DIT IS EEN VOORBEELD

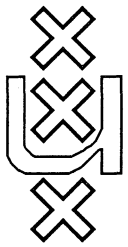
Institute for Language, Logic and Information

GOING PARTIAL IN MONTAGUE GRAMMAR

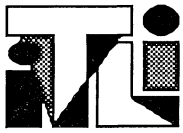
Reinhard Muskens

ITLI Prepublication Series

for Logic and Philosophy of Language LP-88-04



University of Amsterdam



Institute for Language, Logic and Information
Instituut voor Taal, Logica en Informatie

GOING PARTIAL IN MONTAGUE GRAMMAR

Reinhard Muskens
Department of Philosophy
University of Amsterdam

Received March 1988

Research supported by Z.W.O.

Correspondence to:

Faculteit der Wiskunde en Informatica
(Department of Mathematics and Computer Science)
Roetersstraat 15
1018WB Amsterdam

or

Faculteit der Wijsbegeerte
(Department of Philosophy)
Grimburgwal 10
1012GA Amsterdam

0. Introduction

In this paper I shall show that Barwise's [1981] analysis of neutral perception verbs and Barwise & Perry's [1983] treatment of the propositional attitudes can be formalized within Montague Grammar. To this end I shall give a new, partialized, modeltheoretic semantics to the fragment of natural language defined in Montague [1973] (PTQ henceforth). The result will clearly fall within the limits of Montague Semantics but can also be seen as a version of Barwise and Perry's system, Situation Semantics, provided that a liberal definition of the latter is employed.

Present-day Montague Grammar is poorly equipped to treat propositional attitudes or naked infinitive constructions well. The reason for this seems to be that while our beliefs, desires and hopes and the things that we see, feel and hear are evidently *partial* in nature (since our powers are limited and the world is large), Montague Semantics is *total*. It treats the objects of the attitudes as total sets of total worlds and since e.g. the sentences

(1) John walks
(2) John walks and Bill talks or doesn't talk

are true in exactly the same total worlds, the sentences

(3) Mary believes that John walks
(4) Mary believes that John walks and Bill talks or doesn't talk

are treated as equivalent too. But Mary may believe (1) without believing anything about Bill at all and so, in particular, without believing (2). Similarly, and perhaps more convincingly, since (1) and (2) are equivalent, by Compositionality

(5) Mary sees John walk
(6) Mary sees John walk and Bill talk or not talk

must be too. But (6) involves that Mary sees Bill, while (5) clearly does not.

Situation Semantics solves these problems in a fundamental way. In this new framework for the semantic analysis of natural languages the meaning of a sentence, a proposition, is no longer equated with a set of total worlds. Instead, partial structures—situations—are offered, in which, for example, (1) may be true but (2) undefined. So these sentences are no longer ('strongly') equivalent and the equivalences between (3) and (4) and between (5) and (6) do no longer follow. In this way Situation Semantics obtains a treatment of the psychological verbs that is at once more real to nature and empirically more adequate than Montague's analysis is. This is an important reason why the new framework offers an interesting alternative to the older one.

Can Montague Grammarians meet the challenge? The question arises whether the basic idea behind Barwise & Perry's analysis—partiality—is

necessarily tied up to Situation Semantics. Cannot this idea be transplanted to the older system?

Although it is nowadays usual to think about possible worlds as total entities deciding every proposition, there is nothing essentially total to the idea of a possible world. The early pioneers of possible world semantics certainly did not think so, as Van Benthem [] has pointed out, nor did all semanticists of a more recent past share the 'totalitarian' perspective: Humberstone [1981], for example, defines a simple and elegant semantics for propositional modal logic based on what he calls *possibilities*—partial possible worlds, entities he associates with regions rather than with points in logical space.

But if we want partiality in Montague Semantics, it will clearly not do to have a partial semantics for propositional modal logic like Humberstone's. We must have a partialized model theory for the full theory of types. It seems hard to find one. The following remarks made by Barwise can be found in Barwise & Perry [1985]:

> It is true that some writers have augmented the theory of possible worlds to add partial possible worlds. However, no one, as far as I know, [...] has worked out the higher-order Montague-like analogue of this theory. I thought about it once. The idea would be to have a part-of relation between partial worlds and look at those higher-type functions that were hereditarily consistent with respect to this part-of relation. However, I found that it became terribly complex once you went beyond the first couple of levels in the type hierarchy, much more complicated than the analogous problem in the theory of partial functions of higher type recursion theory.

Why does it seem so difficult to partialize type theory? I think it is because people try to generalize a version of it that is not very well suited to any generalization at all since it is formulated in a rather unnatural way. Once this unnaturalness is removed, and it can be removed at no cost, generalization to a partial theory of types is completely straightforward.

The standard formulation of type theory, given in Church [1940], is based on unary functions only. Of course, in applications one generally needs functions and relations in more than one argument, but these, we are told, can be *coded* by unary functions. Two steps are needed to code a multi-argument relation. The first is to identify it with its characteristic function, a multi-argument function. This identification is very simple and unobjectionable. The second step—I submit—is highly tricked. It is based on Schönfinkel's observation that there is a one-to-one correspondence between multi-argument functions and certain unary functions of higher type. So, an ordinary three-place relation on individuals like the relation 'to give' is equated with its characteristic function, which, in its turn, is identified with a function from entities to functions from entities to functions from entities to truth-values.

Now, if we consider type hierarchies consisting of partial rather than total functions, Schönfinkel's one-to-one correspondence between multi-argument functions and unary ones breaks down, as the following example, adapted from Tichy [1982], suggests: Let a be some object of type e. Consider two partial functions F_1 and F_2, both of type $(e(ee))$, defined as follows: $F_1(x) = F_2(x)$ = the identity function, if $x \neq a$. $F_1(a)$ is undefined;

$F_2(a)$ is defined as the (ee) function that is undefined for all its arguments. Clearly, $F_1 \neq F_2$. The function F_2 codes the two-place partial function F such that $F(a,y)$ is undefined and $F(x,y) = y$ if $x \neq a$. But if F_1 codes anything at all, it must code F too.

In general, I think it is not a very good idea to put intricate codifications[1] like Schönfinkel's into your logic if they are not absolutely necessary: they complicate the theory. If you confine yourself to applications of the logic you may get used to the complications; but if you are trying to prove things about the logic and try to generalize it, you will find them a hindrance to any real progression. Lambek & Scott [1981] even blame some of type theory's bad reception among mathematicians on the employment of Schönfinkel's Trick:

> Type theory for the foundations of mathematics was used by Russell and Whitehead in 'Principia Mathematica'. However, it did not catch on among mathematicians, who preferred to use ad hoc methods for circumnavigating the paradoxes, as in the systems of Zermelo–Fränkel and Gödel–Bernays, even though very elegant formulations of type theory became available in the work of Church and Henkin.
> One reason for this failure to catch on was an unwise application of Ockham's razor, which got rid of product types with the help of special tricks in one way or another. The resulting system of types, although extremely economical, was awkward to handle.

The general idea, then, is that we should first give a formulation of type theory that is not based on Schönfinkel's identification before we can generalize it to a partial theory of types. This will be done in the next section. In section 2, we'll use the result to obtain a new, but still total, semantics for the PTQ fragment. It will give a consequence relation on the fragment that is provably the same as the one defined in Dowty, Wall & Peters [1981] (DWP henceforth). At that point we'll be ready to generalize. The logic that is defined in section 3 will be a partial type theory in the sense that its models will have partial objects in all their non-basic domains.

The system will be a higher order generalization of existing partial logics (see e.g. Dunn [1976], Belnap [1977], Woodruff [1984], Blamey [1986] and Langholm [1987]). In section 4 we'll show that the logic gives a strong consequence relation on the fragment that cuts out the exceptionable irrelevant entailments. Sentences (1) and (2) will no longer be equivalent (but they will be 'weakly equivalent'). Some of the crucial concepts of classical Situation Semantics will be seen to be available within a generalized version of Montague Grammar and it will turn out to be possible to give an analysis of the psychological verbs within the latter theory along lines suggested by the former.

---

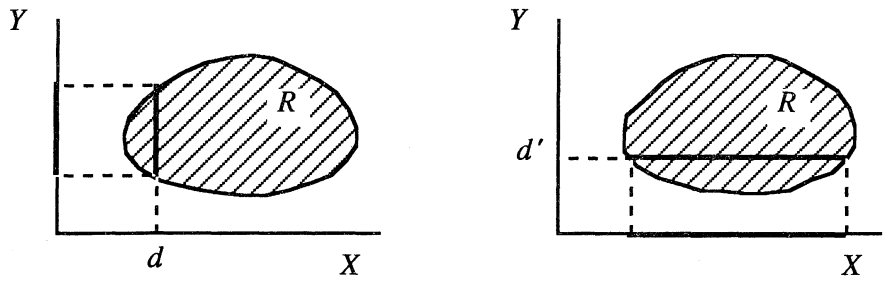
[1] The question whether the Schönfinkel encoding is intricate or not is of course a matter of taste and opinion. Note however that a double recursion is needed if one wants to define the correspondence in any precise way. See the definition of the functions S_α in the proof of Theorem 1. in the Appendix below.

1. Relational type theory

If you want to formalize the theory of types without making use of Schönfinkel's Trick three options are open: First, you can consider type hierarchies consisting of both multi-argument functions and multi-argument relations. This is the most general solution, but we will not need this generality here. Second, you may consider only multi-argument functions. Relations can then be coded by their characteristic functions. Third, you can take only relations. This is the course we shall in fact follow, since it is somewhat simpler than the first and slightly better suited to our present purposes than the second.

At first glance it might seem that the last solution is not liberal enough. In natural language semantics it is often useful to take a functional perspective on things. For example, the *intension* of an expression can fruitfully be seen as a function from possible worlds to extensions. Another example is that functional application seems to be the correct semantic correlate of many, indeed of most, syntactic constructions.

But the obvious fact that functions are indispensable needn't prevent us from using only relations in setting up our logic. It is possible to view any relation as a function. Moreover, it is possible to do this in at least as many ways as the relation has argument places. The following pictures illustrate this phenomenon geometrically:



Let R be some binary relation on the reals, or, equivalently, a set of points in the Euclidean plane. With any point d on the X-axis a set of points $\{y \mid \langle d,y\rangle \in R\}$ on the Y-axis corresponds. This is illustrated in the left picture. Conversely, any d' on the Y-axis gives a set $\{x \mid \langle x,d'\rangle \in R\}$ on the X-axis, as is illustrated in the right picture. So there are two natural ways to see R as a function from the reals to the power set of the reals.

The procedure is of course entirely general:

DEFINITION 1 (Slice Functions). Let R be an n-ary relation and let $0 < k \leq n$. Define the *k-th slice function* F^k_R of R by:
$F^k_R(d) = \{\langle d_1,\ldots,d_{k-1},d_{k+1},\ldots,d_n\rangle \mid \langle d_1,\ldots,d_{k-1},d,d_{k+1},\ldots,d_n\rangle \in R)\}$.

So $F^k_R(d)$ is the n-1-ary relation that is obtained from R by fixing its k-th argument place by d. We shall often want to see relations as functions in this way. Note that by the usual set-theoretic definitions (remember that $\langle a\rangle = a$, $\langle\rangle = \varnothing$, $\varnothing = 0$ and $\{\varnothing\} = 1$), if R is a one-place relation, then F^1_R is its characteristic function.

An example: Let *love* be a ternary relation, *love xyi* meaning '*y* loves *x* at index *i*'. This is a relation-in-intension. Relations-in-intension are nor-

mally thought of as functions from possible worlds to extensions; F^3_{love} is this function for *love*. On the other hand, it is natural to see the relation as the function that, when applied to an entity 'Mary' gives the property '*y* loves Mary at index *i*'; this is the function F^1_{love}.

A relational formulation of higher-order logic was given in Orey [1959] (See also Gallin [1975] and Van Benthem & Doets [1983].) The following two definitions give a two-sorted version of Orey's type hierarchies:

DEFINITION 2. The set of *types* is to be the smallest set of strings[2] such that:

i. e and s are types,

ii. if $\alpha_1,...,\alpha_n$ are types ($n \geq 0$), then $\langle\alpha_1...\alpha_n\rangle$ is a type.

DEFINITION 3. A *(standard) frame* is a family of sets $\{D_\alpha \mid \alpha$ is a type$\}$ such that $D_e \neq \varnothing$, $D_s \neq \varnothing$ and $D_{\langle\alpha_1...\alpha_n\rangle} = P(D_{\alpha_1}\times\cdots\times D_{\alpha_n})$.

The domains D_e and D_s are thought to consist of (*possible*) *individuals* and *possible worlds* respectively. Domains $D_{\langle\alpha_1...\alpha_n\rangle}$ consist of all n-ary relations having D_{α_i} as their i-th domain. Note that by definition 2 the string $\langle\rangle$ is a type, and that $D_{\langle\rangle} = P(\{\varnothing\}) = \{0,1\}$, the set of *truth-values*.

Orey used these frames to interpret the formulae of higher-order predicate logic on. These formulae have a syntax that is essentially that of ordinary predicate logic, be it that quantification over objects of arbitrary type is allowed. There is no λ–abstraction and the syntax allows only one type (type $\langle\rangle$, the type of formulae) of complex expressions, while Montague Grammar assigns many different types to linguistic phrases. It is therefore clear that the higher-order predicate logic as it stands does not fit our purposes. On the other hand, the syntax of ordinary functional type logic does satisfy our needs. So let's keep Church's syntax but attach Orey's models to it. This can be done and the result, a logic I have dubbed TT (for 'type theory'), TT_2 in the two-sorted case, is described in Muskens [1986]. I'll give a brief exposition of it in this section.

We start with the syntax. Assume for each type the existence of a denumerable infinity of variables of that type; also assume for each type a countably infinite set of constants, ordered in the way of the natural numbers. From these we can build up terms by the following clauses, virtually the usual ones.

DEFINITION 4. Define, for each α, the set of *terms* of that type by the following inductive definition:

i. Every constant or variable of any type is a term of that type;

ii. If φ and ψ are terms of type $\langle\rangle$ (*formulae*) then $\neg\varphi$ and $(\varphi \wedge \psi)$ are formulae;

iii. If φ is a formula and x is a variable of any type, then $\forall x\varphi$ is a formula;

---

[2] A technical subtlety: types are strings of symbols over the alphabet $\{e, s, \langle, \rangle\}$, the angled brackets do not form part of the usual notation for ordered tuples. This is important since e.g. we don't want to equate the type $\langle e\rangle$ with e, while we do equate the ordered 1-tuple $\langle a\rangle$ with a.

iv. If A is a term of type $\langle\beta\alpha_1...\alpha_n\rangle$ and B is a term of type β, then (AB) is a term of type $\langle\alpha_1...\alpha_n\rangle$;
v. If A is a term of type $\langle\alpha_1...\alpha_n\rangle$ and x is a variable of type β then $\lambda x(A)$ is a term of type $\langle\beta\alpha_1...\alpha_n\rangle$;
vi. If A and B are terms of the same type, then $(A = B)$ is a formula.

Logical operators that are not mentioned here will have their usual definitions. Parentheses will be omitted where this can be done without creating confusion, on the understanding that association is to the left. So instead of writing $(...(AB_1)...B_n)$ I'll write $AB_1...B_n$.

We define *models* to be tuples $\langle F, I\rangle$, consisting of a frame $F = \{D_\alpha\}_\alpha$ and an *interpretation* function I, having the set of constants as its domain, such that $I(c) \in D_\alpha$ for each constant c of type α. An *assignment* a for a model $\langle\{D_\alpha\}_\alpha, I\rangle$ is a function from the set of all variables such that $a(x) \in D_\alpha$ for each variable x of type α. The assignment $a[d/x]$ is just like a but with the possible exception that $a[d/x](x) = d$.

To evaluate our terms on these relational models we can use the slice functions defined above. We simply let the value of a term AB be the result of applying (the first slice function of) the value of A to the value of B. Terms of the form λxA we evaluate by an inverse procedure.

DEFINITION 5. The *value* $\|A\|^{M,a}$ of a term A on a model M under an assignment a is defined in the following way (To improve readability I shall sometimes write $\|A\|$ for $\|A\|^{M,a}$):

i. $\|c\| = I(c)$ if c is a constant; $\|x\| = a(x)$ if x is a variable;
ii. $\|\neg\varphi\| = 1 \setminus \|\varphi\|$; $\|\varphi \wedge \psi\| = \|\varphi\| \cap \|\psi\|$;
iii. $\|\forall x_\alpha\varphi\|^{M,a} = \bigcap_{d \in D_\alpha} \|\varphi\|^{M,a[d/x]}$;
iv. $\|AB\| = F^1_{\|A\|}(\|B\|)$;
v. $\|\lambda x_\beta A\|^{M,a}$ = the unique relation R such that for all $d \in D_\beta$: $F^1_R(d) = \|A\|^{M,a[d/x]}$;
vi. $\|A = B\| = 1$ iff $\|A\| = \|B\|$.

For reasons that will become clear in section 3, I have taken care to couch clauses ii. and iii. completely in terms of Boolean operations, but clearly the clauses are equivalent to any of the more usual ones. Note that the following two identities hold.

$\|AB\| = \{\langle d_1,...,d_n\rangle \mid \langle\|B\|,d_1,...,d_n\rangle \in \|A\|\}$
$\|\lambda x_\beta A\|^{M,a} = \{\langle d,d_1,...,d_n\rangle \mid d \in D_\beta \text{ and } \langle d_1,...,d_n\rangle \in \|A\|^{M,a[d/x]}\}$.

The notion of logical consequence is again expressed in terms of Boolean operations. It is defined for terms of arbitrary relational type, not only for formulae.

DEFINITION 6. Let Γ and Δ be sets of terms of some type $\alpha = \langle\alpha_1...\alpha_n\rangle$. Γ *entails* Δ , $\Gamma \models \Delta$, if $\bigcap_{A\in\Gamma}\|A\|^{M,a} \subseteq \bigcup_{B\in\Delta}\|B\|^{M,a}$ for all models M and assignments a to M.

What is the relation between this logic and the more standard formulations of type theory? Let's compare the logic with Gallin's system TY_2 (which is just a two-sorted version of Church's original theory). First note that we may associate TY_2 types with our relational ones (I write the functional types with round parentheses).

DEFINITION 7. Define the function Σ (Σ is for Schönfinkel) taking types to TY_2 types by the following double recursion:

I $\quad \Sigma(e) = e,\ \Sigma(s) = s$

II i. $\quad \Sigma(\langle\rangle) = t$

ii. $\quad \Sigma(\langle\alpha_1 \ldots \alpha_n\rangle) = (\Sigma(\alpha_1)\Sigma(\langle\alpha_2 \ldots \alpha_n\rangle))$ if $n \geq 1$.

It is not difficult to see that, if we stipulate that all constants (variables) of any type α are constants (variables) of TY_2 type $\Sigma(\alpha)$ as well, then all terms of any type α are TY_2 terms of TY_2 type $\Sigma(\alpha)$. So our syntax is just a part of the TY_2 syntax. Of course not all TY_2 terms are TT_2 terms by this identification since Σ is not onto: only those functional types in which no e or s immediately precedes a right bracket are values of Σ. However, on TT_2 sentences both logics give the same entailment relation.

THEOREM 1. Let $\Gamma \cup \{\varphi\}$ be a set of (TT_2) sentences then
$\Gamma \models \varphi$ in TT_2 iff $\Gamma \models \varphi$ in TY_2.

All proofs will be given in the Appendix below.

2. A relational semantics for PTQ

Is it possible to do Montague semantics using the relational logic introduced in the previous section? To this question we shall address ourselves now and the answer will turn out to be positive: there is a simple way to translate the PTQ fragment[3] of English—that paradigm of Montague grammar—into the logic TT_2. Moreover, the relation of entailment that is induced on the fragment by the new translation is provably orthodox: it equals the entailment relation given in the text book DWP[4,5].

---

[3]For reasons of exposition I omit Montague's 'rules of tense'. They can however be included without difficulty.

[4]Although it is equivalent to the DWP version of Montague Grammar, my formalization of PTQ will be considerably simpler. Montague Grammar is often considered to be a difficult and perhaps even somewhat esoteric subject, but much of the apparent complexity is really a consequence of certain infelicitous choices in its formalization. We shall remove some of these infelicities.

[5]The main difference between Montague's own semantics for the PTQ fragment and that given in DWP concerns the use of 'individual concepts' (type (se) functions). Montague employed these in order to circumvent certain difficulties with sentences like Partee's *The temperature is ninety but it is rising*, but DWP skip the use of individual concepts alltogether, following Bennett [1974], who saw that these did not only complicate the theory considerably, but moreover created as many problems as they were supposed to solve. For a careful discussion see DWP; for a contrary opinion Janssen [1984]. Those who do not wish to accept Bennett's Simplification should note that we could easily reintroduce individual concepts in the present relational setting: simply treat them as $\langle se\rangle$ type relations that happen to be functional.

In order to be able to prove the last statement (which I'll do in the Appendix), we must give a somewhat altered (but equivalent) definition of the PTQ syntax. Let's start with redefining the categories:

DEFINITION 8 (Categories).
i. e is a category; t is a category;
ii. If A and B are categories and $A \neq e$, then A/B and $A//B$ are categories.

This is of course very close to Montague's definition, but we don't allow categories to end in e. The reason will become apparent below. Next, using the familiar abbreviations for category notations, we define the sets B_A of *basic expressions of category A*:

DEFINITION 9 (Basic Expressions).

B_{IV}	=	{run, walk, talk}
B_T	=	{John, Mary, Bill, he_0, he_1, he_2,...}
B_{TV}	=	{find, lose, eat, love, date, be, seek, conceive}
B_{IAV}	=	{rapidly, slowly, voluntarily, allegedly}
B_{CN}	=	{man, woman, park, fish, pen, unicorn}
B_{DET}	=	{every, the, a}
$B_{t/t}$	=	{necessarily}
$B_{IAV/T}$	=	{in, about}
$B_{IV/t}$	=	{believe that, assert that}
$B_{IV//IV}$	=	{try to, wish to}
B_A	=	$\varnothing$ if A is any category other than those mentioned above.

As is well known, in the PTQ paper Montague deviated slightly from the general program he had set out in 'Universal Grammar' (Montague [1970]). Meanings in PTQ are not attached to an intermediate 'disambiguated language' but are assigned to the ambiguous expressions of English. From a formal point of view this is unfortunate, if only because it robs us of an important method of definition—definition by induction on the complexity of syntactic objects. We can have no such induction without unique readability.

Montague himself remarks that this situation can be remedied by taking the *analysis trees* rather than the phrases of the fragment as the elements of the syntactic algebra. But while this solves the problem theoretically it hardly solves it in any practical sense. Since meaningful expressions of English can be directly read from the nodes of Montague's analysis trees, the syntactical peculiarities of that language must be encoded in the latter's definition. A complete and reasonably precise definition of these trees consequently turns out to be rather formidable[6].

We therefore follow Dowty [1982] in separating the *grammatical rules* of the language from its *syntactical operations*. We let the grammatical rules generate the analysis trees of the fragment, analysis trees that are now stripped of much syntactical information.

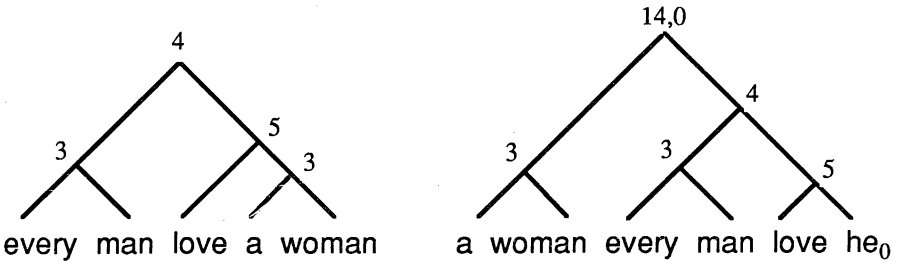
---

[6]See DWP, Chapter 8, where two clauses in the definition—out of the seventeen that would be needed—take up half a printed page.

DEFINITION 10 (Analysis Trees). For each category A the set AT_A of *analysis trees of category A* is defined recursively with the help of the grammatical rules G1 – G17 below:

G1. $B_A \subseteq AT_A$ for every category A.
G2. If $\xi \in AT_{CN}$ and $\vartheta \in AT_t$ then $[\xi\vartheta]^{2,n} \in AT_{CN}$ (for each n).
G3. If $\xi \in AT_{DET}$ and $\vartheta \in AT_{CN}$ then $[\xi\vartheta]^3 \in AT_T$.
G4. If $\xi \in AT_T$ and $\vartheta \in AT_{IV}$ then $[\xi\vartheta]^4 \in AT_t$.
G5. If $\xi \in AT_{TV}$ and $\vartheta \in AT_T$ then $[\xi\vartheta]^5 \in AT_{IV}$.
G6. If $\xi \in AT_{IAV/T}$ and $\vartheta \in AT_T$ then $[\xi\vartheta]^6 \in AT_{IAV}$.
G7. If $\xi \in AT_{IV/t}$ and $\vartheta \in AT_t$ then $[\xi\vartheta]^7 \in AT_{IV}$.
G8. If $\xi \in AT_{IV//IV}$ and $\vartheta \in AT_{IV}$ then $[\xi\vartheta]^8 \in AT_{IV}$.
G9. If $\xi \in AT_{t/t}$ and $\vartheta \in AT_t$ then $[\xi\vartheta]^9 \in AT_t$.
G10. If $\xi \in AT_{IV/IV}$ and $\vartheta \in AT_{IV}$ then $[\xi\vartheta]^{10} \in AT_{IV}$.
G11. If $\xi, \vartheta \in AT_t$ then $[\xi\vartheta]^{11a}, [\xi\vartheta]^{11b} \in AT_t$.
G12. If $\xi, \vartheta \in AT_{IV}$ then $[\xi\vartheta]^{12a}, [\xi\vartheta]^{12b} \in AT_{IV}$.
G13. If $\xi, \vartheta \in AT_T$ then $[\xi\vartheta]^{13} \in AT_T$.
G14. If $\xi \in AT_T$ and $\vartheta \in AT_t$ then $[\xi\vartheta]^{14,n} \in AT_t$ (for each n).
G15. If $\xi \in AT_T$ and $\vartheta \in AT_{CN}$ then $[\xi\vartheta]^{15,n} \in AT_{CN}$ (for each n).
G16. If $\xi \in AT_T$ and $\vartheta \in AT_{IV}$ then $[\xi\vartheta]^{16,n} \in AT_{IV}$ (for each n).
G17. If $\xi \in AT_T$ and $\vartheta \in AT_{IV}$ then $[\xi\vartheta]^{17} \in AT_t$.

This definition produces numbered bracketings like [[every man]^3[love [a woman]^3]^5]^4 and [[a woman]^3[[every man]^3[love he$_0$]^5]^4]^{14,0} (both elements of AT_t), dully keeping track of the basic expressions that were used and the rules by which these were combined. The results are just ordinary analysis trees with all meaningful expressions taken from the non-terminal nodes. In a more tree-like representation the two analysis trees just mentioned would look like this:



To obtain phrases of English, Montague's syntactic operations $F_3 - F_{11}$ (given in PTQ) can be used.

DEFINITION 11 (Phrases). For each analysis tree ξ, define a phrase $\sigma(\xi)$ by induction on the complexity of analysis trees:

S1. $\sigma(\xi) = \xi$ if $\xi \in B_A$
S2. $\sigma([\xi\vartheta]^{2,n}) = F_{3,n}(\sigma(\xi),\sigma(\vartheta))$
S3. $\sigma([\xi\vartheta]^3) = F_6(\sigma(\xi),\sigma(\vartheta))$
S4. $\sigma([\xi\vartheta]^4) = F_4(\sigma(\xi),\sigma(\vartheta))$
S5. $\sigma([\xi\vartheta]^5) = F_5(\sigma(\xi),\sigma(\vartheta))$
S6. $\sigma([\xi\vartheta]^6) = F_5(\sigma(\xi),\sigma(\vartheta))$
S7. $\sigma([\xi\vartheta]^7) = F_6(\sigma(\xi),\sigma(\vartheta))$
S8. $\sigma([\xi\vartheta]^8) = F_6(\sigma(\xi),\sigma(\vartheta))$

S9. $\sigma([\xi\vartheta]^{9}) = F_6(\sigma(\xi),\sigma(\vartheta))$
S10. $\sigma([\xi\vartheta]^{10}) = F_7(\sigma(\xi),\sigma(\vartheta))$
S11. $\sigma([\xi\vartheta]^{11a}) = F_8(\sigma(\xi),\sigma(\vartheta))$
$\sigma([\xi\vartheta]^{11b}) = F_9(\sigma(\xi),\sigma(\vartheta))$
S12. $\sigma([\xi\vartheta]^{12a}) = F_8(\sigma(\xi),\sigma(\vartheta))$
$\sigma([\xi\vartheta]^{12b}) = F_9(\sigma(\xi),\sigma(\vartheta))$
S13. $\sigma([\xi\vartheta]^{13}) = F_9(\sigma(\xi),\sigma(\vartheta))$
S14. $\sigma([\xi\vartheta]^{14,n}) = F_{10,n}(\sigma(\xi),\sigma(\vartheta))$
S15. $\sigma([\xi\vartheta]^{15,n}) = F_{10,n}(\sigma(\xi),\sigma(\vartheta))$
S16. $\sigma([\xi\vartheta]^{16,n}) = F_{10,n}(\sigma(\xi),\sigma(\vartheta))$
S17. $\sigma([\xi\vartheta]^{17}) = F_{11}(\sigma(\xi),\sigma(\vartheta))$

The function σ *ambiguates*, it is not one-to-one: By repeated applications of these rules we may see that σ([[every man]^3[love [a woman]^3]^5]^4) = σ([[a woman]^3[[every man]^3[love he$_0$]^5]^4]^{14,0}) = every man loves a woman.

The advantage of the little detour is not only technical. Disconnecting the syntactical operations of the language from its grammatical rules, an idea Dowty traces back to Curry, makes a rather clean distinction between those parts of the grammar that are language-independent (the 'tectogrammatics') and those that are not (the 'phenogrammatics'). While definition 10 could be used in setting up a fragment of any language, be it English, Dutch or Swahili, definitions 9 and 11 are of course highly particular to the English language. For more motivation along these lines see Dowty's work.

This ends our discussion of the PTQ syntax and we now turn to the semantics of the fragment. Wishing to formalize the way meanings are attached to expressions of English, we shall give a translation function $^\circ$ sending analysis trees to terms of our relational logic TT_2 (the latter standing proxy for meanings). In this way each phrase Φ will be associated with a set of meanings $\{\xi^\circ \mid \sigma(\xi) = \Phi\}$.

Analysis trees of a category A will be sent to terms of a fixed type that can be obtained from A using the following rule:

DEFINITION 12 (Category-to-type Rule)[7].
i. $\tau(e) = e$; $\tau(t) = \langle s\rangle$;
ii. $\tau(A/B) = \tau(A//B) = \tau(B)*\tau(A)$,
where $\beta*\langle\alpha_1\ldots\alpha_n\rangle = \langle\beta\alpha_1\ldots\alpha_n\rangle$ for all $\beta,\alpha_1,\ldots,\alpha_n$.

The idea is that the meaning of a sentence is a proposition (a type $\langle s\rangle$ object, a set of indices), that the meaning of an e-type name (not actually occurring in the fragment) is a possible individual and that the meaning of an expression of any category expecting a B to form an A has a type that expects a $\tau(B)$ to form a $\tau(A)$. Note that, contrary to what is usual[8], the translation of an expression will be its meaning, its intension, not its ex-

[7]Lewis [1974] gives the following category-to-type rule (using functional types of course):
i. $\tau_L(e) = (se)$; $\tau_L(t) = (st)$;
ii. $\tau_L(A/B) = \tau_L(A//B) = (\tau_L(B)\tau_L(A))$
Adopting Bennett's Simplification one obtains the following rule:
i. $\tau'(e) = e$; $\tau'(t) = (st)$;
ii. $\tau'(A/B) = \tau'(A//B) = (\tau'(B)\tau'(A))$
Our rule is equivalent to this last one in the sense that $\tau'(A) = \Sigma(\tau(A))$, where the function Σ is as in the previous section.

[8]See however Lewis [1974] and Thomason's Introduction to Montague [1974].

tension. Of course, the extension of an expression at any index can always be obtained from its intension.

TABLE 1.

Category A		$\tau(A)$	$f(A)$
t		$\langle s \rangle$	t
t / e	(IV or VP)	$\langle es \rangle$	(et)
$t // e$	(CN)	$\langle es \rangle$	(et)
t / t		$\langle\langle s \rangle s \rangle$	$((st)t)$
t / IV	(T or NP)	$\langle\langle es \rangle s \rangle$	$((s(et))t)$
IV / t		$\langle\langle s \rangle es \rangle$	$((st)(et))$
IV / IV	(IAV)	$\langle\langle es \rangle es \rangle$	$((s(et))(et))$
$IV // IV$		$\langle\langle es \rangle es \rangle$	$((s(et))(et))$
IV / T	(TV)	$\langle\langle\langle es \rangle s \rangle es \rangle$	$((s((s(et))t))(et))$
T / CN	(DET)	$\langle\langle es \rangle\langle es \rangle s \rangle$	$((s(et))((s(et))t))$
IAV / T		$\langle\langle\langle es \rangle s \rangle\langle es \rangle es \rangle$	$((s((s(et))t))((s(et))(et)))$

In Table 1. above the values of the function τ are written out for those categories that are actually used in the PTQ fragment. In the third column we give the IL types that are assigned to these categories in DWP for comparison.

To see that objects of type $\tau(A)$ are indeed the kind of objects one would like to assign to expressions of category A we may use the slice functions discussed in the previous section. For example, one would like the intension of a CN or an IV to be a property (of individuals), a function from possible worlds to sets of entities. The second slice function of any type $\langle es \rangle$ object is just this kind of thing. In a similar way we see that the meaning of a term is a property of properties (a quantifier), the meaning of an IAV a function from properties to properties etcetera. The intension of a determiner can either be seen as a function from properties to quantifiers (use the first slice function) or as a relation-in-intension between properties (use the third). Both perspectives have been advocated in the literature.

We are now ready to give our translation function.

DEFINITION 13 (Translation). Let h be some one-to-one function having the set of basic expressions other than those mentioned explicitly in rule T1 below as its domain, such that $h(\xi)$ is a constant of type $\tau(A)$ if ξ is of category A. Let $john_e, find_{\langle ees \rangle}$ etc. be constants of the types indicated, mutually distinct and distinct from all the values of h. For each analysis tree ξ define its *translation* ξ° by induction on the complexity of analysis trees:

T1. If ξ is in the domain of h then $\xi^\circ = h(\xi)$;
$\text{John}^\circ = \lambda P(P\ john_e)$, $\text{Mary}^\circ = \lambda P(P\ mary_e)$, $\text{Bill}^\circ = \lambda P(P\ bill_e)$,
$\text{he}_n{}^\circ = \lambda P(Px_n)$;
$\text{find}^\circ = \lambda Q\lambda y(Q\lambda x(find_{\langle ees \rangle} xy))$, $\text{lose}^\circ = \lambda Q\lambda y(Q\lambda x(lose_{\langle ees \rangle} xy))$,
$\text{eat}^\circ = \lambda Q\lambda y(Q\lambda x(eat_{\langle ees \rangle} xy))$, $\text{love}^\circ = \lambda Q\lambda y(Q\lambda x(love_{\langle ees \rangle} xy))$,
$\text{date}^\circ = \lambda Q\lambda y(Q\lambda x(date_{\langle ees \rangle} xy))$,
$\text{be}^\circ = \lambda Q\lambda y(Q\lambda x\lambda i(x=y))$,
$\text{seek}^\circ = \lambda Q(try_{\langle\langle es \rangle es \rangle}\ \lambda y(Q\lambda x(find_{\langle ees \rangle} xy)))$;

every° = $\lambda P_1 \lambda P_2 \lambda i\, \forall x(P_1 xi \rightarrow P_2 xi)$,
a° = $\lambda P_1 \lambda P_2 \lambda i\, \exists x(P_1 xi \wedge P_2 xi)$,
the° = $\lambda P_1 \lambda P_2 \lambda i\, \exists x(\forall y(P_1 yi \leftrightarrow x{=}y) \wedge P_2 xi)$;
necessarily° = $\lambda p \lambda i\, \forall j(pj)$;
in° = $\lambda Q \lambda P \lambda y(Q \lambda x(in_{\langle e\langle es\rangle es\rangle} xPy))$;

T2. $([\xi\vartheta]^{2,n})^\circ = \lambda x_n \lambda i(\xi^\circ x_n i \wedge \vartheta^\circ i)$;

T3 – T10. $([\xi\vartheta]^k)^\circ = \xi^\circ \vartheta^\circ$ if $3 \leq k \leq 10$;

T11. $([\xi\vartheta]^{11a})^\circ = \lambda i(\xi^\circ i \wedge \vartheta^\circ i)$;
$([\xi\vartheta]^{11b})^\circ = \lambda i(\xi^\circ i \vee \vartheta^\circ i)$;

T12. $([\xi\vartheta]^{12a})^\circ = \lambda x \lambda i(\xi^\circ xi \wedge \vartheta^\circ xi)$;
$([\xi\vartheta]^{12b})^\circ = \lambda x \lambda i(\xi^\circ xi \vee \vartheta^\circ xi)$;

T13. $([\xi\vartheta]^{13})^\circ = \lambda P \lambda i(\xi^\circ Pi \vee \vartheta^\circ Pi)$;

T14. $([\xi\vartheta]^{14,n})^\circ = \xi^\circ \lambda x_n(\vartheta^\circ)$;

T15. $([\xi\vartheta]^{15,n})^\circ = \lambda y(\xi^\circ \lambda x_n(\vartheta^\circ y))$;

T16. $([\xi\vartheta]^{16,n})^\circ = \lambda y(\xi^\circ \lambda x_n(\vartheta^\circ y))$;

T17. $([\xi\vartheta]^{17})^\circ = \lambda i \neg \xi^\circ \vartheta^\circ i$;

Some notational conventions. In the above definition as well as in the sequel of this paper we let x, y and z range over individuals, i and j over indices, p and q over propositions (type $\langle s\rangle$ objects), P over properties (type $\langle es\rangle$ objects, type $(s(et))$ objects in IL and TY_2) and Q over quantifiers (type $\langle\langle es\rangle s\rangle$ or type $(s((s(et))t))$ objects). I write *run*, *love* etc. for the values of h. If Ξ is a set of analysis trees I denote $\{\xi^\circ \mid \xi \in \Xi\}$ by Ξ° and use similar conventions in similar cases.

It is now a simple exercise in lambda-conversion to see for example that the translation of the tree [[every man]^3[love[a woman]^3]^5]^4 is equivalent to the term $\lambda i \forall x(man\ xi \rightarrow \exists y(woman\ yi \wedge love\ xyi))$ while ([[a woman]^3[[every man]^3[love he$_0$]^5]^4]^{14,0})° is equivalent to $\lambda i \exists y(woman\ yi \wedge \forall x(man\ xi \rightarrow love\ xyi))$.

Using the somewhat generalized form of the definition of entailment given in the previous section, we say that an analysis tree ϑ of any category *follows from* a set of trees Ξ of the same category if and only if $\Xi^\circ \models \vartheta^\circ$. For analysis trees of the sentence category t this amounts to stipulating that at each world at which all propositions expressed by the premises are true the proposition expressed by the conclusion is true.

This entailment relation is equivalent to the one given in DWP. The reader may verify this either by working out some of his favourite examples or by reading the proof of the following theorem in the Appendix.

THEOREM 2. For each analysis tree ξ let ξ' be the translation it is given in DWP. Let Δ be the following set of IL-sentences (all meaning postulates in DWP):

$\exists x \Box[x = c]$, where c is any constant of type e;
$\exists S_{(s(e(et)))} \forall x \forall Q \Box[\delta(x,Q) \leftrightarrow Q\{\hat{}\lambda y[S\{x,y\}]\}]$, where δ is find', love', lose', eat' or date';
$\exists G_{(s(e(\langle s(et))(et))))} \forall Q \forall P \forall x \Box[\mathsf{in}'(Q)(P)(x) \leftrightarrow Q\{\hat{}\lambda y[[\check{}G](y)(P)(x)]\}]$;
$\forall x \forall Q \Box[\mathsf{seek}'(x,Q) \leftrightarrow \mathsf{try}'(x, \hat{}[\mathsf{find}'(Q)])]$.

Let $\Xi \cup \{\vartheta\}$ be a set of analysis trees, then:

$\Xi^\circ \models \vartheta^\circ$ in TT_2 iff $\Xi' \cup \Delta \models \vartheta'$ in IL.

3. Partial type theory

Let's go partial. One of the most basic assumptions of classical logic is the assumption that a sentence is false if and only if it is not true. Most authors of logic texts simply let the word 'false' be an abbreviation of 'not true', thus excluding two possibilities: (a) that a sentence is neither true nor false, and (b) that a sentence is both true and false. Partial logics are logics in which this central assumption has been dropped and in which at least one of (a) and (b) is allowed.

Should we allow both (a) and (b) or will it do to allow just (a) or just (b)? Between the last two possibilities there is not much to choose since under certain reasonable assumptions a logic that allows sentences to be both true and false but does not allow them to be neither will be isomorphic to a logic that does the reverse. But the choice between a logic that allows all four possible combinations of truth-values (true and not false, false and not true, neither true nor false and both true and false) and one that allows only three of these is real. It is a choice that must be made in any paper on partial logic and that has been made in a different way by different authors.

I shall allow both overdefinedness and underdefinedness of sentences here. The same choice has been made in Situation Semantics (see Perry [1984]) and is perhaps preferable from a purely formal, esthetic point of view. However, readers who would prefer to go the other way will have no difficulty in making the minimal adaptations to the logic presented below that are needed to make it three-valued. They will also find that most of the applications of the four-valued logic in the next section will carry over to their three-valued variant without any problems.

If there are four combinations of truth-values, then how do the truth and falsity of a complex sentence depend on the truth and falsity of its parts? A stunningly simple answer to this question for the case of propositional logic was given by Dunn [1976]. Truth and falsity can be computed just as it is done in classical logic, but one has to separate truth conditions and falsity conditions:

i. $\neg\varphi$ is true if and only if φ is false,
   $\neg\varphi$ is false if and only if φ is true;
ii. $\varphi \wedge \psi$ is true if and only if φ is true and ψ is true,
   $\varphi \wedge \psi$ is false if and only if φ is false or ψ is false;
iii. $\varphi \vee \psi$ is true if and only if φ is true or ψ is true,
   $\varphi \vee \psi$ is false if and only if φ is false and ψ is false.

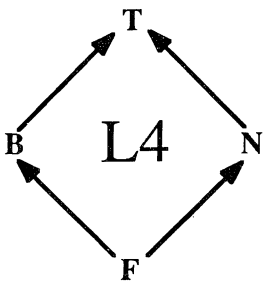
These evaluation rules lead to the following truth-tables, known as the *Extended Strong Kleene* truth-tables for obvious reasons (see Anderson & Belnap [1975], Belnap [1977] for these tables). I write **T** for the combination of values 'true and not false', **F** for 'false and not true', **N** for 'neither true nor false' and **B** for 'both true and false'.

$\wedge$	T	F	N	B
T	T	F	N	B
F	F	F	F	F
N	N	F	N	F
B	B	F	F	B

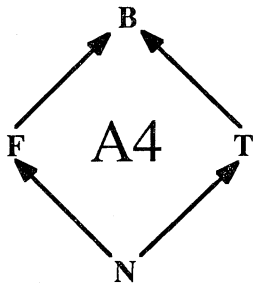
$\vee$	T	F	N	B
T	T	T	T	T
F	T	F	N	B
N	T	N	N	T
B	T	B	T	B

$\neg$	
T	F
F	T
N	N
B	B

It is easy to see that $\wedge$ and $\vee$, or rather the operations associated with these symbols by the truth-tables given above, now form a distributive lattice on the set {**T**,**F**,**N**,**B**}. This lattice is depicted as L4 below.



Logical lattice



Approximation lattice

L4 is discussed extensively in Belnap [1977], who also discusses a second important distributive lattice, borrowed from Scott and shown as A4 above. It can be obtained if we order the truth-combinations **T**,**F**,**N** and **B** by the relation 'approximates the information in'. Meet and join in this lattice give rise to the following truth-tables:

$\otimes$	T	F	N	B
T	T	N	N	T
F	N	F	N	F
N	N	N	N	N
B	T	F	N	B

$\oplus$	T	F	N	B
T	T	B	T	B
F	B	F	F	B
N	T	F	N	B
B	B	B	B	B

The connective $\otimes$ is Blamey's [1986] so-called *interjunction*. It has the truth-conditions of $\wedge$ but the falsity-conditions of $\vee$, while its dual has the truth-conditions of $\vee$ and the falsity-conditions of $\wedge$.

If we add negation to L4, we obtain a structure that conforms to almost all the customary axioms of the theory of Boolean algebras. In fact the following list of axioms can then be seen to hold:

i.	The axioms for distributive lattices;		
ii.	$0 + a = a$,	$1 \cdot a = a$;	(zero and one)
iii.	$a'' = a$;		(double negation)
iv.	$(a \cdot b)' = a' + b'$,	$(a + b)' = a' \cdot b'$,	(De Morgan)
v.	$0' = 1$,	$1' = 0$.	

There are two axioms that do not hold and that, of course, we don't want to hold: $a + a' = 1$ and $a \cdot a' = 0$. If we would add them, we would get a full set of axioms for Boolean algebras.

Adding negation to A4 on the other hand gives a structure that, although it obeys i. – iii., does not satisfy any of the axioms in iv. and v. (for example $(\mathbf{T} \cdot \mathbf{F})' = \mathbf{N}$ but $\mathbf{T}' + \mathbf{F}' = \mathbf{B}$). However, it does conform to the following closely related axioms:

iv'. $(a + b)' = a' + b'$, $\quad (a \cdot b)' = a' \cdot b'$,
v'. $0' = 0$, $\quad 1' = 1$.

I shall call any structure that obeys axioms i. – v. above a *Kleene algebra*, while structures that satisfy i. – iii., iv'. and v'. will be called *approximation algebras*.

Now let us revise the logic given in section 1 keeping the preceding discussion in mind. The most basic notion that was used in that section was the notion of a relation; here is a partialized version of it.

DEFINITION 14 (Partial relations). An n-ary *partial relation* R on domains $D_1,...,D_n$ is a tuple $\langle R^+, R^-\rangle$ of relations $R^+, R^- \subseteq D_1 \times \cdots \times D_n$. The relation R^+ is called R's *denotation*; R^- is called R's *antidenotation*; $D_1 \times \cdots \times D_n / (R^+ \cup R^-)$ its *gap*; and $R^+ \cap R^-$ its *glut*. A partial relation is *coherent* if its glut is empty, *total* if its gap is empty, *incoherent* if it is not coherent and *classical* if it is both coherent and total. A unary partial relation is called a *partial set*.

If D is some set then the *partial power set* of D, $PP(D)$, is the set $\{\langle R^+, R^-\rangle \mid R^+, R^- \subseteq D\}$ (or, equivalently, $P(D) \times P(D)$) of all partial sets over D.

The idea is that it is true of a tuple of objects that they stand in a partial relation R if they are in R's denotation and that it is false that they stand in R if they are in its anti-denotation. This of course leaves open the possibility that it is neither true nor false that a given tuple stand in R or that it is both true and false that they do.

The natural structures on the set of truth-combinations $\{\mathbf{T},\mathbf{F},\mathbf{N},\mathbf{B}\}$ described above can be extended to the class of partial relations. The Extended Strong Kleene valuation scheme for example leads to a generalization of the usual Boolean operations on ordinary relations.

DEFINITION 15.
Let $R_1 = \langle R_1^+, R_1^-\rangle$ and $R_2 = \langle R_2^+, R_2^-\rangle$ be partial relations. Define:

$-R_1$	$:= \langle R_1^-, R_1^+\rangle$	(partial complementation)
$R_1 \cap R_2$	$:= \langle R_1^+ \cap R_2^+, R_1^- \cup R_2^-\rangle$	(partial intersection)
$R_1 \cup R_2$	$:= \langle R_1^+ \cup R_2^+, R_1^- \cap R_2^-\rangle$	(partial union)
$R_1 \subseteq R_2$	iff $R_1^+ \subseteq R_2^+$ and $R_2^- \subseteq R_1^-$	(partial inclusion)

Let A be some set of partial relations. Define:

$\bigcap A := \langle \bigcap\{R^+ \mid R \in A\}, \bigcup\{R^- \mid R \in A\}\rangle$
$\bigcup A := \langle \bigcup\{R^+ \mid R \in A\}, \bigcap\{R^- \mid R \in A\}\rangle$

Consider that it is true of a tuple of objects that they stand in the partial relation $-R$ iff it is false that they stand in R, false that they stand in $-R$ iff it is true that they stand in R. It is true of a tuple of objects that they stand in $R_1 \cap R_2$ iff it is both true that they stand in R_1 and that they stand in R_2; false that they stand in $R_1 \cap R_2$ iff it is false that they stand in R_1 or false that they stand in R_2, etcetera. Note that by elementary reasoning a set of partial relations on domains $D_1,...,D_n$ that is closed under the operations $\cap$, $\cup$ and $-$ and that contains the partial relations $\langle \varnothing, D_1 \times \cdots \times D_n \rangle$ and $\langle D_1 \times \cdots \times D_n, \varnothing \rangle$ forms a Kleene algebra. We shall call any Kleene algebra that has this particular form a *natural Kleene algebra on a set of partial relations*.

In a similar way the operations in the approximation lattice A4 lead to the definition of some more operations on partial relations.

DEFINITION 16.
Let $R_1 = \langle R_1^+, R_1^- \rangle$ and $R_2 = \langle R_2^+, R_2^- \rangle$ be partial relations. Define:
$R_1 \sqcap R_2 := \langle R_1^+ \cap R_2^+, R_1^- \cap R_2^- \rangle$
$R_1 \sqcup R_2 := \langle R_1^+ \cup R_2^+, R_1^- \cup R_2^- \rangle$
$R_1 \sqsubseteq R_2$ iff $R_1^+ \subseteq R_2^+$ and $R_1^- \subseteq R_2^-$ (R_1 approximates R_2)
Let A be some set of partial relations. Define:
$\bigsqcap A := \langle \bigcap \{R^+ \mid R \in A\}, \bigcap \{R^- \mid R \in A\} \rangle$
$\bigsqcup A := \langle \bigcup \{R^+ \mid R \in A\}, \bigcup \{R^- \mid R \in A\} \rangle$

This time we see that a set of partial relations on domains $D_1,...,D_n$ that is closed under the operations $\sqcap$, $\sqcup$ and $-$ and that contains the partial relations $\langle \varnothing, \varnothing \rangle$ and $\langle D_1 \times \cdots \times D_n, D_1 \times \cdots \times D_n \rangle$ forms an approximation algebra. Any approximation algebra of this form I'll call a *natural approximation algebra on a set of partial relations*.

By a trivial adaptation of the proof of the Stone Representation Theorem converses to the two statements given above can be seen to hold.

THEOREM 3 (Representation Theorems).

I. Every Kleene algebra is isomorphic to a natural Kleene algebra on a set of partial relations.

II. Every approximation algebra is isomorphic to a natural approximation algebra on a set of partial relations.

Now that we have partial relations and some structure on them let's build up our frames again.

DEFINITION 17 (Frames). A *frame* is a set $\{D_\alpha \mid \alpha \text{ is a type}\}$ such that $D_e \neq \varnothing$, $D_s \neq \varnothing$ and $D_{\langle \alpha_1 ... \alpha_n \rangle} \subseteq PP(D_{\alpha_1} \times \cdots \times D_{\alpha_n})$. A frame is *standard* if $D_{\langle \alpha_1 ... \alpha_n \rangle} = PP(D_{\alpha_1} \times \cdots \times D_{\alpha_n})$ for all $\alpha_1,...,\alpha_n$.

In a (standard) frame each domain $D_{\langle \alpha_1 ... \alpha_n \rangle}$ consists of (all the) partial relations on domains $D_{\alpha_1},...,D_{\alpha_n}$. Note that since the relational domains of a frame are defined as arbitrary subsets of the relevant partial power set they need not be closed under the operations $\cap$, $\cup$, $-$, $\sqcap$ and $\sqcup$. However we

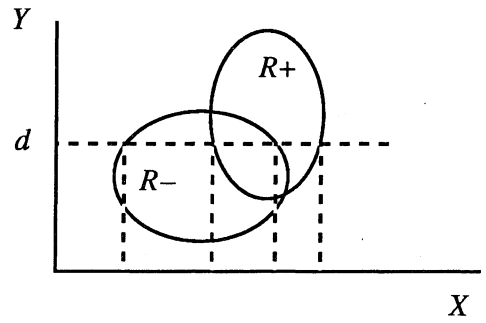
shall shortly restrict our attention to a class of frames in which each relational domain is thus closed.

Checking the set $PP(\{\varnothing\})$ we see the truth-combinations **T**,**F**,**N** and **B** reappear. The set's four elements are $\langle 1,0\rangle$, $\langle 0,1\rangle$, $\langle 0,0\rangle$ and $\langle 1,1\rangle$, which we'll interpret as 'true and not false', 'false and not true', 'true nor false' and 'both true and false' respectively. (If a value's first element is *1*, it gets the interpretation 'true'; if its second element is *1*, it gets the interpretation 'false'.) The operations $\cap$ and $\cup$ give the lattice L4 on the set and that $\sqcap$ and $\sqcup$ give A4; partial complementation gives the Extended Strong Kleene negation.

In section 1. we have seen that there is a converse to Schönfinkel's way of identifying relations with certain unary functions. Instead of trading relations for functions, we have decided to keep the relations and do away with the functions there. But the possibility of doing this rested of course completely on the existence of the slice functions we defined. And so, since we now want to base the logic on partial instead of classical relations, the definition of slice functions must be extended to the former.

DEFINITION 18 (Slice Functions). Let R be an n-ary partial relation and let $0 < k \leq n$. The *k-th slice function* F^k_R of R is defined by $F^k_R(d) = \langle F^k_{R^+}(d), F^k_{R^-}(d)\rangle$.

A picture may again help to see what is going on.



A binary partial relation on the reals can this time be identified with a pair of sets (a partial set) in the Euclidian plane. This pair of sets can be viewed as a (total) function that sends any point on the *Y*-axis to a pair of sets of points on the *X*-axis. So it is a function from points on the *Y*-axis to partial sets of points on the *X*-axis.

We are ready to give a Tarski truth definition evaluating the syntax of ordinary type theory on partial frames. To the language of TT_2 we add two logical constants, # and *, both of formula type $\circ$, that will denote the top and bottom elements of the approximation lattice on $D_\circ$. A *very general model* is a tuple $\langle F, I\rangle$ where $F = \{D_\alpha\}_\alpha$ is a partial frame and I is an interpretation function for F. A very general model is called *standard* if its frame is standard[9].

[9]The name *standard* model is conventional, it should not be taken to imply any preference on our part. On the contrary, it is well known that a restriction to standard models introduces all kinds of inconstructivities into the theory. One gains expressive power and as a consequence one loses the recursive axiomatizability of entailment on the one

When we gave the semantics for TT_2 in section 1, we only made use of Boolean operations, slice functions and identity. We now have partial analogues of the first two of these. So, letting identity be the completely two-valued relation such that $A = B$ is true and not false if the values of A and B are equal and false and not true if they are not, we can define the semantics for our partial type theory PT_2, using virtually the same clauses as in definition 5 above.

DEFINITION 19 (Tarski truth definition). The *value* $\|A\|^{M,a}$ of a term A on a very general model M under an assignment a is defined in the following way:

i. $\|c\| = I(c)$ if c is a constant;
$\|x\| = a(x)$ if x is a variable;

ii. $\|\neg\varphi\| = -\|\varphi\|$;
$\|\varphi \wedge \psi\| = \|\varphi\| \cap \|\psi\|$;
$\|\#\| = \langle 1,1\rangle$;
$\|*\| = \langle 0,0\rangle$;

iii. $\|\forall x_\alpha \varphi\|^{M,a} = \bigcap_{d \in D_\alpha} \|\varphi\|^{M,a[d/x]}$;

iv. $\|AB\| = F^I_{\|A\|}(\|B\|)$;

v. $\|\lambda x_\beta A\|^{M,a}$ = the R such that for all $d \in D_\beta : F^I_R(d) = \|A\|^{M,a[d/x]}$;

vi. $\|A = B\| = \langle 1,0\rangle$ if $\|A\| = \|B\|$
$= \langle 0,1\rangle$ if $\|A\| \neq \|B\|$;

Note that in general there is no guarantee that the value of a term will be an element of the appropriate domain. Still the notion is well-defined. We are however mainly interested in very general models M such that $\|A_\alpha\|^{M,a} \in D_\alpha$ for all terms A and all assignments a; we'll call these *general models* or just *models*.

The definition makes the universal quantifier behave Extended Strong Kleene, just like it makes conjunction and negation do: A formula $\forall x\varphi$ will be true under an assignment a if and only if φ is true under $a[d/x]$ for all objects d in the appropriate domain; $\forall x\varphi$ is false under a if and only if φ is false under $a[d/x]$ for some d.

Note that relational domains of general models are closed under the operations $\cap$, $\cup$, $-$, $\sqcap$ and $\sqcup$. For example if the variables R and R' denote partial relations of some n–ary type, then the term $\lambda x_1 ... x_n (Rx_1 ... x_n \vee R'x_1 ... x_n)$ will denote their partial disjunction. The other cases go similar be it that for the cases of $\sqcap$ and $\sqcup$ we need the connectives $\otimes$ and $\oplus$ discussed above. These we define thus.

$\varphi \otimes \psi \quad := \quad (\varphi \wedge \psi) \vee ((\varphi \vee \psi) \wedge *)$
$\varphi \oplus \psi \quad := \quad ((\varphi \vee \psi) \wedge \#) \vee (\varphi \wedge \psi)$

---

hand and the Löwenheim-Skolem theorem on the other. In view of this we prefer not to make such a restriction at all in the present context. (See Van Benthem&Doets [1983] for a discussion of this point.) Allowing domains to be only a part of the full power set of the relevant Cartesian product seems to fit in nicely with the general spirit of partial semantics.

We define the notion of entailment again in a way that is completely analogous to the way we defined the notion for the total logic, using the partial analogues of the Boolean operations instead of these operations themselves.

DEFINITION 20. Let Γ and Δ be sets of terms of some type $\alpha = \langle \alpha_1 ... \alpha_n \rangle$. Γ *(s–) entails* Δ , $\Gamma \models \Delta$ ($\Gamma \models_s \Delta$), if $\bigcap_{A \in \Gamma} \|A\|^{M,a} \subseteq \bigcup_{B \in \Delta} \|B\|^{M,a}$ for all (standard) models M and assignments a to M.

The resulting notions of logical consequence are 'double-barrelled' in the sense of Blamey [1986], who discusses a similar entailment relation for first-order partial logic. In the case of sentences a set of premises entails a set of conclusions if and only if two conditions hold. The first of these is that in each model in which all premises are true some conclusion is true and the second is that in each model in which all conclusions are false some premise is. This coincides with the definition given in Belnap [1977], who lets entailment go up hill in the logical lattice L4.

Readers familiar with relevance logic and the notion of 'tautological entailment' defined in Anderson & Belnap [1975] will note that there is a close similarity between that relation of logical consequence and the one given here. Let $\varphi \twoheadrightarrow \psi$ be an abbreviation of $(\varphi \wedge \psi) = \varphi$ (I reserve the singleheaded arrow for its ordinary purpose, $\varphi \rightarrow \psi$ is an abbreviation for $\neg\varphi \vee \psi$), then the first of the following matrices lists all answers to the question "does φ entail ψ?" for all possible combinations of values of φ and ψ and the second is a truth-table for $\twoheadrightarrow$.

$\models$	**T**	**F**	**N**	**B**
T	*yes*	*no*	*no*	*no*
F	*yes*	*yes*	*yes*	*yes*
N	*yes*	*no*	*yes*	*no*
B	*yes*	*no*	*no*	*yes*

$\twoheadrightarrow$	**T**	**F**	**N**	**B**
T	**T**	**F**	**F**	**F**
F	**T**	**T**	**T**	**T**
N	**T**	**F**	**T**	**F**
B	**T**	**F**	**F**	**T**

Anderson & Belnap do not wish to distinguish between the meta-language $\models$ and the object-language $\twoheadrightarrow$, we do. But clearly the behaviour of $\twoheadrightarrow$ mirrors that of $\models$. Moreover, our truth-tables for conjunction, disjunction, negation and $\twoheadrightarrow$ are just the characteristic matrices for the logic $\mathbf{E}_{\mathrm{fde}}$ of tautological entailment (see Anderson & Belnap [1975, pp. 161 – 162]).

How wild is the notion $\models$? Not very. We'll characterize its restriction to formulae with the help of a rather familiar-looking calculus of sequents. (In these rules the notation $[A/x]B$ will presuppose that A is substitutable for x in B. I write $\top$ for $* = *$ and $\bot$ for $\neg\top$.)

R

$$\varphi \Rightarrow \varphi$$

Cut

$$\frac{\Gamma, \varphi \Rightarrow \Delta \qquad \Gamma \Rightarrow \varphi, \Delta}{\Gamma \Rightarrow \Delta}$$

Thinning

$$\frac{\Gamma \Rightarrow \Delta}{\Gamma, \varphi \Rightarrow \Delta} \qquad \frac{\Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \varphi}$$

Negation Rules

$$\frac{\{\neg\gamma \mid \gamma \in \Gamma\} \Rightarrow \Delta}{\{\neg\delta \mid \delta \in \Delta\} \Rightarrow \Gamma} \qquad \frac{\Gamma \Rightarrow \{\neg\delta \mid \delta \in \Delta\}}{\Delta \Rightarrow \{\neg\gamma \mid \gamma \in \Gamma\}}$$

Conjunction Rules

$$\text{E}\wedge\ \frac{\Gamma, \varphi, \psi \Rightarrow \Delta}{\Gamma, \varphi \wedge \psi \Rightarrow \Delta} \qquad \text{I}\wedge\ \varphi, \psi \Rightarrow \varphi \wedge \psi$$

Truth value Rules

$\Rightarrow \varphi = \mathsf{T}, \varphi = \bot, \varphi = \#, \varphi = *$ (Excluded Fifth)

$\Rightarrow * = \neg*$

$\Rightarrow \# = \neg\#$

$\Rightarrow \#, *$

Quantifier Rules

$$\text{I}\forall\ \frac{\Gamma \Rightarrow [c/x]\varphi, \Delta}{\Gamma \Rightarrow \forall x\varphi, \Delta} \qquad \text{E}\forall\ \forall x\varphi \Rightarrow [A/x]\varphi$$

provided c doesn't occur in Γ or Δ.

Identity Rules

$\Rightarrow A = A$

$A = B, [A/x]\varphi \Rightarrow [B/x]\varphi$ (Leibniz' Law)

$\Rightarrow A = B, \neg(A = B)$

Lambda Conversion

$\Rightarrow \lambda x(A)B = [B/x]A$

Extensionality

$\forall x(Ax = Bx) \Rightarrow A = B$, provided x is not free in A or B

By inspection it is seen that the rules of this calculus preserve entailment. That is, if $\Gamma_1 \Rightarrow \Delta_1,\ldots,\Gamma_n \Rightarrow \Delta_n \,/\, \Gamma_{n+1} \Rightarrow \Delta_{n+1}$ is a rule (n possibly being equal to 0) and if $\Gamma_1 \models \Delta_1,\ldots,\Gamma_n \models \Delta_n$ hold, then $\Gamma_{n+1} \models \Delta_{n+1}$ holds as well. To see this one must in some cases use the following theorem, which is proved by a straightforward induction on term complexity:

THEOREM 4 (Substitution Theorem). If A is substitutable for x in B then $\|[A/x]B\|^{M,a} = \|B\|^{M,a[d/x]}$, where $d = \|A\|^{M,a}$.

We say that a sequent is *provable* if it belongs to the smallest set of sequents that is closed under the rules above. Note that if $\Gamma \Rightarrow \Delta$ is provable, then Γ and Δ are both finite. Write $\Pi \vdash \Sigma$ if there are Π_0 and Σ_0 such that $\Pi_0 \subseteq \Pi$, $\Sigma_0 \subseteq \Sigma$ and $\Pi_0 \Rightarrow \Sigma_0$ is provable. Clearly, if $\Pi \vdash \Sigma$ then $\Pi \models \Sigma$ (and hence $\Pi \models_s \Sigma$).

In the case of |= the converse also holds.

THEOREM 5 (Generalized Completeness Theorem). Let Π and Σ be sets of formulae then:

$$\Pi \vdash \Sigma \Leftrightarrow \Pi \models \Sigma.$$

4. Situations, persistence and the attitudes

In this section we shall see how some of the basic concepts of Situation Semantics can be formalized within our system and how this formalization can be used to obtain analyses of the propositional attitudes and the neutral perception verbs.

4.1 Situations and the part-of relation

Note that there is now a close correspondence between (abstract) situations and indices. While in the total theory every proposition was either true or false at an index (and never both), propositions may now be neither true nor false or both true and false at a given index.

More formally, we can identify indices with certain partial models. In order to see this, define the *extension* of a term A of type $\langle\alpha_1...\alpha_n s\rangle$ *at* index i in a partial model $M = \langle\{D_\alpha\}_\alpha, I\rangle$ under an assignment a as the partial relation $F_R^{n+1}(i)$ of type $\langle\alpha_1...\alpha_n\rangle$, where $R = \|A\|^{M,a}$. With each $i \in D_s$ we associate a partial model $M_i = \langle\{D_\alpha\}_\alpha, I_i\rangle$, defining the interpretation function I_i thus:

$I_i(c) = I(c)$, if c is a constant of type e or type s

$I_i(c) =$ the extension of c' at index i, if c is the m-th constant of type $\langle\alpha_1...\alpha_n\rangle$, where c' is the m-th constant of type $\langle\alpha_1...\alpha_n s\rangle$

The idea is that we want interpretation functions I_i to link the constants used in the translation of our PTQ fragment ($find_{\langle ees\rangle}$, $man_{\langle es\rangle}$, $try_{\langle\langle es\rangle es\rangle}$,...) to their extensions at index i. Since this cannot be done in a direct way for the obvious reason that a constant and its extension will have different types, we let the constants correspond to constants of the right type (say: $\text{FIND}_{\langle ee\rangle}$, $\text{MAN}_{\langle e\rangle}$, $\text{TRY}_{\langle\langle es\rangle e\rangle}$,...) in a one-to-one fashion and send the latter to the extensions of the former.

The partial models M_i are possible worlds, but they are not necessarily fully specified or coherent worlds, extensions at any index need not be classical. Hence they are partial possible worlds, or, as I shall call them, (possible) situations. Since each M_i is completely given by its index i (given a partial model M), we can identify indices with possible situations as well.

Once we have identified indices with situations, our category-to-type assignment automatically associates categories of expressions in natural language with certain kinds of partial objects that are constructed out of situations and possible individuals. For example, sentences are now associated with partial sets of situations (type $\langle s\rangle$ objects, propositions); common nouns and intransitive verb phrases with functions from situations to partial sets of individuals (properties, type $\langle es\rangle$ objects); noun phrases with

quantifiers, functions from situations to partial sets of properties (type ‹‹*es*›*s*› objects) and so on. The translation function ° defined in section 2 will assign a partial relation of some kind to each analysis tree in each partial model.

There is a natural structure on situations. We say that a partial model $M_i = \langle \{D_\alpha\}_\alpha, I_i \rangle$ is *part of* a partial model $M_j = \langle \{D_\alpha\}_\alpha, I_j \rangle$, or that M_j *extends* M_i, if $I_i(c) = I_j(c)$ for all constants c of types e or s and $I_i(c) \sqsubseteq I_j(c)$ for all constants c of a relational type. Intuitively, M_j extends M_i if M_j is at least as defined as M_i is.

The part-of relation is a relation between *models*, but since we have identified the indices *in* a model with certain models, the type s domain of any model inherits the structure and is ordered by a part-of relation too. In order to be able to express things about this relation in the object language, we introduce a non-logical constant $\leq$ of type ‹*ss*› and want to make it behave like the part-of relation. First let's adopt the following abbreviatory definitions:

$T\varphi$	$:= (\varphi = \mathsf{T} \vee \varphi = \#)$	(φ is true)
$F\varphi$	$:= (\varphi = \bot \vee \varphi = \#)$	(φ is false)
$\varphi \sqsubseteq \psi$	$:= (T\varphi \rightarrow T\psi) \wedge (F\varphi \rightarrow F\psi)$	(φ is an approximation of ψ)

The formula $\forall x_1 \ldots x_n (cx_1 \ldots x_n i \sqsubseteq cx_1 \ldots x_n j)$ expresses that the extension of the constant c at index i approximates its extension at index j. Define Ψ as the conjunction of all formulae $\forall x_1 \ldots x_n (cx_1 \ldots x_n i \sqsubseteq cx_1 \ldots x_n j)$ where c is an element of the finite set LC defined below, which consists of all those constants of relational type that were used in translating the basic expressions of the PTQ fragment plus some extra ones that we shall need in the sequel.

LC := $\{run_{\langle es \rangle}, walk_{\langle es \rangle}, talk_{\langle es \rangle}, find_{\langle ees \rangle}, lose_{\langle ees \rangle}, eat_{\langle ees \rangle}, love_{\langle ees \rangle}, date_{\langle ees \rangle}, conceive_{\langle\langle es \rangle s \rangle es \rangle}, rapidly_{\langle\langle es \rangle es \rangle}, slowly_{\langle\langle es \rangle es \rangle}, voluntarily_{\langle\langle es \rangle es \rangle}, allegedly_{\langle\langle es \rangle es \rangle}, man_{\langle es \rangle}, woman_{\langle es \rangle}, park_{\langle es \rangle}, fish_{\langle es \rangle}, pen_{\langle es \rangle}, unicorn_{\langle es \rangle}, in_{\langle e\langle es \rangle es \rangle}, about_{\langle\langle\langle es \rangle s \rangle\langle es \rangle es \rangle}, believe_{\langle\langle s \rangle es \rangle}, assert_{\langle\langle es \rangle es \rangle}, try_{\langle\langle es \rangle es \rangle}, wish_{\langle\langle es \rangle es \rangle}, BO_{\langle ess \rangle}, KO_{\langle ess \rangle}, AO_{\langle ess \rangle}, see_{\langle ess \rangle}\}$

So Ψ is defined as the finite conjunction of $\forall xy(find\ xyi \sqsubseteq find\ xyj)$, $\forall x(man\ xi \sqsubseteq man\ xj)$, $\forall P \forall x(try\ Pxi \sqsubseteq try\ Pxj)$ etcetera. The following axiom makes $\leq$ behave the way we want it to.

AX1 $\quad \forall ij(i \leq j = \Psi)$

Clearly, in any model satisfying AX1 the value of $\leq$ is a reflexive and transitive classical relation.

4.2 Persistence

Let AX be the set of axioms and meaning postulates AX1 – AX6 (AX2 – AX6 will be defined below). A term A is said to be *truth persistent* if $AX \models \forall ij(i \leq j \rightarrow \forall x_1 \ldots x_n (T\ Ax_1 \ldots x_n i \rightarrow T\ Ax_1 \ldots x_n j))$; A is *falsity*

persistent if $AX \models \forall ij(i \leq j \rightarrow \forall x_1 \ldots x_n(F\ Ax_1 \ldots x_n i \rightarrow F\ Ax_1 \ldots x_n j))$; and A is simply called *persistent* if it is both truth persistent and falsity persistent, or, in other words, if the extension of A at any index i approximates its extension at j, if i is part of j. By definition, all constants in the fragment persist, but the property also holds for translations of sentences.

THEOREM 6 (Persistency Theorem). If ϑ is an analysis tree of category t and $^\circ$ is defined as it is in section 2, then the translation ϑ° persists:

$$AX1 \models \forall ij(i \leq j \rightarrow (\vartheta^\circ i \sqsubseteq \vartheta^\circ j)).$$

So if a sentence is true (or false) at a certain situation it will remain so at all situations that extend that situation. Truth and falsity of sentences are retained under increase of definedness.

This might however not be exactly the result we want. There are sentences in our fragment, sentences formed with the help of the determiners every, a and the, that are usually thought not to persist. While sentence (7), once it is false in some situation, will always remain false if that situation is enlarged, it could be true at an index that is part of a situation at which it is no longer true. Conversely, sentence (8) could be false in some situation, but no longer so in an extension of it, while, once true, it will always remain true. Sentence (9) behaves even worse, it may lose its truth if the situation at which it is evaluated is extended, but in similar cases it may lose falsity too.

(7) Every man loves Mary
(8) Some woman talks
(9) The woman doesn't talk

The explanation is perhaps that the range of quantification at any situation is not the whole domain D_e of all possible individuals but only a classical subset of this set consisting of those individuals that are *actual* at the given situation. If a situation i extends a situation j then the set of individuals that are actual at i includes those that are actual at j, but there may be individuals that are actual at i but not actual at j. In order to formalize this, we introduce an existence predicate E, a non-logical constant of type $\langle es \rangle$ and stipulate that it is classical and that its denotation may not decrease under growth of information by adopting the following as axioms.

AX2 $\forall xi(Exi = \mathsf{T} \vee Exi = \bot)$
AX3 $\forall xij(i \leq j \rightarrow (Exi \rightarrow Exj)$

So E is truth persistent but not falsity persistent. Quantification at any index can now be restricted to the denotation of E at that index by redefining the translations of the determiners every, a and the as follows.

every° = $\lambda P_1 \lambda P_2 \lambda i\ \forall x(Exi \rightarrow (P_1 xi \rightarrow P_2 xi))$
a° = $\lambda P_1 \lambda P_2 \lambda i\ \exists x(Exi \wedge P_1 xi \wedge P_2 xi)$
the° = $\lambda P_1 \lambda P_2 \lambda i\ \exists x(\forall y((Eyi \wedge P_1 yi) \leftrightarrow x{=}y) \wedge P_2 xi)$

The result is that the translation of (7) is falsity persistent but not truth persistent, the translation of (8) is truth persistent but not falsity persistent and that the translation of (9) is neither.

In fact our notion of persistence is a combination of two notions that can be distinguished: persistence under growth of information and persistence under growth of domain. Theorem 6 says that all sentences of the fragment persist under growth of information, provided that domains remain fixed. Of course in general quantified sentences will not persist under growth of domain.

4.3 Strong consequence and weak consequence

The first thing to note when we look at the logic that the verb believe receives in our new system is that we have achieved what we have set out to achieve: the inference from sentence (3) to sentence (4) is blocked. (For ease of reference sentences (1) – (4) are repeated below.) We can now easily find a partial model such that the translation of (1), *walk john*, is true at some index while the translation of (2), λi(*walk john i* $\wedge$ (*talk bill i* $\vee \neg$*talk bill i*)), is undefined at that index. The inference from (1) to (2) is blocked for the reason that it is irrelevant. As a consequence the unwanted inference from (3) to (4) is invalid too.

(1) John walks
(2) John walks and Bill talks or Bill doesn't talk
(3) Mary believes that John walks
(4) Mary believes that John walks and Bill talks or Bill doesn't talk

Now some readers may find that the price that was paid for blocking the unwanted entailments has been somewhat high. Did we really want to cut out the inference from (1) to (2) as well as the one from (3) to (4)? It may be true that in a strong sense of the word (2) does not follow from (1) since the converse inference is valid and the sentences are not synonymous, but certainly there is a weaker sense in which the entailment, although irrelevant, nevertheless seems unobjectionable: If (1) is true, true in the real world, then (2) is true as well and if (2) is actually false then (1) is actually false too. So it seems that there are two natural conceptions of the notion of logical consequence here, a weak, classical notion and a strong, relevant one. Sentence (2) follows from sentence (1) in the weak but not in the strong sense.

Both notions can be formalized within our system. We say that a model is an *intended* model if it satisfies all the axioms in AX. If Γ and Δ are sets of terms of type $\langle \alpha_1 \ldots \alpha_n \rangle$, then we write $\Gamma \models_{AX} \Delta$ for $\Gamma, \{\lambda x_{\alpha_1} \ldots \lambda x_{\alpha_n} \varphi \mid \varphi \in \mathrm{AX}\} \models \Delta$. Since we shall take care to have only bivalent sentences (sentences that are either true and not false or false and not true in any model) as axioms, this gives a notion of logical consequence that takes only intended models into account: $\Gamma \models_{AX} \Delta$ if and only if in each intended model the partial intersection of the values of the terms in Γ is partially included in the partial union of the values of the terms in Δ. An analysis tree ϑ of any category is said to *follow (strongly) from* a set of trees Ξ of that category if $\Xi^\circ \models_{AX} \vartheta^\circ$. Analysis trees ξ and ϑ are called

(*strongly*) *synonymous* if ξ strongly follows from ϑ and ϑ strongly follows from ξ.

Note that if Γ and Δ consist of terms of some type ending in s we can alternatively describe $\Gamma \models_{AX} \Delta$ as: in each intended model and at each situation i in that model the partial intersection of the extensions of the terms in Γ at i is partially included in the partial union of the extensions of the terms in Δ at i. This suggests that we can weaken the notion of entailment by imposing constraints on the kind of situations at which evaluation can take place. The constraint that we shall impose in order to get the second notion of consequence is that these situations are *worlds*, total and coherent situations.

To this end, let *world* be some constant of type ‹*s*› and let Ω be the conjunction of all formulae of the form $\forall x_1 \ldots x_n (cx_1 \ldots x_n i = \top \vee cx_1 \ldots x_n i = \bot)$ where $c \in$ LC and i is some fixed type s variable. We stipulate that the constant *world* will denote the set of situations that are total and coherent by adopting the following axiom.

AX4 $\quad \forall i(world\, i = \Omega)$

Now we say that an analysis tree ϑ of any category A such that $\tau(A) = \langle \alpha_1 \ldots \alpha_n s \rangle$ *weakly follows from* a set of trees Ξ of that category if $\Xi^\circ, \lambda x_{\alpha_1} \ldots \lambda x_{\alpha_n} world \models_{AX} \vartheta^\circ$. That is, a tree ϑ weakly follows from a tree ξ if and only if in each intended model, at each world i in that model, the extension of ξ° at i is partially included in the extension of ϑ° at i. Analysis trees ξ and ϑ are called *weakly synonymous* if ξ weakly follows from ϑ and ϑ weakly follows from ξ.

Note that while (2) weakly follows from (1) under this definition, still (4) does not even weakly follow from (3). The extra premise, the constant *world*, whose meaning we can informally describe as 'The present index is total and coherent', has a purely local effect. It will give a coarse-grained semantics to sentences that have no intensional expressions occurring in them, but it will have no effect on the logic of those expressions that occur within the scope of an intensional operator.

4.4 Belief, doubt, knowledge and assertion

But the above treatment of the verb believe is still not adequate. While we have been able to get rid of certain entailments that we did not want there are others that we do want but that are treated as invalid by our analysis thus far. For example, (3) and (11) should both follow from (10), but on the account given they do not. Similarly, (12) should follow from (3) as well as from (11); (10) should follow from the conjunction of (3) and (11); and (11) plus (13) should have (14) as a consequence.

(10) Mary believes that John walks and Bill talks
(11) Mary believes that Bill talks
(12) Mary believes that John walks or Bill talks
(13) Mary believes that Bill is a man
(14) Mary believes that a man talks

We can get a theory of belief sentences that predicts this behaviour by following classical Situation Semantics very closely and treating the ex-

pression believe that in Hintikka's way. To this end we have introduced a constant *BO* of type ‹*ess*› into our set LC of lexical constants. The informal interpretation of a formula of the form *BOxji* is that in situation *i* situation *j* is classified as compatible with *x*'s beliefs. Since *BO* is a partial relation like any other, the classification need not be total: there may be situations *j* that are neither classified as compatible nor as incompatible with *x*'s beliefs at *i*. Nor need the relation be coherent: some indices may be counted both as compatible and as incompatible with *x*'s beliefs at some incoherent situation. We call a situation *j* a *doxastic option* of *x* in *i* if *BOxij* is true in the model under consideration and we call it a *doxastic alternative* of *x* in *i* if *BOxij* is not false.

Now redefine the translation of believe that as follows:

believe that° = $\lambda p \lambda x \lambda i \forall j (BOxji \rightarrow pj)$

The resulting analysis of belief sentences is roughly equivalent to the analyses of these expressions given in B & P. For example, the translation of sentence (11), $\lambda i \forall j (BOmary\ ji \rightarrow walk\ bill\ j)$, will be true at some index *i* just if Bill walks at all Mary's doxastic alternatives in *i*; it will be false at index *i* just if Bill doesn't walk at some of Mary's doxastic options there.

Belief is no longer merely closed under substitution of strong synonyms but is closed under strong consequence now as well: if a person believes that A_1, believes that A_2,..., and believes that A_n, and if *B* follows strongly from $A_1,...,A_n$, then it follows that the person in question believes that *B* too. Hence the entailments cited above come out valid: the conjunction of (3) and (11) follows from (10) (*Conjunction Distribution*), (12) follows from (3) as well as from (11), (10) follows from (3) plus (11) and (14) from the conjunction of (11) and (13).

The treatment will work for other propositional attitudes too, provided that we make the necessary modifications in each individual case. Here are some extra translations:

assert that° = $\lambda p \lambda x \lambda i \forall j (AOxji \rightarrow pj)$
doubt that° = $\lambda p \lambda x \lambda i \exists j (BOxji \wedge \neg pj)$
know that° = $\lambda p \lambda x \lambda i \forall j (KOxji \rightarrow pj)$

The expression assert that is treated just like believe that, except that in its translation the relation *BO* is replaced by a relation *AO* of the same type. The denotation of $\lambda j\ AOxji$ consists of those situations *j* at which that what is asserted by *x* at *i* is not false; its antidenotation is formed by those indices at which that what is asserted by *x* at *i* is not true. Clearly, assert that and believe that will have the same logic. The translation of doubt that is chosen in a way that will assure that e.g. sentences (15) and (16) are equivalent. Conjunction Distribution fails under this analysis: Neither (16) nor (17) follows from (18). The conjunction of (16) and (17) can on the other hand be inferred from sentence (19). (Compare the discussion in B&P, pages 215 and 216; see also the table on page 195.)

(15) Mary doesn't believe that John walks
(16) Mary doubts that John walks

(17) Mary doubts that Bill talks
(18) Mary doubts that John walks and Bill talks
(19) Mary doubts that John walks or Bill talks
(20) Mary knows that John walks
(21) Mary knows that John doesn't walk
(22) Mary doesn't know that John walks

To get the facts about the expression know that right, we translate it in a way that is analogous to the translations of believe that and assert that but we let AX5 be a meaning postulate. Any agent in any situation will find that situation itself among his epistemic options as well as among his epistemic alternatives. This will assure for example that sentence (1) follows weakly from sentence (20) (*Veridicality*) and that (22) is weakly entailed by (21) (*Negation*). The axiom AX6, in which the double-headed arrow of the previous section is put to use, stipulates that an agents doxastic options are among his epistemic options and that any of his doxastic alternatives is one of his epistemic alternatives. We use it to derive that knowledge involves belief, that is that sentences like (20) entail sentences like (3).

AX5 $\forall x \forall i(KOxii = \mathsf{T})$
AX6 $\forall x \forall i \forall j(BOxji \twoheadrightarrow KOxji)$

4.5 Neutral perception

Lastly I turn to the treatment of the neutral perception verb see. Although I shall treat this expression as a verb that takes sentential complements and that forms intransitive verb phrases with these, I shall not assign it to the same category *IV / t* that the other attitude verbs are assigned to. Instead, we list it as an expression of category *IV // t* . To the definition of *analysis tree* we add an extra clause which runs as follows.

G18. If $\xi \in AT_{IV//t}$ and $\vartheta \in AT_t$, then $[\xi\vartheta]^{18} \in AT_{IV}$

The definition of the translation function ° must also get an extra clause; treating G18 as an ordinary application rule, we define:

T18. $([\xi\vartheta]^{18})^\circ = \xi^\circ\vartheta^\circ$

Following B&P again we shall say that a sentence of the form *a* sees φ (where *a* is a noun phrase and φ is an embedded uninflected sentence) is true at a situation *i* just in case it is true that *a* sees some part of *i* at which φ is true; *a* sees φ is false if at none of *a*'s visual scenes the embedded sentence φ is true. In other words, we choose the following translation for the new lexical element see (where *see xji* is to be read as: "*x* sees *j* at *i*"):

$\mathsf{see}^\circ = \lambda p \lambda x \lambda i \exists j(see\ xji \wedge j \leq i \wedge pj = \mathsf{T})$

This analysis works well enough, as long as the embedded sentences do not contain any determiners (for that case see below). First note that our treatment of the irrelevant entailments in the case of the epistemic and doxastic attitudes carries over to the attitudes of perception without any

difficulty: Sentences (5) and (6) are not synonymous, not even weakly so. Second, our theory predicts roughly the same entailment phenomena as the B&P theory does. For example, from the fact that all determiner-free sentences persist we see that sentence (5) weakly entails its embedded sentence (1) (Veridicality). In a similar way we see that (24) follows weakly from (23) (Negation). We also have logical principles like Conjunction Reduction and Disjunction Distribution: for example (25) entails (26) and (28) follows from (27).

(5) Mary sees John walk
(6) Mary sees John walk and Bill talk or Bill not talk
(23) Mary sees John not walk
(24) Mary doesn't see John walk
(25) Mary sees John walk and Bill talk
(26) Mary sees John walk and Mary sees Bill talk
(27) Mary sees John walk or Bill talk
(28) Mary sees John walk or Mary sees Bill talk

To make the treatment work for arbitrary sentences of the fragment, including those that have a determiner occurring in the complement of a neutral perception verb, some extra care must be taken. Consider sentence (29). According to our analysis thus far, but contrary to intuition, this sentence is ambiguous. It has a *de re* reading, given in (29^a), in which the noun phrase a man takes scope over the perception verb see, as well as a *de dicto* reading, given in (29^b), in which see has scope over a man. A similar remark can be made about sentences (30) - (32): each has a reading (given in (30^a), (31^a) and (32^a) respectively) obtained by 'quantifying-in' as well as a reading (given in (30^b), (31^b) and (32^b)), that can be got by more direct means.

(29) Mary sees a man walk
(29^a) [[a man]^3 [Mary [see [he$_0$ walk]^4]^{18}]^4]^{14,0}
$\lambda i \exists x(Exi \wedge man\ xi \wedge \exists j(see\ mary\ ji \wedge j \leq i \wedge walk\ xj = T)$
(29^b) *[Mary [see [[a man]^3 walk]^4]^{18}]^4
$\lambda i \exists j(see\ mary\ ji \wedge j \leq i \wedge \exists x(Exj \wedge man\ xj \wedge walk\ xj) = T)$

(30) Mary sees the man walk
(30^a) [[the man]^3 [Mary [see [he$_0$ walk]^4]^{18}]^4]^{14,0}
$\lambda i \exists x(\forall y((Eyi \wedge man\ yi) \leftrightarrow x{=}y) \wedge \exists j(see\ mary\ ji \wedge j \leq i \wedge walk\ xj = T)$
(30^b) *[Mary [see [[the man]^3 walk]^4]^{18}]^4
$\lambda i \exists j(see\ mary\ ji \wedge j \leq i \wedge \exists x(\forall y((Eyj \wedge man\ yj) \leftrightarrow x{=}y) \wedge walk\ xj) = T)$

(31) Mary sees every man walk
(31^a) [[every man]^3 [Mary [see [he$_0$ walk]^4]^{18}]^4]^{14,0}
$\lambda i \forall x(Exi \wedge man\ xi) \rightarrow \exists j(see\ mary\ ji \wedge j \leq i \wedge walk\ xj = T))$
(31^b) *[Mary [see [[every man]^3 walk]^4]^{18}]^4
$\lambda i \exists j(see\ mary\ ji \wedge j \leq i \wedge \forall x((Exj \wedge man\ xj) \rightarrow walk\ xj) = T)$

(32) Mary sees no man walk
(32^a) [[no man]^3 [Mary [see [he$_0$ walk]^4]^{18}]^4]^{14,0}
$\lambda i \neg \exists x(Exi \wedge man\ xi \wedge \exists j(see\ mary\ ji \wedge j \leq i \wedge walk\ xj = T)$
(32^b) *[Mary [see [[no man]^3 walk]^4]^{18}]^4
$\lambda i \exists j(see\ mary\ ji \wedge j \leq i \wedge \exists x(Exj \wedge man\ xj \wedge walk\ xj) = \bot)$

Looking at the semantics of these sentences we see that in each case the *de re* but not the *de dicto* readings give the right truth conditions. Consider a case in which Mary sees Bill walk. Bill is a man, but he is too far away to enable Mary to see this. Then (29) is true, but (29^b) could be false since Bill is not in the positive extension of the predicate 'man' at Mary's visual scene. Similarly, we can easily imagine situations in which the truth values of (30^b), (31^b) and (32^b) differ from the intuitive semantics of sentences (30), (31) and (32), the expressions they are supposed to formalize respectively.

Note that we cannot characterize the correct readings by simply stipulating that neutral perception verbs may not take scope over determiners. In (33^a) for example, a perfectly acceptable non-specific reading of sentence (33), the verb see has scope over the determiner a. Sentence (33) has a correct specific reading (33^b), in which a pen takes scope over both see and seek, as well. But the intermediate reading (33^c), that has a pen taking scope over seek, but itself being in the scope of see, is out: There is no natural reading of (33) that implies the existence of a pen in Mary's visual field.

(33) Mary sees Bill seek a pen

(33^a) [Mary [see [Bill [seek [a pen]^3]^5]^4]^{18}]^4
$\lambda i \exists j(see\ mary\ ji \wedge j \leq i \wedge try\ \lambda y \lambda i \exists x(Exi \wedge pen\ xi \wedge find\ xyi)\ bill\ j = T)$

(33^b) [[a pen]^3 [Mary [see [Bill [seek he_0]^5]^4]^{18}]^4]^{14,0}
$\lambda i \exists x(Exi \wedge pen\ xi \wedge \exists j(see\ mary\ ji \wedge j \leq i \wedge try\ (find\ x)\ bill\ j = T)$

(33^c) *[Mary [see [[a pen]^3 [Bill [seek he_0]^5]^4]^{14,0}]^{18}]^4
$\lambda i \exists j(see\ mary\ ji \wedge j \leq i \wedge \exists x(Exj \wedge pen\ xj \wedge try\ (find\ x)\ bill\ j) = T)$

Then what is it that the bad readings (30^b), (31^b), (32^b) and (33^c) have in common but that distinguishes them from the good guys (30^a), (31^a), (32^a), (33^a) and (33^b)? It is a semantical property: in the translation of each of the bad trees a quantification occurs over the domain of individuals occurring in somebody's visual scene. We must conclude with Asher & Bonevac [1985, 1987] that no such quantification is allowed. Quantifiers are not interpreted in scenes. In normal situations the domain of objects associated with a person's field of vision changes rapidly over time; even the tiniest movement of the eye can cause objects to be introduced into ones domain of vision and can cause other objects to be expelled from it. It seems that language users are therefore extremely reluctant to interpret non-persistent expressions in their visual fields.

Mathematically we can characterize the good readings by adopting the following definition.

DEFINITION 21. An analysis tree is called *admissible* if for all its subtrees of the form $[\xi\vartheta]^{18}$ the term ϑ° is persistent.

Meanings are now associated with admissible readings only. For example (33^c) is now ruled inadmissible and is not associated with a meaning because it has [see [[a pen]^3 [Bill [seek he_0]^5]^4]^{14,0}]^{18} as a subtree and the translation of [[a pen]^3 [Bill [seek he_0]^5]^4]^{14,0}]^{18}, $\lambda i \exists x(Exi \wedge pen\ xi$

$\wedge$*try* (*find x*) *bill i*) is not persistent. On the other hand, (33[a]) and (33[b]) are admissible since both *try* $\lambda y \lambda i \exists x$(*Exi* $\wedge$ *pen xi* $\wedge$ *find xyi*) *bill* and the open term *try* (*find x*) *bill* persist. Similarly, the a-readings in (29) – (32) can be seen to be admissible but the b-readings are out; the verb see takes non-persistent complements in them.

(34) A man walks
(35) The man walks
(36) Every man walks
(37) No man walks
(38) Bill seeks a pen
(39) John is the man
(40) John is a man

If we restrict the set of readings thus, the basic facts about entailments involving neutral perception sentences can be predicted. The theory gets the Veridicality phenomena right: (34), (35) and (36) follow weakly from (29), (30) and (31) respectively. But of course (37) is not weakly entailed by (32), nor should it be. Notice that (38) follows weakly from (33), or, to be more precise, its *de dicto* reading follows weakly from (33)'s *de dicto* reading (33[a]) and its *de re* reading is weakly entailed by (33)'s *de re* reading (33[b]). *Substitutivity* also holds: for example from (5) and (39) sentence (30) follows. Relatedly, we may see that (29) follows from sentences (5) and (40). Lastly, note that the theory predicts the facts about *Exportation* in a trivial way. As was noted in Barwise [1981], the noun phrase a man in (29) can be 'exported' to give it widest possible scope without change of meaning. Our theory predicts this, since we claim that the NP already has wide scope in the sentence's unique (up to equivalence) legal reading and so there is nothing to export in the first place. Of course by a similar reasoning we predict that *all* noun phrases show this behaviour, which conforms to an observation made in Higginbotham [1983].

Appendix

Proof of Theorem 1.

THEOREM 1 (repeated). Let $\Gamma \cup \{\varphi\}$ be a set of (TT_2) sentences then

$$\Gamma \models \varphi \text{ in } TT_2 \text{ iff } \Gamma \models \varphi \text{ in } TY_2.$$

Let $F = \{D_\alpha \mid \alpha \text{ is a type}\}$ be a frame and $F' = \{D'_\alpha \mid \alpha \text{ is a } TY_2 \text{ type}\}$ the TY_2 frame such that $D_e = D'_e$ and $D_s = D'_s$. For each type α define a function S_α: $D_\alpha \to D'_{\Sigma(\alpha)}$ by the following double recursion:

I $S_e(d) = d$, if $d \in D_e$; $S_s(d) = d$, if $d \in D_s$;
II i. $S_{\langle\rangle}(d) = d$, if $d \in D_{\langle\rangle}$;
ii. If $n > 0$, $\alpha = \langle\alpha_1 \ldots \alpha_n\rangle$ and $R \in D_\alpha$, then $S_\alpha(R)$ is the function G of type $(\Sigma(\alpha_1)\Sigma(\langle\alpha_2 \ldots \alpha_n\rangle))$ such that $G(f) = S_{\langle\alpha_2 \ldots \alpha_n\rangle}(F^1_R(S_{\alpha_1}{}^{-1}(f)))$ for each $f \in D'_{\Sigma(\alpha_1)}$.

It is easy to prove that the functions S_α are bijections and that hence the definition is correct. I'll suppress subscripts on S in the rest of the proof.

Let I be an interpretation function for the frame F and let I' be an interpretation function for F' such that $I'(c) = S(I(c))$ for all TT_2 constants c. Let $M = \langle F, I\rangle$ and let $M' = \langle F', I'\rangle$. Write $|A|^{N,b}$ for the value of a term A in a TY_2 model N under an assignment b. Then, for all TT_2 terms A and all assignments a for M and a' for M' such that $a'(x) = S(a(x))$ for all TT_2 variables x, we have $|A|^{M',a'} = S(\|A\|^{M,a})$, as can be seen by an induction on the complexity of A (I'll sometimes write $|A|$ for $|A|^{M',a'}$, just as I write $\|A\|$ for $\|A\|^{M,a}$, to avoid too much mathematical clutter).

i. $S(\|c\|) = S(I(c)) = I'(c)$ if c is a constant;
$S(\|x\|) = S(a(x)) = a'(x)$ if x is a variable;

ii. Since $S_{\langle\rangle}$ is the identity function on $\{0,1\}$ it holds that $S(\|\neg\varphi\|) = 1 - S(\|\varphi\|) = 1 - |\varphi| = |\neg\varphi|$ and that $S(\|\varphi \wedge \psi\|) = S(\|\varphi\|) \cap S(\|\psi\|) = |\varphi| \cap |\psi| = |\varphi \wedge \psi|$;

iii. $S(\|\forall x_\alpha \varphi\|^{M,a}) = \bigcap_{d \in D_\alpha} S(\|\varphi\|^{M,a[d/x]}) = \bigcap_{d \in D_\alpha} \|\varphi\|^{M',a'[S(d)/x]} = \bigcap_{d' \in D'_{\Sigma(\alpha)}} |\varphi|^{M',a'[d'/x]} = |\forall x \varphi|^{M',a'}$;

iv. $S(\|AB\|) = S(F^1_{\|A\|}(\|B\|)) = S(F^1_{\|A\|}(S^{-1}(|B|^{M',a'}))) = S(\|A\|)(|B|) = |A|(|B|) = |AB|$;

v. Since $\|\lambda x_\beta A\|^{M,a}$ is the relation R such that for all $d \in D_\beta$: $F^1_R(d) = \|A\|^{M,a[d/x]}$, we see that $S(\|\lambda x_\beta A\|^{M,a})$ is the function G such that for all $d' \in D'_{\Sigma(\beta)}$: $G(d') = S(F^1_R(S^{-1}(d'))) = S(\|A\|^{M,a[d/x]})$, where $d = S^{-1}(d')$. So for each $d' \in D'_{\Sigma(\beta)}$, $G(d') = \|A\|^{M',a'[d'/x]}$ and hence $S(\|\lambda x A\|^{M,a}) = \|\lambda x A\|^{M',a'}$;

vi. $S(\|A = B\|) = 1$ iff $\|A\| = \|B\|$ iff $S(\|A\|) = S(\|B\|)$ iff $|A| = |B|$ iff $|A = B| = 1$.

Now let M be a TT_2 model such that all $\psi \in \Gamma$ are true in M but φ is false in M. Then all $\psi \in \Gamma$ are true in M' as constructed above but φ is false in M'. So if $\Gamma \models \varphi$ in TY_2 then $\Gamma \models \varphi$ in TT_2. Conversely, let $M' = \langle \{D'_\alpha\}_\alpha, I'\rangle$ be a TY_2 model such that all $\psi \in \Gamma$ are true in M' but φ is false in M'. Let $M = \langle \{D_\alpha\}_\alpha, I\rangle$, where $D_e = D'_e$, $D_s = D'_s$ and I is defined as the interpretation function such that $I(c) = S^{-1}(I'(c))$ for all constants c. Then all $\psi \in \Gamma$ are true in M but φ is false in M. So if $\Gamma \models \varphi$ in TT_2 then $\Gamma \models \varphi$ in TY_2. ■

Proof of Theorem 2.

THEOREM 2 (repeated). For each analysis tree ξ let ξ' be the translation it is given in DWP. Let Δ be the following set of IL-sentences (all meaning postulates in DWP):

$\exists x \Box[x = c]$, where c is any constant of type e;
$\exists S_{(s(e(et)))} \forall x \forall Q \Box[\delta(x,Q) \leftrightarrow Q\{\hat{}\lambda y[S\{x,y\}]\}]$, where δ is find', love', lose', eat' or date';
$\exists G_{(s(e((s(et))(et))))} \forall Q \forall P \forall x \Box[\text{in}'(Q)(P)(x) \leftrightarrow Q\{\hat{}\lambda y[[\check{}G](y)(P)(x)]\}]$;
$\forall x \forall Q \Box[\text{seek}'(x,Q) \leftrightarrow \text{try}'(x, \hat{}[\text{find}'(Q)])]$.

Let $\Xi \cup \{\vartheta\}$ be a set of analysis trees, then:

$$\Xi^\circ \models \vartheta^\circ \text{ in } TT_2 \text{ iff } \Xi' \cup \Delta \models \vartheta' \text{ in IL.}$$

For the sake of definiteness and ease of reference, we give a definition of the translation function ' as it is employed in DWP.

DEFINITION (DWP Translations). Let g be some fixed one-to-one function having the set of basic expressions other than those mentioned explicitly in rule T1' below as its domain, such that $g(\xi)$ is an IL constant of type $f(A)$ (defined in Table 1) if ξ is of category A. For each analysis tree ξ define its *DWP translation* ξ' by induction on the complexity of analysis trees:

T1'. If ξ is in the domain of g then $\xi' = g(\xi)$;
$\text{John}' = \lambda P[\check{}P\ john_e]$, $\text{Mary}' = \lambda P[\check{}P\ mary_e]$, $\text{Bill}' = \lambda P[\check{}P\ bill_e]$,
$\text{he}_n' = \lambda P[\check{}Px_n]$;
$\text{be}' = \lambda Q\lambda y[\check{}Q\hat{}\lambda x[x=y]]$;
$\text{every}' = \lambda P_1\lambda P_2\forall x[\check{}P_1x \rightarrow \check{}P_2x]$,
$\text{a}' = \lambda P_1\lambda P_2\ \exists x[\check{}P_1x \wedge \check{}P_2x]$,
$\text{the}' = \lambda P_1\lambda P_2\ \exists x[\forall y[\check{}P_1y \leftrightarrow x=y]\wedge\check{}P_2x]$;
$\text{necessarily}' = \lambda p\Box[\check{}p]$;

T2'. $([\xi\vartheta]^{2,n})' = \lambda x_n[\xi'(x_n) \wedge \vartheta']$;

T3' – T10'. $([\xi\vartheta]^k)' = \xi'(\hat{}\vartheta')$ if $3 \leq k \leq 10$;

T11'. $([\xi\vartheta]^{11a})' = \xi' \wedge \vartheta'$;
$([\xi\vartheta]^{11b})' = \xi' \vee \vartheta'$;

T12'. $([\xi\vartheta]^{12a})' = \lambda x[\xi'(x) \wedge \vartheta'(x)]$;
$([\xi\vartheta]^{12b})' = \lambda x[\xi'(x) \vee \vartheta'(x)]$;

T13'. $([\xi\vartheta]^{13})' = \lambda P[\xi'(P) \vee \vartheta'(P)]$;

T14'. $([\xi\vartheta]^{14,n})' = \xi'(\hat{}\lambda x_n\vartheta')$;

T15'. $([\xi\vartheta]^{15,n})' = \lambda y[\xi'(\lambda x_n[\vartheta'(y)])]$;

T16'. $([\xi\vartheta]^{16,n})' = \lambda y[\xi'(\lambda x_n[\vartheta'(y)])]$;

T17'. $([\xi\vartheta]^{17})' = \neg\xi'(\hat{}\vartheta')$.

It is well known that the intensional logic IL is in fact a part of the much simpler logic TY_2. Gallin [1975] gives an embedding. The following translation is essentially his, but for the way individual constants are dealt with:

DEFINITION (Gallin's Embedding). Let i be some TY_2 variable of type s. The function *, sending IL terms to TY_2 terms, is defined with the help of the following clauses:

i. $x^* = x$, if x is a variable

ii. $c^* = c$, if c is a constant of type e.
$c^* = ki$, where k is the n-th constant of type $(s\alpha)$ in some fixed ordering, if c is the n-th constant of type $\alpha \neq e$.

iii. $(\varphi \wedge \psi)^* = \varphi^* \wedge \psi^*$ $\quad$ $(\neg\varphi)^* = \neg\varphi^*$

iv. $(\forall x\varphi)^* = \forall x\varphi^*$

v. $(AB)^* = A^*B^*$

vi. $(\lambda xA)^* = \lambda xA^*$

vii. $(A = B)^* = A^* = B^*$

viii. $(\hat{}A)^* = \lambda iA^*$

ix. $(\check{}A)^* = A^*i$

x. $(\Box\varphi)^* = \forall i\varphi^*$

It is not difficult to prove that the translation * is an embedding of IL plus some rigid designator meaning postulates into TY_2:

THEOREM (Gallin). Let $\Gamma \cup \{\varphi\}$ be a set of IL formulae, let $\Gamma^{\clubsuit}$ be the set $\{\gamma^{\clubsuit} \mid \gamma \in \Gamma\}$ and let Θ be the set of IL formulae $\{\exists x \Box[x = c] \mid c$ is a constant of type $e\}$. Then:

$$\Gamma, \Theta \models \varphi \text{ in IL} \Leftrightarrow \Gamma^{\clubsuit} \models \varphi^{\clubsuit} \text{in } TY_2.$$

By an easy induction on the complexity of analysis trees we verify that for each tree ξ the TY_2 term $\xi'^{\clubsuit}$ is in fact a TT_2 term: none of its subterms will have a type in which an e or an s immediately precedes a right parenthesis. So in virtue of Theorem 1. we have that $\Xi'^{\clubsuit} \models \vartheta'^{\clubsuit}$ in TT_2 iff $\Xi', \Theta \models \vartheta'$ in IL.

But in general $\xi'^{\clubsuit}$ will not be equivalent to ξ°; the types will not match. The main reason for this is that our category-to-type rule places ss immediately before right brackets, while in the type of any $\lambda i(\xi'^{\clubsuit})$ these ss occupy a place immediately following the corresponding left brackets. So we have to permute types.

DEFINITION. Define, for each α, the type $\alpha^{\blacklozenge}$ by:

I $\quad e^{\blacklozenge} = e$; $s^{\blacklozenge} = s$;

II i. $\quad \langle\rangle^{\blacklozenge} = \langle\rangle$

ii. $\quad \langle\alpha_1 \ldots \alpha_n \alpha_{n+1}\rangle^{\blacklozenge} = \langle\alpha_{n+1}\alpha_1 \ldots \alpha_n\rangle$

And we embed TT_2 in itself.

DEFINITION. Define for each TT_2 term A of type α a TT_2 term $A^{\blacktriangledown}$ of type $\alpha^{\blacklozenge}$ by the following induction.

i. $\quad c^{\blacktriangledown} = k$, where k is the n-th constant of type $\alpha^{\blacklozenge}$ in some fixed ordering, if c is the n-th constant of type α;
$x^{\blacktriangledown} = y$, where y is the n-th variable of type $\alpha^{\blacklozenge}$ in some fixed ordering, if x is the n-th variable of type α;

ii. $\quad (\neg\varphi)^{\blacktriangledown} = \neg\varphi^{\blacktriangledown}$;
$(\varphi \wedge \psi)^{\blacktriangledown} = \varphi^{\blacktriangledown} \wedge \psi^{\blacktriangledown}$;

iii. $\quad (\forall x\varphi)^{\blacktriangledown} = \forall x^{\blacktriangledown}\varphi^{\blacktriangledown}$;

iv. $\quad (A_{\langle\beta,B_{\beta}})^{\blacktriangledown} = A^{\blacktriangledown}B^{\blacktriangledown}$,
$(A_{\langle\beta\alpha_1 \ldots \alpha_{n+1}\rangle}B_{\beta})^{\blacktriangledown} = \lambda x(A^{\blacktriangledown}xB^{\blacktriangledown})$, where x is the first variable of type $\alpha_{n+1}^{\blacklozenge}$ that does not occur free in $A^{\blacktriangledown}$ or $B^{\blacktriangledown}$;

v. $\quad (\lambda x\varphi_{\langle\rangle})^{\blacktriangledown} = (\lambda x^{\blacktriangledown}\varphi^{\blacktriangledown})$,
$(\lambda x_{\beta}A_{\langle\alpha_1 \ldots \alpha_{n+1}\rangle})^{\blacktriangledown} = \lambda y \lambda x^{\blacktriangledown}(A^{\blacktriangledown}y)$, where y is the first variable of type $\alpha_{n+1}^{\blacklozenge}$ that does not occur free in $A^{\blacktriangledown}$;

vi. $\quad (A = B)^{\blacktriangledown} = A^{\blacktriangledown} = B^{\blacktriangledown}$.

We prove that $^{\blacktriangledown}$ preserves entailment. Let $M = \langle\{D_{\alpha}\}_{\alpha}, I\rangle$ be a model and let a be an assignment for it. Define $\pi(d) = d$ if $d \in D_e$, $d \in D_s$ or $d \in D_{\langle\rangle}$ and define $\pi(R) = \{\langle\pi(d_{n+1}),\pi(d_1),\ldots,\pi(d_n)\rangle \mid \langle d_1,\ldots,d_n,d_{n+1}\rangle \in R\}$ if $R \in D_{\langle\alpha_1 \ldots \alpha_n\alpha_{n+1}\rangle}$. It is easily seen that the restriction of π to D_{α} is a bijection between D_{α} and $D_{\alpha^{\blacklozenge}}$. Let I' be an interpretation function such that $I'(c^{\blacktriangledown}) = \pi(I(c))$ for all constants c. Define $M' := \langle\{D_{\alpha}\}_{\alpha}, I'\rangle$. Then by an induction on term complexity that we leave to the reader: $\|A^{\blacktriangledown}\|^{M',a'} = \pi(\|A\|^{M,a})$ if a' is an assignment such that $a'(x^{\blacktriangledown}) = \pi(a(x))$ for all variables x. It readily follows that $\Gamma \models \Delta$ if and only if $\Gamma^{\blacktriangledown} \models \Delta^{\blacktriangledown}$.

Note that if A is an $n+1$–ary term then $(AB_1...B_nB_{n+1})^\blacktriangledown$ is equivalent to $A^\blacktriangledown B_{n+1}{}^\blacktriangledown B_1{}^\blacktriangledown ...B_n{}^\blacktriangledown$. Also note that $(\lambda x_1...\lambda x_n\lambda x_{n+1}\varphi_{\langle\rangle})^\blacktriangledown$ is equivalent to $\lambda x_{n+1}{}^\blacktriangledown\lambda x_1{}^\blacktriangledown ...\lambda x_n{}^\blacktriangledown\varphi_{\langle\rangle}$.

Let $^\spadesuit$ be a function from analysis trees to TT_2 terms that is defined just as $^\circ$ with the exception that $\mathsf{find}^\spadesuit$, $\mathsf{love}^\spadesuit$, $\mathsf{lose}^\spadesuit$, $\mathsf{eat}^\spadesuit$, $\mathsf{date}^\spadesuit$, $\mathsf{in}^\spadesuit$ and $\mathsf{seek}^\spadesuit$ are all constants (of the appropriate type) instead of complex logical expressions. Let Ω be the following set of terms:

$\lambda j\exists S_{\langle ees\rangle}\forall x\forall Q(\delta Qx = Q\lambda y(Syx))$, where δ is $\mathsf{find}^\spadesuit$, $\mathsf{love}^\spadesuit$, $\mathsf{lose}^\spadesuit$, $\mathsf{eat}^\spadesuit$ or $\mathsf{date}^\spadesuit$;
$\lambda j\exists G_{\langle e\langle es\rangle es\rangle}\forall Q\forall P\forall x(\mathsf{in}^\spadesuit QPx = Q\lambda y(GyPx))$;
$\lambda j\forall Q(\mathsf{seek}^\spadesuit Q = \mathsf{try}^\spadesuit(\mathsf{find}^\spadesuit Q))$.

Clearly, $\Xi^\circ \models \vartheta^\circ \Leftrightarrow \Xi^\spadesuit, \Omega \models \vartheta^\spadesuit$.

Without loss of generality we may assume that if δ is any basic expression of category A other than the ones mentioned explicitly in clause T1' of the definition of ' above, then δ' is the n-th IL constant of type $f(A)$ if and only if $\delta^\spadesuit$ is the n-th TT_2 constant of type $\tau(A)$. We can now bridge the gap between the DWP translation and ours.

LEMMA. $\lambda i(\xi'^\spadesuit)$ (where i is the variable in Gallin's Embedding) is equivalent to $\xi^{\spadesuit\blacktriangledown}$ for all analysis trees ξ.

The proof is a straightforward but tedious induction on the complexity of analysis trees which I leave to the reader.

Combining our findings thus far, we see that $\Xi', \Theta \models \vartheta'$ in IL $\Leftrightarrow$ $\Xi'^\spadesuit \models \vartheta'^\spadesuit \Leftrightarrow \{\lambda i(\xi'^\spadesuit) \mid \xi \in \Xi\} \models \lambda i(\vartheta'^\spadesuit) \Leftrightarrow \Xi^{\spadesuit\blacktriangledown} \models \vartheta^{\spadesuit\blacktriangledown} \Leftrightarrow \Xi^\spadesuit \models \vartheta^\spadesuit$. Since it can easily be verified that for each $\varphi \in \Omega$ there is an $A \in \Delta\backslash\Theta$ such that $\lambda i(\varphi^\spadesuit)$ is equivalent to $A^\blacktriangledown$ and vice versa, we see that $\Xi', \Delta \models \vartheta'$ in IL $\Leftrightarrow \Xi^\spadesuit, \Omega \models \vartheta^\spadesuit \Leftrightarrow \Xi^\circ \models \vartheta^\circ$, which proves the theorem. ■

Proof of Theorem 3

THEOREM 3 (repeated).

I. Every Kleene algebra is isomorphic to a natural Kleene algebra on a set of partial relations.

II. Every approximation algebra is isomorphic to a natural approximation algebra on a set of partial relations.

In distributive lattices with zero and one a *prime filter* can be defined as a set ∇ such that the following hold:

$a + b \in \nabla \Leftrightarrow a \in \nabla$ or $b \in \nabla$;
$a \cdot b \in \nabla \Leftrightarrow a \in \nabla$ and $b \in \nabla$;
$0 \notin \nabla$; $\quad 1 \in \nabla$.

In the sequel we let ∇ range over prime filters.

To prove I., let K be a Kleene algebra. We'll show that there is a natural Kleene algebra, with partial sets of prime filters on K as its elements, that is isomorphic with K. Define a function f with the domain

of K as its domain by: $f(a) = \langle \{\nabla \mid a \in \nabla\}, \{\nabla \mid a' \in \nabla\}\rangle$ for each a. Then by Stone's Theorem f is 1-1. Using the properties of prime filters cited above and the axioms of Kleene algebras it is easily verified that moreover f is an isomorphism:

$f(a') = \langle \{\nabla \mid a' \in \nabla\}, \{\nabla \mid a'' \in \nabla\}\rangle = \langle \{\nabla \mid a' \in \nabla\}, \{\nabla \mid a \in \nabla\}\rangle =$
$-\langle \{\nabla \mid a \in \nabla\}, \{\nabla \mid a' \in \nabla\}\rangle = -f(a)$;
$f(a + b) = \langle \{\nabla \mid a + b \in \nabla\}, \{\nabla \mid (a + b)' \in \nabla\}\rangle =$
$\langle \{\nabla \mid a + b \in \nabla\}, \{\nabla \mid a' \cdot b' \in \nabla\}\rangle =$
$\langle \{\nabla \mid a \in \nabla \text{ or } b \in \nabla\}, \{\nabla \mid a' \in \nabla \text{ and } b' \in \nabla\}\rangle =$
$\langle \{\nabla \mid a \in \nabla\}, \{\nabla \mid a' \in \nabla\}\rangle \cup \langle \{\nabla \mid b \in \nabla\}, \{\nabla \mid b' \in \nabla\}\rangle =$
$f(a) \cup f(b)$;
$f(a \cdot b) = f(a) \cap f(b)$ by dual reasoning;
$f(0) = \langle \{\nabla \mid 0 \in \nabla\}, \{\nabla \mid 1 \in \nabla\}\rangle = \langle \varnothing, \{\nabla \mid \nabla \text{ is an prime filter on } K\}\rangle$;
$f(1) = \langle \{\nabla \mid 1 \in \nabla\}, \{\nabla \mid 0 \in \nabla\}\rangle = \langle \{\nabla \mid \nabla \text{ is an prime filter on } K\}, \varnothing\rangle$.

Proposition II. is proved in a very similar way. ■

Proof of Theorem 5

THEOREM 5 (repeated). Let Π and Σ be sets of formulae then:

$$\Pi \mid\!\!- \Sigma \Leftrightarrow \Pi \mid\!\!= \Sigma.$$

Before we can give the proof we must first state two lemmas about the syntactical consequence relation $\mid\!\!-$. The first lemma says that certain formulae behave in a classical way. Its proof is an easy induction.

LEMMA. Define a *2-formula* to be a formula that is built up from formulae of the form $A = B$ and the logical operators $\neg$, $\wedge$ and $\forall$ solely. If φ is a 2-formula then $\mid\!\!- \varphi, \neg\varphi$.

The second lemma lists some provable sequents that we shall either need below or need in the proof of the lemma itself.

LEMMA. The following sequents are provable.

I	$\Rightarrow \neg\neg\mathsf{T}$	(use the negation rules)
II	$\Rightarrow \neg\neg\mathsf{T} = \mathsf{T}$	(use I and the Truth-Value rules)
III	$\Rightarrow \neg\neg\bot = \bot$	(use II and the definition of $\bot$)
IV	$\Rightarrow \neg\neg* = *$	(use the Truth-Value rules)
V	$\Rightarrow \neg\neg\# = \#$	(use the Truth-Value rules)
VI	$\Rightarrow \neg\neg\varphi = \varphi$	(use II, III, IV, V)
VII	$\varphi = \mathsf{T} \Rightarrow \varphi$	
VIII	$\varphi = \# \Rightarrow \varphi, *$	(use the Truth-Value rules)
IX	$\varphi = \# \Rightarrow \neg\varphi, *$	(use VI and VIII)
X	$\varphi = *, \varphi \Rightarrow *$	
XI	$\varphi = *, \neg\varphi \Rightarrow *$	(use VI and X)
XII	$\varphi = \bot, \varphi \Rightarrow$	(use the definition of $\bot$)
XIII	$\varphi = \mathsf{T}, \neg\varphi \Rightarrow$	
XIV	$\varphi = \bot \Rightarrow \neg\varphi$	

XV	$\varphi \Rightarrow \varphi = \mathsf{T}, \varphi = \#, \varphi = *$	(use XII)
XVI	$\varphi \Rightarrow \varphi = \mathsf{T}, \varphi = \#, *$	(use X and XV)
XVII	$\Rightarrow \varphi, \neg\varphi, \varphi = *, *$	(use VII, VIII and XIV)
XVIII	$\varphi, \neg\varphi \Rightarrow \varphi = \#, *$	(use XIII and XVI)
XIX	$\varphi \Rightarrow \neg\varphi, \varphi = \mathsf{T}, *$	(use IX and XVI)
XX	$\neg\varphi \Rightarrow \varphi, \varphi = \bot, *$	(use XIX)

We now come to the main part of the proof, which is a generalization of the standard Henkin generalized completeness proof for type theory. A tset of formulae Γ is called **–consistent* if it does not hold that $\Gamma \vdash *$.

THEOREM (Star-Consistency Theorem). If a set of formulae is *–consistent then it has a general model.

PROOF. Let Γ be a *–consistent set of formulae. We construct a general model for Γ. Add denumerably many constants of each type to the language of Γ and let $\varphi_0,\ldots,\varphi_n,\ldots$ be some enumeration of all formulae in the extended language. For each natural number n, define a set of formulae Γ_n by the following induction.

Γ_0	=	Γ	
Γ_{n+1}	=	Γ_n,	if $\Gamma_n, \varphi_n \vdash *$ and φ_n is not of the form $\forall x\psi$;
	=	$\Gamma_n \cup \{\varphi_n\}$,	if not $\Gamma_n, \varphi_n \vdash *$ and φ_n is not of the form $\neg\forall x\psi$;
	=	$\Gamma_n \cup \{\neg T[c/x]\psi\}$,	where c is the first constant of type α (in some fixed enumeration) that does not occur in any of the sentences in $\Gamma_n \cup \{\psi\}$, if $\Gamma_n, \varphi_n \vdash *$ and $\varphi_n \equiv \forall x_\alpha\psi$;
	=	$\Gamma_n \cup \{\varphi_n, [c/x]\neg\psi\}$,	where c is the first constant of type α that does not occur in any of the sentences in $\Gamma_n \cup \{\psi\}$if not $\Gamma_n, \varphi_n \vdash *$ and $\varphi_n \equiv \neg\forall x_\alpha\psi$;

We show by induction that each Γ_n is *-consistent. By assumption Γ_0 is *-consistent. The first two cases in the proof of the induction step are trivial, so let $\Gamma_n, \forall x_\alpha\psi \vdash *$ and suppose that $\Gamma_n, \neg T[c/x]\psi \vdash *$. Then since $T[c/x]\psi$ is a 2-formula we have that $\Gamma_n \vdash T[c/x]\psi, *$, from which it follows that $\Gamma_n \vdash [c/x]\psi, *$. Since c doesn't occur in Γ_n or ψ we see that $\Gamma_n \vdash \forall x\psi, *$ by I$\forall$. By the Cut rule $\Gamma_n \vdash *$, which contradicts the induction hypothesis. To prove the last step, use I$\forall$ and the negation rules to see that $\Gamma_n \cup \{\neg\forall x\psi, [c/x]\neg\psi\}$ is *-consistent if $\Gamma_n \cup \{\neg\forall x\psi\}$ is.

Define $\Delta := \bigcup_n \Gamma_n$. Then, since all Γ_n are *-consistent, Δ is. Moreover, Δ is *maximal* in the sense that if $\varphi \notin \Delta$ then $\Delta, \varphi \vdash *$. From Δ's maximal *-consistency it follows that if $\Delta \vdash \psi_0,\ldots,\psi_n, *$ then $\psi_i \in \Delta$ for some ψ_i. So $T\varphi \in \Delta$ iff $\varphi \in \Delta$, since $\varphi \vdash T\varphi, *$ and $T\varphi \vdash \varphi, *$. In a similar way the following equivalences are seen to hold:

i. $\varphi \in \Delta$ and $\neg\varphi \in \Delta \quad \Leftrightarrow \quad \varphi = \# \in \Delta$ (use VIII, IX, XVIII)
ii. $\varphi \notin \Delta$ and $\neg\varphi \notin \Delta \quad \Leftrightarrow \quad \varphi = * \in \Delta$ (use X, XI, XVII)
iii. $\varphi \in \Delta$ and $\neg\varphi \notin \Delta \quad \Leftrightarrow \quad \varphi = \top \in \Delta$ (use VII, XIII, XIX)
iv. $\varphi \notin \Delta$ and $\neg\varphi \in \Delta \quad \Leftrightarrow \quad \varphi = \bot \in \Delta$ (use X, XIV, XX)

As a consequence $\varphi = \psi \in \Delta$ if it both holds that $\varphi \in \Delta \Leftrightarrow \psi \in \Delta$ and $\neg\varphi \in \Delta \Leftrightarrow \neg\psi \in \Delta$.

The maximal *-consistent set of formulae Δ satisfies the following form of the *Henkin property*: if $[c/x]\psi \in \Delta$ for all constants c_α then $\forall x_\alpha \psi \in \Delta$ and if $\neg\forall x_\alpha \psi \in \Delta$ then $\neg[c/x]\psi \in \Delta$ for some constant c_α. To prove the first of these statements, assume that $\forall x\psi \notin \Delta$. Then $\neg T[c/x]\psi \in \Delta$ for some c, whence, since $T[c/x]\psi$ is a 2-formula, $T[c/x]\psi \notin \Delta$ and $[c/x]\psi \notin \Delta$. The proof of the second statement is straightforward.

Define the relation ~ between terms by $A \sim B := A = B \in \Delta$. Using the identity axioms we see that this is an equivalence relation. The equivalence class $\{B \mid A \sim B\}$ of a term A under this relation we denote with $[A]$. Note that by the Henkin property and the fact that $\vdash \exists x(x = A)$, for each term A there is a constant c such that $[A] = [c]$. Now, by induction on type complexity define for each α a function Φ_α having the set of equivalence classes $\{[A] \mid A \text{ is of type } \alpha\}$ as its domain:

$\Phi_e([A]) = [A]; \qquad \Phi_s([A]) = [A];$
$\Phi_{\langle\alpha_1\ldots\alpha_n\rangle}([A]) = \langle R^+, R^-\rangle$, where
$R^+ = \{\langle\Phi_{\alpha_1}([c_1]),\ldots,\Phi_{\alpha_n}([c_n])\rangle \mid Ac_1\ldots c_n \in \Delta\}$
$R^- = \{\langle\Phi_{\alpha_1}([c_1]),\ldots,\Phi_{\alpha_n}([c_n])\rangle \mid \neg Ac_1\ldots c_n \in \Delta\}$

This is well-defined; the identity axioms assure that $\Phi_\alpha([A]) = \Phi_\alpha([A'])$ if $[A] = [A']$. To prove the converse (the injectivity of Φ_α) assume that $\Phi_{\langle\alpha_1\ldots\alpha_n\rangle}([A]) = \Phi_{\langle\alpha_1\ldots\alpha_n\rangle}([A'])$. Then for all suitable constants $c_1,\ldots,c_n$: $Ac_1\ldots c_n \in \Delta$ iff $A'c_1\ldots c_n \in \Delta$ and $\neg Ac_1\ldots c_n \in \Delta$ iff $\neg A'c_1\ldots c_n \in \Delta$. Hence $Ac_1\ldots c_n = A'c_1\ldots c_n \in \Delta$ for all $c_1,\ldots,c_n$. Suppose $n > 0$. By the Henkin property $\forall x_n(Ac_1\ldots c_{n-1}x_n = A'c_1\ldots c_{n-1}x_n) \in \Delta$, whence $Ac_1\ldots c_{n-1} = A'c_1\ldots c_{n-1} \in \Delta$ by Extensionality. Repeating this procedure as long as is necessary we find that $A = A' \in \Delta$.

We can now construct the canonical model from the equivalence classes of constants. Define D_α to be $\{\Phi_\alpha([c]) \mid c \text{ is a constant of type } \alpha\}$ and define $I(c) = \Phi_\alpha([c])$ for each constant c of type α. Then $M = \langle\{D_\alpha\}_\alpha, I\rangle$ is a very general model. Let a be an assignment for M such that $a(x) = \Phi_\alpha([x])$ for each variable x of type α. We prove by term induction that $\Phi_\alpha([A]) = \|A\|^{M,a}$ for each term A and so that M is a general model of Γ.

i. $\|c\|^{M,a} = I(c) = \Phi([c])$ if c is a constant;
$\|x\|^{M,a} = a(x) = \Phi([x])$ if x is a variable;
ii. $\|\neg\varphi\|^{M,a} = — \|\varphi\|^{M,a} = \langle\{\langle\rangle \mid \neg\varphi \in \Delta\}, \{\langle\rangle \mid \varphi \in \Delta\}\rangle =$
$\langle\{\langle\rangle \mid \neg\varphi \in \Delta\}, \{\langle\rangle \mid \neg\neg\varphi \in \Delta\}\rangle = \Phi([\neg\varphi])$;
$\|\varphi \wedge \psi\|^{M,a} = \|\varphi\|^{M,a} \cap \|\psi\|^{M,a} =$
$\langle\{\langle\rangle \mid \varphi \in \Delta \text{ and } \psi \in \Delta\}, \{\langle\rangle \mid \neg\varphi \in \Delta \text{ or } \neg\psi \in \Delta\}\rangle =$
$\langle\{\langle\rangle \mid \varphi \wedge \psi \in \Delta\}, \{\langle\rangle \mid \neg(\varphi \wedge \psi) \in \Delta\}\rangle = \Phi([\varphi \wedge \psi])$;
$\|\#\|^{M,a} = \langle 1,1\rangle = \langle\{\langle\rangle \mid \# \in \Delta\}, \{\langle\rangle \mid \neg\# \in \Delta\}\rangle = \Phi([\#])$;

$\|*\|^{M,a} = \langle 0,0\rangle = \langle \{\langle\rangle \mid * \in \Delta\}, \{\langle\rangle \mid \neg * \in \Delta\}\rangle = \Phi([*])$.

iii. Since all elements of any D_α can be written as $I(c)$ for some constant c of type α, we have $\|\forall x_\alpha \varphi\|^{M,a} = \bigcap_{d \in D_\alpha} \|\varphi\|^{M,a[d/x]} = \bigcap_{c_\alpha} \|\varphi\|^{M,a[I(c)/x]} = \bigcap_{c_\alpha} \|[c/x]\varphi\|^{M,a} = \bigcap_{c_\alpha} \Phi([[c/x]\varphi]) = \langle \bigcap_{c_\alpha} \{\langle\rangle \mid [c/x]\varphi \in \Delta\}, \bigcup_{c_\alpha} \{\langle\rangle \mid [c/x]\neg\varphi \in \Delta\}\rangle$ = (by the Henkin property) $\langle \{\langle\rangle \mid \forall x_\alpha \varphi \in \Delta\}, \{\langle\rangle \mid \neg\forall x_\alpha \varphi \in \Delta\}\rangle = \Phi([\forall x_\alpha \varphi])$.

iv. $\|AB\| = F^I_{\|A\|}(\|B\|) = F^I_{\Phi([A])}(\Phi([B])) = \langle R^+, R^-\rangle$, where
$R^+ = \{\langle d_1,\ldots,d_n\rangle \mid \langle \Phi([B]), d_1,\ldots,d_n\rangle \in \Phi([A])^+\}$
$R^- = \{\langle d_1,\ldots,d_n\rangle \mid \langle \Phi([B]), d_1,\ldots,d_n\rangle \in \Phi([A])^-\}$
We see that
$R^+ = \{\langle \Phi([c_1]),\ldots,\Phi([c_n])\rangle \mid ABc_1\ldots c_n \in \Delta\}$
$R^- = \{\langle \Phi([c_1]),\ldots,\Phi([c_n])\rangle \mid \neg ABc_1\ldots c_n \in \Delta\}$
and so that $\|AB\| = \Phi([AB])$.

v. $\|\lambda x_\beta A\|^{M,a}$ = the R such that for all $d \in D_\beta$: $F^I_R(d) = \|A\|^{M,a[d/x]}$ = the R such that for all c_β: $F^I_R(I(c)) = \|A\|^{M,a[I(c)/x]} = \|[c/x]A\|^{M,a}$ $= \Phi([[c/x]A])$. Hence $\|\lambda x_\beta A\|^{M,a} = \langle R^+, R^-\rangle$, where:
$R^+ = \{\langle \Phi([c]), \Phi([c_1]),\ldots,\Phi([c_n])\rangle \mid [c/x](A)c_1\ldots c_n \in \Delta\}$
$R^- = \{\langle \Phi([c]), \Phi([c_1]),\ldots,\Phi([c_n])\rangle \mid \neg[c/x](A)c_1\ldots c_n \in \Delta\}$.
By Lambda Conversion $[c/x]A = \lambda x_\beta(A)c \in \Delta$ and so:
$R^+ = \{\langle \Phi([c]), \Phi([c_1]),\ldots,\Phi([c_n])\rangle \mid \lambda x_\beta(A)c\ c_1\ldots c_n \in \Delta\}$
$R^- = \{\langle \Phi([c]), \Phi([c_1]),\ldots,\Phi([c_n])\rangle \mid \neg\lambda x_\beta(A)c\ c_1\ldots c_n \in \Delta\}$,
from which it follows that $\|\lambda x_\beta A\|^{M,a} = \Phi([\lambda x_\beta A])$.

vi. Since $A = B$ is a 2-formula it holds that $\neg(A = B) \in \Delta$ iff $A = B \notin \Delta$. So $\Phi([A = B]) = \langle \{\langle\rangle \mid A = B \in \Delta\}, \{\langle\rangle \mid \neg(A = B) \in \Delta\}\rangle = \langle \{\langle\rangle \mid [A] = [B]\}, \{\langle\rangle \mid [A] \neq [B]\}\rangle = \langle \{\langle\rangle \mid \Phi([A]) = \Phi([B])\}, \{\langle\rangle \mid \Phi([A]) \neq \Phi([B])\}\rangle = \langle \{\langle\rangle \mid \|A\| = \|B\|\}, \{\langle\rangle \mid \|A\| \neq \|B\|\}\rangle = \|A = B\|$. ■

We are now ready to prove Theorem 5. Suppose $\Pi \models \Sigma$. Then each general model of Π is a general model of some $\sigma \in \Sigma$, and so the set of sentences $\Pi \cup \{\neg T\sigma \mid \sigma \in \Sigma\}$ has no general model. By the star-consistency theorem it is seen that Π, $\{\neg T\sigma \mid \sigma \in \Sigma\} \vdash *$. Hence Π_0, $\{\neg T\sigma \mid \sigma \in \Sigma_0\} \Rightarrow *$ is a provable sequent for some finite Π_0, Σ_0 such that $\Pi_0 \subseteq \Pi$ and $\Sigma_0 \subseteq \Sigma$. Using and we find that $\Pi_0 \Rightarrow \{T\sigma \mid \sigma \in \Sigma_0\}$, $*$ and $\Pi_0 \Rightarrow \Sigma_0$, $*$ are provable sequents as well.
From $\Pi \models \Sigma$ it also follows that each general model of $\{\neg\sigma \mid \sigma \in \Sigma\}$ is a general model of the negation of some $\pi \in \Pi$. By an argument analogous to the one above we find that there are finite $\Pi_1 \subseteq \Pi$ and $\Sigma_1 \subseteq \Sigma$ such that $\{\neg\sigma \mid \sigma \in \Sigma_1\} \Rightarrow \{\neg\pi \mid \pi \in \Pi_1\}$, $*$ is a provable sequent and hence that $\Pi_1, * \Rightarrow \Sigma_1$ is. Now use the Cut Rule to see that $\Pi_0, \Pi_1 \Rightarrow \Sigma_0, \Sigma_1$ is provable, whence $\Pi \vdash \Sigma$. ■

Proof of Theorem 6

THEOREM 6 (repeated). If ϑ is an analysis tree of category t and $^\circ$ is defined as it is in section 2, then the translation ϑ° persists:

$$\text{AX1} \models \forall ij(i \leq j \rightarrow (\vartheta^\circ i \sqsubseteq \vartheta^\circ j)).$$

I'll give a sketch of the proof, leaving details to the reader. For each term A such that in A's type every right bracket is immediately preceded by an s

define a formula $\mathsf{QP}(A, i, j)$, saying that A *quasi-persists* from i to j, by the following induction:

i. $\mathsf{QP}(A, i, j) = \mathsf{T}$, if A is of type e or type s;

ii. $\mathsf{QP}(A, i, j) = \forall x_{\alpha_1}...\forall x_{\alpha_n}((\mathsf{QP}(x_{\alpha_1}, i, j) \wedge ... \wedge \mathsf{QP}(x_{\alpha_n}, i, j)) \rightarrow (Ax_{\alpha_1}...x_{\alpha_n} i \sqsubseteq Ax_{\alpha_1}...x_{\alpha_n} j))$,
if A is of type $\langle \alpha_1...\alpha_n s \rangle$.

By an easy but long induction on the complexity of analysis trees we can prove that AX1 $\models \forall ij(i \leq j \rightarrow \mathsf{QP}(\vartheta^\circ, i, j))$ for every analysis tree ϑ. From this the theorem follows immediately. ■

Acknowledgements

I would like to thank Johan van Benthem and Dick de Jongh for their encouragement, their patience and their good guidance. Tore Langholm made very helpful comments on an earlier version of this paper.

References

Anderson, A.R. and Belnap, N.D. Jr.: 1975, *Entailment: the Logic of Relevance and Necessity*, Vol I, Princeton University Press, Princeton.

Asher, N. and Bonevac, D.: 1985, How Extensional is Extensional Perception?, *Linguistics and Philosophy*, **8**, 203-228.

Asher, N. and Bonevac, D.: 1987, Determiners and Resource Situations, *Linguistics and Philosophy*, **10**, 567-596.

Barwise, J.: 1981, Scenes and Other Situations, *The Journal of Philosophy*, **78**, 369-397.

Barwise, J. and Perry J.: 1983, *Situations and Attitudes*, MIT Press, Cambridge, Massachusetts.

Barwise, J. and Perry J.: 1985, Shifting Situations and Shaken Attitudes, *Linguistics and Philosophy*, **8**, 103-161.

Belnap, N.D. Jr.: 1977, A Useful Four-Valued Logic, in Dunn, J.M. and Epstein, G. (eds.), *Modern Uses of Multiple-Valued Logic*, Reidel, Dordrecht, 8-37.

Bennett, M.: 1974, *Some Extensions of a Montague Fragment of English*, UCLA, Dissertation.

Blamey, S.: 1986, Partial Logic, in Gabbay & Guenthner [1983] Vol III, 1-70.

Church, A.: 1940, A Formulation of the Simple Theory of Types, *The Journal of Symbolic Logic* **5**, 56-68.

Dowty, D.R.: 1982: Grammatical Relations and Montague Grammar, in Jacobson, P. and Pullum, G.K. (eds.), *The Nature of Syntactic Representation*, Reidel, Dordrecht, 79-130.

Dowty, D.R., Wall, R.E. and Peters, S.: 1981, *Introduction to Montague Semantics*, Reidel, Dordrecht.

Dunn, J.M.: 1976, Intuitive Semantics for First-Degree Entailments and 'Coupled Trees', *Philosophical Studies* **29**, 149-168.

Gabbay, D. and Guenthner, F. (eds.): 1983, *Handbook of Philosophical Logic*, Reidel, Dordrecht.

Gallin, D.: 1975, *Intensional and Higher-Order Modal Logic*, North-Holland, Amsterdam.

Humberstone, I. L.: 1981, From Worlds to Possibilities, *Journal of Philosophical Logic*, **10**, 313-339.

Janssen, T.: 1984, Individual Concepts are Useful, in Landman & Veltman [1984], 171-192.

Lambek, J. and Scott, P.J.: Intuitionist Type Theory and Foundations, *Journal of Philosophical Logic*, **10**, 101-115.

Landman, A. and Veltman, F. (eds.): 1984, *Varieties of Formal Semantics*, Foris, Dordrecht.

Langholm, T.: 1987, *Partiality, Truth and Persistence*, Dissertation, Stanford University.

Montague, R.: 1970, Universal Grammar, reprinted in Montague [1974], 222-246.

Montague, R.: 1973, The Proper Treatment of Quantification in Ordinary English, reprinted in Montague [1974], 247-270.

Montague, R.: 1974, *Formal Philosophy*, Yale University Press, New Haven.

Muskens, R.A.: to appear, A Relational Formulation of the Theory of Types, *Linguistics and Philosophy*.
Orey, S.: 1959, Model Theory for the Higher Order Predicate Calculus, *Transactions of the American Mathematical Society* **92**, 72-84.
Perry, J.: 1984, Contradictory Situations, in Landman & Veltman [1984], 313-323.
Tichy, P.: 1982, Foundations of Partial Type Theory, *Reports on Mathematical Logic*, **14**, 59-72.
Van Benthem, J.F.A.K. and Doets, K.: 1983, Higher-Order Logic, in Gabbay & Guenthner [1983] Vol I, 275-329.
Woodruff, P.W.: 1984, Paradox, Truth and Logic. Part I: Paradox and Truth, *Journal of Philosophical Logic* **13**, 213-232.