5

A persistent notion of truth in dynamic semantics

Abstract. For a certain interpretation of first-order formulas A as binary (input/output) relations $[\![A]\!]$ on a set S of states (specifying state transitions induced by A), a notion $|A| \subseteq S$ of truth for A is investigated, arising from a so-called double (intuitionistic) negation translation of the domain of $[\![A]\!]$ (or of, equivalently, as it turns out, the fixed points of $[\![A]\!]$). A global, Boolean-valued analysis is presented alongside a local, three-valued non-compositional approximation of it. Complications with existence and identity are exposed, and suitable generic models constructed. The analysis is carried over to a determinization of transitions between "disjunctive" sets of output states.

1. Introduction

The essential idea behind "dynamic semantics" (e.g., [**14**]) is that the semantic content of a formula (or statement) lies not so much in its truth relative to a state (or context) but in the change in state that it induces (that change being the reason the statement is *used*). That is, over a given set S of states, a formula A is interpreted as a binary relation $[\![A]\!] \subseteq S \times S$ describing transitions between states s and s'

$$s[\![A]\!]s' \quad \text{iff} \quad \text{on input } s,\ A \text{ can output } s' \ .$$

By contrast, let us understand the truth value $|A|$ of a formula A to be the subset $\{s \in S : A \text{ is true at } s\}$ of S, given some definition of

$$(*) \quad A \text{ is true at } s \ .$$

The problem addressed by the present work is to define a notion $(*)$ of truth, given a relational interpretation $[\![A]\!]$ of formulas A.

1.1. Related work. Two obvious candidates for $(*)$ mentioned in [**4**] amount essentially to what are called acceptance and acceptability in [**19**]

(i) A is *accepted at* s if $s[\![A]\!]s$ (i.e., $s \in fix([\![A]\!])$)

(ii) A is *acceptable at* s if there is some state s' for which $s[\![A]\!]s'$ (i.e., $s \in dom([\![A]\!])$).

Neither definition mentions any notion of partiality on states, measuring say information content — a feature that is particularly prominent in [**19**], where a state is formed from a set of worlds. If the partiality of a state s is understood to refer to the possibility of a formula A as well as its negation being (separately) acceptable in s, then adopting (ii) for $(*)$ yields an incoherent notion of truth. Option (i) would, on the other hand, mean that asserting a truth can have *no* effect on the initial state, thereby building a certain logical omniscience into states, as well as trivializing the informational significance of truths.

Proceeding more concretely, consider, for example, *Dynamic Predicate Logic* (DPL, [**7**]), where a state is a function from a set of variables to the universe of some first-order model. It is easy to see (under definitions reviewed in §2.1 below) that over a first-order model M where the unary relation symbol R has a non-empty denotation R^M, and a function f that *fails* to map the variable x to an element of R^M,

$$\begin{array}{ll} R(x) & \text{is neither accepted nor acceptable at } f \ , \\ \exists x \ R(x) & \text{is not accepted but is acceptable at } f \ , \end{array}$$

and

$$(\exists x \ R(x)) \ \& \ R(x) \quad \text{is not accepted but is acceptable at } f \ .$$

Hence, if the formula $\exists x R(x)$ ought to be true at f, then stipulating that A is true at s precisely when A is accepted at s is too strong to account for the dynamic effects of quantification. As for identifying truth with acceptability, it is slightly embarrassing to assert that the conjunction of a true formula with an untrue formula is true. It is only slightly embarrassing in view of the somewhat novel "dynamic" interpretation of conjunction as relational composition. Even so, however, when breaking new ground, there is every reason to pause before throwing out hard-won fruits of past labor, among which one might count a Boolean-valued notion of truth.

Bringing into the picture a pre-order $\sqsubseteq$ on the set S of states (comparing information content), a natural intuition is that if $s \sqsubseteq s'$ then every formula true at s is also true at s'. That is, the truth set $|A|$ of a formula A ought to be persistent, where a set $U \subseteq S$ of states is understood to be *persistent* if for all $s \in U$, $s' \sqsupseteq s$ implies $s' \in U$. As there is no reason to expect that $fix([\![A]\!])$ or $dom([\![A]\!])$ should be persistent relative to natural measures $\sqsubseteq$ of information content (a point confirmed in section 4 below), a bit more work needs to be put into the definition of truth. And also, as it turns out, into the relational interpretation $[\![A]\!]$ of formulas A, which will be defined below to be more faithful to [**14**], extended with witness constructs $\epsilon x : A$ for explicit quantification $\exists x \ A$.

1.2. Summary of present work. Building on the idea that a state $s \in S$ is *encompassed by* a set $U \subseteq S$ of states if

$$(\forall s' \sqsupseteq s) \ (\exists s'' \sqsupseteq s') \qquad s'' \in U \ ,$$

a formula A is defined to be *true at* s if s is encompassed by $fix([\![A]\!])$ — i.e.,

$$(\forall s' \sqsupseteq s) \ (\exists s'' \sqsupseteq s') \qquad s''[\![A]\!]s'' \ .$$

Then the truth set $|A| = \{s \in S : A \text{ is true at } s\}$ is manifestly persistent, and is, in fact, a persistent negation applied twice. That is, $|A|$ is the image

$j(fix([\![A]\!]))$ of $fix([\![A]\!])$ under the the so-called *double negation translation*[1] j that maps subsets U of S into persistent subsets of S according to

$$j(U) \quad = \quad \{s \in S \ : \ s \text{ is encompassed by } U\} \ .$$

A certain interpretation $[\![A]\!]$ of first-order formulas A as binary relations on finite functions from a set *Var* of variables to the universe $|M|$ of a first-order model M is defined supporting the following facts.

(i) Although acceptance is, as it stands, an inappropriate notion of truth for the dynamic effects introduced to handle anaphora, j makes acceptance acceptable in the sense that

$$j(fix([\![A]\!])) \quad = \quad j(dom([\![A]\!])) \ .$$

(ii) $|A|$ is compositional (relative to the logical connectives), and is moreover Boolean-valued (e.g., $|A\&B| = |A| \wedge |B|$).

(iii) The truth of A at s is determined by the behavior of A at completions of s, or more precisely, the "generic models" of s

$$A \text{ is true at } s \quad \text{iff} \quad \text{every generic model of } s \text{ satisfies } A \ .$$

(iv) $|A|$ supports truth gaps implicated in presupposition failure.

(v) Under a passage to a richer notion of ("information") state (compatible with [**19**]), j is well-behaved.

The relational interpretation $[\![A]\!]$ is essentially the part of [**14**] that was meant to be reformulated by DPL, plus certain constructs introduced to handle complications with existence and identity. These constructs are reminiscent of Hilbert's epsilon terms as well as restricted parameters in situation theory.

2. Preliminaries

The linguistic motivation for the present work can be traced back to *Discourse Representation Theory* (DRT, [**14**]). Although DRT covers a good deal of ground (see, for example, [**13**]), only a small part of it need concern us here. That part is henceforth referred to as DRT/DPL, having been analyzed in DPL, modulo certain simplifications that unfortunately constitute significant differences from DRT. Nonetheless, it is instructive to reverse the

[1] "Negation" here refers to Kripke's semantic interpretation of the intuitionistic negation $\overline{A}$ of a formula A under a forcing relation $|\!\vdash$ between states s partially ordered by $\leq$ and formulas

$$s \,|\!\vdash \overline{A} \quad \text{iff} \quad \text{not } (\exists s' \geq s) \ s' \,|\!\vdash A \ .$$

The double negation translation $\overline{\overline{A}}$ of a formula A then yields a so-called *weak forcing* relation $|\!\vdash^w$ (e.g., [**15**])

$$\begin{aligned} s \,|\!\vdash^w A \quad & \text{iff} \quad s \,|\!\vdash \overline{\overline{A}} \\ & \text{iff} \quad (\forall s' \geq s)(\exists s'' \geq s') \ s'' \,|\!\vdash A \end{aligned}$$

typically subject to Boolean logic. In the present context, the terms "weak acceptance" and "weak acceptability" are somewhat inappropriate because neither acceptance nor acceptability is persistent, as pointed out in section 4, and thus neither implies truth.

historical order, and present DPL (in its simplicity) before DRT/DPL. Towards that end, fix a signature **L** with equality, and a countable set *Var* of variables.

2.1. Dynamic Predicate Logic. At the heart of DPL is a translation $\cdot^{DPL}$ of first-order **L**-formulas with variables from *Var* into programs from (quantified) dynamic logic (see, for instance, [**9**]) according to

$$\begin{array}{rcll} A^{DPL} & = & A? & \text{for atomic formulas } A \\ (\neg A)^{DPL} & = & \neg(A^{DPL}) & \\ (A\&B)^{DPL} & = & A^{DPL}\ ;\ B^{DPL} & \\ (\exists x\ A)^{DPL} & = & x :=?\ ;\ A^{DPL}\ , & \end{array}$$

where the programs p (in the right hand side) are interpreted relative to an **L**-model M as binary relations $\rho(p)$ on functions $(f, g, h, \ldots)$ from *Var* to the universe $|M|$ of M as follows

$$\begin{array}{rcl} f\ \rho(A?)\ g & \text{iff} & f = g \text{ and } M \models A[f] \\ f\ \rho(x :=?)\ g & \text{iff} & (\forall x' \neq x)\ f(x') = g(x') \\ f\ \rho(p;q)\ g & \text{iff} & (\exists h)\ f\ \rho(p)\ h \ \text{ and } \ h\ \rho(q)\ g \\ f\ \rho(\neg p)\ g & \text{iff} & f = g \text{ and } f \notin dom(\rho(p))\ . \end{array}$$

The reader familiar with dynamic logic may code $\neg p$ up as the test $([p]\bot)?$, where $[p]$ is the universal (box) modality labelled by the program p, and $\bot$ is some contradictory formula. As in classical logic, disjunction, implication and universal quantification can be defined in DPL from negation, conjunction and existential quantification

$$\begin{array}{rcl} A \vee B & \equiv & \neg(\neg A\ \&\ \neg B) \\ A \supset B & \equiv & \neg(A\ \&\ \neg B) \\ \forall x\ A & \equiv & \neg \exists x\ \neg A\ , \end{array}$$

where $A \equiv B$ means $\rho(A^{DPL}) = \rho(B^{DPL})$.

2.2. DRT/DPL. A consequence of the use of total functions as inputs and outputs in dynamic logic is that the value of a variable can be destroyed by a program interpreting quantification. One can argue, however, that there is no compelling reason for such destruction of information, and that, indeed, the use of total functions in dynamic logic obscures the expansive growth of information implicit in the introduction of discourse markers in DRT. Without getting into the details of DRT, let us switch from total functions to finite functions, and call a function from a finite subset of *Var* to $|M|$ an (M-)*substitution*. (We will often say "substitution" without mentioning M, with the understanding that M is fixed in the background.) The interpretations $\rho(p)$ are characterized as before (i.e., $A?$ as a test for A, sequential composition $p;q$ as relational composition $\rho(p) \circ \rho(q)$, and negation $\neg p$ as a test for the complement of the domain of $\rho(p)$), except that the relation $\rho(x :=?)$ is no longer symmetric:

$$f \; \rho(x :=?) \; g \quad \text{iff} \quad \begin{array}{l} dom(g) = dom(f) \cup \{x\} \text{ and} \\ (\forall x' \in dom(f) - \{x\}) \; f(x') = g(x') \end{array}$$

for all substitutions f and g. Now, to protect a pre-existent binding on a variable $x \in Var$, let us define a *guarded assignment* $x := *$ to behave exactly like $x :=?$ on input substitutions where x is uninitialized, and to do nothing elsewhere. That is, let $x := *$ abbreviate

$$x = x? \quad + \quad \neg(x = x?) \; ; \; x :=? \; ,$$

where $+$ is non-deterministic choice, and interpreted as union:

$$f \; \rho(p + q) \; g \quad \text{iff} \quad f \; \rho(p) \; g \; \text{ or } \; f \; \rho(q) \; g$$

for all substitutions f and g. Observe that $\neg\neg$ reduces $+$ to DPL disjunction

$$\rho(\neg\neg(p + q)) \quad = \quad \rho(\neg(\neg p; \neg q)) \; ,$$

or, put in another way, $+$ provides a "dynamic" alternative to DPL disjunction. Note also that because a substitution need not be defined on a variable occurring in a formula A, the program $A? + \overline{A}?$ (or $\neg(\neg A?; \neg\overline{A}?)$), where $\overline{A}$ is interpreted as the complement of A, may fail to return an output. That is, bivalence may break down because of variables undefined at the input state.

The precise relationship between the modification of DPL above and DRT is described in [**6**]. As detailed there, guarded assignments provide an implicit treatment of existential quantification, in the sense that the fragment of DRT covered by DPL is given by the following translation $\cdot^\bullet$ of quantifier-free first-order formulas

$$\begin{array}{rcl} (R(\overline{t}))^\bullet & = & \overline{x} := * \; ; \; R(\overline{t})? \quad \text{for all relation} \\ & & \qquad\qquad \text{symbols } R \text{ including } = \\ (A \& B)^\bullet & = & A^\bullet; B^\bullet \\ (\neg A)^\bullet & = & \neg(A^\bullet) \end{array}$$

where the list $\overline{x} = x_1, \ldots, x_n$ in the first line for $\cdot^\bullet$ above consists of all variables occurring in $\overline{t}$, and $\overline{x} := *$ is $x_1 := *; \ldots; x_n := *$. An explicit treatment of quantification will be presented in the next section that extends this translation conservatively.

3. An interpretation of formulas as programs

To overcome complications with identity and existence, it is useful to introduce terms that witness existential formulas $\exists x \; A$. Such terms can be viewed as "labelled variables" $y_{A,x}$ or as epsilon terms $\epsilon x : A$, similar to those of Hilbert mentioned in [**12**] (except that here such terms will not always be defined, thus blocking the treatment of universal quantification intended by Hilbert). (These terms can also be viewed situation-theoretically as restricted parameters; see [**5**].) We will use the notation $y_{A,x}$ and $\epsilon x : A$ interchangably.

3.1. A language. To avoid pesky variable clashes, let us be careful to partition the set *Var* of variables into two disjoint sets X and Y, where X is a countable set of "unlabelled variables" given at the outset (along with a fixed signature $\mathbf{L}$) and the set Y of fresh "labelled" variables is defined simultaneously with the set $\mathbf{L}(\mathit{Var})$ of formulas as follows.

(i) If R is an n-ary relation symbol of $\mathbf{L}$ (including $=$), and $\overline{t}$ is a list of n $\mathbf{L}$-terms with variables from $X \cup Y$, then $R(\overline{t})$ belongs to $\mathbf{L}(\mathit{Var})$.

(ii) If $x \in X$, and A and B belong to $\mathbf{L}(\mathit{Var})$, then $A \& B$, $A \vee B$, $\neg A$, and $\exists x\ A$ belong to $\mathbf{L}(\mathit{Var})$.

(iii) If $x \in X$ and $A \in \mathbf{L}(\mathit{Var})$, then the fresh variable $y_{A,x}$ (= epsilon term $\epsilon x\!:\!A$) belongs to Y.

Observe that $\mathbf{L}(\mathit{Var})$ is *not* closed under $\supset$ or $\forall$. The expressions $A \supset B$ and $\forall x A$ can, if the reader insists, be understood as abbreviations of $\neg(A \& \neg B)$ and $\neg\exists x \neg A$, respectively.

3.2. A translation. Next, modifying the DRT/DPL translation slightly, define a translation $\cdot^u$ from formulas $A \in \mathbf{L}(\mathit{Var})$ into programs by

$$
\begin{array}{lrcl}
(1) & (R(\overline{t}))^u & = & \overline{x} := * \ ; \ R(\overline{t})? \\
 & (A \& B)^u & = & A^u ; B^u \\
 & (A \vee B)^u & = & A^u + B^u \\
 & (\neg A)^u & = & \neg(A^u) \\
 & (\exists x\ A)^u & = & y_{A,x} := * \ ; \ A[y_{A,x}/x]^u \ ; \\
(2) & & & \mathit{always}(A[y_{A,x}/x]) \ ,
\end{array}
$$

where the list $\overline{x}$ in (1) consists of all variables in X occurring in $\overline{t}$, the witness variable $y_{A,x}$ in (2) belongs to Y, and the program $\mathit{always}(A[y_{A,x}/x])$ will be described shortly. Observe that variables in X occuring freely in a formula A are automatically initialized in A^u (in accordance with (1)), whereas other variables (in Y) are initialized only when they are bound (according to (2)). That is, the choice between using unlabelled and labelled variables (to represent discourse markers) in a formula depends on how much one wishes to fuss over variable initializations.[2]

[2]For example, the donkey sentence

`If a farmer owns a donkey then he beats it`

can be translated as

$$\mathtt{farmer}(x)\ \&\ \mathtt{own}(x,x')\ \&\ \mathtt{donkey}(x') \quad \supset \quad \mathtt{beat}(x,x')$$

(because of effects on initializing unlabelled variables reminiscent of [**17**]), or (among other possibilities) as

$$(\exists x)(\exists x')(\mathtt{farmer}(x)\ \&\ \mathtt{own}(x,x')\ \&\ \mathtt{donkey}(x')) \quad \supset \quad \mathtt{beat}(\epsilon x\!:\!\exists x' A, \epsilon x'\!:\!A') \ ,$$

To insure that a labelled variable $y_{A,x}$ is initialized only if it will always witness the formula, a program $always(A[y_{A,x}/x])$ is appended[3] to the translation of $\exists xA$. (For example, if A were $v = v \supset R(x)$ — i.e., $\neg(v = v \& \neg R(x))$ —, where $v \in Y$, then any object would witness $\exists xA$ at a state f where $v \notin dom(f)$.) Note that a witness to A can be spoiled only if variables mentioned in A^u are given new values. Fortunately, there are only finitely many such variables. This allows us to construct $always(A)$ as

$$always(A) \quad = \quad \neg\ (in(A)\ ;\ \neg(A^u))\ ,$$

where the program $in(A)$ is the non-deterministic choice

$$\sum In(A)$$

of the set $In(A)$ of all "legal" initializations that have some bearing on A^u. Just what the set $In(A)$ is, we turn to next. The reader is warned that the description of $In(A)$ is annoyingly technical, and is perhaps best skipped on a first reading. (Afterall, once all free variables in A which can be initialized are initialized, the test $always(A[y_{A,x}/x])$ added to (2) can be ignored.) With this in mind, the definition of $In(A)$ is indented below to mark it from the rest of the text.

> For $v \in Y$, let us write W_v for the formula that v should witness
>
> $$W_v \quad = \quad C[v/x] \qquad \text{where } v \text{ is } y_{C,x}\ .$$
>
> Then, given an $A \in \mathbf{L}(Var)$, define the sets $Y_A \subset Y$ and $Fmla_A \subset \mathbf{L}(Var)$ inductively by
>
> > if $v \in Y$ occurs in A or in some $B \in Fmla_A$ then v belongs to Y_A and W_v belongs to $Fmla_A$.
>
> **Proposition**. *For every $A \in \mathbf{L}(Var)$, $A \notin Fmla_A$ and the sets $Fmla_A$ and Y_A are finite. Moreover, for every $B \in Fmla_A$, $Fmla_B \subset Fmla_A$ and $Y_B \subseteq Y_A$.*
>
> **Proof**. That $A \notin Fmla_A$ follows from the inductive construction of Var and $\mathbf{L}(Var)$. The inclusions $Fmla_B \subset Fmla_A$ and $Y_B \subseteq Y_A$ for $B \in Fmla_A$ are easy consequences of the definitions of $Fmla_{(\cdot)}$ and $Y_{(\cdot)}$. An application of König's Lemma to the derivation trees of $Fmla_A$ and Y_A yields the finiteness of the sets $Fmla_A$ and Y_A. $\dashv$

---

where A is $\texttt{farmer}(x)\ \&\ \texttt{own}(x,x')\ \&\ \texttt{donkey}(x')$, A' is $A[(\epsilon x : \exists x' A)/x]$, and $B \supset C$ is understood as an abbreviation of $\neg(B \& \neg C)$. Observe that we lose the DPL equivalences

$$(\exists xA)\ \&\ B \ \equiv\ \exists x(A\ \&\ B)$$
$$(\exists xA)\ \supset\ B \ \equiv\ \forall x(A\ \supset\ B)$$

because of variable labelling, although $\exists x$ still introduces a variable available for reference outside $\exists$'s usual scope.

[3] An alternative might be to initialize all variables in A that can be initialized before choosing a witness for $\exists xA$

$$(\exists x\ A)^u \quad = \quad init(A)\ ;\ y_{A,x} := *\ ;\ A[y_{A,x}/x]^u$$

(for some suitable program $init(A)$) but this possibility will not be pursued here.

Next, let X_A be

$$\{x \in X \;:\; x \text{ occurs freely in } A \text{ or in some } B \in \mathit{Fmla}_A\} \;,$$

and finally let $In(A)$ be the (finite) set of programs of the form

$$\begin{array}{l} x_1 := * \;;\; x_2 := * \;;\; \cdots \;;\; x_m := * \;; \\ v_1 := * \;;\; v_2 := * \;;\; \cdots \;;\; v_n := * \;; \\ (W_{v_1})^u \;;\; (W_{v_2})^u \;;\; \cdots \;;\; (W_{v_n})^u \end{array}$$

where $x_1, \ldots, x_m$ is some list of elements from X_A (that does not necessarily include every $x \in X_A$), and (similarly) $v_1, \ldots, v_n$ is some list of elements from Y_A. (The sequence $(W_{v_1})^u \;;\; (W_{v_2})^u \;; \cdots \;;\; (W_{v_n})^u$ is appended to verify that the initializations of labelled variables are "legal".) The astute reader concerned about the inductive character of our definition of $\cdot^u$ may draw from the previous proposition the

Corollary. *The binary relation $\succ$ on $\mathbf{L}(\mathit{Var})$ given by*

$$A \succ B \quad \text{iff} \quad B \in \mathit{Fmla}_A$$

is well-founded (and transitive), and for every $A \in \mathbf{L}(\mathit{Var})$, the set $\{B : A \succ B\}$ is finite.

Having defined the translation $\cdot^u$, let us isolate the set S of *states* that will interest us

$$S \quad = \quad \{f \;:\; \text{for some } A \in \mathbf{L}(\mathit{Var}),\; \emptyset \;\rho(A^u)\; f\} \;.$$

For every $A \in \mathbf{L}(\mathit{Var})$, define

$$[\![A]\!] \quad = \quad \rho(A^u) \text{ restricted to } S \;.$$

Notice that for every $A \in \mathbf{L}(\mathit{Var})$, the program A^u is *guarded* in that all occurrences of random assignments in it are within guarded assignments.

Proposition 1 (Monotonicity). *For every guarded program p, and for all substitutions f and g, if $f \;\rho(p)\; g$ then $f \subseteq g$. In particular, for all $A \in \mathbf{L}(\mathit{Var})$, $f[\![A]\!]g$ implies $f \subseteq g$.*

By contrast, under DPL, the output state of $[\![\exists x A]\!]$ may destroy a variable binding in the input state.

4. Truth of formulas *qua* programs

Let $\sqsubseteq$ be the restriction of $\subseteq$ to S. A natural property to require of a definition of "A is true at f" is that truth persist through $\sqsubseteq$-larger states g. Neither acceptance nor acceptability is persistent; indeed

$$f[\![A]\!]f \text{ and } f \sqsubseteq g \quad \text{need not imply} \quad g \in \mathit{dom}([\![A]\!]) \;,$$

since a variable may be assigned a value at g that precludes the acceptability of A. (For example, take A to be $\neg x = x$.) To secure persistence, let us recall the double negation translation j mentioned in the introduction.

Definition. A formula A of $\mathbf{L}(\mathit{Var})$ is *true at* a state $f \in S$ if $f \in j(fix([\![A]\!]))$ — i.e.,

$$(\forall g \sqsupseteq f)\ (\exists h \sqsupseteq g) \qquad h[\![A]\!]h \ .$$

Proposition 2 (Persistence). *If A is true at f and $f \sqsubseteq g$ then A is true at g.*

Proposition 2 is immediate from the transitivity of $\sqsupseteq$. The next lemma may seem rather technical, but will nevertheless prove important, particularly as $fix([\![A]\!])$ is not necessarily persistent.

Lemma 3 (Eventual stability). *Given an $A \in \mathbf{L}(\mathit{Var})$ and a $g \in S$, there is a $\hat{g} \sqsupseteq g$ such that*

$$(\exists h \sqsupseteq \hat{g})\ h \in \mathit{dom}([\![A]\!]) \quad \textit{implies} \quad (\forall h \sqsupseteq \hat{g})\ h[\![A]\!]h \ .$$

Proof. The idea is to choose $\hat{g}$ such that $g[\![A']\!]\hat{g}$, where A' is some tautology that translates (under $\cdot^u$) to a program initializing all variables in A^u which can be initialized (given input g). Towards this end, recall the definitions of Fmla_A, Y_A and X_A from §3.2, and choose (by the inductive construction of $\mathbf{L}(\mathit{Var})$ along with Var) an enumeration $B_0, \dots, B_n$ of Fmla_A such that for all i and j with $0 \leq i < j \leq n$,

$$\text{if } B_j \text{ is } W_v \text{ then } v \text{ does not occur in } B_i.$$

Take A' to be

$$\bigwedge_{0 \leq j \leq m} x_j = x_j \quad \& \quad \bigwedge_{0 \leq i \leq n} (B_i \vee \neg B_i) \ ,$$

where $x_0, \dots, x_m$ is an (any) enumeration of the set X_A of all variables in X occurring freely in A or in some $B \in \mathit{Fmla}_A$.[4] Let $\mathit{Var}_A = Y_A \cup X_A$ and F be the set of substitutions in S with domain contained in Var_A

$$F \quad = \quad \{f \in S \ : \ \mathit{dom}(f) \subseteq \mathit{Var}_A\} \ .$$

Observe that for every $\hat{g}$ such that $g[\![A']\!]\hat{g}$, the restriction of $\hat{g}$ to Var_A is $\sqsubseteq$-maximal among all substitutions in F, yielding the desired property. $\dashv$

We can now provide three alternatives U to $fix([\![A]\!])$ satisfying $j(U) = j(fix([\![A]\!]))$.

Theorem 4. *Given an $A \in \mathbf{L}(\mathit{Var})$ and an $f \in S$, the following are equivalent.*

(i) *A is true at f.*
(ii) $f \in j(\mathit{dom}([\![A]\!]))$.
(iii) $f \in j(\{h \in S : (\forall f' \sqsupseteq h)\ f'[\![A]\!]f'\})$.
(iv) $f \in j(\{h \in S : (\forall f' \sqsupseteq h)\ f' \in \mathit{dom}([\![A]\!])\})$.

[4]This proof exploits the "dynamic" interpretation of disjunction as union, $[\![A \vee B]\!] = [\![A]\!] \cup [\![B]\!]$. If disjunction $A \vee B$ is instead reduced "statically" to $\neg(\neg A \& \neg B)$, then the formula A' in the proof above will have to be replaced by one of the 2^{n+1} combinations of negated or unnegated B_i's that is satisfiable (i.e., returns an output).

Proof. Clearly, (i) implies (ii), and (iii) implies (i). To see that (ii) implies (iii), assume (ii) and suppose $g \sqsupseteq f$. Appealing to Lemma 3, pick a $\hat{g} \sqsupseteq g$ such that (by (ii)) $(\forall f' \sqsupseteq \hat{g})\ f'[\![A]\!]f'$, thus validating (iii). Finally, note that (iii) implies (iv), while (iv) implies (ii). $\dashv$

As $dom([\![A]\!]) = dom([\![\neg\neg A]\!])$, Theorem 4 yields

Corollary 5 (Static projection). *For every formula A and state f, A is true at f iff for some B such that $[\![B]\!] = [\![\neg\neg A]\!]$, B is true at f.*

As $\neg\neg p$ never returns an output state different from the input, Corollary 5 raises the question as to whether or not the truth of a formula at a state f can be predicted from information about the truth of its (proper) parts at f. As the double negation translation j involves the totality of all states, it is hardly surprisingly that the answer, as the next subsection shows, is negative. Nevertheless, this exercise will prove useful warm-up for the global analysis that then follows.

4.1. A local three-valued analysis of truth. The present subsection is concerned with the question of predicting the truth of a formula A at a state f from information about the truth at the same state f of (proper) subformulas of A. Complementing the notion of truth, a notion of falsehood will be considered that builds on certain harmless assumptions, yielding a negation map $\sim$. Let $\mathbf{L}$ be a signature such that a bijective map $\hat{\cdot}$ on the relation symbols in $\mathbf{L}$ can be defined assigning to every R its "complement" $\hat{R}$ (also in $\mathbf{L}$) with the same arity, such that $\hat{\hat{\cdot}}$ is the identity on the relation symbols of $\mathbf{L}$. (A simple doubling of $\mathbf{L}$ will suffice.) An $\mathbf{L}$-model M is understood always to satisfy

$$(\dagger) \qquad R^M \subseteq |M|^n - \hat{R}^M$$

for every n-ary relation symbol R in $\mathbf{L}$, which is easy enough to express in the first-order language of $\mathbf{L}$. Next, let $\sim$ be the map on $\mathbf{L}(\mathit{Var})$ defined recursively by

$$\begin{array}{rcl} \sim R(\bar{t}) & = & \hat{R}(\bar{t}) \\ \sim(A\&B) & = & \sim A \vee \sim B \\ \sim(A \vee B) & = & \sim A \;\&\; \sim B \\ \sim\neg A & = & A \\ \sim(\exists x\ A) & = & \neg\exists x\ A \end{array}$$

(e.g., [**16**], except that the clause for $\exists$ is modified, based on an identification of $\forall xA$ with $\neg\exists x\neg A$, as $\mathbf{L}(\mathit{Var})$ is not closed under $\forall$). Because of ($\dagger$) above,

Lemma 6. *If $f[\![A]\!]f$ then not $f[\![\sim A]\!]f$.*

We can now assign a truth value $|A|_f$ to a formula $A \in \mathbf{L}(\mathit{Var})$ at a state f from the 3-element set $\{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$. Define

$|A|_f = \mathbf{t}$ if A is true at f,
A is false at f, $|A|_f = \mathbf{f}$, if $|\sim A| = \mathbf{t}$,

and

A is undetermined at f, $|A|_f = \mathbf{u}$, otherwise.

That is, $|A|_f = \mathbf{u}$ iff

$$(\exists g \sqsupseteq f)(\forall h \sqsupseteq g) \text{ not } h[\![A]\!]h \quad \text{and}$$
$$(\exists g \sqsupseteq f)(\forall h \sqsupseteq g) \text{ not } h[\![\sim A]\!]h \ .$$

A formula may be neither true nor false at a state not only because a variable may be uninitialized at that state, but also because the inclusion $\subseteq$ in (†) may be proper. Moreover, from the preceding lemma,

Corollary 7 (3-valuedness). *A cannot be both true and false at the same state.*

Although the truth values **t** and **f** persist, **u** need not. Holding a state f fixed, however, the definitions above yield the following truth tables, where $A \supset B$ is taken as an abbreviation of $\neg(A \& \neg B)$.

A	B	$\&$	$\sim$	$\vee$	$\supset$
t	**t**	**t**	**f**	**t**	**t**
t	**f**	**f**		**t**	**f**
t	**u**	**u**		**t**	**u**
f	**t**	**f**	**t**	**t**	**t**
f	**f**	**f**		**f**	**t**
f	**u**	**f**		**u**	**t**
u	**t**	**u**	**u**	**t**	**t**
u	**f**	**f**		**u**	$\neq$ **f**
u	**u**	$\neq$ **t**		$\neq$ **f**	$\neq$ **f**

The truth tables are strong Kleene except on certain entries where one of the subformulas has an undetermined truth value and no truth value can be predicted for the formula as a whole. For instance, if $|A|_f = |B|_f = \mathbf{u}$, then $|A \supset B|_f$ can be **t**, as with

$$|x = 1 \supset x = 1|_\emptyset \ ,$$

or **u**, as with

$$|x = 1 \supset x = 0|_\emptyset \ .$$

Observe that $|A \supset B|_f$ can never be **f**, or else $\sim(A \supset B)$ would be true at f; i.e., $|A \, \& \neg B|_f = \mathbf{t}$, which is absurd since $|A|_f = \mathbf{u}$.

4.2. A global Boolean-valued analysis. The non-compositionality in §4.1 can be repaired through an analysis of truth that is global in that the relativization of the truth value $|A|_f$ to a state f is dropped.[5] Instead, we define

$$|A| \ = \ \{f \in S \ : \ A \text{ is true at } f\} \quad (= \ j(fix([\![A]\!]))) \ ,$$

with the intuition (if the reader likes) that the truth value **u** breaks up into many different values lying (properly) between $\emptyset$ and S.

[5] An alternative would be to keep the truth value local, but to set $|A|_f = \{g \in S : f[\![A]\!]g\} \in \text{Power}(S)$. There is then, however, no real difference with the dynamic interpretation $[\![A]\!]$, the compositionality of which carries over to $|A|$.

Rather than proceeding directly to $|A|$, let us try more slowly to relate $j(fix([\![A]\!]))$ to standard constructions in algebraic semantics. Towards this end, define

$$\begin{aligned}\Omega &= \{U \subseteq S \;:\; (\forall f \in U)(\forall g \sqsupseteq f)\; g \in U\} \\ \Omega_j &= \{j(U) \;:\; U \subseteq S\}\;,\end{aligned}$$

the first set Ω consisting of subsets of S persistent with respect to $\sqsubseteq$, and the second set Ω_j of j-sets.

Theorem (Folklore).

(a) $\langle \Omega, \cap, \cup, \rightarrow, \emptyset \rangle$ *is a complete Heyting algebra, where*

$$U \rightarrow U' \;=\; \{f \in S \;:\; (\forall g \sqsupseteq f)\;\; \text{if } g \in U \text{ then } g \in U'\}$$

for $U, U' \in \Omega$, and least upper bounds are given by union, and greatest lower bounds by intersection.

(b) $\langle \Omega_j, \cap, \sqcup, c, S, \emptyset \rangle$ *is a complete Boolean algebra, where*

$$\begin{aligned}U \sqcup U' &= j(U \cup U') \\ c(U) &= \{f \in S \;:\; (\forall g \sqsupseteq f)\; g \notin U\}\end{aligned}$$

for U and $U' \in \Omega_j$, and

$$\begin{aligned}\coprod_{i \in I} U_i &= j(\bigcup_{i \in I} U_i) \\ \prod_{i \in I} U_i &= j(\bigcap_{i \in I} U_i)\end{aligned}$$

for $\{U_i \;:\; i \in I\} \subseteq \Omega_j$.

Next, to provide an algebraic semantics for **L**-sentences relative to a fixed **L**-model M, it is customary to extend the signature **L** to a signature $\mathbf{L}_M$ by throwing into **L** constants naming elements in $|M|$ (identifying $m \in |M|$ with its name). Such a step will not be required here, although **L** can, if the reader wishes, be assumed to have names for all objects in $|M|$. In any case, we will interpret not only **L**-sentences (without free variables), but also arbitrary formulas from $\mathbf{L}(\mathit{Var})$ (possibly with free variables from Var). Or, put another way, the variables in Var will be treated as constants, which, as we will see, will serve as the usual witnesses.

Another point the reader familiar with algebraic semantics might raise is that the double negation translation j is commonly applied to a Heyting algebra — for example, the Heyting algebra Ω at hand. Although neither $fix([\![A]\!])$ nor $dom([\![A]\!])$ is necessarily in Ω, Theorem 4 suggests defining a (persistent) forcing relation $\Vdash$ between $f \in S$ and $A \in \mathbf{L}(\mathit{Var})$ by either

$$f \Vdash A \quad \text{iff} \quad (\forall g \sqsupseteq f)\; g \in dom([\![A]\!])\;,$$

or (non-equivalently)

$$f \Vdash A \quad \text{iff} \quad (\forall g \sqsupseteq f)\; g[\![A]\!]g\;.$$

Unfortunately, neither alternative provides (for instance) a decomposition of $\{f : f \Vdash A \vee B\}$ as the union (the join in the Heyting algebra Ω) of $\{f : f \Vdash A\}$ and of $\{f : f \Vdash B\}$. (Consider the case where A is $(\epsilon x : C) = (\epsilon x : C)$ and B is $\neg A$.) Some Heyting algebra other than Ω is called for; but the only natural one the present author has found is the Boolean algebra Ω_j of j-sets!

A final point before presenting our Boolean-valued analysis of truth concerns our treatment of quantification. Following our coding of guarded assignments in terms of equality, the range of quantification is restricted to objects m that "exist" in the sense that $m = m$. Also, note that for every family $\mathcal{U} \subseteq \Omega_j$, if $\bigcup \mathcal{U} \in \Omega_j$ (i.e., $\bigcup \mathcal{U} = j(\bigcup \mathcal{U})$), then $\bigcup \mathcal{U} = \coprod \mathcal{U}$.

Theorem 8 (Boolean-valued semantics). *For all $A, B \in \mathbf{L}(Var)$,*

$$
\begin{aligned}
|A \& B| &= |A| \cap |B| \\
|A \vee B| &= |A| \sqcup |B| \quad (= j(|A| \cup |B|)) \\
|\neg A| &= c(|A|) \\
&= j(S - |A|) \\
|\exists x\ A| &= |(\epsilon x : A) = (\epsilon x : A)| \\
&= \bigcup \{|v = v\ \&\ A[v/x]| \ :\ v \in Var\} \\
&= \bigcup \{|A[m/x]| \ :\ m \in |M|\} \qquad \textit{if } \mathbf{L} \textit{ names all objects in } M.
\end{aligned}
$$

Proof. Use monotonicity (Proposition 1) and the various characterizations of $|A|$ provided by Theorem 4. For example, to see that $f \in |A\&B|$ implies $f \in |B|$, fix a $g \sqsupseteq f$ and conclude (as required by Theorem 4(ii)) that there must be some $h \sqsupseteq g$ in $dom([\![B]\!])$ since $f \in |A\&B|$. Assuming that $|A\&B| = |A| \cap |B|$ has been established, then we can appeal to static projection (Corollary 5) to reduce $|A \vee B| = |A| \sqcup |B|$ to $|\neg A| = c(|A|)$, since $[\![\neg\neg(A \vee B)]\!] = [\![\neg(\neg A \& \neg B)]\!]$. By definition, $|\neg A| = j(S - dom([\![A]\!]))$, and so by eventual stability (Lemma 3), $|\neg A| = \{f \in S : (\forall g \sqsupseteq f)\ g \notin |A|\}$. Note that eventual stability also yields $j(S - dom([\![A]\!])) = j(S - |A|)$.

Turning finally to quantification, these equations are consequences of the careful bookkeeping and coding that has gone into labelling, as well as our restriction of S to the substitutions accessible from $\emptyset$. The point is that witnesses are persistent in the sense that for all $f \in S$,

$$(\epsilon x : A) \in dom(f) \quad \text{implies} \quad f \in dom([\![\exists x\ A]\!]) \ .$$

Note the restriction in quantification to $v = v$ is required since quantified formulas translate to programs that always initialize the (translations of) variables bound by quantifiers. (Consider, for example, $\exists x\ \neg x = x$.) $\dashv$

Theorem 8 provides some evidence that our notion of truth is natural. A further argument is presented next, consisting of a construction along the partial order $\sqsubseteq$ on S, that will again force us to face the complications with existence and identity above.

4.3. Constructing generic models from states. To understand just what the double negation translation j is, we might ask what constraints on information growth from f are implied by $f \in j(U)$? Call a sequence $\{f_\alpha : \alpha < \kappa\} \subseteq S$ (for some cardinal κ) a *maximal* $\sqsubseteq$*-chain from* f if

(i) $f \sqsubseteq f_0$,
(ii) for all $\alpha < \kappa$ and $\beta < \alpha$, $f_\beta \sqsubseteq f_\alpha$, and
(iii) there is *no* $g \in S$ such that for all $\alpha < \kappa$, $f_\alpha \sqsubset g$.

Now, it is true enough that if every maximal $\sqsubseteq$-chain from f intersects U, then $f \in j(U)$. On the other hand, the converse need not hold: given a maximal $\sqsubseteq$-chain $f = f_0 \sqsubset f_1 \sqsubset f_2 \sqsubset \cdots$, the maximal chain $\{f_{2i} : i \in \omega\}$ does not intersect $U = \{f_{2i+1} : i \in \omega\}$. For a more pleasing fit with information growth, we must consider not only the partial order $\sqsubseteq$ on S, but also the formulas A in $\mathbf{L}(Var)$. In particular, we will adapt the theory of generic models described in [**15**] (going back to [**18**]).

It will be convenient to define the basic notions from [**15**] in our setting as follows. A subset U of S is a *generic set* if

(i) for all $f, f' \in U$, there is a $g \in U$ such that $f \sqsubseteq g$ and $f' \sqsubseteq g$, and
(ii) for every $A \in \mathbf{L}(Var)$, there is an $f \in U$ such that $f \in |A| \cup |\neg A|$.

Defining the *theory* $Th(U)$ *of* U to be

$$Th(U) \;=\; \{A \in \mathbf{L}(Var) \;:\; (\exists f \in U)\; f \in |A|\}\ ,$$

observe that condition (i) (in the definition of a generic set) is a consistency criterion, while (ii) is a completeness criterion. Next, given a signature $\mathbf{L}$, define an $\mathbf{L}$-*free model* to be an $\mathbf{L}$-model N, except that equality is construed as a non-logical symbol $\approx$ that is interpreted by N as some restriction of true equality

$$\approx^N \;=\; \{(n,n) \;:\; n \in dom(\approx^N)\}\ .$$

The term "free" here should be construed not in the usual algebraic sense but rather as in "free logic" (e.g. [**3**]), the intuition being that $dom(\approx^N)$ consists of the objects that exist. The notion of an $\mathbf{L}$-free model N *satisfying* an $\mathbf{L}$-sentence A, $N \models A$, is defined as in the familiar Tarskian manner (with $\approx$ regarded as a non-logical symbol), except that quantification is restricted to the domain of $\approx^N$

$$\begin{aligned} N \models \forall x A \quad &\text{iff} \quad (\forall n \in dom(\approx^N))\; N \models A[n/x] \\ N \models \exists x A \quad &\text{iff} \quad (\exists n \in dom(\approx^N))\; N \models A[n/x]\ . \end{aligned}$$

Now, treating Var as a set of constants, a *generic model for a* generic set U is an $\mathbf{L}(Var)$-free model N such that all its objects are named in Var

$$|N| \;=\; \{v^N \;:\; v \in Var\}\ ,$$

and every $\mathbf{L}(Var)$-sentence in $Th(U)$ is satisfied by N. If, moreover, $f \in U$, then N is said to be a *generic model for* f.

Theorem 9 (Generic Model Theorem). *Every* $f \in S$ *has a generic model; i.e., for every* $f \in S$*, there is a generic model for* f.

Proof. The proof is similar to pp. 101 and 102 of [**15**] except that (i) a generic set is constructed in a different way below (to avoid requiring that $\mathbf{L}(Var)$ be countable), (ii) conjunction, rather than disjunction, is taken to be primitive (since $\vee$ cannot be interpreted as union), and (iii) quantification must be restricted to objects n such that $n \approx n$. (Note that j turns DRT/DPL negation into intuitionistic negation.) More precisely, Theorem 9 is an immediate consequence of the following two lemmas (applied in sequence).

Lemma. *Every* $f \in S$ *belongs to a generic set.*

Proof. From an enumeration $\{A_\alpha : \alpha < \kappa\}$ of $\mathbf{L}(Var)$,construct a generic set $U = \{f_\alpha : \alpha < \kappa\}$ by taking f_0 to be f, and choosing f_α as follows, for $0 < \alpha < \kappa$. Following the notation of the proof of Lemma 3, let Var_α be the (finite) set of all variables occuring in some $B \in Fmla_{A_\alpha}$. Let g be the substitution

$$\{(v,m) \in Var_\alpha \times |M| \ : \ (\exists \beta < \alpha)\ v \in dom(f_\beta) \text{ and } f_\beta(v) = m\}$$

and pick a $\hat{g}$ such that $g[\![A'_\alpha]\!]\hat{g}$, where A'_α is the A' in the proof of Lemma 3, with A replaced by A_α. Set f_α to $\hat{g}$. $\dashv$

Lemma. *Every generic set has a generic model.*

Proof. Given a generic set U, define a (partial equivalence) relation $\equiv$ on Var by

$$v \equiv v' \quad \text{iff} \quad v \approx v' \in Th(U)\ ,$$

and form an $\mathbf{L}(Var)$-free model N by taking as its universe $|N|$ the set $\{v^\equiv : v \in Var\}$ of (partial equivalence) classes $v^\equiv = \{v' \in Var : v \equiv v'\}$ for $v \in Var$. (Note that $\emptyset \in |N|$.) Interpret the k-ary relation symbols R of $\mathbf{L}$ (inlcuding $\approx$) and the l-ary function (including constant) symbols F of $\mathbf{L} \cup Var$ according to

$$\begin{array}{rcl} R^N(v_1^\equiv, \ldots, v_k^\equiv) & \text{iff} & R(v_1, \ldots, v_k) \in Th(U) \\ F^N(v_1^\equiv, \ldots, v_l^\equiv) & = & y_{x=F(v_1,\ldots,v_l),x}{}^\equiv \ \text{for some} \\ & & \qquad x \in X - \{v_1, \ldots, v_l\}\ . \end{array}$$

These are legitimate definitions (i.e., independent of the choice of representatives v of $v^\equiv$), since $Th(U)$ respects the laws of equality restricted to $dom(\approx^N)$ and the witnessing on variables. Notice that every term τ has a denotation τ^N, and that $\tau^N = \emptyset$ precisely if $\tau \approx \tau \notin Th(U)$. Luckily, $\emptyset \not\approx^N \emptyset$, and we have no way of expressing equality in $\mathbf{L}(Var)$ except by its approximation $\approx$.) The expected equivalence

$$N \models A \quad \text{iff} \quad A \in Th(U)$$

can be established by a routine induction on A, appealing to the consistency criterion (i) of a generic set. ⊣

⊣

Corollary 10. *Let $A \in \mathbf{L}(Var)$ and $f \in S$. A is true at f (i.e., $f \in |A|$) iff A is satisfied by every generic model for f.*

Proof. As in pp. 102 and 103 of [**15**], the forward direction is trivial, while the converse follows from Theorem 9. ⊣

Lastly, let us remark that the underlying $\mathbf{L}$-model M (relative to which $[\![\cdot]\!]$ and S are defined) is an elementary extension of N°, for every generic model N (based on M), where $\cdot^\circ$ is the obvious map from $\mathbf{L}(Var)$-free models to $\mathbf{L}$-models (obtained by throwing out all "non-existent" objects). Conversely, every countable elementary substructure of M can be obtained as such.

5. Discussion

A basic objection that might be raised against the double negation translation j on which our notion of truth rests is that it abstracts away the temporary instability of a formula (as Lemma 3 and ultimately Theorem 8 suggest). How is the doctrine that

the meaning of a formula is the change it induces

served by trivializing that very change? But, ideologies aside, if we refrain from reducing meaning to truth, and instead agree that truth is a "static" abstraction of meaning, then the objection loses much of its force. The question becomes, quite innocently, what does our notion of truth fail to abstract away? One answer is the partiality of $\mathbf{L}(Var)$. For although a good deal of effort was put above into developing the Boolean-valued character of the notion of truth proposed, the formally bivalent (i.e. 2-valued) interpretation in §4.3 is achieved only at the cost of a non-logical treatment of equality and a somewhat non-standard interpretation of quantification. Furthermore, as demonstrated in §4.1, one can define translations that have the effect of re-interpreting the logical connectives (for instance, negation[6]) to support some partiality. Let us conclude by briefly considering these points in relation to presuppositions, and turning to a richer notion of state similar to that in [**19**] (concerning which an additional sense in which the double negation translation is persistent will be established).

[6] Observe, however, that even though a negation $\sim$ different from DRT/DPL negation $\neg$ is introduced in §4.1, DRT/DPL negation is still useful for $\forall$ and presumably implication.

5.1. Truth gaps. With regard to truth gaps implicated in presupposition failure, two sources of partiality then remain:

(i) the partiality of the substitutions f, particularly relative to labelled variables (or epsilon terms), some of which cannot be initialized, and
(ii) the possibility of defining negation translations based on extension/anti-extension pairs that need not be full Boolean complements.

Comparing DRT/DPL negation $\neg$ with the negation translation $\sim$ employed in §4.1 to define falsehood, note that whereas $A \vee \neg A$ is always true, $A \vee \sim A$ may fail to be true even if for every k-ary $R \in \mathbf{L}$, $R^M = |M|^k - \hat{R}^M$ (in which case, $R^N = dom(\approx^N)^k - \hat{R}^N$ holds for every generic model N based on M). There remains the possibility of an epsilon term that can never be initialized — for example, ϵx : wife(x,John), where John has no wife (in M). Such truth gaps provide natural opportunities for giving accounts of simple (existential) cases of presupposition failure. Interpreting $\hat{R}$ to be less than the full complement of the interpretation of R offers further possibilities.

5.2. Information states. The analysis above is predicated on a notion of state that is somewhat impoverished, next to [**10**]'s set of "sequence-world-pairs", commonly used in accounts of presuppositions. (This is a non-trivial matter inasmuch as our notion of truth is conditioned by it.) A sequence-world-pair is simply a substitution f, coupled with the first-order model M that structures the range of f. (It becomes important to keep track of M, when a state is built from different M's.) For a sufficiently large set $\mathcal{M}$ of $\mathbf{L}$-models, let

$$\mathcal{S} \;=\; \{(f, M) \mid M \in \mathcal{M} \text{ and } f \text{ is an } M\text{-substitution}\} \;.$$

The richer notion of a state as a subset of $\mathcal{S}$ can be obtained from the interpretation $[\![\cdot]\!]$ above, as in the classic construction of deterministic finite automata from non-deterministic ones (e.g. [**11**]). More precisely, given an $A \in \mathbf{L}(Var)$, lift the binary relation

$$[A] \;=\; \{((f, M), (g, M)) \mid M \in \mathcal{M} \text{ and } f[\![A]\!]_M g\}$$

on $\mathcal{S}$ to the binary relation $\mathcal{D}[A]$ on $\mathsf{Pow}_+(\mathcal{S})$ $(= \mathsf{Pow}(\mathcal{S}) - \{\emptyset\})$, given by

$$\mathcal{D}[A] \;=\; \{(U, V) \in \mathsf{Pow}_+(\mathcal{S}) \times \mathsf{Pow}_+(\mathcal{S}) \mid V = \{t \mid (\exists s \in U) s[A]t\}\}.$$

(Let us agree to exclude the empty set from the field of $\mathcal{D}[A]$, as that arises from the impossibility of an $[A]$-transition.) This lifting is feasible because all programs involved above are so-called distributive (i.e., definable at the level of single substitutions). We will attend later to the non-distributive programs $\mathsf{might}\ A$ of [**19**]. As $\mathcal{D}$ internalizes the non-determinism of $[\![A]\!]$ in states (from $\mathsf{Pow}_+(\mathcal{S})$), it is natural to lift a pre-order $\leq$ on $\mathcal{S}$ comparing information content to its so-called Smyth pre-order

$$\leq_{\mathcal{D}} \;=\; \{(U, V) \in \mathsf{Pow}_+(\mathcal{S}) \times \mathsf{Pow}_+(\mathcal{S}) \mid (\forall t \in V)\ (\exists s \in U)\ s \leq t\} \;.$$

Next, let us write J for the double negation translation $j^{\sqsubseteq_{\mathcal{D}}}$ defined relative to the Smyth pre-order $\sqsubseteq_{\mathcal{D}}$ of

$$\{((f,M),(g,M)) \mid M \in \mathcal{M} \text{ and } f \sqsubseteq_M g\} ,$$

where $\sqsubseteq_M$ is the partial order $\sqsubseteq$ used to define truth in section 3 for a fixed **L**-model M. Finally, define an $A \in \mathbf{L}(\mathit{Var})$ to be *true at* a state $U \in \mathsf{Pow}_+(\mathcal{S})$ if $U \in J(fix(\mathcal{D}[A]))$.

Theorem 11. Given an $A \in \mathbf{L}(\mathit{Var})$ and a $U \in \mathsf{Pow}_+(\mathcal{S})$, the following are equivalent.

(i) A is true at U.
(ii) $U \in J(dom(\mathcal{D}[A]))$.
(iii) For every $s \in U$, A is true at $\{s\}$.
(iv) For every $s \in U$, A is true at s (according to section 4).

Proof. Argue that (i) implies (ii), (ii) implies (iii), (iii) implies (iv), and finally (iv) implies (i). The first implication is immediate. The others hold because every $V \in \mathsf{Pow}_+(\mathcal{S})$ is $\sqsubseteq_{\mathcal{D}}$-dominated by some singleton (even though every singleton is also dominated by a non-singleton) and by appealing to Theorem 4. The universal quantifier in J leads to the universal (rather than existential) force in (iii) and (iv). (Observe that the theorem is quite robust under revisions of the interpretation $[\![A]\!]_M$.) $\dashv$

A final remark (which the present author plans to expand upon elsewhere) concerns [**19**]'s program construct might, two features of which would appear to be at odds with the analysis above — viz., non-persistence and non-distributivity. Just how useful our persistent notion of truth is in analyzing the non-persistence of might remains to be seen. As for the second feature, in the same way that modality can be treated syntactically (an approach advocated in [**1**] for propositional attitudes), a distributive analysis of might is possible, so long as updates are added to the underlying first-order model (exactly as sets are treated in generalized Henkin models for higher-order logic). The introduction of abstract objects into first-order models is argued and investigated at length in [**2**]. In the case of might, the alternative syntactic approach promises to be of interest, given complications with quantification defined at the richer level of $\mathsf{Pow}_+(\mathcal{S})$-states (addressed in [**8**]) — complications that the present author suspects vanish in a distributive setting.

Acknowledgments. Although the present paper was completed in Stuttgart, much of the work was carried out while I was a visiting researcher at ITK, Tilburg, where I profited from discussions with Emiel Krahmer. I also thank David Israel and Fernando Pereira for comments following my talk on this subject at the fourth conference on situation theory and its applications.

References

[1] N. Asher and H. Kamp. Self-reference, attitudes, and paradox. In G. Chierchia, B.H. Partee, and R. Turner, editors, *Properties, Types and Meaning*, volume 1. Kluwer Academic Publishers, Dordrecht, 1989.

[2] Nicholas Asher. *Reference to Abstract Objects in Discourse.* Kluwer Academic Publishers, Dordrecht, 1993.

[3] Ermanno Bencivenga. Free logics. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 3. Reidel, Dordrecht, 1986.

[4] Johan van Benthem. Logic and the flow of information. In D. Prawitz et al., editor, *Proc. 9th International Congress of Logic, Methodology and Philosophy of Science.* North-Holland, Amsterdam, 1991.

[5] Robin Cooper. Situation theoretic discourse representation theory. Manuscript, HCRC, Edinburgh, 1992.

[6] Tim Fernando. What is a DRS? In R. Cooper and J. Groenendijk, editors, *Integrating Semantic Theories II.* Dyana deliverable R2.1.B, 1994. Presented in a Computational Semantics workshop at Tilburg, December 1994.

[7] J. Groenendijk and M. Stokhof. Dynamic predicate logic. *Linguistics and Philosophy*, 14, 1991.

[8] J. Groenendijk, M. Stokhof, and F. Veltman. This might be it. In J. Seligman and D. Westerståhl, editors, *Logic, Language and Computation*, volume 1. CSLI Lecture Notes Number 58, Stanford, 1996. Also appears in Dyana deliverable R2.1.B, *Integrating Semantic Theories II*, (R. Cooper and J. Groenendijk, editors), 1994.

[9] David Harel. Dynamic logic. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 2. Reidel, Dordrecht, 1984.

[10] Irene Heim. On the projection problem for presuppositions. In M. Barlow, D.P. Flickinger, and M.T. Westcoat, editors, *Proc. West Coast Conference on Formal Linguistics*, volume 2. Stanford Linguistics Association, California, 1983.

[11] J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Language and Computation.* Addison-Wesley, 1979.

[12] David Israel. The very idea of dynamic semantics. In P. Dekker and M. Stokhof, editors, *Proc. Ninth Amsterdam Colloquium.* ILLC, University of Amsterdam, 1994.

[13] H. Kamp and U. Reyle. *From Discourse to Logic.* Kluwer Academic Publishers, Dordrecht, 1993.

[14] Hans Kamp. A theory of truth and semantic representation. In J. Groenendijk, T. Janssen, and M. Stokhof, editors, *Formal Methods in the Study of Language.* Mathematical Centre Tracts 135, Amsterdam, 1981.

[15] H. Jerome Keisler. Forcing and the omitting types theorem. In M. Morley, editor, *Studies in Model Theory.* The Mathematical Association of America, 1973.

[16] David Nelson. Constructible falsity. *J. Symbolic Logic*, 14, 1949.

[17] P. Pagin and D. Westerståhl. Predicate logic with flexibly binding operators and natural language semantics. *J. Logic, Language and Information*, 2(2), 1993.

[18] H. Rasiowa and R. Sikorski. *The Mathematics of Metamathematics.* Warsaw, 1963.

[19] Frank Veltman. Defaults in update semantics. In H. Kamp, editor, *Conditionals, Defaults and Belief Revision.* Edinburgh, Dyana deliverable R2.5.A, 1990.