Nice Embedding in Classical Logic*

Peter Verdée[†], Diderik Batens
Centre for Logic and Philosophy of Science
Ghent University, Belgium
PVerdee@GMail.com, Diderik.Batens@UGent.be

embed_fin2.tex

Abstract

It is shown that a set of semi-recursive logics, including many fragments of **CL** (Classical Logic), can be embedded within **CL** in an interesting way. A logic belongs to the set iff it has a certain type of semantics, called nice semantics. The set includes many logics presented in the literature. The embedding reveals structural properties of the embedded logic. The embedding turns finite premise sets into finite premise sets. The partial decision methods for **CL** that are goal directed with respect to **CL** are turned into partial decision methods that are goal directed with respect to the embedded logics.

Keywords: embedding, translations, classical logic, gluts and gaps, bi-valued semantics.

1 Aim of this Paper

Let $\mathcal{W}$ and $\mathcal{W}'$ be the sets of formulas of the languages of logics $\mathbf{L}$ and $\mathbf{L}'$ respectively. An *embedding* of $\mathbf{L}$ within $\mathbf{L}'$ is a recursive function emb: $\wp(\mathcal{W}) \times \mathcal{W} \rightarrow \wp(\mathcal{W}') \times \mathcal{W}'$ such that $\Gamma \vdash_{\mathbf{L}} A$ iff $\Gamma' \vdash_{\mathbf{L}'} A'$, whenever $(\Gamma', A') = \mathrm{emb}(\Gamma, A)$. In this paper, we propose a specific type of embedding of logics within **CL**, which we shall call *nice embedding*.

This type of embedding has several advantages over other types of embedding. In the case of a nice embedding, (i) Γ' is finite whenever Γ is so, (ii) Γ' is a recursive set whenever Γ is so, and (iii) partial decision methods that are goal directed with respect to **CL** are turned into partial decision methods that are goal directed with respect to the embedded logics.

Nice embedding provides easy means to show the presence or absence of meta-properties like reflexivity, transitivity, monotonicity, compactness, structurality, interpolation, and so on. It allows one to devise tableau methods and

*We are indebted to Kristof De Clercq for participating in the early research for this paper and for many careful comments on early drafts. We also express our gratitude to a referee whose comments enabled us to clarify our text at several points.

[†]During the research of this paper Peter Verdée was a post-doctoral fellow of the Fund for Scientific Research – Flanders at Ghent University and later of FAPESP at the State University of Campinas, SP, Brazil.

other partial decision methods for the embedded logic. For many logics, their nice embedding also provides heuristic insights (with respect to proofs, tableaux, etc.).

In Section 7 we shall compare nice embedding with other types of embedding in **CL** and highlight its advantages.

It was shown in [7] that a set of propositional logics which are paraconsistent (A and $\neg A$ can be jointly true) or paracomplete (A and $\neg A$ can be jointly false) have (what we now call) a nice embedding. In the present paper we generalize this result not only to the predicative version of those logics, but also to a large set of fragments of **CL** that allow for gluts and/or gaps with respect to other logical symbols than negation—that $\neg A$ is true while A is also true is a negation glut; that $A \wedge B$ is false while A and B are true is a conjunction gap. The embedding result is further generalized to all logics that have (what we shall define as) a *nice semantics*. So we show that, where **L** has a nice semantics, A is a formula, and Γ is a recursive set of formulas, there exists a formula A' and a recursive Γ' such that $\Gamma \vdash_{\mathbf{L}} A$ iff $\Gamma' \vdash_{\mathbf{CL}} A'$. We moreover show that Γ' is finite whenever Γ is so.

Every nice semantics is deterministic and two-valued. The metalanguage of a nice semantics is always classical. A nice semantics defines a semantic consequence relation that is semi-recursive—this follows from the fact that every logic **L** that has a nice semantics is embeddable within **CL**.

Incidentally, the notion of a nice semantics is interesting in itself. In past work, we experienced the notion's usefulness in devising a logic in terms of its semantics, axiomatizing it, studying its properties, employing it as the underlying logic of theories, and so on.[1] We shall present an assignment that can serve as a common basis for every nice semantics. Before getting there, we need a few technicalities.

2 Preliminaries

Let $\mathcal{L}_s$ be the language of **CL** with the logical symbols $\neg$, $\wedge$, $\vee$, $\supset$, $\equiv$, $\forall$, $\exists$, and $=$ (but without function symbols); $\mathcal{C}$ is the set of (letters for) individual constants, $\mathcal{V}$ the set of individual variables, and $\mathcal{P}^r$ the set of predicates of rank $r \geq 0$—predicates of rank 0 will function as sentential letters. Officially, the members of $\mathcal{P}^r$ will be $P^r, Q^r, R^r, P^r_1, \ldots$, but we shall often write the superscripts invisibly, relying on the usual convention that we write only well-formed strings. Let $\mathcal{F}_s$ and $\mathcal{W}_s$ denote respectively the set of formulas and the set of closed formulas of $\mathcal{L}_s$.

By $\mathcal{L}$ we shall refer to any language that has the same non-logical symbols as $\mathcal{L}_s$ and an arbitrary set of logical symbols—these may but need not be symbols of $\mathcal{L}_s$. $\mathcal{L}$ will be the language of the logic that is embedded in **CL**. In some contexts $\mathcal{L}$ will be a variable for such languages, in others it will refer to a specific such language. Let $\mathcal{F}$ and $\mathcal{W}$ denote respectively the set of formulas and the set of closed formulas of $\mathcal{L}$.

The easiest way to present the embedding is to consider a language $\mathcal{L}_\sharp$, which extends $\mathcal{L}_s$ in view of the specific language $\mathcal{L}$. We first introduce some functions that have $\mathcal{F}$ as their domain. Let $f(A)$ be the string obtained by

[1] For the sake of an example: all the non-defeasible logics from [6].

replacing in $A \in \mathcal{F}$ every occurrence of an individual constant and every free occurrence of an individual variable by a centred dot. Thus $f(\exists y(Pay \supset Qbx)) = f(\exists y(Pxy \supset Qxx)) = \exists y(P \cdot y \supset Q \cdot \cdot)$. Let $h(A)$ be the number of centred dots that occur in $f(A)$—for example $h(\exists y(Pay \supset Qbx)) = 3$. Let $g(A)$ be the (possibly empty) string obtained by deleting from A all symbols except for occurrences of individual constants and free occurrences of individual variables. Thus $g(\exists y(Pay \supset Qbx)) = abx$, and $g(\exists y(Pxy \supset Qxx)) = xxx$. Finally, let the functions $g_i(A)$ denote the ith item in $g(A)$, $g_i(A)$ being undefined for $i \notin \{1, ..., h(A)\}$. For example, $g_2(\exists y(Pay \supset Qbx)) = b$ and $g_4(\exists y(Pay \supset Qbx))$ is undefined.

The language $\mathcal{L}_\sharp$ is obtained from $\mathcal{L}_s$ by adding (i) a new binary predicate I and (ii) a set of new predicates containing, for every $A \in \mathcal{W}$, a predicate $P^{h(A)}_{f(A)}$. Thus P^0_P and $P^0_{\forall x(Px \supset Qx)}$ are new predicates of rank 0, $P^1_{P\cdot}$ and $P^1_{\forall x(P\cdot \supset Qx)}$ are new predicates of rank 1, etc. Let, for every $r \in \mathbb{N}$, $\mathcal{P}^r_\sharp$ be the set of all predicates of rank r of $\mathcal{L}_\sharp$. Let $\mathcal{F}_\sharp$ and $\mathcal{W}_\sharp$ denote respectively the set of formulas and the set of closed formulas of $\mathcal{L}_\sharp$.

In order to simplify the characterization of the semantic systems, we introduce pseudo-languages. Consider sets $\mathcal{O}$ of *pseudo-constants*; for each model M, a set $\mathcal{O}$ should have at least the cardinality of the domain of M. A *pseudo-language* ${}^+\mathcal{L}$ is obtained from $\mathcal{L}$ by replacing $\mathcal{C}$ by $\mathcal{C} \cup \mathcal{O}$. We always tacitly presume that, in every context, a ${}^+\mathcal{L}$ is chosen with sufficiently large $\mathcal{O}$. Let ${}^+\mathcal{F}$ and ${}^+\mathcal{W}$ denote respectively the set of formulas and the set of closed formulas of ${}^+\mathcal{L}$. In a similar way one defines the pseudo-languages ${}^+\mathcal{L}_s$ and ${}^+\mathcal{L}_\sharp$ from $\mathcal{L}_s$ and $\mathcal{L}_\sharp$ respectively. Their sets of formulas are respectively ${}^+\mathcal{F}_s$ and ${}^+\mathcal{F}_\sharp$, their sets of closed formulas respectively ${}^+\mathcal{W}_s$ and ${}^+\mathcal{W}_\sharp$.

Extend the functions f, g, h, and g_i to the pseudo-languages ${}^+\mathcal{L}$, ${}^+\mathcal{L}_s$, and ${}^+\mathcal{L}_\sharp$ by letting them refer to $\mathcal{C} \cup \mathcal{O} \cup \mathcal{V}$ instead of to $\mathcal{C} \cup \mathcal{V}$. Let $\mathcal{Z}^0 = \{f(A) \mid A \in {}^+\mathcal{W}; h(A) = 0\} \cup {}^+\mathcal{W}$ and, for all $r > 0$, $\mathcal{Z}^r = \{f(A) \mid A \in {}^+\mathcal{W}; h(A) = r\}$. The sets $\mathcal{Z}^r_s$ and $\mathcal{Z}^r_\sharp$ are defined similarly (for all $r \geq 0$), replacing ${}^+\mathcal{W}$ by the suitable set of closed pseudo-formulas. Also extend f, g, h, and the g_i to the metalanguage in the standard way.

In the semantic systems, the assignment function v will assign a set of $h(A)$-tuples of members of the domain to every $f(A)$ for which A is a closed pseudo-formula of the language. So $v(f(P^2ab)) = v(P^2 \cdot\cdot)$ is a set of couples. If v were to assign a value to P^2, one would obviously require that $v(P^2) = v(P^2 \cdot\cdot)$. For this reason we identify, for every $\pi^r \in \mathcal{P}^r$, π^r with $\pi^r \cdot \ldots \cdot$ (in which $\cdot \ldots \cdot$ denotes r centred dots). As an effect, $\mathcal{P}^r \subset \mathcal{Z}^r$. Moreover, a 0-tuple will be identified with $\emptyset$—see, for example, clause C2.1 in Section 4. So, if $h(\neg A) = 0$, $\langle v(g_1(\neg A)), \ldots, v(g_{h(\neg A)}(\neg A))\rangle$ is a 0-tuple, and hence is identified with $\emptyset$—see, for example, clause C2.3$^{o=}$ in Section 4.

Let, for every $r > 0$, $D^{(r)}$ denote the r-th Cartesian product of D and let $D^{(0)} = \{\emptyset\}$.

Let $\mathbb{P} \subset \mathcal{F}$ be the set of formulas that do not contain any logical symbols (not even identity), and let $\mathbb{P}^= = \mathbb{P} \cup \{\alpha = \beta \mid \alpha, \beta \in \mathcal{C} \cup \mathcal{V}\}$. Let ${}^+\mathbb{P}$ and ${}^+\mathbb{P}^=$ be defined analogously in terms of ${}^+\mathcal{F}$ and $\mathcal{C} \cup \mathcal{O} \cup \mathcal{V}$. Finally, let ${}^m\mathbb{P}$ be the set of metalinguistic formulas that do not contain any logical symbols and ${}^m\mathbb{P}^=$ the set of metalinguistic formulas that do not contain any logical symbols different from identity.

The further use of symbols will be self-explanatory, except (perhaps) for the following. ${}^{m}\mathcal{F}$ will denote the set of metalinguistic formulas (which contain only metavariables and logical symbols of the object language) and ${}^{m}\mathcal{W}$ the set of closed metalinguistic formulas. We shall use the following metametalinguistic variables: A and B as variables for metalinguistic formulas, P^r as a variable for metavariables for predicates of rank r, a, b, $\mathsf{a}_1, \ldots,$ as variables for metavariables for individual constants and individual pseudo-constants, and x as a variable for metavariables for individual variables. The symbols $\mathfrak{A}$, $\mathfrak{B}$, $\mathfrak{A}'$, ... will be used as variables for metalinguistic statements that occur in semantic clauses (we shall call these statements *semantic statements*).

3 Nice semantics

All semantic systems will have the same type of models—$\mathcal{L}$ is a variable in the following definition.

Definition 1 *A model M (for the language ${}^{+}\mathcal{L}$ and hence for $\mathcal{L}$) is a couple $\langle D, v\rangle$ in which D is a non-empty set and the assignment v is as follows:*

C1.1 $v\colon \mathcal{C} \cup \mathcal{O} \to D$ *(where $D = \{v(\alpha) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$)*
C1.2 $v\colon \mathcal{Z}^r \to \wp(D^{(r)})$ *(for every $r \in \mathbb{N}$)*

Let $\mathcal{M}$ be the set comprising the metavariables for non-logical symbols and the metavariables for formulas. Let $\overline{\mathsf{A}}$ be the set of members of $\mathcal{M}$ that occur in A. Let m be an *instantiation function* iff m maps every member of $\mathcal{M}$ on a symbol or formula from the object language for which it is a variable. The formula $m(\mathsf{A})$ is obtained by replacing every metavariable $\mu \in \mathcal{M}$ by $m(\mu)$. Let $i(\mathsf{A})$ be the set of all $A \in {}^{+}\mathcal{W}$ such that $m(\mathsf{A}) = A$ for an instantiation function m. A *logical form* ψ will be identified with a couple $\langle \mathsf{A}, \{\mathsf{B}_1, \ldots, \mathsf{B}_n\}\rangle$ ($n \geq 0$) and a formula A will be said to have the form $\psi = \langle \mathsf{A}, \{\mathsf{B}_1, \ldots, \mathsf{B}_n\}\rangle$ iff $A \in i(\mathsf{A}) - (i(\mathsf{B}_1) \cup \ldots \cup i(\mathsf{B}_n))$. If $n = 0$, we shall also say that A *has the form of* A.

We shall distinguish between two kinds of nice semantics: those for logics that follow the RoI schema and those for logics that do not. A logic $\mathbf{L}$ follows the RoI schema iff it validates the rule of replacement of identicals: $\alpha = \beta, A(\alpha) \vdash_{\mathbf{L}} A(\beta)$ for all $\alpha, \beta \in \mathcal{C}$.

Definition 2 *By* semantic elements *we shall mean the expressions that occur in quotation marks in (i)–(vi):*
(i) "$v_M(\mathsf{B}) = 1$", *with* $\mathsf{B} \in {}^{m}\mathcal{W}$,
(ii) "$\langle v(\mathsf{a}_1), \ldots, v(\mathsf{a}_r)\rangle \in v(\mathsf{P}^r)$",
(iii) "$v(\mathsf{a}) = v(\mathsf{b})$",
(iv) "$0 = 0$",
(v) "$\langle v(g_1(\mathsf{B})), \ldots, v(g_{h(\mathsf{B})}(\mathsf{B}))\rangle \in v(f(\mathsf{B}))$" *with* $\mathsf{B} \in {}^{m}\mathcal{W}$ *and* B *not of the form* $\mathsf{a} = \mathsf{b}$,
(vi) "$v(\mathsf{B}) = \{\emptyset\}$" *with* $\mathsf{B} \in {}^{m}\mathcal{W}$.
The semantic elements from (i)–(v) are RoI-semantic elements*, those from (i)–(iv) and (vi) are* non-RoI-semantic elements*; those from (ii)–(vi) are* semantic base elements.[2]

[2]Semantic base elements do not refer to the (pre-)valuation value of other formulas.

Definition 3 *A* RoI-valuation-defining clause *has officially the following structure:*

$$\textit{[Where } \mathsf{A} \textit{ has the form } \psi\textit{,] } v_M(\mathsf{A}) = 1 \textit{ iff } \mathfrak{A}.$$

provided (i) A *is the first element of* ψ*, (ii)* $\mathfrak{A}$ *is a finite semantic statement made up by parentheses, occurrences of "not", "or", "and", "for all* $\mathsf{a} \in \mathcal{C} \cup \mathcal{O}$*", "for at least one* $\mathsf{a} \in \mathcal{C} \cup \mathcal{O}$*", and one or more RoI-semantic elements, and (iii) every metavariable that occurs in* $\mathfrak{A}$ *either occurs in* A *or is bound by (or occurs in) a metaquantifier of the form "for all* $\mathsf{a} \in \mathcal{C} \cup \mathcal{O}$*" or "for at least one* $\mathsf{a} \in \mathcal{C} \cup \mathcal{O}$*".*

A non-RoI-valuation-defining clause *is defined similarly in terms of non-RoI-semantic elements.*

In the examples of semantic systems presented in subsequent sections, we shall often specify the form of A in a shorter way, leaving it to the reader to rephrase the clause in official structure. If $\psi = \langle \mathsf{A}, \emptyset \rangle$, the part between square brackets will be dropped altogether. Another example is clause C2.3$^{c\neg\wedge 1}$ of the $\mathbf{C_1}$-semantics. It reads "where $B \neq \neg A$, $v_M(\neg(A \wedge B)) = 1$ iff ...", whereas its official structure is "where $\neg(A \wedge B)$ has the form $\langle \neg(A \wedge B), \{\neg(A \wedge \neg A)\} \rangle$, $v_M(\neg(A \wedge B)) = 1$ iff ...".

Where $\mathfrak{A}$ is a semantic statement and m is an instantiation function, define $m^*(\mathfrak{A})$ as the result of replacing every metalinguistic variable μ that occurs in $\mathfrak{A}$, by $m^*(\mu)$, which is defined as follows: (i) if μ occurs in a quantifier "for all $\mu \in \mathcal{C} \cup \mathcal{O}$" or "for at least one $\mu \in \mathcal{C} \cup \mathcal{O}$" or is bound by such a quantifier, then $m^*(\mu) = \mu$, (ii) otherwise $m^*(\mu) = m(\mu)$. $m^*(\mathfrak{A})$ will be called an *instance* of $\mathfrak{A}$.

Definition 4 *An* instance *of the valuation-defining clause*

$$\textit{[Where } \mathsf{A} \textit{ has the form } \psi\textit{,] } v_M(\mathsf{A}) = 1 \textit{ iff } \mathfrak{A}.$$

is a statement

$$v_M(m(\mathsf{A})) = 1 \textit{ iff } m^*(\mathfrak{A}).$$

provided m *is an instantiation function and* $m(\mathsf{A})$ *has the form* ψ*.*

Let $^{++}\mathcal{W}$ comprise all members of $^{+}\mathcal{W}$ together with the formulas that result from replacing in a member of $^{+}\mathcal{W}$ one or more members of $\mathcal{C} \cup \mathcal{O}$ by metavariables for individual constants. Let the *form* of the result be identical to the form of the formula from which it is obtained.

Definition 5 *A recursive set* Ψ *is a* complete set of logical forms *for* $^{+}\mathcal{L}$ *iff* $\bigcup\{A \mid A \in {}^{++}\mathcal{W} \text{ has the form } \psi; \psi \in \Psi\} = {}^{++}\mathcal{W}$ *and no formula of a form* $\psi_1 \in \Psi$ *has also a different form* $\psi_2 \in \Psi$*.*

Note that, in the following definition, α is an arbitrary variable for individual constants and a is an arbitrary metametavariable for those.

Definition 6 *A* regular complexity function *for* $^{+}\mathcal{L}$ *is a function* $c\colon {}^{++}\mathcal{W} \to \mathbb{N}$ *such that, if* $B(\xi) \in {}^{+}\mathcal{F}$*, then* $c(B(\mathsf{a})) = c(B(\alpha))$*.*

We shall say that a semantics is *complex* iff a pre-valuation function v_M is defined in terms of the assignment function v and the valuation function V_M is defined by $V_M(A) = v_M(\phi(A))$, in which $\phi\colon \mathcal{W} \to \mathcal{W}$ is a computable function. A semantics is *simple* if the valuation value of a formula coincides with its pre-valuation value. For the sake of uniformity, we shall then say that $V_M(A) = v_M(A)$. A special common case is that equivalence classes are defined by a (recursive) partition of all closed formulas and that all members of an equivalence class receive the same valuation value V_M in a model M. To realize this, let $s[\![A]\!]$ select an element from the equivalence class $[\![A]\!]$ and define $V_M(A) = v_M(s[\![A]\!])$.

Definition 7 *A semantics for a logic* **L** *with language* $\mathcal{L}$ *is* nice *iff (i) it has models in the sense of Definition 1, (ii) there is a complete set of logical forms* Ψ *for* ${}^{+}\mathcal{L}$ *such that, for every* $\psi \in \Psi$*, the semantics has a unique valuation-defining clause*

$$\textit{[Where } \mathsf{A} \textit{ has the form } \psi\textit{,]}\ v_M(\mathsf{A}) = 1 \textit{ iff } \mathfrak{A}. \tag{1}$$

(iii) all the semantics' valuation-defining clauses are RoI or all are non-RoI, (iv) there is a regular complexity function c such that, for every instance

$$v_M(m(\mathsf{A})) = 1 \textit{ iff } m^*(\mathfrak{A}).$$

of every valuation-defining clause (1), $c(B) < c(m(\mathsf{A}))$ *whenever* $B \in {}^{++}\mathcal{W}$ *occurs in* $m^*(\mathfrak{A})$*, and (v) there is a recursive function that maps every formula to the clause that applies to it.*

For all logics with a nice semantics, we shall sometimes write $M \Vdash A$ to express that M verifies A, which is defined as $V_M(A) = 1$; semantic consequence, and validity are defined from there as usual. In this paragraph (and in similar passages later on) we follow the usual convention to let "model" refer to an entity comprising a model $M = \langle D, v\rangle$ in the strict sense *plus* the valuation function for the specific logic, which is here **CL**.

A *transparent semantic statement* is compounded from instances of semantic base elements by the connectives "(... and ...)", "(... or ...)", and "not ..." and by restricted quantifiers of the form "for all $\alpha \in \mathcal{C} \cup \mathcal{O}$" and "for at least one $\alpha \in \mathcal{C} \cup \mathcal{O}$". A *reduction statement* is a statement of the form "$v_M(A) = 1$ iff $\mathfrak{A}$", in which $\mathfrak{A}$ is a transparent semantic statement.[3]

Lemma 1 *In a nice semantics for* **L***, a reduction statement "*$v_M(A) = 1$ *iff* $\mathfrak{A}$*" holds for every* $A \in {}^{+}\mathcal{W}$*, and there is an algorithm for constructing it.*

Proof. Let the semantics be nice with respect to the complete set of logical forms Ψ and let c be a regular complexity function suitable for the nice semantics. We prove by an induction on $d(A) = c(A) - \min\{c(B) \mid B \in {}^{+}\mathcal{W}\}$ that a reduction statement "$v_M(A) = 1$ iff $\mathfrak{A}$" holds for every $A \in {}^{++}\mathcal{W}$. We also show the way in which the reduction statement is constructed.

For the basis, let $d(A) = 0$ and let $\psi \in \Psi$ be the form of A. As there is no $B \in {}^{++}\mathcal{W}$ for which $c(B) < c(A)$, the clause for ψ cannot contain semantic

[3]Non-logical symbols of the object language occur in this occurrence of $\mathfrak{A}$ (and in future occurrences of similar expressions) and this was not the case for former occurrences. The context disambiguates everywhere.

non-base elements in view of Definition 7. So the instance "$v_M(A) = 1$ iff $\mathfrak{A}$" of this clause is a reduction statement.

For the induction step, suppose that there is a reduction statement "$v_M(B) = 1$ iff $\mathfrak{B}$" for all $B \in {}^{++}\mathcal{W}$ for which $d(B) < n$. Consider an A for which $d(A) = n$ and let $\psi \in \Psi$ be the form of A. Consider the instance "$v_M(A) = 1$ iff $\mathfrak{A}'$" of the clause for ψ. In view of Definition 7, $d(B) < n$ for every $v_M(B) = 1$ that occurs in $\mathfrak{A}'$. So, for every such $v_M(B) = 1$, there is a reduction statement "$v_M(B) = 1$ iff $\mathfrak{B}$" in view of the induction hypothesis. Replacing in "$v_M(A) = 1$ iff $\mathfrak{A}'$", for every $B \in {}^{++}\mathcal{W}$, every $v_M(B) = 1$ by $\mathfrak{B}$ one obtains a reduction statement "$v_M(A) = 1$ iff $\mathfrak{A}$". ■

Corollary 1 *In every nice semantics, $v_M(A)$ is, for every A, a function of the model M.*

Corollary 2 *In a nice semantics for* **L***, whether simple or complex, a reduction statement "$V_M(A) = 1$ iff $\mathfrak{A}$" holds for every closed formula A, and there is an algorithm for constructing it.*

Obviously, there may be several $\mathfrak{A}$ for which "$V_M(A) = 1$ iff $\mathfrak{A}$" is a reduction statement. If that is so, we shall take one such $\mathfrak{A}$ to be *selected*—$\mathfrak{A}$ will then be called the selected transparent statement.

4 Classical Logic and Its Basic Fragments

We begin with a nice semantics for **CL**.[4] Its models are as defined at the outset of the previous section (here for the language ${}^{+}\mathcal{L}_s$). The valuation function $v_M \colon {}^{+}\mathcal{W} \to \{0, 1\}$, determined by M, is defined by:

C2.1 $v_M(\pi^r \alpha_1 \ldots \alpha_r) = 1$ iff $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)$ $(r \geq 0)$[5]
C2.2 $v_M(\alpha = \beta) = 1$ iff $v(\alpha) = v(\beta)$
C2.3 $v_M(\neg A) = 1$ iff $v_M(A) = 0$
C2.4 $v_M(A \supset B) = 1$ iff $v_M(A) = 0$ or $v_M(B) = 1$
C2.5 $v_M(A \wedge B) = 1$ iff $v_M(A) = 1$ and $v_M(B) = 1$
C2.6 $v_M(A \vee B) = 1$ iff $v_M(A) = 1$ or $v_M(B) = 1$
C2.7 $v_M(A \equiv B) = 1$ iff $v_M(A) = v_M(B)$
C2.8 $v_M(\forall \xi A(\xi)) = 1$ iff $v_M(A(\alpha)) = 1$ for all $\alpha \in \mathcal{C} \cup \mathcal{O}$
C2.9 $v_M(\exists \xi A(\xi)) = 1$ iff $v_M(A(\alpha)) = 1$ for at least one $\alpha \in \mathcal{C} \cup \mathcal{O}$

For all $A \in \mathcal{W}$, $M \Vdash A$ iff $v_M(A) = 1$.

In order to extend the semantics to $\mathcal{L}_\sharp$, replace $\mathcal{Z}^r$ by $\mathcal{Z}^r_\sharp$ and ${}^{+}\mathcal{W}$ by ${}^{+}\mathcal{W}_\sharp$ in the definition of the assignment. This version will be used for the embedding.

Each of C2.1–9 specifies the valuation values of all formulas of a certain logical form. Let us call these nine logical forms the *simple logical forms*.

The basic fragments are obtained by removing one or both directions of the equivalences in the clauses C2.1–9. Thus, by removing "if $v_M(\neg A) = 1$, then $v_M(A) = 0$" some models will display negation gluts, by removing "$v_M(\neg A) = 1$ if $v_M(A) = 0$" some models will display negation gaps, and by removing both,

[4] Materials for this section are taken from [6].

[5] As stipulated in Section 2, $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle = \emptyset$ if $r = 0$. So $v_M(\pi^0) = 1$ iff $v(\pi^0) = \{\emptyset\}$.

some models will display both negation gluts and negation gaps.[6] Similarly, by removing "If $v_M(A \wedge B) = 1$, then $v_M(A) = 1$ and $v_M(B) = 1$" some models will display conjunction gluts and by removing "$v_M(\exists\xi A(\xi)) = 1$ if $v_M(A(\alpha)) = 1$ for at least one $\alpha \in \mathcal{C} \cup \mathcal{O}$" some models will display existential gaps. A semantics that allows for predicative gluts or gaps is obtained by removing one or both directions of C2.1.

The resulting semantic systems are indeterministic: the valuation values of formulas are not functions of the assignment values of their components. We shall devise equivalent nice (and hence deterministic) semantics, but first point to another peculiarity.

If some models of a logic **L** display gluts or gaps, RoI does not hold in **L**. In view of C2.1 and C2.2, $v_M(a = b) = 1$ warrants that $v_M(Pa) = v_M(Pb)$. But if there is, for example, a negation glut or gap, $v_M(a = b) = 1$ does not warrant that $v_M(\neg Pa) = v_M(\neg Pb)$. For some purposes, however, one will want to combine gluts or gaps with RoI. It is indeed possible to do so, as we now shall show.

An obvious example concerns gluts and gaps for negation. The six basic fragments handle negation gluts, negation gaps, or both negation gluts and gaps respectively. RoI does not hold in the first three logics, but holds in the last three (that have identity in the superscript). The nice semantics of the six logics is obtained from the above **CL**-semantics for $\mathcal{L}$ by replacing C2.3 according to the following table:

CL	**CLoN**	**CLuN**	**CLaN**	**CLoN**$^=$	**CLuN**$^=$	**CLaN**$^=$
C2.3	C2.3^o	C2.3^u	C2.3^a	C2.3$^{o=}$	C2.3$^{u=}$	C2.3$^{a=}$

The replacing clauses are:

C2.3^o $v_M(\neg A) = 1$ iff $v(\neg A) = \{\emptyset\}$
C2.3^u $v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $v(\neg A) = \{\emptyset\}$
C2.3^a $v_M(\neg A) = 1$ iff $v_M(A) = 0$ and $v(\neg A) = \{\emptyset\}$
C2.3$^{o=}$ $v_M(\neg A) = 1$ iff $\langle v(g_1(\neg A)), \ldots, v(g_{h(\neg A)}(\neg A))\rangle \in v(f(\neg A))$
C2.3$^{u=}$ $v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $\langle v(g_1(\neg A)), \ldots, v(g_{h(\neg A)}(\neg A))\rangle \in v(f(\neg A))$
C2.3$^{a=}$ $v_M(\neg A) = 1$ iff $v_M(A) = 0$ and $\langle v(g_1(\neg A)), \ldots, v(g_{h(\neg A)}(\neg A))\rangle \in v(f(\neg A))$

Other gluts and gaps are handled similarly. Suppose that one wants to allow for gluts or gaps with respect to some logical symbol. In the above **CL**-semantics, the symbol is characterized by a simple form A and the clause for it reads "$v_M(\mathsf{A}) = 1$ iff Z" for some Z. In this clause, one replaces the expression "Z" by "Z or Y" to allow for gluts, by "Z and Y" to allow for gaps, and by "Y" to allow for both. In these expressions, Y is either $v(\mathsf{A}) = \{\emptyset\}$, in which case RoI is invalidated, or $\langle v(g_1(\mathsf{A})), \ldots, v(g_{h(\mathsf{A})}(\mathsf{A}))\rangle \in v(f(\mathsf{A}))$ in which case RoI is validated. Consider the clause for the universal quantifier as an example. Only gluts are allowed by the clause

$$v_M(\forall\xi A(\xi)) = 1 \text{ iff } v_M(A(\alpha)) = 1 \text{ for all } \alpha \in \mathcal{C} \cup \mathcal{O} \text{ or } v(\forall\xi A(\xi)) = \{\emptyset\}\,,$$

which invalidates RoI. Both gluts and gaps are allowed by the clause

[6] The resulting logics are called **CLuN** (for example in [4]), **CLaN** and **CLoN** respectively—they are like **CL** except in that they allow for, respectively, gl*u*ts, g*a*ps, and b*o*th gluts and gaps with respect to *n*egation.

$$v_M(\forall\xi A(\xi)) = 1 \text{ iff } \\ \langle v(g_1(\forall\xi A(\xi))), \ldots, v(g_{h(\forall\xi A(\xi))}(\forall\xi A(\xi)))\rangle \in v(f(\forall\xi A(\xi)))\,,$$

which makes sure that RoI is validated.

Some special cases deserve a comment. The first case concerns predicative gluts or gaps. Consider the RoI variant of what the clause for predicative gluts would be:

$$v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1 \text{ iff } \langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r) \text{ or} \\ \langle v(g_1(\pi^r\alpha_1 \ldots \alpha_r)), \ldots, v(g_{h(\pi^r\alpha_1\ldots\alpha_r)}(\pi^r\alpha_1 \ldots \alpha_r))\rangle \in v(f(\pi^r\alpha_1 \ldots \alpha_r))$$

which is equivalent to

$$v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1 \text{ iff } \langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r) \text{ or} \\ \langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(f(\pi^r\alpha_1 \ldots \alpha_r)).$$

In Section 2 we have identified $v(\pi^r)$ with $v(f(\pi^r\alpha_1 \ldots \alpha_r))$, but suppose we did not do so. Where $M = \langle D, v\rangle$, there obviously is a model $M' = \langle D, v'\rangle$ that is exactly as M except that $v'(\pi^r) = v(\pi^r) \cup v(f(\pi^r\alpha_1 \ldots \alpha_r))$ and in which there are no predicative gluts in that the corresponding clause there reads:

$$v_{M'}(\pi^r\alpha_1 \ldots \alpha_r) = 1 \text{ iff } \langle v'(\alpha_1), \ldots, v'(\alpha_r)\rangle \in v'(\pi^r)\,.$$

It is easily seen that $v_{M'}(A) = v_M(A)$ for all $A \in {}^{+}\mathcal{W}$.

So the semantics is equivalent to (defines the same consequence relation as) a simpler semantics. As this simpler semantics does not introduce predicative gluts, it follows at once that the original semantics does not introduce any identity gluts that show at the level of the consequence relation. By the same reasoning, one immediately sees that predicative gaps, either by themselves or combined with predicative gluts, are a useless complication if the semantics follows the RoI schema. So there is no harm in identifying $v(\pi^r)$ with $v(f(\pi^r\alpha_1 \ldots \alpha_r))$ as we did.

If the logic does not follow the RoI schema, predicative gluts and gaps do have effect. Consider a semantics that is exactly like that for **CL** except that clause C2.1 is modified in order to allow for gluts and/or gaps. It is easily seen that RoI is not valid on this semantics.

The second special case is identity. As we are not interested here in the study of the basic logics themselves, two comments are sufficient. First, the RoI variant of the clause for identity gluts, which reads

$$v_M(\alpha = \beta) = 1 \text{ iff } v(\alpha) = v(\beta) \text{ or } \langle v(\alpha), v(\beta)\rangle \in v(\cdot = \cdot)\,,$$

obviously does not warrant the validity of RoI. Indeed, it allows for models in which $v(a) \neq v(b)$, $\langle v(a), v(b)\rangle \in v(\cdot = \cdot)$, $v(a) \in v(P)$, $v(b) \notin v(P)$, and hence $v_M(a = b) = v_M(Pa) = 1$ and $v_M(Pb) = 0$. Similarly for the RoI variant of the clause that allows for both identity gluts and gaps. The resulting logics have a semantics that follows the non-RoI schema.

We shall show that all basic fragments of **CL** can be embedded in **CL**. The same holds for certain extensions and fragments of them, which we discuss in the next section.

5 Other Logics that Have a Nice Semantics

An extension of a logic **L** may be defined in terms of axiom schemata. If one adds to the semantics of **L** a clause $v_M(\mathsf{A}) = 1$ for every new axiom schema A, the result will not be sensible because the new clauses may (and for some models will) contradict one of the original clauses for **L**. This, however may sometimes be repaired by first considering the original clause as a default (which is overruled by the new clauses) and next turning the semantics into a consistent and recursive one.

As a simple example, consider the extension of **CLoN** with the axiom schema $\neg\neg A \supset A$. The new semantic clause is $v_M(\neg\neg A \supset A) = 1$, which is contextually equivalent to "$v_M(\neg\neg A) = 0$ if $v_M(A) = 0$." It readily turns out that C2.3^o should be replaced by C2.3$'$ and C2.3$''$:

C2.3$'$ if A is not of the form $\neg B$, then $v_M(\neg A) = 1$ iff $v(\neg A) = \{\emptyset\}$
C2.3$''$ $v_M(\neg\neg A) = 1$ iff $v_M(A) = 1$ and $v(\neg\neg A) = \{\emptyset\}$

5.1 Some Maximal Fragments of CL

Two sets of logics between **CL** and those listed in the table in Section 4 will be considered.[7]

The first six will be called Schütte logics because their propositional fragments were first presented in [20]—their names are formed by appending a "s" to the systems they extend. The nice semantics for these systems is obtained from the **CL**-semantics of Section 4 by adding C2.3$^{\neg\neg}$–C2.3$^{\neg\exists}$ and by replacing C2.3 according to the following table:

CL	**CLoNs**	**CLuNs**	**CLaNs**	**CLoNs**$^=$	**CLuNs**$^=$	**CLaNs**$^=$
C2.3	C2.3^{op}	C2.3^{up}	C2.3^{ap}	C2.3$^{o=p}$	C2.3$^{u=p}$	C2.3$^{a=p}$

Here are the clauses:

C2.3^{op} If $A \in {}^+\mathbb{P}^=$, $v_M(\neg A) = 1$ iff $v(\neg A) = \{\emptyset\}$
C2.3^{up} If $A \in {}^+\mathbb{P}^=$, $v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $v(\neg A) = \{\emptyset\}$
C2.3^{ap} If $A \in {}^+\mathbb{P}^=$, $v_M(\neg A) = 1$ iff $v_M(A) = 0$ and $v(\neg A) = \{\emptyset\}$
C2.3$^{o=p}$ If $A \in {}^+\mathbb{P}^=$, $v_M(\neg A) = 1$ iff $\langle v(g_1(\neg A)), \ldots, v(g_{h(\neg A)}(\neg A))\rangle \in v(f(\neg A))$
C2.3$^{u=p}$ If $A \in {}^+\mathbb{P}^=$, $v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $\langle v(g_1(\neg A)), \ldots, v(g_{h(\neg A)}(\neg A))\rangle \in v(f(\neg A))$
C2.3$^{a=p}$ If $A \in {}^+\mathbb{P}^=$, $v_M(\neg A) = 1$ iff $v_M(A) = 0$ and $\langle v(g_1(\neg A)), \ldots, v(g_{h(\neg A)}(\neg A))\rangle \in v(f(\neg A))$
C2.3$^{\neg\neg}$ $v_M(\neg\neg A) = v_M(A)$
C2.3$^{\neg\supset}$ $v_M(\neg(A \supset B)) = v_M(A \wedge \neg B)$
C2.3$^{\neg\vee}$ $v_M(\neg(A \wedge B)) = v_M(\neg A \vee \neg B)$
C2.3$^{\neg\wedge}$ $v_M(\neg(A \vee B)) = v_M(\neg A \wedge \neg B)$
C2.3$^{\neg\equiv}$ $v_M(\neg(A \equiv B)) = v_M((A \vee B) \wedge (\neg A \vee \neg B))$
C2.3$^{\neg\forall}$ $v_M(\neg\forall\xi A(\xi)) = v_M(\exists\xi\neg A(\xi))$
C2.3$^{\neg\exists}$ $v_M(\neg\exists\xi A(\xi)) = v_M(\forall\xi\neg A(\xi))$

[7]All logics considered in this section have a characteristic three-valued semantics and their propositional fragments are maximally paraconsistent—see [3].

These six systems ‘drive’ negations of complex formulas ‘inwards’. On six other systems negations behave classically in front of complex formulas. The logics are called **CLoNv**, **CLuNv**, **CLaNv**, **CLoNv**$^{=}$, **CLuNv**$^{=}$, and **CLaNv**$^{=}$—the “v” refers to Arruda’s so-called Vasil’ev system from [2], which is the propositional fragment of **CLuNv** and **CLuNv**$^{=}$.

The semantics of these logics is the same as that of the corresponding Schütte logic, except that C2.3^{v} is added instead of C2.3$^{\neg\neg}$–C2.3$^{\neg\exists}$:

C2.3^{v} where $A \in {}^{+}\mathcal{W} - {}^{+}\mathbb{P}^{=}$, $v_M(\neg A) = 1$ iff $v_M(A) = 0$,

5.2 Linguistic Extensions and Fragments

Several logics are fragments of the aforementioned ones, obtained by removing some logical symbols from the language. Their semantics is obtained by selecting the relevant valuation clauses from the logics of which they are fragments. Examples are **LP** from [18] (obtained from **CLuNs**$^{=}$ by removing $\supset$ from the language), the predicative version of $\mathbf{SK_3}$ from [16], etc.

Other logics are obtained by adding logical symbols that are definable in **CL** to logics from the previous paragraph. Typical examples are logics extended with the missing classical connectives. Thus, if a logic handles negation gluts or gaps, the language may be extended with classical negation, say $\sim$. If it handles conjunction gaps or gluts, the language may be extended with classical conjunction, say $\sqcap$. The advantage of this linguistic extension is that it often greatly simplifies the metatheory.

The easiest way to handle linguistic extensions is to extend the language $\mathcal{L}$ and the pseudo-language ${}^{+}\mathcal{L}$ with the new symbol, and to extend the **CL**-semantics with an appropriate clause for the new symbol. In the case of added classical symbols, this clause will duplicate that for the original symbol (except for the single occurrence of the new symbol itself).

5.3 Other Roads to Gluts and Gaps

Many more logics than the ones described in this paper have a nice semantics and can be embedded in **CL** by the method described below.

Consider the result of replacing, in the **CL**-semantics from Section 4, C2.1 and C2.2 by

C2.1^{o} $v_M(\pi^r\alpha_1 \ldots \alpha_r) = \{\emptyset\}$ iff $v(\pi^r\alpha_1 \ldots \alpha_r) = 1$ $(r \geq 0)$
C2.2^{o} $v_M(\alpha = \beta) = 1$ iff $v(\alpha = \beta) = \{\emptyset\}$

and C2.3 by C2.3$^{\neg\neg}$–C2.3$^{\neg\exists}$ together with

C2.3$^{\neg p}$ $v_M(\neg\pi^r\alpha_1 \ldots \alpha_r) = 1$ iff $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \notin v(\pi^r)$ $(r \geq 0)$
C2.3$^{\neg =}$ $v_M(\neg\alpha = \beta) = 1$ iff $v(\alpha) \neq v(\beta)$

Suppose moreover that classical negation, $\sim$, is added to the language and correctly defined within the semantics—see the previous subsection.

The resulting logic allows for predicative gluts and gaps, for identity gluts and gaps, but also for negation gluts and gaps. The logic is not equivalent to any of the logics considered before, even if these are extended with classical negation. Indeed, unlike all previously considered logics, the present logic validates “If $\Gamma \vdash {\sim}\neg a = b$, then $\Gamma \vdash \neg A(a) \equiv \neg A(b)$.”

Our proofs in Section 6 can handle gluts and gaps for different simple forms, provided all of them are RoI variants—we then say that the logic *follows the* RoI *schema*—or all of them are non-RoI variants—we then say that the logic *follows the non*-RoI *schema*.

5.4 A nice semantics for $\mathbf{C_1}$

A set of logics that have a nice semantics are the well-known $\mathbf{C}_n$-systems ($n \in \mathbb{N}$) from [11], further studied in [12] and many other papers—$\mathbf{C}_0$ is **CL**.

The nice semantics for $\mathbf{C_1}$ is like the one for **CL**, apart from the fact that v_M is not equal to V_M for $\mathbf{C_1}$ and that clause C2.3 must be replaced by clauses C2.3$^{c\neg p}$ to C2.3$^{c\neg\exists}$.

C3 $V_M(A) = v_M(T(A))$, where $T(A)$ is the result of first deleting all the vacuous quantifiers in A and then uniformly replacing all variables by the first variables of the alphabet in alphabetic order.
C2.3$^{c\neg p}$ Where $A \in \mathbb{P}^{=}$: $v_M(\neg A) = 1$ iff $v(A) = 0$ or $v(\neg A) = \{\emptyset\}$
C2.3$^{c\neg\neg}$ $v_M(\neg\neg A) = 1$ iff $v_M(\neg A) = 0$ or $(v_M(\neg A) = v_M(A) = 1$ and $v(\neg\neg A) = \{\emptyset\})$
C2.3$^{c\neg\supset}$ $v_M(\neg(A \supset B)) = 1$ iff $v_M(A \supset B) = 0$ or $((v_M(\neg A) = v_M(A) = 1$ or $v_M(\neg B) = v_M(B) = 1)$ and $v(\neg(A \supset B)) = \{\emptyset\})$
C2.3$^{c\neg\vee}$ $v_M(\neg(A \vee B)) = 1$ iff $v_M(A \vee B) = 0$ or $((v_M(\neg B) = v_M(B) = 1$ or $v_M(\neg C) = v_M(C) = 1)$ and $v(\neg(A \vee B)) = \{\emptyset\})$
C2.3$^{c\neg\wedge 1}$ where $B \neq \neg A$: $v_M(\neg(A \wedge B)) = 1$ iff $v_M(A \wedge B) = 0$ or $((v_M(\neg A) = v_M(A) = 1$ or $v_M(\neg B) = v_M(B) = 1)$ and $v(\neg(A \wedge B)) = \{\emptyset\})$
C2.3$^{c\neg\wedge 2}$ $v_M(\neg(A \wedge \neg A)) = 1$ iff $v_M(\neg A) \neq v_M(A)$
C2.3$^{c\neg\forall}$ $v_M(\neg\forall\alpha A(\alpha)) = 1$ iff $v_M(\forall\alpha A(\alpha)) = 0$ or $(v_M(\neg A(\beta)) = v_M(A(\beta)) = 1$ for at least one $\beta \in \mathcal{C} \cup \mathcal{O}$ and $v(\neg\forall\alpha A(\alpha)) = \{\emptyset\})$
C2.3$^{c\neg\exists}$ $v_M(\neg\exists\alpha A(\alpha)) = 1$ iff $v_M(\exists\alpha A(\alpha)) = 0$ or $(v_M(\neg A(\beta)) = v_M(A(\beta)) = 1$ for at least one $\beta \in \mathcal{C} \cup \mathcal{O}$ and $v(\neg\exists\alpha A(\alpha)) = \{\emptyset\})$

Adjusting the semantics to any logic $\mathbf{C}_n$ ($n \in \mathbb{N}$) is straightforward.

5.5 A nice semantics for AN

The logic **AN** was presented in [17] by means of an elegant three-valued semantics. Its peculiarity is that paraconsistency is realized by weakening disjunction. **AN** validates all 'analysing rules' at the expense of giving up some 'constructive rules'.

The nice semantics for **AN** is like the one for **CL**, except that v_M is not equal to V_M, that the clauses for negation and disjunction are replaced as shown below, and that the clauses for implication and equivalence are removed—they are useless in view if C3.

C3 $V_M(A) = v_M(B)$, where B is the prenex conjunctive normal form of A—see, for example [9].
C2.3^{up} If $A \in {}^{+}\mathbb{P}^{=}$, $v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $v(\neg A) = \{\emptyset\}$
C2.3$^{\neg\neg}$ $v_M(\neg\neg A) = v_M(A)$
C2.3$^{\neg\vee}$ $v_M(\neg(A \wedge B)) = v_M(\neg A \vee \neg B)$
C2.3$^{\neg\wedge}$ $v_M(\neg(A \vee B)) = v_M(\neg A \wedge \neg B)$
C2.3$^{\neg\forall}$ $v_M(\neg\forall\xi A(\xi)) = v_M(\exists\xi\neg A(\xi))$

C2.3$^{\neg\exists}$ $v_M(\neg\exists\xi A(\xi)) = v_M(\forall\xi\neg A(\xi))$
C2.6$^{a\vee}$ $v_M(A \vee B) = 1$ iff ($v_M(A) = 1$ and $v_M(\neg A) = 0$) or ($v_M(B) = 1$ and $v_M(\neg B) = 0$) or ($v_M(A) = v_M(B) = 1$)

5.6 A nice semantics for Łukasiewicz's m-valued logic Ł$_m$

The logical symbols of the language of Ł$_m$ ($m > 2$) are $\supset$, $\neg$ and $\forall$. The language is denoted by $\mathcal{L}_{\text{Ł}}$, its set of well formed formulas by $\mathcal{W}_{\text{Ł}}$, and their extensions with pseudo-constants respectively by $\mathcal{L}^+_{\text{Ł}}$ and $\mathcal{W}^+_{\text{Ł}}$.

Let $N_m = \{i \in \mathbb{N} \mid 1 \leq i \leq m\}$. An m-valued semantics (with truth values in N_m, of which 1 is the only designated value) for Ł$_m$ is defined by considering models $L = \langle D, v\rangle$ where the domain D is a non-empty set and the assignment v is a function $v : \mathcal{C} \cup \mathcal{O} \to D$ and $v : \mathcal{P}^r \to (D^{(r)} \to N_m)$.

Such a model $L = \langle D, v\rangle$ determines the valuation function $v_L : \mathcal{W}^+_{\text{Ł}} \to N_m$, defined by the following clauses: $v_L(\pi^r\alpha_1 \dots \alpha_r) = (v(\pi^r))(\langle v(\alpha_1), \dots, v(\alpha_r)\rangle)$, $v_L(A \supset B) = max(1, 1 + v_L(B) - v_L(A))$, $v_L(\neg A) = m - v_L(A) + 1$ and $v_L(\forall\xi A(\xi)) = max\{v_L(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$. Truth in a model L is defined by $L \Vdash A$ iff $v_L(A) = 1$.

In order to transform this many valued semantics into a nice semantics we proceed by making use of specific symbols which are definable in Ł$_m$. As was established in [19], it is possible to define a set of symbols $\{I_k \mid k \in N_m\}$ within Ł$_m$ such that

$$v_L(I_k(A)) = \begin{cases} 1 & v_L(A) = k \\ m & \text{otherwise.} \end{cases} \qquad (*)$$

Although the precise way in which these symbols are defined is of no importance to the semantics we are about to present (we are only interested in their property (*)), we mention the definitions for the sake of completeness. To avoid clutter, we first define some other symbols as is usual in the literature.

D1 $A \& B =_{df} \neg(A \supset \neg B)$
D2 $A \wedge B =_{df} A\&(A \supset B)$
D3 $A \vee B =_{df} ((A \supset B) \supset B)\&((B \supset A) \supset A)$
D4 $A^i =_{df} \overbrace{A\&(A\&(\dots(A\&A)\dots)}^{i \text{ times}}$

Let $f_m(k)$ denote the least integer $n \geq \frac{m-k}{k-1}$.

D5 Define $I_k(A)$ recursively by:
(i) $I_1(A) = A^{m-1}$,
(ii) $I_m(A) = (\neg A)^{m-1}$,
(iii) if $1 < k < m$ and $k \leq \max(1, m - ((k-1) \times f_m(k)))$, then $I_k(A) = ((\neg A^{f_m(k)} \vee A) \supset (\neg A^{f_m(k)} \wedge A))^{m-1}$, and
(iv) if $1 < k < m$ and $k > \max(1, m - ((k-1) \times f_m(k)))$, then $I_k(A) = I_{f_m(k)}(\neg(A^{f_m(k)}))$.

The clauses for the valuation function of the nice semantics, which is two-valued, are the following.

C2.1^k Where $k \in N_m - \{m\}$, $v_M(I_k(\pi^r\alpha_1 \dots \alpha_r)) = 1$ iff $\langle v(\alpha_1), \dots, v(\alpha_r)\rangle \in v(f(I_k(\pi^r\alpha_1 \dots \alpha_r)))$ and for every $i < k$, $v_M(I_i(\pi^r\alpha_1 \dots \alpha_r)) = 0$ ($r \geq 0$)

C2.1^m $v_M(I_m(\pi^r\alpha_1\dots\alpha_r)) = 1$ iff for every $i < m$, $v_M(I_i(\pi^r\alpha_1\dots\alpha_r)) = 0$ $(r \geq 0)$
C2.3$^{Ł\neg}$ $v_M(I_k(\neg A) = 1$ iff $v_M(I_{m+1-k}(A)) = 1$
C2.4$^{Ł\supset}$ $v_M(I_k(A \supset B)) = 1$ iff $v_M(I_i(A)) = 1$ and $v_M(I_j(B)) = 1$ for some $i, j \in N_m$ such that $k = max(1, 1 + j - i)$
C2.8$^{Ł\forall}$ $v_M(I_k(\forall\alpha A(\alpha))) = 1$ iff, for all $\beta \in \mathcal{C}\cup\mathcal{O}$, there is an $l \leq k$ such that $v_M(I_l A(\beta)) = 1$ and, for at least one $\beta \in \mathcal{C}\cup\mathcal{O}$, $v_M(I_k A(\beta)) = 1$
C2.10 where A is not of the form $I_k(B)$, $v_M(A) = v_M(I_1(A))$

For each specific choice of m, the clauses C2.1^k–C2.10 define the valuation as required for a nice semantics. The matter is most easily understood by considering the formulas that contain only the primitive symbols. Some of these correspond by the definitions to one of the $m \times 4$ forms handled by C2.1^k–C2.8$^{Ł\forall}$. All other formulas belong to the residual category handled by C2.10.

We better also present a suitable regular complexity function. In order to do this in a transparent way we first define an auxiliary language $\mathcal{L}_D$, with symbols $\supset$, $\neg$, $\forall$ and I_k, for each, $k \in N_m$. The set $\mathcal{W}_D$ of formulas of $\mathcal{L}_D$ is the set $\{I_k(A) \mid k \in N_m; A \in \mathcal{W}_Ł\}$. We recursively define a complexity function $c' : \mathcal{W}_D \to \mathbb{N}$ for this language.

CoFn1 $c'(I_1(A)) = 1$ where A does not contain logical symbols
CoFn2 $c'(I_1(A \supset B)) = c'(I_1(A)) + c'(I_1(B)) + 2m + 1$
CoFn3 $c'(I_1(\neg A)) = c'(I_1(A)) + m + 1$
CoFn4 $c'(I_1(\forall x A(x))) = c'(I_1(A(x))) + m + 1$
CoFn5 $c'(I_k(A)) = c'(I_1(A)) + k - 1$

We define a function $red : \mathcal{W}_Ł \to \mathcal{W}_D$ in order to obtain a complexity function for members of $\mathcal{W}_Ł$ from c'. Consider again the members of $\mathcal{W}_Ł$ that contain only primitive symbols. If $A \in \mathcal{W}_Ł$ is of the form of the definiens of $I_k(B)$ for some $k \in N_m$, $red(A) = I_k(B)$—remember that I_k is a symbol of $\mathcal{W}_D$. Otherwise, let $red(A) = I_1(A)$. This function is complete and well defined. We define a complexity function $c : \mathcal{W}_Ł \to \mathbb{N}$ for $\mathcal{W}_Ł$ by $c(A) = c'(red(A)) + 1$.

Unlike the other nice semantic systems defined in this paper, it is not completely obvious why this one is adequate. This requires some explanation.

To every Ł$_m$-model $L = \langle D, v\rangle$ corresponds a nice semantics model $M = \langle D', v'\rangle$, as is warranted by the following procedure. Let $D' = D$ and $v'(\alpha) = v(\alpha)$ for all $\alpha \in \mathcal{C}\cup\mathcal{O}$. Now let $v'(I_k(\pi^r \cdot \dots \cdot))$ for each $k < m$ be the set $\{\langle\alpha_1, \dots \alpha_r\rangle \in D^{(r)} \mid (v(\pi^r))(\langle\alpha_1, \dots \alpha_r\rangle) = k\}$. For all other cases v' can get arbitrary values.

The following procedure transforms every nice semantics model $M = \langle D', v'\rangle$ into an Ł$_m$-model $L = \langle D, v\rangle$. Let $D = D'$ and $v(\alpha) = v'(\alpha)$ for all $\alpha \in \mathcal{C}\cup\mathcal{O}$. Now let $v(\pi^r)$ be the function such that

(1) for every $k < m$: $(v(\pi^r))(\langle\alpha_1, \dots \alpha_r\rangle) = k$ iff

(1.a) $\langle\alpha_1, \dots, \alpha_r\rangle \in v(I_k(\pi^r \cdot \dots \cdot)))$ and

(1.b) for every $i < k$, $\langle\alpha_1, \dots, \alpha_r\rangle \notin v(I_i(\pi^r \cdot \dots \cdot)))$ and

(2) $(v(\pi^r))(\langle\alpha_1, \dots \alpha_r\rangle) = m$ iff $\langle\alpha_1, \dots, \alpha_r\rangle \notin v(f(I_i(\pi^r\alpha_1 \dots \alpha_r)))$, for every $i < m$.

An original Ł_m-model L *corresponds to* a nice semantics Ł_m-model M iff M is the result of transforming L according to the first procedure defined above or L is the result of transforming M according to the second. The transformation procedures are defined in such a way that if L corresponds to M, we have that

$$v_M(I_k(A)) = 1 \text{ iff } v_L(A) = k \qquad (^{**})$$

and therefore also $v_M(I_1(A)) = 1$ iff $v_L(A) = 1$, or, by C2.10, $v_M(A) = 1$ iff $v_L(A) = 1$, whence $M \Vdash A$ iff $L \Vdash A$. (**) can be proven by a straightforward induction on the (ordinary) complexity of formulas in view of the fact that clauses C2.3$^{\text{Ł}\neg}$, C2.4$^{\text{Ł}\supset}$, and C2.8$^{\text{Ł}\forall}$ are precisely formulated in such a way that they correspond to the clauses of the original semantics, the main difference being that the numerical operations are replaced from the truth values to the subscripts k of the defined symbols of the form I_k.

The reader can easily verify that the transformation procedures construct a corresponding model of the one type for every model of the other type. This fact together with fact that $M \Vdash A$ iff $L \Vdash A$ in case M corresponds to L immediately implies that the nice semantics defines the same semantic consequence relation as the original semantics. We have thus proven the adequateness of this nice semantics.

6 The Embedding

Let **L** be a logic that has an adequate nice semantics. In order to show that **L** can be embedded in **CL**, we shall first turn the **L**-semantics into a NE-function (nice embedding function) tr which maps formulas (and sets of formulas) from $\mathcal{W}$ to formulas (and sets of formulas) from $\mathcal{W}_\sharp$, thus taking care of the embedding. We shall distinguish between two cases according as **L** follows the RoI schema or not. The second case is slightly more complicated.

6.1 Logics Following the RoI Schema

Let **L** be a logic that follows the RoI schema and has a nice semantics—so without gluts or gaps for either predicates or identity. We shall prove that, where $\Gamma \subset \mathcal{W}$ and $A \in \mathcal{W}$, $\Gamma \vdash_{\mathbf{L}} A$ iff $\text{tr}(\Gamma) \vdash_{\mathbf{CL}} \text{tr}(A)$, in which $\text{tr}(\Gamma)$ is a finite set whenever Γ is finite.

Definition 8 *Where* **L** *is a logic that follows the RoI schema and has a nice semantics, the NE-function* $\text{tr}\colon \mathcal{W} \to \mathcal{W}_\sharp$ *for* **L** *is defined as:*

$$\text{tr}(A) = \text{TRoI}(\mathfrak{A}),$$

where $\mathfrak{A}$ *is the selected transparent semantic statement for which "*$V_M(A) = 1$ *iff* $\mathfrak{A}$*" is a reduction statement*[8] *and* TRoI *is the function from transparent semantic statements to* $\mathcal{W}_\sharp$*-formulas that is recursively defined in Table 1. The NE-function is extended to sets by* $\text{tr}(\Gamma) = \{\text{tr}(A) \mid A \in \Gamma\}$.

Together with Definition 7, the definition of the NE-function warrants that tr is a total function.

[8] See Corollary 2 and the subsequent paragraph.

(i)	$\mathrm{TRoI}((\mathfrak{A})) = (\mathrm{TRoI}(\mathfrak{A}))$
(ii)	$\mathrm{TRoI}(\text{not } \mathfrak{A}) = \neg\mathrm{TRoI}(\mathfrak{A})$
(iii)	$\mathrm{TRoI}(\mathfrak{A} \text{ and } \mathfrak{B}) = \mathrm{TRoI}(\mathfrak{A}) \wedge \mathrm{TRoI}(\mathfrak{B})$
(iv)	$\mathrm{TRoI}(\mathfrak{A} \text{ or } \mathfrak{B}) = \mathrm{TRoI}(\mathfrak{A}) \vee \mathrm{TRoI}(\mathfrak{B})$
(v)	$\mathrm{TRoI}(\mathfrak{A}(\mathsf{a}) \text{ for all } \mathsf{a} \in \mathcal{C} \cup \mathcal{O}) = \forall\xi\, \mathrm{TRoI}(\mathfrak{A}(\xi))$, where ξ is the first variable that does not occur in $\mathfrak{A}(\mathsf{a})$
(vi)	$\mathrm{TRoI}(\mathfrak{A}(\mathsf{a}) \text{ for at least one } \mathsf{a} \in \mathcal{C} \cup \mathcal{O}) = \exists\xi\, \mathrm{TRoI}(\mathfrak{A}(\xi))$, where ξ is the first variable that does not occur in $\mathfrak{A}(\mathsf{a})$
(vii)	$\mathrm{TRoI}(\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)) = \pi^r\alpha_1 \ldots \alpha_r$
(viii)	$\mathrm{TRoI}(v(\alpha) = v(\beta)) = \alpha = \beta$
(ix)	if $A \notin {}^{+}\mathbb{P}^{=}$, $\mathrm{TRoI}(\langle v(g_1(A)), \ldots, v(g_{h(A)}(A))\rangle \in v(f(A))) = P^{h(A)}_{f(A)} g(A)$
(x)	$\mathrm{TRoI}(0 = 0) = (P^0 \vee \neg P^0)$

Table 1: RoI schema: from the semantics to tr

Let us at once consider a complex example, viz. the NE-function for $\mathbf{CLuNs}^{=}$. The NE-function is recursively defined as follows:[9]

T1	$\mathrm{tr}(\pi^r\alpha_1 \ldots \alpha_r) = \pi^r\alpha_1 \ldots \alpha_r \quad (r \geq 0)$
T2	$\mathrm{tr}(\alpha = \beta) = \alpha = \beta$
T3	$\mathrm{tr}(A \supset B) = \mathrm{tr}(A) \supset \mathrm{tr}(B)$
T4	$\mathrm{tr}(A \wedge B) = \mathrm{tr}(A) \wedge \mathrm{tr}(B)$
T5	$\mathrm{tr}(A \vee B) = \mathrm{tr}(A) \vee \mathrm{tr}(B)$
T6	$\mathrm{tr}(A \equiv B) = \mathrm{tr}(A) \equiv \mathrm{tr}(B)$
T7	$\mathrm{tr}(\forall\xi A) = \forall\xi\, \mathrm{tr}(A)$
T8	$\mathrm{tr}(\exists\xi A) = \exists\xi\, \mathrm{tr}(A)$
$\mathrm{T9}^{u=p}$	If $A \in {}^{+}\mathbb{P}^{=}$, $\mathrm{tr}(\neg A) = \neg\mathrm{tr}(A) \vee P^{h(\neg A)}_{f(\neg A)} g(\neg A)$
$\mathrm{T9}^{s\neg\neg}$	$\mathrm{tr}(\neg\neg A) = \mathrm{tr}(A)$
$\mathrm{T9}^{s\neg\supset}$	$\mathrm{tr}(\neg(A \supset B)) = \mathrm{tr}(A) \wedge \mathrm{tr}(\neg B)$
$\mathrm{T9}^{s\neg\wedge}$	$\mathrm{tr}(\neg(A \wedge B)) = \mathrm{tr}(\neg A) \vee \mathrm{tr}(\neg B)$
$\mathrm{T9}^{s\neg\vee}$	$\mathrm{tr}(\neg(A \vee B) = \mathrm{tr}(\neg A) \wedge \mathrm{tr}(\neg B)$
$\mathrm{T9}^{s\neg\equiv}$	$\mathrm{tr}(\neg(A \equiv B)) = (\mathrm{tr}(A) \vee \mathrm{tr}(B)) \wedge (\mathrm{tr}(\neg A) \vee \mathrm{tr}(\neg B))$
$\mathrm{T9}^{s\neg\forall}$	$\mathrm{tr}(\neg\forall\xi A) = \exists\xi\, \mathrm{tr}(\neg A)$
$\mathrm{T9}^{s\neg\exists}$	$\mathrm{tr}(\neg\exists\xi A) = \forall\xi\, \mathrm{tr}(\neg A)$

Definition 9 *Where $M = \langle D, v\rangle$ is a* **L***-model for ${}^{+}\mathcal{L}$ and $M' = \langle D, v'\rangle$ is a* **CL***-model for ${}^{+}\mathcal{L}_\sharp$, let RMM' iff the following conditions are fulfilled:*
R1 If $\alpha \in \mathcal{C} \cup \mathcal{O}$, then $v'(\alpha) = v(\alpha)$.
R2 If $A \in {}^{+}\mathbb{P}$, then $v'(A) = v(A)$.
R3 If $A \notin {}^{+}\mathbb{P}^{=}$, then $v'(P^{h(A)}_{f(A)}) = v(f(A))$.

Lemma 2 *(i) For every* **L***-model $M = \langle D, v\rangle$ for ${}^{+}\mathcal{L}$ there is a* **CL***-model $M' = \langle D, v'\rangle$ for ${}^{+}\mathcal{L}_\sharp$ such that RMM' and (ii) for every* **CL***-model $M' = \langle D, v'\rangle$ for ${}^{+}\mathcal{L}_\sharp$ there is a* **L***-model $M = \langle D, v\rangle$ for ${}^{+}\mathcal{L}$ such that RMM'.*

Proof. Immediate in view of Definition 9. ■

[9]There is some notational abuse in T7 and T8. However, it is easily seen that the correct formulation of the formula to the right of the identity in T7 is a relettering of $\forall\xi\, \mathrm{tr}(A)$. Analogously for T8.

Lemma 3 *If $M = \langle D, v\rangle$ is a* **L**-*model for* $^+\mathcal{L}$, $M' = \langle D, v'\rangle$ *is a* **CL**-*model for* $^+\mathcal{L}_\sharp$, RMM', *and* $\mathfrak{A}$ *is an instance of a semantic base element, then* $\mathfrak{A}$ *holds true in* M *iff* $v_{M'}(\mathrm{TRoI}(\mathfrak{A})) = 1$.

Proof. Suppose that the antecedent is true. There are four cases.

Case 1: $\mathfrak{A}$ has the form $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)$, whence $\mathrm{TRoI}(\mathfrak{A})$ is $\pi^r\alpha_1 \ldots \alpha_r$. The consequent of the lemma follows in view of R1, R2, and C2.1.

Case 2: $\mathfrak{A}$ has the form $v(\alpha) = v(\beta)$, whence $\mathrm{TRoI}(\mathfrak{A})$ is $\alpha = \beta$. The consequent of the lemma follows in view of R1 and C2.2.

Case 3: $\mathfrak{A}$ has the form $\langle v(g_1(A)), \ldots, v(g_{h(A)}(A))\rangle \in v(f(A))$ and $A \notin {}^+\mathbb{P}^=$, whence $\mathrm{TRoI}(\mathfrak{A})$ is $P^{h(A)}_{f(A)}g(A)$. The consequent of the lemma follows in view of R1, R3, and C2.1.

Case 4: $\mathfrak{A}$ has the form $0 = 0$, whence $\mathrm{TRoI}(\mathfrak{A})$ is $P^0 \vee \neg P^0$. The consequent of the lemma follows in view of C2.6 and C2.3. ■

Lemma 4 If $M = \langle D, v\rangle$ *is a* **L**-*model for* $^+\mathcal{L}$, $M' = \langle D, v'\rangle$ *is a* **CL**-*model for* $^+\mathcal{L}_\sharp$, RMM', *and* $\mathfrak{A}$ *is a transparent semantic statement,* then $\mathfrak{A}$ *holds true in* M *iff* $v_{M'}(\mathrm{TRoI}(\mathfrak{A})) = 1$.

Proof. Suppose $M = \langle D, v\rangle$ is a **L**-model for $^+\mathcal{L}$, $M' = \langle D, v'\rangle$ a **CL**-model for $^+\mathcal{L}_\sharp$ and RMM'. We prove that

$$\mathfrak{A} \text{ holds true in } M \text{ iff } v_{M'}(\mathrm{TRoI}(\mathfrak{A})) = 1 \tag{2}$$

for every transparent semantic statement $\mathfrak{A}$, by means of an induction on the complexity of transparent semantic statements[10].

Base case: $\mathfrak{A}$ is an instance of a semantic base element. So (2) follows by Lemma 3.

For the induction step, suppose that, for every transparent semantic statement $\mathfrak{B}$ that is less complex than $\mathfrak{A}$, $\mathfrak{B}$ holds true in M just in case $v_{M'}(\mathrm{TRoI}(\mathfrak{B})) = 1$. There are five cases.

Case 1–3. $\mathfrak{A}$ is of the form "not $\mathfrak{B}$", "$\mathfrak{B}_1$ and $\mathfrak{B}_2$", or "$\mathfrak{B}_1$ or $\mathfrak{B}_2$", whence respectively $\mathrm{TRoI}(\mathfrak{A}) = \neg\mathrm{TRoI}(\mathfrak{B})$, $\mathrm{TRoI}(\mathfrak{A}) = \mathrm{TRoI}(\mathfrak{B}_1) \wedge \mathrm{TRoI}(\mathfrak{B}_2)$ and $\mathrm{TRoI}(\mathfrak{A}) = \mathrm{TRoI}(\mathfrak{B}_1) \vee \mathrm{TRoI}(\mathfrak{B}_2)$. (2) follows in view of C2.3, C2.5, C2.6 and the induction hypothesis.

Case 4 and 5. $\mathfrak{A}$ is of the form "$\mathfrak{B}(\mathsf{a})$ for at least one $\mathsf{a} \in \mathcal{C} \cup \mathcal{O}$" or "$\mathfrak{B}(\mathsf{a})$ for every $\mathsf{a} \in \mathcal{C} \cup \mathcal{O}$", whence respectively $\mathrm{TRoI}(\mathfrak{A}) = \exists\xi\mathrm{TRoI}(\mathfrak{B}(\xi))$ and $\mathrm{TRoI}(\mathfrak{A}) = \forall\xi\mathrm{TRoI}(\mathfrak{B}(\xi))$. In view of C2.9 and C2.8, $v_{M'}(\exists\xi\mathrm{TRoI}(\mathfrak{B}(\xi))) = 1$ just in case $v_{M'}(\mathrm{TRoI}(\mathfrak{B}(\alpha))) = 1$ for at least one $\alpha \in \mathcal{C} \cup \mathcal{O}$ and $v_{M'}(\forall\xi\mathrm{TRoI}(\mathfrak{B}(\xi))) = 1$ iff $v_{M'}(\mathrm{TRoI}(\mathfrak{B}(\alpha))) = 1$ for all $\alpha \in \mathcal{C} \cup \mathcal{O}$. The induction hypothesis entails that for all $\alpha \in \mathcal{C} \cup \mathcal{O}$, $\mathfrak{B}(\alpha)$ holds true in M iff $v_{M'}(\mathrm{TRoI}(\mathfrak{B}(\alpha))) = 1$. Hence, $v_{M'}(\exists\xi\mathrm{TRoI}(\mathfrak{B}(\xi))) = 1$ iff $\mathfrak{B}(\alpha)$ holds true in M for at least one $\alpha \in \mathcal{C} \cup \mathcal{O}$ and $v_{M'}(\forall\xi\mathrm{TRoI}(\mathfrak{B}(\xi))) = 1$ iff $\mathfrak{B}(\alpha)$ holds true in M for all $\alpha \in \mathcal{C} \cup \mathcal{O}$. So we have established (2). ■

Lemma 5 *If* tr *is the NE-function for* **L**, $M = \langle D, v\rangle$ *is a* **L**-*model for* $^+\mathcal{L}$, $M' = \langle D, v'\rangle$ *a* **CL**-*model for* $^+\mathcal{L}_\sharp$, *and* RMM', *then* $v_{M'}(\mathrm{tr}(A)) = v_M(A)$.

[10] The complexity of a transparent semantic statement, which should not be confused with the complexity function of a nice semantics, is the number of connectives and quantifiers that occur (in English) in the statement.

(i)	$\text{TNRoI}((\mathfrak{A})) = (\text{TNRoI}(\mathfrak{A}))$
(ii)	$\text{TNRoI}(\text{not } \mathfrak{A}) = \neg\text{TNRoI}(\mathfrak{A})$
(iii)	$\text{TNRoI}(\mathfrak{A} \text{ and } \mathfrak{B}) = \text{TNRoI}(\mathfrak{A}) \wedge \text{TNRoI}(\mathfrak{B})$
(iv)	$\text{TNRoI}(\mathfrak{A} \text{ or } \mathfrak{B}) = \text{TNRoI}(\mathfrak{A}) \vee \text{TNRoI}(\mathfrak{B})$
(v)	$\text{TNRoI}(\mathfrak{A}(\mathsf{a}) \text{ for all } \mathsf{a} \in \mathcal{C} \cup \mathcal{O}) = \forall\xi\, \text{TNRoI}(\mathfrak{A}(\xi))$, where ξ is the first variable that does not occur in $\mathfrak{A}(\mathsf{a})$
(vi)	$\text{TNRoI}(\mathfrak{A}(\mathsf{a}) \text{ for at least one } \mathsf{a} \in \mathcal{C} \cup \mathcal{O}) = \exists\xi\, \text{TNRoI}(\mathfrak{A}(\xi))$, where ξ is the first variable that does not occur in $\mathfrak{A}(\mathsf{a})$
(vii)	$\text{TNRoI}(\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)) = \pi^r\alpha_1 \ldots \alpha_r$
(viii)	$\text{TNRoI}(v(\alpha) = v(\beta)) = I\alpha\beta$
(ix)	$\text{TNRoI}(v(A) = \{\emptyset\}) = P^{h(A)}_{f(A)} g(A)$
(x)	$\text{TNRoI}(0 = 0) = (P^0 \vee \neg P^0)$

Table 2: Without RoI: from the semantics to tr

Proof. Immediate in view of Definition 8, Lemma 4 and Lemma 3. ■

Theorem 1 *If* **L** *has a nice semantics that follows the RoI schema and* tr *is the NE-function for* **L**, *then* $\Gamma \vDash_{\mathbf{L}} A$ *iff* $\text{tr}(\Gamma) \vDash_{\mathbf{CL}} \text{tr}(A)$.

Proof. By Lemmas 2 and 5, if a **L**-model M verifies Γ and falsifies A, then there is a **CL**-model M' that verifies $\text{tr}(\Gamma)$ and falsifies $\text{tr}(A)$, and vice versa. ■

If Γ is a finite set, then so is $\text{tr}(\Gamma)$.

6.2 Logics Following the Non-RoI Schema

Let $\Delta^= = \{\forall x Ixx, \forall x\forall y\forall z(Ixy \supset (Ixz \equiv Iyz))\} \cup \{\forall x\forall y(Ixy \supset (A(x) \equiv A(y))) \mid A(x) \in \mathbb{P}\}$.[11] The main general difference with the previous subsection is that, whenever **L** has a nice semantics, the NE-function tr will be such that where $\Gamma \subset \mathcal{W}$ and $A \in \mathcal{W}$, $\Gamma \vdash_{\mathbf{L}} A$ iff $\text{tr}(\Gamma) \cup \Delta^= \vdash_{\mathbf{CL}} \text{tr}(A)$. $\Delta^=$ is an infinite set, but we shall also be able to show that, under the above conditions, $\Gamma \vdash_{\mathbf{L}} A$ iff $\text{tr}(\Gamma) \cup \Delta^=_{\Gamma\cup\{A\}} \vdash_{\mathbf{CL}} \text{tr}(A)$, in which $\text{tr}(\Gamma) \cup \Delta^=_{\Gamma\cup\{A\}}$ is a finite set whenever Γ is finite.

Definition 10 *Where* **L** *is a logic that follows the non-RoI schema and has a nice semantics, the NE-function* $\text{tr}\colon \mathcal{W} \to \mathcal{W}_\sharp$ *for* **L** *is defined as:*

$$\text{tr}(A) = \text{TNRoI}(\mathfrak{A})\,,$$

where $\mathfrak{A}$ *is the selected transparent semantic statement for which "*$V_M(A) = 1$ *iff* $\mathfrak{A}$*" is a reduction statement and* TNRoI *is the function from transparent semantic statements to* $\mathcal{W}_\sharp$*-formulas that is recursively defined in Table 2. The NE-function is extended to sets by* $\text{tr}(\Gamma) = \{\text{tr}(A) \mid A \in \Gamma\}$.

Let us at once consider a complex example, viz. the NE-function for $\mathbf{C_1}$. The NE-function is defined by:[12]

[11] $\Delta^=$ **CL**-entails $\forall x\forall y(Ixy \supset Iyx)$ as well as $\forall x\forall y\forall z(Ixy \supset (Iyz \supset Ixz))$.
[12] The notational abuse for the quantifiers is as described in footnote 9.

T1	$\mathrm{tr}(A) = \mathrm{tr}'(T(A))$, where $T(A)$ is the result of first deleting all the vacuous quantifiers in A and then uniformly replacing all variables by the first variables of the alphabet in alphabetic order.
T2	$\mathrm{tr}'(\pi^r\alpha_1 \ldots \alpha_r) = \pi^r\alpha_1 \ldots \alpha_r \quad (r \geq 0)$
T3	$\mathrm{tr}'(\alpha = \beta) = I\alpha\beta$
T4	$\mathrm{tr}'(A \supset B) = \mathrm{tr}'(A) \supset \mathrm{tr}'(B)$
T5	$\mathrm{tr}'(A \wedge B) = \mathrm{tr}'(A) \wedge \mathrm{tr}'(B)$
T6	$\mathrm{tr}'(A \vee B) = \mathrm{tr}'(A) \vee \mathrm{tr}'(B)$
T7	$\mathrm{tr}'(A \equiv B) = \mathrm{tr}'(A) \equiv \mathrm{tr}'(B)$
T8	$\mathrm{tr}'(\forall\xi A) = \forall\xi\, \mathrm{tr}'(A)$
T9	$\mathrm{tr}'(\exists\xi A) = \exists\xi\, \mathrm{tr}'(A)$
$\mathrm{T10}^{u=p}$	If $A \in {}^{+}\mathbb{P}^{=}$, $\mathrm{tr}'(\neg A) = \neg\mathrm{tr}'(A) \vee P^{h(\neg A)}_{f(\neg A)} g(\neg A)$
$\mathrm{T10}^{s\neg\neg}$	$\mathrm{tr}'(\neg\neg A) = \neg\mathrm{tr}'(\neg A) \vee (\mathrm{tr}'(\neg A) \wedge \mathrm{tr}'(A) \wedge P^{h(\neg\neg A)}_{f(\neg\neg A)} g(\neg\neg A))$
$\mathrm{T10}^{s\neg\supset}$	$\mathrm{tr}'(\neg(A \supset B)) = \neg\mathrm{tr}'(A \supset B) \vee (((\mathrm{tr}'(\neg A) \wedge \mathrm{tr}'(A)) \vee (\mathrm{tr}'(\neg B) \wedge \mathrm{tr}'(B))) \wedge P^{h(\neg(A\supset B))}_{f(\neg(A\supset B))} g(\neg(A \supset B)))$
$\mathrm{T10}^{s\neg\wedge 1}$	if B is not of the form $\neg A$: $\mathrm{tr}'(\neg(A \wedge B)) = \neg\mathrm{tr}'(A \wedge B) \vee (((\mathrm{tr}'(\neg A) \wedge \mathrm{tr}'(A)) \vee (\mathrm{tr}'(\neg B) \wedge \mathrm{tr}'(B))) \wedge P^{h(\neg(A\wedge B))}_{f(\neg(A\wedge B))} g(\neg(A \wedge B)))$
$\mathrm{T10}^{s\neg\wedge 2}$	$\mathrm{tr}'(\neg(A \wedge \neg A)) = \neg(\mathrm{tr}'(A) \wedge \mathrm{tr}'(\neg A))$
$\mathrm{T10}^{s\neg\vee}$	$\mathrm{tr}'(\neg(A \vee B)) = \neg\mathrm{tr}'(A \vee B) \vee (((\mathrm{tr}'(\neg A) \wedge \mathrm{tr}'(A)) \vee (\mathrm{tr}'(\neg B) \wedge \mathrm{tr}'(B))) \wedge P^{h(\neg(A\vee B))}_{f(\neg(A\vee B))} g(\neg(A \vee B)))$
$\mathrm{T10}^{s\neg\equiv}$	$\mathrm{tr}'(\neg(A \equiv B)) = \neg\mathrm{tr}'(A \equiv B) \vee (((\mathrm{tr}'(\neg A) \wedge \mathrm{tr}'(A)) \vee (\mathrm{tr}'(\neg B) \wedge \mathrm{tr}'(B))) \wedge P^{h(\neg(A\equiv B))}_{f(\neg(A\equiv B))} g(\neg(A \equiv B)))$
$\mathrm{T10}^{s\neg\forall}$	$\mathrm{tr}'(\neg\forall\xi A(\xi)) = \neg\mathrm{tr}'(\forall\xi A(\xi)) \vee (\exists\xi(\mathrm{tr}'(\neg A(\xi)) \wedge \mathrm{tr}'(A(\xi))) \wedge P^{h(\neg\forall\xi A(\xi))}_{f(\neg\forall\xi A(\xi))} g(\neg\forall\xi A(\xi)))$
$\mathrm{T10}^{s\neg\exists}$	$\mathrm{tr}'(\neg\exists\xi A(\xi)) = \neg\mathrm{tr}'(\exists\xi A(\xi)) \vee (\exists\xi(\mathrm{tr}'(\neg A(\xi)) \wedge \mathrm{tr}'(A(\xi))) \wedge P^{h(\neg\exists\xi A(\xi))}_{f(\neg\exists\xi A(\xi))} g(\neg\exists\xi A(\xi)))$

Definition 11 *Where $M = \langle D, v\rangle$ is a* **L**-*model for ${}^{+}\mathcal{L}$ and $M' = \langle D, v'\rangle$ a* **CL**-*model for ${}^{+}\mathcal{L}_{\sharp}$, let SMM' hold iff the following conditions are fulfilled:*
S1 If $\alpha \in \mathcal{C} \cup \mathcal{O}$, then $v'(\alpha) = \alpha$.
S2 If $\pi^r \in \mathcal{P}^r$, then $v'(\pi^r) = \{\langle\alpha_1, \ldots, \alpha_r\rangle \mid \langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)\}$.
S3 $v'(I) = \{\langle\alpha, \beta\rangle \mid v(\alpha) = v(\beta)\}$.
S4 For all $A \in {}^{+}\mathcal{W}$, if $f(A) \neq I$, then $v'(P^{h(A)}_{f(A)}) = \{\langle\alpha_1, \ldots, \alpha_r\rangle \mid$ for some $B \in {}^{+}\mathcal{W}$, $f(B) = f(A)$, $v(B) = \{\emptyset\}$, and $g(B) = \alpha_1 \ldots \alpha_r\}$.

Two models (for the same language) are *equivalent* iff they verify the same set of formulas. Where $M = \langle D, v\rangle$ is a **CL**-model for ${}^{+}\mathcal{L}_{\sharp}$, a predicate $\pi^2 \in \mathcal{P}^2_{\sharp}$ will be called *an identity relation over ${}^{+}\mathcal{L}_{\sharp}$ in M* iff $v(\pi^2)$ is reflexive, symmetric and transitive and, for all $\rho^r \in \mathcal{P}^r$, if $\langle v(\alpha_1), \ldots, v(\alpha_i), \ldots, v(\alpha_r)\rangle \in v(\rho^r)$ and $\langle v(\alpha_i), v(\beta)\rangle \in v(\pi^2)$, then $\langle v(\alpha_1), \ldots, v(\beta), \ldots, v(\alpha_r)\rangle \in v(\rho^r)$.

Lemma 6 *(i) For every* **L**-*model $M = \langle D, v\rangle$ for ${}^{+}\mathcal{L}$ there is a* **CL**-*model $M' = \langle\mathcal{C} \cup \mathcal{O}, v'\rangle$ for ${}^{+}\mathcal{L}_{\sharp}$ such that SMM', and (ii) for every* **CL**-*model $M'' = \langle D'', v''\rangle$ for ${}^{+}\mathcal{L}_{\sharp}$ in which I is an identity relation over ${}^{+}\mathcal{L}_{\sharp}$, there is an equivalent* **CL**-*model $M' = \langle\mathcal{C} \cup \mathcal{O}, v'\rangle$ for ${}^{+}\mathcal{L}_{\sharp}$ and there is a* **L**-*model $M = \langle D, v\rangle$ for ${}^{+}\mathcal{L}$ such that SMM'.*

Proof. The proof of (i) is immediate in view of the definition of SMM'. For the proof of (ii), consider a **CL**-model $M'' = \langle D'', v'' \rangle$ for ${}^{+}\mathcal{L}_\sharp$ in which I is an identity relation over ${}^{+}\mathcal{L}_\sharp$. Let $M' = \langle \mathcal{C} \cup \mathcal{O}, v' \rangle$ be a **CL**-model for ${}^{+}\mathcal{L}_\sharp$ in which v' fulfills the following conditions:

(a) Where $\alpha \in \mathcal{C} \cup \mathcal{O}$, $v'(\alpha) = \alpha$.
(b) Where $\pi^r \in \mathcal{P}^r_\sharp$, $v'(\pi^r) = \{\langle \alpha_1, \ldots, \alpha_r \rangle \mid \langle v''(\alpha_1), \ldots, v''(\alpha_r) \rangle \in v''(\pi^r)\}$.

We leave it to the reader to prove that M' is equivalent with M'' and that I is an identity relation over ${}^{+}\mathcal{L}_\sharp$ in M'.

Let, for all $\alpha \in \mathcal{C} \cup \mathcal{O}$, $[\![\alpha]\!] = \{\beta \in \mathcal{C} \cup \mathcal{O} \mid \langle \alpha, \beta \rangle \in v'(I)\}$. Define a **L**-model $M = \langle D, v \rangle$ in which $D = \{[\![\alpha]\!] \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$ and v fulfills the following conditions:

$v1$ Where $\alpha \in \mathcal{C} \cup \mathcal{O}$, $v(\alpha) = [\![\alpha]\!]$.
$v2$ Where $\pi^r \in \mathcal{P}^r$, $v(\pi^r) = \{\langle [\![\alpha_1]\!], \ldots, [\![\alpha_r]\!] \rangle \mid \langle \alpha_1, \ldots, \alpha_r \rangle \in v'(\pi^r)\}$.
$v3$ Where $A \in {}^{+}\mathcal{W}$ and $g(A) = \alpha_1 \ldots \alpha_{h(A)}$, $v(A) = \{\emptyset\}$ iff $\langle \alpha_1, \ldots, \alpha_{h(A)} \rangle \in v'(P^{h(A)}_{f(A)})$.

SMM' holds because M and M' are models of the right sorts, (a) warrants S1, S2 is warranted by $v1$ together with $v2$ and the fact that I is an identity relation over ${}^{+}\mathcal{L}_\sharp$ in M', $v3$ warrants S4, and, given the way in which D is defined, $v1$ warrants S3. ■

Lemma 7 *If SMM', then $M' \Vdash \Delta^{=}$.*

Proof. Suppose that SMM'. In view of S1 and S3, C2.1 warrants that $v_{M'}(\forall x Ixx) = v_{M'}(\forall x \forall y \forall z(Ixy \supset (Ixz \equiv Iyz))) = 1$.

Suppose moreover that $v_{M'}(\forall x \forall y(Ixy \supset (A(x) \equiv A(y)))) = 0$ for some $A(x) \in \mathbb{P}$, whence $v_{M'}(\forall x \forall y(Ixy \supset (A(x) \supset A(y)))) = 0$ or $v_{M'}(\forall x \forall y(Ixy \supset (A(y) \supset A(x)))) = 0$. We only consider the first possibility. It follows that there are $\alpha, \beta \in \mathcal{C} \cup \mathcal{O}$ such that $v_{M'}(I\alpha\beta) = v_{M'}(A(\alpha)) = 1$ and $v_{M'}(A(\beta)) = 0$. We shall show that this is impossible.

As $v_{M'}(I\alpha\beta) = 1$ (for those α and β), $v(\alpha) = v(\beta)$ by S1, S3 and C2.1. As $A(x) \in \mathbb{P}$, and hence $A(\alpha) \in {}^{+}\mathbb{P}$, it follows that $A(\alpha)$ has the form $\pi^r \gamma_1 \ldots \gamma_r$ and that α is one of the γ_i ($1 \leq i \leq r$). Let us represent this by $\pi^r \gamma_1 \ldots \alpha \ldots \gamma_r$. The following equivalences obtain:

	$v_{M'}(\pi^r \gamma_1 \ldots \alpha \ldots \gamma_r) = 1$
iff (by C2.1)	$\langle v'(\gamma_1), \ldots, v'(\alpha), \ldots, v'(\gamma_r) \rangle \in v'(\pi^r)$
iff (by S1)	$\langle \gamma_1, \ldots, \alpha, \ldots, \gamma_r \rangle \in v'(\pi^r)$
iff (by S2)	$\langle v(\gamma_1), \ldots, v(\alpha), \ldots, v(\gamma_r) \rangle \in v(\pi^r)$
iff (as $v(\alpha) = v(\beta)$)	$\langle v(\gamma_1), \ldots, v(\beta), \ldots, v(\gamma_r) \rangle \in v(\pi^r)$
iff (by S1 and S2)	$\langle v'(\gamma_1), \ldots, v'(\beta), \ldots, v'(\gamma_r) \rangle \in v'(\pi^r)$
iff (by C2.1)	$v_{M'}(\pi^r \gamma_1 \ldots \beta \ldots \gamma_r) = 1$.

As $\pi^r \gamma_1 \ldots \beta \ldots \gamma_r$ is $A(\beta)$, this contradicts $v_{M'}(A(\beta)) = 0$. ■

Lemma 8 *If $M = \langle D, v \rangle$ is a* **L***-model for ${}^{+}\mathcal{L}$, $M' = \langle D, v' \rangle$ is a* **CL***-model for ${}^{+}\mathcal{L}_\sharp$, SMM', and $\mathfrak{A}$ is an instance of a semantic base element, then $\mathfrak{A}$ holds true in M iff $v_{M'}(\mathrm{TNRoI}(\mathfrak{A})) = 1$.*

Proof. Suppose that the antecedent is true. There are four cases.

Case 1: $\mathfrak{A}$ has the form $\langle v(\alpha_1),\ldots,v(\alpha_r)\rangle \in v(\pi^r)$, whence TNRoI($\mathfrak{A}$) is $\pi^r\alpha_1\ldots\alpha_r$. The consequent of the lemma follows in view of S1, S2, and C2.1.

Case 2: $\mathfrak{A}$ has the form $v(\alpha) = v(\beta)$, whence TNRoI($\mathfrak{A}$) is $I\alpha\beta$. The consequent of the lemma follows in view of S1, S3 and C2.2.

Case 3: $\mathfrak{A}$ has the form $v(A) = \{\emptyset\}$ whence TNRoI($\mathfrak{A}$) is $P^{h(A)}_{f(A)}g(A)$. The consequent of the lemma follows in view of S1, S4, and C2.1.

Case 4: $\mathfrak{A}$ has the form $0 = 0$, whence TNRoI($\mathfrak{A}$) is $P^0 \vee \neg P^0$. The consequent of the lemma follows in view of C2.6 and C2.3. ■

Lemma 9 *If $M = \langle D, v\rangle$ is a* **L***-model for* $^+\mathcal{L}$*,* $M' = \langle D, v'\rangle$ *is a* **CL***-model for* $^+\mathcal{L}_\sharp$*,* SMM'*, and* $\mathfrak{A}$ *is a transparent semantic statement, then* $\mathfrak{A}$ *holds true in* M *iff* $v_{M'}(\text{TNRoI}(\mathfrak{A})) = 1$.

Proof. The proof is identical to the proof of Lemma 4, apart from the following two aspects: the reference to Lemma 3 should be changed into a reference to Lemma 8 and every occurrence of TRoI should be replaced by TNRoI. ■

Lemma 10 *If* tr *is the NE-function for* **L***,* $M = \langle D, v\rangle$ *is a* **L***-model for* $^+\mathcal{L}$*,* $M' = \langle D, v'\rangle$ *a* **CL***-model for* $^+\mathcal{L}_\sharp$*, and* SMM'*, then* $v_{M'}(\text{tr}(A)) = v_M(A)$.

Proof. Immediate in view of Definition 10 and Lemmas 9 and 8. ■

Theorem 2 *If* **L** *has a nice semantics that follows the non-RoI schema and* tr *is the NE-function for* **L***:* $\Gamma \vDash_{\mathbf{L}} A$ *iff* $\text{tr}(\Gamma) \cup \Delta^= \vDash_{\mathbf{CL}} \text{tr}(A)$.

Proof. For the first direction, suppose that there is a **CL**-model $M'' = \langle D'', v''\rangle$ for $^+\mathcal{L}_\sharp$ such that $M'' \Vdash \text{tr}(\Gamma) \cup \Delta^=$ and $M'' \nVdash \text{tr}(A)$. As $M'' \Vdash \Delta^=$, I is an identity relation over $^+\mathcal{L}_\sharp$ in M''. Hence, by Lemma 6, there is an equivalent **CL**-model $M' = \langle \mathcal{C} \cup \mathcal{O}, v'\rangle$ for $^+\mathcal{L}_\sharp$ and there is a **L**-model M for $^+\mathcal{L}$ such that SMM'. In view of Lemma 10, $M \Vdash \Gamma$ and $M \nVdash A$.

For the second direction, suppose that there is a **L**-model M such that $M \Vdash \Gamma$ and $M \nVdash A$. By Lemma 6, there is a **CL**-model M' such that SMM'. $M' \Vdash \Delta^=$ in view of Lemma 7; $M' \Vdash \text{tr}(\Gamma)$ and $M' \nVdash \text{tr}(A)$ in view of Lemma 10. ■

Even if Γ is a finite set, $\text{tr}(\Gamma) \cup \Delta^=$ is an infinite set, which is inconvenient from a computational point of view. Let $\mathcal{P}_{\Gamma\cup\{A\}}$ be the set of members of $\mathcal{P}$ that occur in Γ or in A, let $\mathsf{Pr}(\Gamma \cup \{A\}) = \{\pi^r x_1 \ldots x_r \mid \pi^r \in \mathcal{P}_{\Gamma\cup\{A\}}\}$, and let $\forall\forall A$ be the universal closure of A (A preceded by a universal quantifier over every variable free in A). Finally, let $\Delta^=_{\Gamma\cup\{A\}} = \{\forall x Ixx, \forall x\forall y\forall z(Ixy \supset (Ixz \equiv Iyz))\} \cup \{\forall\forall(Ixy \supset (B(x) \equiv B(y))) \mid B(x) \in \mathsf{Pr}(\Gamma \cup \{A\})\}$. Clearly $\Delta^=_{\Gamma\cup\{A\}}$ is finite whenever Γ is so.

Theorem 3 $\text{tr}(\Gamma) \cup \Delta^= \vDash_{\mathbf{CL}} \text{tr}(A)$ *iff* $\text{tr}(\Gamma) \cup \Delta^=_{\Gamma\cup\{A\}} \vDash_{\mathbf{CL}} \text{tr}(A)$.

Proof. As $\Delta^=_{\Gamma\cup\{A\}} \subseteq \Delta^=$, the right–left direction is obvious. For the left–right direction, suppose that $\text{tr}(\Gamma) \cup \Delta^=_{\Gamma\cup\{A\}} \nvDash_{\mathbf{CL}} \text{tr}(A)$. It follows that there is a **CL**-model $M = \langle D, v\rangle$ for $^+\mathcal{L}_\sharp$ that verifies $\text{tr}(\Gamma) \cup \Delta^=_{\Gamma\cup\{A\}}$ and falsifies $\text{tr}(A)$. Let $M' = \langle D, v'\rangle$ be exactly as M, except that $v'(\pi^r) = \emptyset$ for all $\pi^r \in \mathcal{P} - \mathcal{P}_{\Gamma\cup\{A\}}$. It follows that M' verifies $\text{tr}(\Gamma) \cup \Delta^=$ and falsifies $\text{tr}(A)$. ■

7 Some Properties of Nice Embeddings

Given that a nice embedding requires a nice semantics, it is easy to prove the following theorem.

Theorem 4 *If there is a nice embedding of* **L** *in* **CL***, then* **L** *is reflexive, transitive, monotonic, and compact and, for every* $\Gamma \subseteq \mathcal{W}$*,* $Cn_{\mathbf{L}}(\Gamma)$ *is a semi-recursive set.*

Proof. That $Cn_{\mathbf{L}}(\Gamma)$ is a semi-recursive set for all $\Gamma \subseteq \mathcal{W}$ is obvious in view of the embedding.

The proof of the other properties is nearly obvious in view of the fact that $\mathrm{tr}(\Gamma) = \{\mathrm{tr}(A) \mid A \in \Gamma\}$ and that $\mathrm{tr}(A)$ is always a single formula.[13] Consider Reflexivity. Where tr is the specific NE-function for **L**, $\Gamma \cup \{A\} \vdash_{\mathbf{L}} A$ is warranted by $[\Delta^{=}\cup]\mathrm{tr}(\Gamma) \cup \{\mathrm{tr}(A)\} \vdash_{\mathbf{L}} \mathrm{tr}(A)$.[14] The proof of the Transitivity and Monotonicity of **L** proceeds similarly.

For Compactness, note that **CL** is compact. So, whenever $\Gamma \vdash_{\mathbf{L}} A$, there are $B_1, \ldots, B_n \in \mathrm{tr}(\Gamma)$ and there is a finite $\Theta \subseteq \Delta^{=}$ ($\emptyset$ if **L** follows the RoI-schema), for which $\Theta \cup B_1, \ldots, B_n \vdash_{\mathbf{CL}} \mathrm{tr}(A)$. As every B_i is $\mathrm{tr}(C_i)$ for a $C_i \in \Gamma$, $C_1, \ldots, C_n \vdash_{\mathbf{L}} A$, whence $\Gamma \vdash_{\mathbf{L}} A$ by the Monotonicity of **L** (proven in the previous paragraph). ■

So there is no point in trying to find a nice embedding of a logic **L** in **CL** if **L** misses any of the properties stated in the theorem.

An embedding of **L** into **CL** reduces questions on **L**-derivability to questions on **CL**-derivability. Different kinds of embedding, however, establish different reductions of questions of the first sort to questions of the second sort. The most important effects of the difference concern the heuristics for questions on **L**-derivability that is offered by the embedding.

A nice embedding of **L** into **CL** reduces, for all Γ and A, the question whether $\Gamma \vdash_{\mathbf{L}} A$ to a specific question whether $\Gamma' \vdash_{\mathbf{CL}} A'$. The latter question is specific in that Γ' and A' belong to a specific fragment of the language of **CL**. The relation between the two kinds of questions has two interesting properties. First, whenever Γ is finite, then so is Γ'. Next, as the heuristics for **CL**-derivability is well-studied and quite efficient, this efficiency is transferred to the original question on **L**-derivability. If the nice semantics is not artificially complicated, then the obtained heuristics for the question whether $\Gamma \vdash_{\mathbf{CL}} A$ will be reasonably efficient.[15] Note that a somewhat efficient heuristics for finding out whether $\Gamma' \vdash_{\mathbf{CL}} A'$ will take properties of Γ' and A' into account. Think about **CL**-tableaux, which provide a very general heuristic method for approaching questions on **CL**-derivability. The tableau rules react typically on the logical form of formulas (premisses, the conclusion, and descendants of these formulas). The efficiency of a heuristic method may be enhanced for example by making it more goal-directed, for example by selecting applications of rules (clauses, instructions, ...) in view of sets of formulas rather than single formulas.[16]

[13] For the proof to go through, it is even sufficient that $\mathrm{tr}(A)$ is a finite set of formulas for all $A \in \mathcal{W}$.

[14] The part in square brackets is only present if **L** follows the non-RoI-schema and it is identical for all Γ and A.

[15] The heuristics cannot be the most efficient one possible and not even the most efficient known one.

[16] See [5, 8, 13] for some examples of goal-directed methods.

In a sense a nice embedding translates A and the members of Γ into formulas that express the meanings of the translated formulas in terms of **CL**, sometimes adding formulas on the premise side to take care of the translated identity. So a nice embedding of **L** in **CL** provides **L** with a heuristics in terms of this translation. The translation also warrants that the thus obtained heuristics for **L** inherits a certain degree of efficiency from **CL**.

The specific properties of a nice embedding are most clearly highlighted by a comparison to other types of embedding. Consider first (what we shall call) a TM-embedding. A logic **L** is semi-recursive iff there is a Turing machine T with the following property: when given the input (A, Γ), T halts after finitely many steps with the answer YES iff $\Gamma \vdash_{\mathbf{L}} A$. The machine T, its initial state, its tape, and the admissible transformations of the machine's state and of the tape can be described in **CL**—see for example [9]. This description can be seen as an embedding of **L** in **CL**, whence every semi-recursive logic can in this way be embedded in **CL**.

Where **L** is semi-recursive, the Turing machine T for **L** may function as follows. Given an input (A, Γ), T considers the natural numbers $n \in \mathbb{N}$ starting from 0. For each n, T first checks wether n is the Gödel number of a list of members of $\mathcal{W}$ and, if so, whether the list is a **L**-proof of A from Γ. If this is the case, T answers YES; if it is not the case, T proceeds to the next $n \in \mathbb{N}$. Note that checking wether n is the Gödel number of a list of members of $\mathcal{W}$ and checking whether the list is a **L**-proof of A from Γ are recursive tasks which require, apart from A and Γ, nothing but the lists of symbols and formation rules of $\mathcal{L}$ and the lists of axioms and rules of **L**.[17] This means that a single Turing machine, call it T_1 may do the job for *all* semi-recursive logics, requiring as input A, Γ, and the four lists. To fix ideas, let the Turing machine require a tape each square of which is blank or filled by a 1 and let the machine start and halt on the leftmost 1 on its tape.

To understand the nature of the TM-embedding derived from T_1, it is instructive to consider $(\Gamma', A') = \text{emb}(\Gamma, A)$. The set Γ' comprises (i) a set of formulas describing (under a convention) the initial state of the tape of the Turing machine on which A, Γ, and the four lists from the previous paragraph are coded and (ii) a set of formulas describing T_1 (under the same convention). The formula A' is a description (under the same convention) of the state of T_1 and its tape that corresponds to the positive answer YES.[18]

One may clearly apply whatever is known about the heuristics of **CL**-proofs to find out whether $\Gamma' \vdash_{\mathbf{CL}} A'$. However, a TM-embedding of **L** in **CL** need not reveal anything about the heuristics of **L**-proofs. Consider T_1. All information it conveys about the logics it handles is that they are semi-recursive. All specific information on the logics is on the respective tapes. Even the specific information, however, concerns checking the well-formedness of formulas and the proofhood of lists of formulas. So in as far as T_1 may be said to embody a heuristics, it is the least goal-directed heuristics one may imagine. T_1 provides some information on recognizing proofs, not on devising **L**-proofs of A from Γ.

[17]The set Γ and the four lists need to be recursive but may be infinite. Note that infinite lists cannot in general be built into the Turing machine itself but have to be coded on the machine's tape.

[18]Basically that T_1 halts reading the leftmost 1 on its tape and that the next square on the tape contains a 1.

As a second example,[19] consider a more specific type of embedding. It looks similar to nice embedding and may be applied to all logics that are reflexive, transitive, and monotonic. By way of example, consider an application to a propositional logic **L**. For any $A \in \mathcal{W}$, $\mathrm{tr}(A) = P^0_A$. It is obvious that $\Gamma \vdash_{\mathbf{L}} A$ iff $\Theta \cup \mathrm{tr}(\Gamma) \vdash_{\mathbf{CL}} \mathrm{tr}(A)$, where Θ contains (i) the translation of all axioms of **L** as well as (ii) the translation of all instances of rules of **L**, expressed by means of material implications. Where the translated logic is the relevant logic **R**, for example, Θ contains (i) P^0_A for every instance $A \in \mathcal{W}$ of an axiom schema of **R** as well as (ii) $P^0_A \supset (P^0_B \supset P^0_{A\wedge B})$ and $P^0_A \supset (P^0_{A\rightarrow B} \supset P^0_B)$ for all $A, B \in \mathcal{W}$.[20]

The difference with a nice embedding is striking. The Compactness of **CL** still warrants that, for every Γ and A, there is a finite Θ' such that $\Theta' \cup \mathrm{tr}(\Gamma) \vdash_{\mathbf{CL}} \mathrm{tr}(A)$ whenever $\Gamma \vdash_{\mathbf{R}} A$. There is no clue, however, for constructing the finite $\Theta' \subset \Theta$.[21] So even for the simplest **R**-derivation, we are facing an infinite premise set in **CL**. Moreover, the embedding offers no insight on heuristic methods for **R**-derivability.

8 Some Further Comments

Our distinction in terms of the RoI schema is useful when one devises a nice semantics for a given logic and moreover simplifies the proof in Section 6. However, it is not difficult to unify the matter. First, modify two clauses in the definition of the function TRoI:

(viii) $\mathrm{TRoI}(v(\alpha) = v(\beta)) = I\alpha\beta$

(ix) if $A \notin {}^{+}\mathbb{P}^{=}$,
$\mathrm{TRoI}(\langle v(g_1(A)), \ldots, v(g_{h(A)}(A))\rangle \in v(f(A))) = Q^{h(A)}_{f(A)} g(A)$

Extend $\mathcal{L}_\sharp$ with predicates $Q^{h(A)}_{f(A)}$ for $A \in \mathcal{W}$ and replace, in the definition of $\Delta^{=}$, $\mathbb{P}$ by $\mathbb{P} \cup \{Q^{h(A)}_{f(A)} g(A) \mid A \in \mathcal{W}\}$. It is easily seen that the embedding still goes through for logics that follow the RoI schema. Moreover, the functions TRoI and TNRoI do not conflict. So they may be replaced by a single function that takes care of the embedding of both kinds of logics (and also of logics that follow the RoI schema at some points and not at others).[22] With these changes, Theorem 1 becomes invalid but Theorem 2 holds for all logics that have a nice semantics.

When one comes across a new logic **L**, and devises a nice semantics for it—or possibly finds **L** by devising a nice semantics—our result provides an embedding of **L** in **CL** and constructively warrants that **L** is a semi-recursive logic.

[19] The example is made up by us. Another example from the literature is [15].

[20] The claim in the text is correct. Yet, given that **R** is defined in [1] (in the worst modal tradition, viz. the one of [14]) as a logic engendering a set of theorems and that the inference relation is only defined indirectly by the “Entailment Theorem”, viz. as $A_1, \ldots, A_n \vdash_{\mathbf{R}} B$ iff $\vdash_{\mathbf{R}} (A_1 \wedge \ldots \wedge A_n) \rightarrow B$, it would take a longer way than we can afford here to show that our claim is correct.

[21] Which formulas should belong to Θ' depends on the question which axioms and rules of **R** are required to derive the conclusion from the premises. So this depends wholly on **R** and the embedding does not offer any help.

[22] This highlights that the difference does not relate to the way in which the metalinguistic identity is translated, but with the fact that I warrants RoI for $Q^{h(A)}_{f(A)} g(A)$-formulas and not for $P^{h(A)}_{f(A)} g(A)$-formulas.

An interesting open problem concerns the delineation of the set of logics that have a nice semantics and the procedure to devise, where this is possible, a nice semantics for a given logic.

In Section 7 "translation" was used in an intuitive way. However, there is a significant literature in which "translation" has a specific meaning and this literature concerns the embedding of one logic into another. An interesting survey is presented by Carnielli and others [10]. The survey is also interesting because it illustrates one aspect of the novelty of our result, as we shall now show.

Phrased in our terminology, a translation is defined in [10] as follows. Let $\mathcal{L}_1$ be the language of $\mathbf{L_1}$ and $\mathcal{L}_2$ be the language of $\mathbf{L_2}$. A *translation* is a function $f\colon \mathcal{W}_1 \to \mathcal{W}_2$ such that, for all $\Gamma \cup \{A\} \subseteq \mathcal{W}_1$ holds: if $\Gamma \vdash_{\mathbf{L_1}} A$ then $f(\Gamma) \vdash_{\mathbf{L_2}} f(A)$ in which $f(\Gamma) = \{f(B) \mid B \in \mathcal{W}_1\}$. A translation is *conservative* iff an equivalence holds instead of an implication. It is easily seen that the NE-function tr is not a translation according to this definition. Moreover, our results from Section 6.2 and hence also our result on the unification introduced at the beginning of the present section cannot be rephrased in terms of a translation function in the sense of [10]. In the non-RoI case, a nice embedding requires one of the following. (i) $\Gamma \vdash_{\mathbf{L}} A$ corresponds to a **CL**-derivability statement in which, on the premise side, the translation of the premises is extended either with an infinite context independent set, as in Theorem 2, or with a set that depends on the conclusion, as in Theorem 3. (ii) If one tries to push the set into the translation function itself, then one needs two translation functions, for example $\mathrm{TR}_1(A) =_{df} \bigwedge(\Delta^{=}_{\{A\}}) \wedge \mathrm{tr}(A)$ for the premise side and $\mathrm{TR}_2(A) =_{df} \bigwedge(\Delta^{=}_{\{A\}}) \supset \mathrm{tr}(A)$ for the conclusion side. In neither case does one obtain a translation in the sense of [10]. Note that (i) and (ii) are exchangeable choices because we consider only embedding within **CL**. If one considers embedding within an arbitrary logic, (i) and (ii) may lead to very different results indeed. And obviously there is nothing wrong with an embedding or a translation that has either structure.

In [10], three 'dimensions' on translations (in the informal sense) are discussed. In view of what is said in the previous paragraph, the dimension concerning conservative translations may be rephrased more generally, viz. in agreement with (i) and (ii). In this way nice embeddings would fall under conservative translations.

References

[1] Anderson, Alan Ross, and Nuel D. Belnap, Jr., *Entailment. The Logic of Relevance and Necessity*, vol. 1, Princeton University Press, 1975.

[2] Arruda, Ayda I., 'On the imaginary logic of N.A. Vasil'ev', in Ayda I. Arruda, Newton C.A. da Costa, and R. Chuaqui, (eds.), *Non-classical Logics, Model Theory and Computability*, North-Holland, Amsterdam, 1977, pp. 3–24.

[3] Batens, Diderik, 'Paraconsistent extensional propositional logics', *Logique et Analyse*, 90–91 (1980), 195–234.

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

[5] Batens, Diderik, 'It might have been Classical Logic', *Logique et Analyse*, 218 (2012), 241–279.

[6] Batens, Diderik, 'Spoiled for choice?', *Journal of Logic and Computation*, (in print). Doi:10.1093/logcom/ext019, 1913.

[7] Batens, Diderik, Kristof De Clercq, and Natasha Kurtonina, 'Embedding and interpolation for some paralogics. The propositional case', *Reports on Mathematical Logic*, 33 (1999), 29–44.

[8] Batens, Diderik, and Dagmar Provijn, 'Pushing the search paths in the proofs. A study in proof heuristics', *Logique et Analyse*, 173–175 (2001), 113–134. Appeared 2003.

[9] Boolos, George S., John P. Burgess, and Richard J. Jeffrey, *Computability and Logic*, Cambridge University Press, 2002. (Fourth edition).

[10] Carnielli, Walter A., Marcelo E. Coniglio, and Itala M. Loffredo D'Ottaviano, 'New dimensions on translations between logics', *Logica Universalis*, 3 (2009), 1–18.

[11] da Costa, Newton C.A., 'Calculs propositionnels pour les systèmes formels inconsistants', *Comptes rendus de l'Académie des sciences de Paris*, 259 (1963), 3790–3792.

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

[13] Gabbay, Dov M., and Nicola Olivetti, *Goal-Directed Proof Theory*, Kluwer, Dordrecht, 2000.

[14] Hughes, G.E., and M.J. Cresswell, *An Introduction to Modal Logic*, Methuen, London, New York, 1972. First published 1968.

[15] Jeřábek, Emil, 'The ubiquity of conservative translations', *The Review of Symbolic Logic*, 5 (2012), 666–678.

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

[17] Meheus, Joke, 'An extremely rich paraconsistent logic and the adaptive logic based on it', in Diderik Batens, Chris Mortensen, Graham Priest, and Jean Paul Van Bendegem, (eds.), *Frontiers of Paraconsistent Logic*, Research Studies Press, Baldock, UK, 2000, pp. 189–201.

[18] Priest, Graham, *In Contradiction. A Study of the Transconsistent*, Nijhoff, Dordrecht, 1987.

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

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