A Natural Term Language

Jan van Eijck

This paper proposes a natural term language, investigates some of its properties, and discusses some of the advantages of natural term logic (NTL) as a medium for natural language semantics over its rivals and ancestors.

1 Introduction

In 1989 Cor Baayen was the prime mover behind the decision to start long-term work on the logic of natural language at CWI. Work in this area had found an occasional refuge at the centre before, witness Janssen [13], but the seed of a full scale research group in 'Logic and Language' was sown in the Autumn of 1989. Right now, five years later, the group consists of five researchers (six if we count a distinguished longtime guest), all but one supported by external funds. Fortunately for the rest of CWI we anticipate that this rate of growth will not be sustained in the future.

The main focus of current CWI research in 'Logic and Language' is on connections between programming language semantics and natural language semantics and on the design and analysis of suitable representation languages for natural language meaning. The connection with programming is explained by the fact that natural language representation should account for incrementality of processing, i.e., for the fact that we tend to understand each natural language utterance in the context of our understanding of what we have heard before. The semantics of a natural language text T consisting of T_1 followed by T_2 will specify that T_1 sets up a context which is passed on as input to T_2, and that the meaning of T can be described as an increment of the meaning of T_1. This has a straightforward parallel in the analysis of computation: the semantics of a computer program P consisting of two parts P_1 and P_2, in that order, will specify that the result of the computation to which P_1 refers is passed on as input to P_2, and that the output of P_2 for this input is the final output of P.

The paper starts with listing some desiderata for natural language representation, and then makes a new proposal for an incremental language for meaning representation.

2 What Makes an NL Representation Language 'Natural'?

If we assume that the meaning of (descriptive uses of) language should reveal itself in the conclusions we can draw from the truth of natural language utterances, the following requirement is possibly the most important:

Suitability of Representation for Reasoning The representation language should come with a sound and complete calculus for reasoning, and preferably with decidable and efficient sound reasoning systems for useful fragments of it.

First order logic meets this requirement quite well, as we know. More esoteric higher order representation languages such as Montague's [18] Intensional Logic and its derivatives score lower in this dimension, as it is not always obvious how such logics should be axiomatized in the first place.

Another natural requirement on NL representation is the following:

Structural Similarity of Representation The structure of the logical representation language should bear a reasonable amount of similarity to that of the 'source' natural language.

At first sight, first order predicate logic does not meet this requirement at all. Consider (1), with its first order representation (2) (disregarding tense for simplicity). In the logical translation the subject–predicate structure of the natural language source seems to have got lost.

1 *A man walked in.*

2 $\exists x(Mx \wedge Wx)$.

But here the appearance of the representation is misleading. If one thinks of the representation as the result of combining, by functional application, the meaning of the subject, $\lambda P \cdot \exists x(Mx \wedge Px)$, with that of the predicate, $\lambda y \cdot Wy$, then the structure of the source text reveals itself in the meaning representation of (1) before lambda reduction:

3 $(\lambda P \cdot \exists x(Mx \wedge Px))(\lambda y \cdot Wy)$.

Still, the end result (2) of normalizing (3) does not have the same subject–predicate structure as the original. A representation where noun phrases reveal themselves in normal form as terms would satisfy the requirement better.

In the representation of the meaning of a very simple natural language example like (4), an extension of (1), we want to capture the fact that the first sentence of the example makes an indefinite reference to a man, while the second sentence picks up the reference to that same individual.

4 *A man walked in. He looked happy.*

The reason why ordinary first order predicate logic is letting us down here is that we also want our representation language to satisfy the following principle of incremental representation (already hinted at in the introduction above):

Incrementality of Representation The representation of a text T consisting of a subtext T_1 followed by a subtext T_2 should be an increment of the representation of T_1.

This principle is closely connected to, although not identical with, the principle of compositional interpretation which is the main preoccupation of Janssen's [13] investigations in Montague grammar.

In ordinary predicate logic, the natural representation of the first sentence of (4) is (2). This is not a suitable basis to construct a representation of the whole text (4). A natural representation of the pronoun *he* would use the variable x, but this choice runs into the problem that the scope of $\exists x$ in (2) has been closed off.

The theory of discourse representation proposed in Kamp [14] tried to remedy this problem by assuming that every indefinite description gives rise to a so-called *discourse marker*, which can be picked up later on by an anaphoric link (*anaphora* is the standard linguistic name for the connection between the pronoun *he* and its antecedent *a man* in example (4)). Discourse representations *à la* Kamp essentially consist of sets or lists of discourse markers followed by lists of conditions. A discourse representation for the first sentence of (4) is given in (5)

5 $\{x\}, \{Mx, Wx\}$.

In an analysis *à la* Kamp, the representation for the second sentence of the example can introduce a new marker y for *he*, and specify that the markers are to be linked:

6 $\{y\}, \{y = x, Hy\}$.

The representation of the complete example text (4) is the result of an obvious process of 'merging' the two representations:

7 $\{x, y\}, \{Mx, Wx, y = x, Hy\}$.

Later on, Groenendijk and Stokhof [8] observed that the essence of Kamp's proposal is already captured by a very simple modification of ordinary predicate logic. Replace Tarski's truth definition for first order logic by a dynamic variant which interprets a first order formula as a two-place relation on the set of variable assignments. The meaning of φ is then given as $s[\varphi]s'$, where s denotes the input assignment and s' the output assignment. All semantic clauses are tests, in the sense of imperative programming (where a test which gets memory state s as input indicates success by returning s as output and failure by giving no output at all), with the exception of $\exists x$, which has the clause $s[\exists x]s'$ iff $s' = s(x|d)$, for some arbitrary d in the domain of the model under consideration.

If the predicate logical meaning of the first part of (1) is read dynamically in the manner indicated, and the pronoun in the second part of (1) is translated with the same variable, then in the end result this 'dangling' variable turns out to be bound after all, due to the continuing dynamic effect of the 'existential switch':

8 $\exists x(Mx \wedge Wx) \wedge Hx$.

It is clear that the requirement of incremental representation leads in a natural way to a representation language with a dynamic semantics, and we can expect such representation languages to be similar to programming languages in interesting ways. For instance, it turned out that the dynamic version of predicate logic can be analysed with the standard tools from the study of imperative programming, such as Hoare logic (Van Eijck and De Vries [4]). Also, it became clear that dynamic predicate logic and its derivatives suffer from the problem of destructive assignment (see Dekker [1], Vermeulen [24] and Visser [25] for discussion and for possible remedies): because $\exists x$ has been effectively replaced by the assignment statement $x :=?$, an existential quantification destroys the old value of its variable, with the result that anaphoric reference to that value by means of the variable (or a pronoun which has that variable as its translation) becomes impossible. The present proposal adds one more item to the long list of possible solutions for this problem.

3 The Basic Idea

The basic idea of this paper is to design a language with complex 'indefinite' terms, with a dynamic semantics based on term valuations rather than variable assignments. This representation language is structurally more similar to natural language than languages which adopt the term structure of predicate logic, it caters for the needs of incremental representation by its dynamic nature, and it also looks like a promising tool for reasoning, due to its link to Hilbert's epsilon calculus [9]. An earlier application of epsilon logic to the concerns of natural language representation is Meyer Viol [16].

The Natural Term Logic (NTL) to be defined in the next section is intended to achieve several goals at once:

- to give an account of the dynamics of left to right processing by means of a relational semantics (an idea from dynamic predicate logic [8], update logic [23], and similar proposals)
- to use intensional choice functions from epsilon logic [9] and instantial logic [6, 17] for the representation of indefinites,
- to account for the existential and universal quantifier in term of choices (friendly for existentials, unfriendly for universals), thus incorporating a key idea from Game Theoretical Semantics [11],
- to link pronouns to descriptions of their antecedents (the key idea of the so-called e-type analysis of pronouns proposed by Evans [5]),
- to treat universal and existential NPs as terms (one half of this idea incorporated in file change semantics and DRT; the full idea plays a role in traditional syllogistics and natural logic (Purdy [19], Sanchez [21],

Sommers [22]) and was all but killed off by Frege's Begriffsschrift analysis of quantification [7]).

4 Semantics of Natural Term Logic

We start with the non-logical vocabulary of a predicate logical language L. This consists of a set

$$\mathrm{C} = \{c_0, c_1, c_2, \dots\}$$

of *names* (or *individual constants*), for each $n > 0$ a set

$$\mathrm{P}^n = \{P_0^n, P_1^n, P_2^n, \dots\}$$

of *n-place predicate constants* and for each $n > 0$ a set

$$\mathrm{f}^n = \{f_0^n, f_1^n, f_2^n, \dots\}$$

of *n-place function constants.*

It is also useful to have $\perp$ for absurdity, $=$ for identity, and $^-$ for predicate negation. The further logical vocabulary we add to this consists of parentheses, the ϵ term operator (borrowed from Hilbert and Bernays [9]), the colon :, an infinitely denumerable set V of individual variables, the sequential composition connective ; and the connective $\Rightarrow$ for dynamic implication.

Terms and formulas are defined by mutual recursion, as follows (assume $c \in C$, $v \in V$, $f \in \mathrm{f}^n$, $P \in \mathrm{P}^n$):

terms $t ::= c \mid v \mid ft_1 \cdots t_n \mid (\epsilon v : \varphi)$.

formulas $\varphi ::= \perp \mid Pt_1 \cdots t_n \mid \bar{P}t_1 \cdots t_n \mid t_1 = t_2 \mid (\varphi_1;\varphi_2) \mid (\varphi_1 \Rightarrow \varphi_2)$.

The translation in this language of Example (4) becomes:

9 $W(\epsilon x : Mx); H(\epsilon x : Mx)$.

Note that in this translation the reference to the previously mentioned individual *a man* gets picked up by just repeating the term which was used to refer to that individual in the first place: the translation of *he* is the same as that of its antecedent.

An occurrence of v is bound in φ if v occurs inside a subformula ψ of the form $(\epsilon v : \psi)$, otherwise it is free in φ. I will write $\varphi(v_1, \dots, v_n)$ to indicate that the free variables of φ are among $v_1, \dots, v_n$. Just as in standard predicate logic one has to take some care with substitution. If one wants to substitute t for free occurrences of v in φ, one should check that t is free for v in φ, i.e., that no free variable inside t is in danger of becoming bound in the result. Substituting $(\epsilon x : Pxy)$ for x in $R(\epsilon y : Sxy)x$, would run into this problem, for instance. The problem can always be remedied by switching to an alphabetic variant. In the example case, the result would be $R(\epsilon z : S(\epsilon x : Pxy)z)(\epsilon x : Pxy)$. I will use $\varphi(t/v)$ for the result of substituting t for all free occurrences of v in φ, with a switch to an alphabetic variant if the need arises. The result of simultaneous substitution of $t_1, \dots, t_n$ for free occurrences of $v_1, \dots, v_n$, respectively, in φ, with renaming of bound variables as the need arises, will be written as $\varphi(t_1/v_1, \dots, t_n/v_n)$.

Let $M = \langle \mathrm{dom}(M), \mathrm{int}(M) \rangle$ be a first order model for the vocabulary of L. I will use c^M, f^M, P^M as shorthand for $\mathrm{int}(M)(c)$, $\mathrm{int}(M)(f)$ and $\mathrm{int}(M)(P)$, respectively.

Let A be the set of variable assignments for L in M, i.e., let A be the set of functions $\mathrm{dom}(M)^V$. We will use a, a' for members of A, and $a(v|d)$ for the assignment a' with $a'(t) = t$ for $t \neq v$ and $a'(t) = d$ for $t = v$.

Let T be the set of terms of L. We consider the set of partial functions

$$\mathrm{dom}(M)^{[A \times T]}$$

as total functions in

$$B = (\mathrm{dom}(M) \cup \{\uparrow\})^{A \times T}.$$

For $T' \subseteq T$ and $s \in B$, let $s \restriction T'$ be the function $s' \in B$ given by:

$$s'(a,t) = s(a,t) \text{ if } t \in T', \text{ and } s'(a,t) = \uparrow \text{ otherwise.}$$

Define $\mathrm{dom}(s)$ as:

$$\{\langle a, t \rangle \in A \times T \mid s(a,t) \neq \uparrow\}.$$

The relation $\leq$ on B is defined as $s \leq s'$ iff $s' \restriction \mathrm{dom}(s) = s \restriction \mathrm{dom}(s)$.

The set $S \subseteq B$ of states for L in M is the set of those $s \in B$ satisfying the following:

- $s(a, v) = a(v)$,
- $s(a, c) = c^M$,
- $s(a, ft_1 \cdots t_n) = \begin{cases} f^M(s(a,t_1), \dots, s(a,t_n)) \\ \quad \text{if } s(a,t_1) \neq \uparrow, \dots, s(a,t_n) \neq \uparrow, \\ \uparrow \quad \text{otherwise.} \end{cases}$
- $s(a, \epsilon v : \varphi) = \begin{cases} d & \text{for some } d \in [\![\varphi]\!]^v_{s,a} \quad \text{if } [\![\varphi]\!]^v_{s,a} \neq \emptyset, \\ \uparrow & \text{otherwise.} \end{cases}$

where $[\![\varphi]\!]^v_{s,a}$ is

$$\{d \in \mathrm{dom}(M) \mid s, a(v|d)[\varphi]\},$$

with $s, a(v|d)[\varphi]$ given by the following clauses (where we assume $s, s', s'' \in S$ and $a \in A$):

$s, a[\varphi]$	iff	$\exists s'$ with $s \leq s'$ and $s, a[\varphi]s'$,
$s, a[\bot]s'$	iff	never,
$s, a[Pt_1 \cdots t_n]s'$	iff	$s \leq s', s'(a,t_1) \neq \uparrow, \dots, s'(a,t_n) \neq \uparrow$, $\langle s'(a,t_1), \dots, s'(a,t_n) \rangle \in P^M$,
$s, a[\bar{P}t_1 \cdots t_n]s'$	iff	$s \leq s', s'(a,t_1) \neq \uparrow, \dots, s'(a,t_n) \neq \uparrow$, $\langle s'(a,t_1), \dots, s'(a,t_n) \rangle \notin P^M$,
$s, a[t_1 = t_2]s'$	iff	$s \leq s', s'(a,t_1) \neq \uparrow, s'(a,t_2) \neq \uparrow$, $s'(a,t_1) = s'(a,t_2)$,
$s, a[\varphi_1; \varphi_2]s'$	iff	$\exists s''$ with $s, a[\varphi_1]s''$ and $s'', a[\varphi_2]s'$,
$s, a[\varphi_1 \Rightarrow \varphi_2]s'$	iff	$s = s'$ and $\forall s''$ with $s, a[\varphi_1]s''$ it holds that $s'', a[\varphi_2]$.

Note that the definition of the state set S for L in M is phrased in terms of S itself, a potentially dangerous situation. The next proposition shows that for every M for L, the set of states for L in M is non-empty.

PROPOSITION 1 *If S is the set of states for L in M, then $S \neq \emptyset$.*

Proof: The proof uses a variation on a standard Skolem expansion argument (see e.g. Hodges [12]).

Start out with the following language L_0:

terms $t ::= c \mid v \mid ft_1 \cdots t_n$.

formulas $\varphi ::= \bot \mid Pt_1 \cdots t_n \mid \bar{P}t_1 \cdots t_n \mid t_1 = t_2 \mid (\varphi_1; \varphi_2) \mid (\varphi_1 \Rightarrow \varphi_2)$.

Let T_0 be the set of terms of L_0. Surely, states for L_0 exist, for a state for L_0 is just a mapping from assignments to classical first order term valuations. Let S_0 be the set of states for L_0. Note that $[\![\varphi]\!]^v_{s,a}$ is well-defined for $\varphi \in L_0$.

Next, expand the language in layers. Assume T_k, the set of terms for layer k, and L_k, the set of formulas for layer k, are given. Then T_{k+1} and L_{k+1} are given by the following clauses:

terms $t ::= c \mid v \mid ft_1 \cdots t_n \mid (\epsilon v : \varphi)$ with $\varphi \in L_k$,

formulas $\varphi ::= \bot \mid Pt_1 \cdots t_n \mid \bar{P}t_1 \cdots t_n \mid t_1 = t_2 \mid (\varphi_1; \varphi_2) \mid (\varphi_1 \Rightarrow \varphi_2)$ with $t \in T_{k+1}$.

We may assume that S_k, the set of states for L_k, is non-empty. Also, we may assume that $[\![\varphi]\!]^v_{s,a}$ is well-defined for $\varphi \in L_k$, $s \in S_k$.

Take some member $s_k \in S_k$ and use it to construct a member s of S_{k+1} as follows.

- if $t \in T_k$, then $s(a,t) := s_k(a,t)$,
- if $t \in T_{k+1} - T_k$, then t has the form $(\epsilon v : \varphi)$, with $\varphi \in L_k$, and we set $s(a, \epsilon v : \varphi) := \begin{cases} d & \text{for some } d \in [\![\varphi]\!]^v_{s_k,a} \text{ if } [\![\varphi]\!]^v_{s_k,a} \neq \emptyset \\ \uparrow & \text{otherwise.} \end{cases}$

Obviously, this can always be done, so we have shown that $S_{k+1} \neq \emptyset$, and moroever, that every $s_k \in S_k$ can be extended to an $s_{k+1} \in S_{k+1}$. Also, if $s \in S_{k+1}$, $[\![\varphi]\!]^v_{s,a}$ will be well-defined for $\varphi \in L_{k+1}$.

The full language L is $\bigcup_{k=0}^{\infty} L_k$, the full set of terms T is $\bigcup_{k=0}^{\infty} T_k$. The set of states S for L in M is given by:

$$\{s \in B \mid s \restriction T_k \in S_k, 0 \leq k < \infty\}.$$

As each S_k is non-empty and each $s_k \in S_k$ has an extension $s_{k+1} \in S_{k+1}$, this proves that $S \neq \emptyset$. ∎

6 Truth, Validity and Entailment

The following definitions of truth, validity and entailment round off the presentation of the semantics of L.

DEFINITION 1 (TRUTH) *φ is true in L-model M if $\exists s \in S$, $\exists a \in A$ with $s, a[\varphi]$, where S is the set of states for L in M and $A = \text{dom}(M)^V$.*

Here are some examples of first order equivalents of NTL formulas to illustrate the definition (where $\models$ denotes NTL truth, and $\models_c$ the classical first order notion of truth).

- $M \models B(\epsilon x : Ax)$ iff $M \models_c \exists x(Ax \wedge Bx)$
- $M \models R(\epsilon x : Ax)(\epsilon x : Bx)$ iff $M \models_c \exists x \exists y(Ax \wedge By \wedge Rxy)$.
- $M \models R(\epsilon x : Ax)(\epsilon x : Ax)$ iff $M \models_c \exists x(Ax \wedge Rxx)$.
- $M \models \bar{R}(\epsilon x : Ax)(\epsilon x : Ax)$ iff $M \models_c \exists x(Ax \wedge \neg Rxx)$.
- $M \models A(\epsilon x : Ax) \Rightarrow B(\epsilon x : Ax)$ iff $M \models_c \forall x(Ax \rightarrow Bx)$.

DEFINITION 2 (VALIDITY) *φ is valid if φ is true in every L-model M.*

Here is an example validity (with $\models \varphi$ for 'φ is valid'):

$$\models A(\epsilon x : Bx) \Rightarrow B(\epsilon x : Ax).$$

DEFINITION 3 (ENTAILMENT) *φ entails ψ if the truth of φ in L-model M entails the truth of $\varphi; \psi$ in L-model M.*

This may sound slightly non-standard. The reason for looking at the conclusion 'in the context of the premise' is of course that the conclusion may contain translations of pronouns that find an antecedent in the premise.

Here is an example entailment (with $\models$ for the entailment relation):

$$\begin{aligned}&(P(\epsilon x : Ax) \Rightarrow P(\epsilon x : Bx)); (P(\epsilon x : Bx) \Rightarrow P(\epsilon x : Cx))\\ &\models P(\epsilon x : Ax) \Rightarrow P(\epsilon x : Cx).\end{aligned}$$

The term language L is a dynamic variant of Hilbert and Bernays' epsilon logic (see [9]). The dynamic epsilon terms are meant to represent the process of referring indefinitely to individual entities (by means of indefinite descriptions) in natural language.

Moreover, it is an intensional version, for two formulas φ and ψ which are logically equivalent (i.e., which entail one another) can give rise to different 'epsilon choices' in the sense that for some state s, $s(\epsilon v : \varphi) \neq s(\epsilon v : \psi)$. In extensional epsilon logic (cf. Leisenring [15]) this situation cannot occur. For our purposes the intensionality of choice is indispensable, for we want to be able to use logically equivalent indefinite descriptions for indefinite reference to different individuals.

Some extra notation is useful for that. Note that according to the semantic clauses, $(\bot \Rightarrow \bot)$ is valid. Let $(\epsilon v : \varphi)_n$ abbreviate the following:

$$(\epsilon v : (\varphi; \underbrace{((\bot \Rightarrow \bot); ((\bot \Rightarrow \bot); \ldots)))}_{n \text{ times}})$$

Then we can use $(\epsilon x : Ax)_0$, $(\epsilon x : Ax)_1$, $(\epsilon x : Ax)_2$, and so on, to translate different occurrences of an indefinite description in a text.

10 *A beer for her, a beer for him, and an orange juice for me.*

In ordering a round of drinks for three, as in (10), a repetetion of the same indefinite description should not entail that the same glass is to be shared by two of your friends, so the translation should use $(\epsilon x : Bx)_0$ and $(\epsilon x : Bx)_1$, for the two different glasses of beer.

7 An Update Formulation of the Semantics

If $I \subseteq S$, where S is the state set for L in some given M, let $I[\varphi]$ be the set of states given by:

$$\{s \in S \mid \exists s' \in I \exists a \in A : s', a[\varphi]s\}.$$

We can use this notion to define a global index elimination procedure for NTL. An index for L is a pair $\langle M, I\rangle$, where M is a model for L and $I \subseteq S, I \neq \emptyset$, with S the state set for L in M.

If U is a set of indices, then define:

$$U|\varphi| = \{\langle M, I[\varphi]\rangle \mid \langle M, I\rangle \in U \text{ and } I[\varphi] \neq \emptyset\}.$$

Let W be the class of all pairs $\langle M, S\rangle$, with M a model for L and S the full state set for L in M. Then φ is valid iff $(W|\varphi|)_0$ equals the class of all models for L; here $(\ )_0$ denotes the operation of taking the first projection.

Let $\mathcal{U}$ be the power set of the class of all indices for L. A natural information ordering on $\mathcal{U}$ can now be given in terms of the local ordering $\leq$ on states for a given model, which we first extend to state sets, as follows:

$$I \leq J \text{ iff for all } s \in J \text{ there is an } s' \in I \text{ with } s' \leq s.$$

Next, we set, for $U_1, U_2 \in \mathcal{U}$:

$$U_1 \leq U_2 \text{ iff for all } \langle M, J\rangle \in U_2 \text{ there is a } I \leq J \text{ with } \langle M, I\rangle \in U_1.$$

This distinction between a global and a local perspective on the semantics should be compared to a similar distinction made for dynamic modal predicate logic, in Van Eijck and Cepparello [3]. The distinction is the key to extending the present proposal with epistemic operators such as *maybe*, an extension which is beyond the scope of the present paper, however.

8 Some Example Meaning Representations

We will now illustrate the potential of the language by a brief discussion of examples, some of them famous from the literature.

11 *Some farmer owns a donkey. He beats it.*

Natural translation:

12 $O(\epsilon x : Fx)(\epsilon y : Dy); B(\epsilon x : Fx)(\epsilon y : Dy)$.

This does have the expected meaning, for it is equivalent to the following first order sentence:

13 $\exists x \exists y(Fx \wedge Dy \wedge Oxy \wedge Bxy)$.

The advantage of the NTL version is the fact that the translation of the second part is an increment of that of the first.

14 *If a farmer owns a donkey, he beats it.*

The translation of this key motivating example for Discourse Representation Theory:

15 $O(\epsilon x : Fx)(\epsilon y : Dy) \Rightarrow B(\epsilon x : Fx)(\epsilon y : Dy)$.

The first order equivalent of this:

16 $\forall x \forall y((Fx \wedge Dy \wedge Oxy) \rightarrow Bxy)$.

This example derives its fame from the fact that its first order translation is so hard to get in a compositional way. The NTL version does not face such a problem.

17 *Every farmer who owns a donkey beats it.*

To treat the example it is useful to have a notation of universal terms. Let $P(\dots(\tau v : \varphi)\dots)$ be shorthand for:

$$(\epsilon v : \varphi = \epsilon v : \varphi) \Rightarrow P(\dots(\epsilon v : \varphi)\dots).$$

Then (17) gets as natural translation:

18 $B(\tau x : Fx; Ox(\epsilon y : Dy))(\epsilon y : Dy)$.

This is shorthand for:

19 $$(\epsilon x : Fx; Ox(\epsilon y : Dy)) = (\epsilon x : Fx; Ox(\epsilon y : Dy)) \\ \Rightarrow B(\epsilon x : Fx; Ox(\epsilon y : Dy))(\epsilon y : Dy),$$

which has the same first order equivalent as (15).

20 *Every farmer owns a donkey. He beats it (regularly).*

The discourse representation literature [14] claims that the example is ill-formed, a nice illustration of the fact that linguistic observation, like all observation in science, is biased by theory. Unlike discourse representation theory, which cannot handle it, we can afford to assume that this example is linguistically acceptable. Here is the translation:

21 $O(\tau x : Fx)(\epsilon y : Dy); B(\tau x : Fx)(\epsilon y : Dy).$

Its first order equivalent:

22 $\forall x(Fx \rightarrow \exists y(Dy \wedge Oxy)) \wedge \forall x(Fx \rightarrow \exists y(Dy \wedge Bxy)).$

If this isn't close enough, we can relax our regime of pronoun translation which says that pronouns are to be translated by repetition of the term translation of their antecedent.

23 $O(\tau x : Fx)(\epsilon y : Dy).$

In fact, from the truth of (23) we get that in every setting the term $(\tau x : Fx)$ can be replaced *salva veritatis* by $(\tau x : Fx; Ox(\epsilon y : Dy))$. Using this as pronoun translation we get:

24 $O(\tau x : Fx)(\epsilon y : Dy);\ B(\tau x : Fx; Ox(\epsilon y : Dy))(\epsilon y : Dy).$

The first order equivalent of (24):

25 $\forall x(Fx \rightarrow \exists y(Dy \wedge Oxy)) \wedge \forall x(Fx \rightarrow \forall y((Dy \wedge Oxy) \rightarrow Bxy)).$

26 *Every farmer owns a donkey. Some farmer beats it.*

Like the previous example, this one is beyond the scope of most current semantic theories. Outside of the mainstream of natural language semantics, Game Theoretical Semantics [10] does sketch an account, however. NTL now incorporates this treatment in standard dynamic semantics. Here is a translation:

27 $O(\tau x : Fx)(\epsilon y : Dy); B(\epsilon x : Fx)(\epsilon y : Dy).$

Its first order equivalent:

28 $\forall x(Fx \rightarrow \exists y(Dy \wedge Oxy)) \wedge \exists x(Fx \wedge \exists y(Dy \wedge Bxy)).$

Again, if this isn't close enough, we can relax the pronoun translation regime and observe that the truth of the first half of (27) guarantees that we can replace the term $(\epsilon x : Fx)$ by $(\epsilon x : Fx \wedge Ox(\epsilon y : Dy))$ without changing truth conditions. This gives the following alternative translation:

29 $O(\tau x : Fx)(\epsilon y : Dy);\ B(\epsilon x : Fx; Ox(\epsilon y : Dy))(\epsilon y : Dy),$

with first order equivalent:

30 $\forall x(Fx \rightarrow \exists y(Dy \wedge Oxy)) \wedge \exists x(Fx \wedge \exists y(Dy \wedge Oxy \wedge Bxy)).$

Of course, all first order equivalents in this section were given *ad hoc.* In the next section the issue of reasoning about and in NTL will be addressed in a more systematic way.

9 Assertion Reasoning For Natural Term Logic

One approach to developing a calculus for a dynamic logic is by using assertions, in the style of Hoare logic or quantified dynamic logic. The statements from the dynamic language to be analyzed then become modalities, and we interpret $\langle\varphi\rangle X$ as: there is some state s' reachable from the current state s with $s, a[\varphi]s'$ and X holds at s', and its dual $[\varphi]X$ as: for all states s' reachable from the current state s with $s, a[\varphi]s'$, X holds at s'.

Here are some axioms for an assertion calculus along these lines (we use X as metavariable over assertion statements, and $\top$ as abbreviation of some abitrary tautology).

A 1 $\langle\boldsymbol{\varphi_1; \varphi_2}\rangle X \leftrightarrow \langle\boldsymbol{\varphi_1}\rangle\langle\boldsymbol{\varphi_2}\rangle X$.

A 2 $\langle\boldsymbol{\varphi_1 \Rightarrow \varphi_2}\rangle X \leftrightarrow (X \wedge [\boldsymbol{\varphi_1}]\langle\boldsymbol{\varphi_2}\rangle\top)$.

A 3 $\langle\boldsymbol{P(\ldots(\epsilon v : \varphi)\ldots)}\rangle X \leftrightarrow \exists x(\langle\boldsymbol{\varphi}\rangle\top \wedge \langle\boldsymbol{P(\ldots x\ldots)}\rangle X(v/(\epsilon v : \varphi)))$.

A 4 $[\boldsymbol{P(\ldots(\epsilon v : \varphi)\ldots)}]X \leftrightarrow \forall x(\langle\boldsymbol{\varphi}\rangle\top \rightarrow [\boldsymbol{P(\ldots x\ldots)}]X(v/(\epsilon v : \varphi)))$.

A 5 $\langle\boldsymbol{Pt_1 \cdots t_n}\rangle X \leftrightarrow (Pt_1 \cdots t_n \wedge X)$.

Condition on A-5: none of the t_i is of the form $(\epsilon v : \varphi)$.

A 6 $[\boldsymbol{Pt_1 \cdots t_n}]X \leftrightarrow (Pt_1 \cdots t_n \rightarrow X)$.

Condition on A-6: none of the t_i is of the form $(\epsilon v : \varphi)$.

Further discussion of these axioms is beyond the present scope (see Van Eijck [2] for a similar calculus for dynamic predicate logic).

Instead, we confine ourselves to illustrating their use by means of the following example.

$$
\begin{array}{l}
\langle\boldsymbol{O(\epsilon x : Fx)(\epsilon y : Dy) \Rightarrow B(\epsilon x : Fx)(\epsilon y : Dy)}\rangle\top \\
\leftrightarrow\ [\boldsymbol{O(\epsilon x : Fx)(\epsilon y : Dy)}]\langle\boldsymbol{B(\epsilon x : Fx)(\epsilon y : Dy)}\rangle\top \\
\leftrightarrow\ \forall x(\langle\boldsymbol{Fx}\rangle\top \rightarrow [\boldsymbol{Ox(\epsilon y : Dy)}]\langle\boldsymbol{Bx(\epsilon y : Dy)}\rangle\top) \\
\leftrightarrow\ \forall x(Fx \rightarrow [\boldsymbol{Ox(\epsilon y : Dy)}]\langle\boldsymbol{Bx(\epsilon y : Dy)}\rangle\top) \\
\leftrightarrow\ \forall x(Fx \rightarrow \forall y(\langle\boldsymbol{Dy}\rangle\top \rightarrow [\boldsymbol{Oxx}]\langle\boldsymbol{Bxy}\rangle\top)) \\
\leftrightarrow\ \forall x(Fx \rightarrow \forall y(Dy \rightarrow [\boldsymbol{Oxy}]\langle\boldsymbol{Bxy}\rangle\top)) \\
\leftrightarrow\ \forall x(Fx \rightarrow \forall y(Dy \rightarrow (Oxy \rightarrow Bxy))).
\end{array}
$$

10 Natural Deduction For Natural Term Logic

A different approach to reasoning with term logic is given by the following example rules from a natural deduction calculus with ordered premises.

A 1 $\dfrac{\varphi; \psi}{\varphi}$

A 2 $\dfrac{\varphi; \psi(\epsilon v : \chi/v)}{\psi(\epsilon v : (\chi \wedge \varphi)/v)}$

Condition on A-2: φ should not contain occurrences of epsilon terms. An example application of the second rule is:

$$\frac{W(\epsilon x : Mx); H(\epsilon x : Mx)}{H(\epsilon x : Mx \wedge Wx)}$$

These rules are for purposes of illustration only. Axiom A-2 needs a more complex formulation to deal with cases where the first member φ of the sequence $\varphi; \psi$ contains more than one epsilon term.

Further work on natural deduction for NTL should establish a connection with natural deduction for standard epsilon logic (see Meyer Viol [17] for a treatment).

11 Conclusion and Further Directions

We have sketched a representation for natural language meaning which treats indefinite descriptions as terms. An obvious first extension is definite descriptions, for which standard logic has a term treatment using the ι term operator (see e.g. Reichenbach [20] for an illuminating discussion). Further extensions of the representation language that seem interesting are epistemic modalities and, in a different direction, plural terms.

References

1. P. Dekker. *Transsentential Meditations.* PhD thesis, Department of Philosophy, University of Amsterdam, 1993.
2. J. van Eijck. Axiomatizing dynamic predicate logic with quantified dynamic logic. In J. van Eijck and A. Visser, editors, *Logic and Information Flow*, pages 30–48. MIT Press, Cambridge Mass, 1994.
3. J. van Eijck and G. Cepparello. Dynamic modal predicate logic. In M. Kanazawa and C.J. Pinón, editors, *Dynamics, Polarity and Quantification*, pages 251–276. CSLI, Stanford, 1994.
4. J. van Eijck and F.J. de Vries. Dynamic interpretation and Hoare deduction. *Journal of Logic, Language, and Information*, 1:1–44, 1992.
5. G. Evans. Pronouns. *Linguistic Inquiry*, 11:337–362, 1980.
6. K. Fine. *Reasoning With Arbitrary Objects.* Blackwell, Oxford, 1985.
7. G. Frege. *Begriffsschrift, eine der arithmetischen nachgebildete Formelprache des reinen Denkens.* Verlag Nebert, Halle, 1879.
8. J. Groenendijk and M. Stokhof. Dynamic predicate logic. *Linguistics and Philosophy*, 14:39–100, 1991.
9. D. Hilbert and P. Bernays. *Grundlagen der Mathematik.* Springer, Berlin, 1939. Second edition: Berlin 1970.
10. J. Hintikka. Quantifiers in natural languages: Some logical problems. In I. Niiniluoto and E.E. Saarinen, editors, *Essays in Mathematical and Philosophical Logic*, pages 295–314. Reidel, Dordrecht, 1979.
11. J. Hintikka and J. Kulas. *Anaphora and Definite Descriptions: Two Applications of Game-Theoretical Semantics.* Reidel, Dordrecht, 1985.

12. W. Hodges. *Model Theory.* Cambridge University Press, 1993.
13. T.M.V. Janssen. *Foundations and Applications of Montague Grammar.* CWI Tract 19. CWI, Amsterdam, 1986.
14. H. Kamp. A theory of truth and semantic representation. In J. Groenendijk et al., editors, *Formal Methods in the Study of Language.* Mathematisch Centrum, Amsterdam, 1981.
15. A.C. Leisenring. *Mathematical Logic and Hilbert's ϵ-symbol.* MacDonald, London, 1969.
16. W. Meyer Viol. Partial objects and DRT. In P. Dekker and M. Stokhof, editors, *Proceedings of the Eighth Amsterdam Colloquium*, pages 381–402. ILLC, Amsterdam, 1992.
17. W. Meyer Viol. *Instantial Logic.* PhD thesis, OTS, Utrecht, to appear.
18. R. Montague. The proper treatment of quantification in ordinary english. In J. Hintikka e.a., editor, *Approaches to Natural Language*, pages 221–242. Reidel, 1973.
19. W.C. Purdy. A variable-free logic for anaphora. Manuscript, Syracuse University, 1992.
20. H. Reichenbach. *Elements of Symbolic Logic.* Macmillan, London, 1947.
21. V. Sánchez. *Studies on Natural Logic and Categorial Grammar.* PhD thesis, University of Amsterdam, 1991.
22. F. Sommers. *The Logic of Natural Language.* Cambridge University Press, 1982.
23. F. Veltman. Defaults in update semantics. Technical report, Department of Philosophy, University of Amsterdam, 1991. To appear in the Journal of Philosophical Logic.
24. C.F.M. Vermeulen. *Explorations of the Dynamic Environment.* PhD thesis, OTS, Utrecht, 1994.
25. A. Visser. The design of dynamic discourse denotations. Lecture notes, Utrecht University, February, 1994.