$	f
t	(other)	$[f, u]$
f	(any)	t

The other tables apply to all members of $\mathcal{W}_{\mathcal{O}}$:

$\neg A$	$\neg\neg A$
t	f
u	$[f, u]$
f	t

$\neg A$	$A^{(1)}$
t	t
u	f
f	t

The expression $[f, u]$ indicates that the value may be f or u—this is an indeterministic semantics. Note that the 'normal' value, the one that agrees with **CL** at this point, is f. So if the value is u, it 'drops from the sky' as far as the indeterministic semantics is concerned—in the deterministic semantics, the assignment function interferes at this point. The 'dropping from the sky' holds for the semantics only; a premise set may require that some values are u in its models. For other logics, a premise set may require some values to be a. The metaphor is helpful, however, because it highlights that the values u and a occur at points where an abnormality is generated.

The application of the semantics should clearly be separated from the underlying idea. Even in indeterministic versions, the occurrence of output values u should be understood in terms of the assignment's interference.

6 An Application: The Flip-Flop Danger

A *logic* assigns a set of consequences to every premise set.[25] A logic **L** is *adaptive* if it adapts itself to the specific premises to which it is applied. One way to explicate this phrase is by saying that there are rules R such that **L** does not validate R, but **L** validates some applications of R to some premise sets. Adaptive logics were developed with the aim to obtain precise formulations of defeasible reasoning forms. These reasoning forms are methodological, rather than logico-deductive.

This is not the place to present an introduction to adaptive logics—many survey papers are available and a new state-of-the-art overview is on its way [11, 12, 14]. I shall merely present an example to give the reader a feel of adaptive logics. The example will allow me to point out a problem that is solved by the new type of many-valued semantics.

Let $\Gamma_1 = \{p, q, \neg p \vee r, \neg q \vee s, \neg q\}$. Note that $\Gamma_1 \nvdash_{\mathbf{CLuN}} s$ and $\Gamma_1 \nvdash_{\mathbf{CLuN}} r$. Yet, there is a clear difference between p and q. While Γ_1 requires that q behaves inconsistently, it does not require that p behaves inconsistently. This holds intuitively and **CLuN** leads to exactly the same insight: $\Gamma_1 \vdash_{\mathbf{CLuN}} q \wedge \neg q$ whereas $\Gamma_1 \vdash_{\mathbf{CLuN}} p$ but $\Gamma_1 \nvdash_{\mathbf{CLuN}} \neg p$. The idea behind inconsistency-adaptive logics is to interpret premise sets as normal as possible, where normality is obviously connected to consistency. Interpreting Γ_1 as normally as possible entails that q is considered as inconsistent whereas p and many other formulas are considered as consistent. Doing so results in r and not in s. Indeed, $\Gamma_1 \vdash_{\mathbf{CLuN}} r \vee (p \wedge \neg p)$ and $p \wedge \neg p$ is considered to be false; $\Gamma_1 \vdash_{\mathbf{CLuN}} s \vee (q \wedge \neg q)$ but $q \wedge \neg q$ is true anyway. Precisely this result is delivered by the inconsistency-adaptive logic $\mathbf{CLuN}^m$.

A crucial component of adaptive logics is their set of abnormalities. In the case of $\mathbf{CLuN}^m$, this set is $\{\exists(A \wedge \neg A) \mid A \in \mathcal{F}\}$, the existential closure of contradictory formulas. As suggested before, the idea is that, if B is an abnormality and A is not, if $A \vee B$ is **CLuN**-derivable, and if the premises allow one on systematic and formal grounds to consider B as false,[26] then A is an adaptive consequence, in this case a $\mathbf{CLuN}^m$-consequence.

Now consider another inconsistency-adaptive logic, $\mathbf{CLuNs}^m$. One difference is that **CLuN** is replaced by **CLuNs**. This has the effect that all **CLuNs** consequences are derivable from the premises independently of the fact that one aims at a maximally normal interpretation. Thus $\Gamma_1 \vdash_{\mathbf{CLuNs}^m} \neg\neg q$, whereas $\Gamma_1 \nvdash_{\mathbf{CLuN}^m} \neg\neg q$.[27]

In defining $\mathbf{CLuNs}^m$, one might be tempted to consider the same set of abnormalities as in the case of $\mathbf{CLuN}^m$, viz. $\{\exists(A \wedge \neg A) \mid A \in \mathcal{F}\}$. If one does so, however, one looses the adaptive effect. Indeed, although $\Gamma_1 \vdash_{\mathbf{CLuNs}} r \vee (p \wedge \neg p)$ and $\Gamma_1 \nvdash_{\mathbf{CLuN}} p \wedge \neg p$, one cannot simply consider p as behaving consistently on Γ_1. This is so because $\Gamma_1 \vdash_{\mathbf{CLuN}} (p \wedge \neg p) \vee ((r \wedge q) \wedge \neg(r \wedge q))$, whereas $\Gamma_1 \nvdash_{\mathbf{CLuN}} (r \wedge q) \wedge \neg(r \wedge q)$. So $(p \wedge \neg p) \vee ((r \wedge q) \wedge \neg(r \wedge q))$ is a minimal disjunction of abnormalities that is **CLuNs**-derivable from Γ_1. One of the disjuncts is false, but we do not know which one and cannot decide on logical grounds for one or

[25] This weak characterization is preferable in order to avoid prejudged narrowing of the domain. It is equivalent to the characterization offered by Béziau [18].

[26] The matter is handled by an adaptive strategy—see the referred survey papers.

[27] If this sounds puzzling, please realize that $\{q, \neg q, \neg\neg q, \neg\neg\neg q\}$ is more inconsistent than $\{q, \neg q, \neg\neg\neg q\}$ and that **CLuN** does not validate $A \supset \neg\neg A$.

for the other *if* both disjuncts count as abnormalities. It can be shown that this reasoning can be generalized. If $\mathbf{CLuNs}^m$ is given $\{\exists(A\wedge\neg A) \mid A \in \mathcal{F}\}$ as its set of abnormalities, the consequences derivable by $\mathbf{CLuNs}^m$ from any inconsistent premise set Γ are identical to the consequences derivable by **CLuNs** from Γ. Such an adaptive logic is called a flip-flop: its consequence set is identical to the **CL**-consequence set if Γ is consistent and to the **CLuNs**-consequence set otherwise.[28]

This does not mean that a decent inconsistency-adaptive logic $\mathbf{CLuNs}^m$ cannot be defined. Such a logic is obtained, e.g., by defining $\{\exists(A \wedge \neg A) \mid A \in \mathcal{F}^a\}$ as the set of abnormalities, $\mathcal{F}^a$ being the set of atomic formulas of $\mathcal{L}$. In this case $p\wedge\neg p$ can be considered as false and r will be a $\mathbf{CLuNs}^m$-consequence of Γ_1.[29]

Abnormality is used here as a technical term. There is obviously a relation to the intuitive sense of the term. The intuitive sense may be seen as defined by **CL**.[30] Every inconsistent theory is *intuitively abnormal*: it has no **CL**-models. This justifies the choice of $\{\exists(A \wedge \neg A) \mid A \in \mathcal{F}^a\}$ as the set of abnormalities for the inconsistency-adaptive logic $\mathbf{CLuNs}^m$. Indeed, if Γ is inconsistent, then there is an $A \in \{\exists(A \wedge \neg A) \mid A \in \mathcal{F}^a\}$ such that $\Gamma \vdash_{\mathbf{CLuNs}} A$. So if Γ is normal in the intuitive sense, then it is $\mathbf{CLuNs}^m$-normal; and vice versa.

The handiest way to describe gluts and gaps is available when all classical logical symbols are present in the language. A negation gap will for example be described by $\check{\neg}A \mathbin{\check{\wedge}} \check{\neg}\neg A$, in which the 'checked' symbols have their **CL**-meaning.[31] Similarly, a disjunction glut may be described by $\check{\neg}A \mathbin{\check{\wedge}} \check{\neg}B \mathbin{\check{\wedge}} (A \vee B)$ or, more transparently, by $\check{\neg}(A \mathbin{\check{\vee}} B) \mathbin{\check{\wedge}} (A \vee B)$.

A possible source of confusion should be clarified here. If $v_M(\neg p) = u$ in a tuaf-semantics, then $M \Vdash \check{\neg}p \wedge \check{\neg}\neg p$—the classical negation of p is true while its standard negation is false. If $v_M(\neg p) = a$ in a tuaf-semantics, then $M \Vdash \check{\neg}\check{\neg}p \wedge \neg p$—the classical negation of p is false while its standard negation is true.[32] Similarly for other logical symbols. So it is important to distinguish between the formula A that 'displays' a glut or gap in a model M, whether on the agreement view or on the interference view, and the formula that 'describes' the glut or gap and is verified by M. Let the $(A)^a$, respectively $(A)^u$, be the formula that 'describes' that A 'displays' a gap, respectively a glut.

An adaptive logic that is a flip-flop may be turned into a non-flip-flop by weakening its set of abnormalities. There is, however a danger to that as well. The danger may be illustrated by considering the inconsistency-adaptive logic—call it **X**—that is obtained by replacing the set of abnormalities of $\mathbf{CLuN}^m$ by the set $\{\exists(A \wedge \neg A) \mid A \in \mathcal{F}^a\}$. Consider the simple but explicit premise set $\Gamma_2 = \{p \vee ((q \wedge r) \wedge \neg(q \wedge r))\}$. The fact is that $\Gamma_2 \nvdash_{\mathbf{X}} p$ and the reason is that there are no abnormalities $A_1, \ldots, A_n \in \{\exists(A \wedge \neg A) \mid A \in \mathcal{F}^a\}$ such

[28] In some exceptional cases, one wants an adaptive logic that is a flip-flop.

[29] The formula $(r \wedge q) \wedge \neg(r \wedge q)$ is not an abnormality but it is a $\mathbf{CLuNs}^m$-consequence of Γ_1. Indeed, $q \wedge \neg q$ is a **CLuNs**-consequence (and hence a $\mathbf{CLuNs}^m$-consequence) of Γ_1, r is a $\mathbf{CLuNs}^m$-consequence of Γ_1, and $q \wedge \neg q, r \vdash_{\mathbf{CLuNs}} (r \wedge q) \wedge \neg(r \wedge q)$.

[30] Still and to the best of my knowledge, paraconsistent logics that do not allow for other gluts or gaps and were proposed to serve a sensible purpose agree with **CL** in classifying Γ as inconsistent.

[31] The 'checked' symbols are metalinguistic names for certain symbols of the language $\mathcal{L}$ of logic **L**. If the $\wedge$ is a classical conjunction in **L** and $\sim$ is a classical negation in **L**, then the formula in the text stands for $\sim A \wedge \sim\neg A$.

[32] Note that $M \Vdash p \wedge \neg p$ will do just as good.

that $\Gamma_2 \vdash_{\mathbf{CLuN}} p \vee (A_1 \vee \ldots \vee A_n)$. So here lurks a different danger: that the technical sense of abnormality is too weak with respect to the intuitive sense of abnormality, whence some consistent sets, like Γ_2, are not assigned all **CL**-consequences. If the aim is to interpret theories as consistently as possible in the sense of **CL**, then the inconsistency-adaptive logic **X** is clearly too weak.

So while there is, on the one hand, the flip-flop danger, there is, on the other hand, (what may be called) the wimp danger. It is typical for adaptive logics that both extending the set of abnormalities and reducing it may lead to a weakening of the consequence set. The matter is too complex to fully discuss it here, but the use of the many-valued logics from the previous section may still be illustrated. They do not offer a single criterion, but rather two criteria. Moreover, they obviously can only be applied if the considered logics have a tuaf semantics. Finally, I cannot show that the (partial) criteria in the subsequent paragraph are correct because this requires much more technical information on adaptive logics—people familiar with adaptive logics will find the matter rather transparent.

Let **L** be the deductive logic that underlies the adaptive logic—like **CLuN** and **CLuNs** in the previous examples—and let the semantic phraseology refer to the tuaf semantics of **L**. Let the adaptive logic be $\mathbf{L}^m$ and $\Omega \subset \mathcal{W}$ its set of abnormalities.

(a) $\mathbf{L}^m$ is not a flip-flop if, for every $A \in \mathcal{W}$, (i) $(A)^a \notin \Omega$ if there is a **L**-tuaf-model M such that $v_M(A) = t$ and $M \Vdash (A)^a$ and (ii) $(A)^u \notin \Omega$ if there is a **L**-tuaf-model M such that $v_M(A) = f$ and $M \Vdash (A)^u$.[33]

(b) $\mathbf{L}^m$ is not a wimp if, for every $A \in \mathcal{W}$, (i) $(A)^a \in \Omega$ if there is a **L**-tuaf-model M such that $v_M(A) = a$ and $M \Vdash (A)^a$ and (ii) $(A)^u \in \Omega$ if there is a **L**-tuaf-model M such that $v_M(A) = u$ and $M \Vdash (A)^u$.[34]

The main antecedent of (a) is not fulfilled for any flip-flops but is fulfilled for some wimps. The main antecedent of (b) is not fulfilled for any wimps, but is fulfilled for some flip-flops. If it sounds confusing, realize that (a) and (b) delineate extremes of the sets of abnormalities, not of the adaptive consequence sets.

7 Some Reflections

The tuaf semantics introduced in Section 5 provides insights that are useful independent of the flip-flop problem.[35] The tuaf semantics delineates the points at which inconsistencies originate—similarly for other non-standard features, but I shall continue to concentrate on inconsistencies. So the information provided by a tuaf semantics is very valuable for comparing different paraconsistent logics and even for understanding specific paraconsistent logics separately. A

[33] The main antecedent warrants that there are **L**-models M and M' of intuitively abnormal Γ such that $\{A \in \Omega \mid M \Vdash A\} \subset \{A \in \Omega \mid M' \Vdash A\}$. That some **L**-models of intuitively abnormal Γ are not minimally abnormal **L**-models of Γ entails that $\mathbf{L}^m$ is not a flip-flop.

[34] The main antecedent warrants that every minimally abnormal **L**-model M of an intuitively normal Γ is such that $\{A \in \Omega \mid M \Vdash A\} = \emptyset$. This entails that, for all intuitively normal Γ, $\Gamma \vdash_{\mathbf{L}^m} A$ iff $\Gamma \vdash_{\mathbf{CL}} A$. So $\mathbf{L}^m$ is not a wimp.

[35] In connection with the flip-flop problem, the result may easily be generalized to, for example, modal logics. There are indeed adaptive logics in which abnormalities have the form $\Diamond A \wedge \neg A$ or the form $\Diamond A \wedge \Diamond \neg A$. This, however, should not be elaborated here.

typical difference between **CLuNs** and **CLuN** is that the former makes all inconsistencies dependent on inconsistencies in atomic formulas whereas **CLuN** makes inconsistencies independent of each other.[36] While it is not difficult to understand the behaviour of inconsistencies in those two logics, the matter is more difficult for other paraconsistent logics, such as $\mathbf{C}_1$ or $\mathbf{CLuN}^{NN}$—and also for other non-standard behaviour as allowed by logics like **CLuCoDaM**. In all such cases, the tuaf semantics is definitely clarifying. Some readers may question the use of these many-valued logics in view of the fact that the information they provide may also be obtained (in a more laborious way) from the two-valued semantics. This objection does not hold water. If it did, it would just as well be an objection against usual many-valued semantic systems.

The main conclusion on semantic systems in general is that one should separate technical features from philosophical ones. That a logic has a many-valued semantics is a technical feature, and so is the fact that the logical symbols are or are not truth-functions in that semantics. Such technicalities do not determine the ontological structure of domains to which the logic may sensibly be applied. The same logic may very well agree with different ontological views and each of these may suggest a different set of valuation values. A nice example is that Priest's **LP** has a three-valued semantics in which all logical symbols are truth-functions, but that the ontology underlying this semantics is clearly at odds with Priest's dialetheism [34, §19.7]—see also footnote 15.

A logic **L** need not to be given an interpretation that agrees with a **L**-semantics in which all logical symbols are truth-functions. In some cases it is hard to imagine an interpretation that would go along with such a semantics—the $\mathfrak{S}$-valued and the $\mathfrak{T}$-valued semantics of **CLuN** are ready examples. Which is the set of truth-values, or more generally of valuation values, that statements may take, is a philosophical question. A sensible person might hold that there are three truth-values, say plain truth, plain falsehood, and inconsistency, and this person might want to allow for complex inconsistencies that have only consistent components, some true, some false. This person might end up with **CLuN** as her preferred logic and might end up with a three-valued **CLuN**-semantics, in which negation is not a truth-function, as the best way to picture the world's ontology.

The aim of this paper was to raise questions, rather than to draw conclusions. The aim of the questions was to criticize prejudices, especially prejudices on many-valued logics, on the use or need to express the valuation values within the object language, on truth-functionality, and on the connection of all this to the semantics' ontological significance.

In a sense this paper concerns consequences of Suszko's aforementioned result. If many-valued logics have a two-valued semantics and if this semantics, unlike the many-valued one, expresses the truth-preservation underlying the consequence relation, then the many-valued semantics is bound to serve a different purpose. But obviously there are several such purposes and these will lead to different many-valued semantic characterizations of the same logic.

Much work remains to be done in connection with the two preceding paragraphs. An obvious topic of research is the generalization of the $\mathfrak{S}$-semantics and of the $\mathfrak{T}$-semantics to other logics than **CLuN** and the study of the prop-

[36] A conjunct of an inconsistency may be an inconsistency itself, as is the case for $(p \wedge \neg p) \wedge \neg(p \wedge \neg p)$. Even then the complex inconsistency is independent of the less complex one.

erties of logics and of classes of logics revealed by this generalization. Coding the information from a worlds-semantics into a many-valued semantics seems a closely related task. A very different topic concerns the articulation of many-valued logics originating from the agreement view, as opposed to the interference view, and the study of insights offered by both types of many-valued semantics. While **CL** was considered as the absolute point of reference in the present paper, shifting to a different point of reference may have enlightening effects. All such research will help us, logicians, to overcome traditional prejudices and to better understand the aims, properties, and uses of logic at the service of reasoning and thus of understanding and action.

References

[1] Alan Ross Anderson, Nuel D. Belnap, Jr., and J. Michael Dunn. *Entailment. The Logic of Relevance and Necessity*, volume 2. Princeton University Press, 1992.

[2] Ofer Arieli, Arnon Avron, and Anna Zamansky. Maximal and premaximal paraconsistency in the framework of three-valued semantics. *Studia Logica*, 97:31–60, 2011.

[3] Arnon Avron. On an implication connective of RM. *Notre Dame Journal of Formal Logic*, 27:201–209, 1986.

[4] Arnon Avron. Natural 3-valued logics—Characterization and proof theory. *The Journal of Symbolic Logic*, 56:276–294, 1991.

[5] Arnon Avron. Non-deterministic matrices and modular semantics of rules. In Jean-Yves Béziau, editor, *Logica Universalis*, pages 149–167. Birkhäuser Verlag, Basel – Boston – Berlin, 2005.

[6] Arnon Avron, Jonathan Ben-Naim, and Beata Konikowska. Cut-free ordinary sequent calculi for logics having generalized finite-valued semantics. *Logica Universalis*, 1:41–70, 2007.

[7] Arnon Avron and Beata Konikowska. Multi-valued calculi for logics based on non-determinism. *Journal of the Interest Group of Pure and Applied Logic*, 10:365–387, 2005.

[8] Diderik Batens. Paraconsistent extensional propositional logics. *Logique et Analyse*, 90–91:195–234, 1980.

[9] Diderik Batens. A bridge between two-valued and many-valued semantic systems: n-tuple semantics. *Proceedings of the 12th International Symposium on Multiple-Valued Logic*, IEEE:Los Angeles, 318–322, 1982.

[10] Diderik Batens. Inconsistency-adaptive logics. In Ewa Orłowska, editor, *Logic at Work. Essays Dedicated to the Memory of Helena Rasiowa*, pages 445–472. Physica Verlag (Springer), Heidelberg, New York, 1999.

[11] Diderik Batens. A general characterization of adaptive logics. *Logique et Analyse*, 173–175:45–68, 2001. Appeared 2003.

[12] Diderik Batens. A universal logic approach to adaptive logics. *Logica Universalis*, 1:221–242, 2007.

[13] Diderik Batens. *Adaptive Logics and Dynamic Proofs. Mastering the Dynamics of Reasoning.* 201x. Forthcoming.

[14] Diderik Batens. Tutorial on inconsistency-adaptive logics. 201x.

[15] Diderik Batens. Spoiled for choice? *Journal of Logic and Computation*, in print. doi:10.1093/logcom/ext019, 1913.

[16] Diderik Batens and Kristof De Clercq. A rich paraconsistent extension of full positive logic. *Logique et Analyse*, 185–188:227–257, 2004. Appeared 2005.

[17] Diderik Batens, Chris Mortensen, Graham Priest, and Jean Paul Van Bendegem, editors. *Frontiers of Paraconsistent Logic.* Research Studies Press, Baldock, UK, 2000.

[18] Jean-Yves Béziau. What is paraconsistent logic? In Batens et al. [17], pages 95–111.

[19] Jean-Yves Béziau. Are paraconsistent negations negations? In Walter A. Carnielli, Marcelo E. Coniglio, and Itala M. Loffredo D'Ottaviano, editors, *Paraconsistency. The Logical Way to the Inconsistent*, pages 465–486. Marcel Dekker, New York, 2002.

[20] Rudolf Carnap. *Meaning and Necessity.* University of Chicago Press, Chicago, 1947.

[21] Rudolf Carnap. *Logical Foundations of Probability.* University of Chicago Press, Chicago, 1950.

[22] Walter A. Carnielli, João Marcos, and Sandra de Amo. Formal inconsistency and evolutionary databases. *Logic and Logical Philosophy*, 8:115–152, 2001. Appeared 2002.

[23] Newton C.A. da Costa. On the theory of inconsistent formal systems. *Notre Dame Journal of Formal Logic*, 15:497–510, 1974.

[24] Itala M. L. D'Ottaviano. *Sobre uma Teoria de Modelos Trivalente* (in Portuguese). PhD thesis, State University of Campinas (Brazil), 1982.

[25] Itala M. L. D'Ottaviano. The completeness and compactness of a three-valued first-order logic. In *Proceedings of the* 5*th Latin American Symposium on Mathematical Logic*, pages 77–94. *Revista Colombiana de Matemáticas*, 1–2, 1985.

[26] Itala M. L. D'Ottaviano. The model extension theorems for $\mathbf{J}_3$-theories. In Carlos A. Di Prisco, editor, *Methods in Mathematical Logic: Proceedings of the* 6*th Latin American Symposium on Mathematical Logic*, Lecture Notes in Mathematics 1130, pages 157–173. Springer-Verlag, 1985.

[27] Itala M. L. D'Ottaviano. Definability and quantifier elimination for $\mathbf{J}_3$-theories. *Studia Logica*, 46(1):37–54, 1987.

[28] Itala M. L. D'Ottaviano and Richard L. Epstein. A paraconsistent many-valued propositional logic: $\mathbf{J}_3$. *Reports on Mathematical Logic*, 22:89–103, 1988.

[29] Olivier Esser. A strong model of paraconsistent logic. *Notre Dame Journal of Formal Logic*, 44:149–156, 2003.

[30] Stephen Cole Kleene. *Introduction to Metamathematics*. North-Holland, Amsterdam, 1952.

[31] Victor Kraft. *Der Wiener Kreis*. Springer, Wien, 1950.

[32] Iddo Lev. Preferential systems for plausible non-classical reasoning. Master's thesis, Department of Computer Science, Tel-Aviv University, 2000. Unpublished M.A. dissertation.

[33] Graham Priest. The logic of paradox. *Journal of Philosophical Logic*, 8:219–241, 1979.

[34] Graham Priest. *In Contradiction. A Study of the Transconsistent*. Oxford University Press, Oxford, 2006. Second expanded edition (first edition 1987).

[35] J. B. Rosser and A. R. Turquette. Axiom schemes for m-valued propositional calculi. *Journal of Symbolic Logic*, 10:61–82, 1945.

[36] Kurt Schütte. *Beweistheorie*. Springer, Berlin, 1960.

[37] Elena D. Smirnova. An approach to the justification of semantics of paraconsistent logics. In Batens et al. [17], pages 255–262.

[38] Roman Suszko. The Fregean axiom and Polish mathematical logic in the 1920s. *Studia Logica*, 36:377–380, 1977.

[39] Peter Verdée and Diderik Batens. Nice embedding in classical logic. Under review, 201x.

[40] Georg Henrik von Wright. Truth-logics. *Acta Philosophica Fennica*, 60:71–91, 1996.