5

Discourse Plans and Linguistic Meaning

Alexander Dikovsky

Linguistic meaning is presented in this paper as a result of dynamic planning. The plans, we call them *discourse plans*, are tree-like structures composed of primitive situations and establishing referential links between the situations and the context. Linguistic meanings are normal form second order Lambda-terms available through a simple morphism from realizable discourse plans.

5.1 Introduction

There are two different approaches to meaning formalization: one *logical*, another *cognitive*. The logical approach, going back to Frege and developed and explicitly applied to natural language semantics by Montague, is characterized by establishing semantics from primitive semantic structures of *sentential type*. This approach is most consistently represented by type logical grammars (see collections (Gamut 1991, van Benthem and ter Meulen 1997)).

The cognitive approach treats linguistic meaning as a structure (e.g., a graph, a DAG, a feature structure) *encoding the content* of a text in enough detail to represent it at the syntactic, morphological and phonological levels, *independent of the propositional attitude*. This is the way it appears in the theory "Meaning ⇔ Text" (Mel'čuk 1997). Formalized or completely formal descriptive definitions of cognitive meaning are well known: cf. semantic networks of Sowa 1991 and situation theory of Barwise and Perry 1983. However, these definitions construct meaning structures *per se* independent of their realization in natural

Proceedings of FGVienna: The 8th Conference on Formal Grammar.
Gerald Penn (ed.).

language. So in fact, they define an information and not a meaning structure. In this paper, we propose a formal definition of linguistic meaning using specific linguistic realization means. Basically, our definition is functional and object-oriented. All meaning elements may be seen as typed functions, the types forming a hierarchy based on a genericity relation. Any realistic implementation of this definition must be feature driven with feature inheritance. In the theoretical model of this paper the features are abstracted. Our primitive meaning structures are in fact very close to the primitive situations of Barwise and Perry 1983. The difference is that we use specific semantical linguistic types and roles. The basic particularity of our definition is that building complex meanings from primitive situations needs planning. The planning becomes necessary for at least three reasons:

1. Meaning composition may be complicated by type conflicts, so it may need type conversion.

2. Name and reference scoping does not always conform with meaning composition.

3. To realize a particular communicative structure (theme / rheme partition, focus) may need in general specific transformations of situation argument structure (semantic diatheses).

In that way, we come to *discourse plans* [1] - meaning specifications transformed into meanings by a simple morphism under certain realizability conditions.

5.2 Roles, Types and Situations

We start with primitives from which the discourse plans are defined. The first primitives are *cognitive roles* marking arguments of meanings. We choose in this paper the following list of cognitive roles, which is of course far from being complete, but largely suffices for the examples below: **ACT** (action), **EVT** (event), **ST** (state), **TNS** (tense), **LOC** (location), **AGT** (agent), **CAG** (counteragent), **PAT** (patent), **OBJ** (object), **EXP** (experiencer), **ADR** (addresser), **ORG** (origin), **RCP** (recipient), **DST** (destination), **EFF** (effect), **GL** (goal), **CND** (condition), **CSE** (cause), **INS** (instrument), **RES** (result), **CLS** (class), **EL** (element), **PRO** (proprietor), **ATR** (attribute), **QUA**(qualia), **DEF** (definiendum), **STR** (strength), **INT** (intensivity).

Types. Next primitives are linguistic meaning types which serve for classification of meanings and for class inheritance.

[1] We came to the idea of planning and to the notion itself of discourse plan with a knowledge of a literature on first language acquisition in little children. In Dikovsky 2003 we state without details a cognitive hypothesis accounting for dynamic planning of meanings.

Primitive types make a finite lattice $(\mathbf{P}, \prec)$ containing four pairwise incomparable elementary types:
− the type of *nominators* $\mathbf{n}$ intuitively corresponding to "things" in the most general sense,
− the type of *sententiator* $\mathbf{s}$ intuitively corresponding to relations between "things"(actions / processes / events, etc.),
− the type of *qualifiers* $\mathbf{q}$, intuitively corresponding to meanings qualifying nominators,
− the type of *circumscriptors* $\mathbf{c}$, intuitively corresponding to meanings, qualifying qualifiers, sententiators, ad and circumscriptors themselves.

$\prec$ is an instance/generic partial order relation on primitive types. For example, the type of nominators $\mathbf{n}$ has the instances $\mathbf{n_a}, \mathbf{n_{\tilde{a}}} \prec \mathbf{n}$ of *animated/inanimated nominators*, the type of sententiators $\mathbf{s}$ has the instance $\mathbf{s_{om}} \prec \mathbf{s}$ of *oriented movement sententiators* (e.g., $\mathbf{run_1}$, **mount**), and also the instance $\mathbf{s_{bf}} \prec \mathbf{s}$ of *belief sententiators*, the type $\mathbf{q}$ of qualifiers has the instance $\mathbf{q_{cp}} \prec \mathbf{q}$ of *comparison qualifiers* (e.g., **better**, **worst**), and the instance $\mathbf{q_{qu}} \prec \mathbf{q}$ of *quantification qualifiers* (e.g., **all**, **neither**, **four**), the type $\mathbf{c}$ of circumscriptors has the instance $\mathbf{c_{dg}} \prec \mathbf{c}$ of *degree circumscriptors* (e.g., **more**, **especially**).

Let $\bot = \wedge\mathbf{P}$ be the *least* and $\top = \vee\mathbf{P}$ be the *greatest* primitive types.

Complex types. An important particularity of linguistic meaning types is that their definition includes options and iteration.

Option types. $\mathbf{O} = \{\mathbf{u^{(0)}} \mid \mathbf{u} \in \mathbf{P} \setminus \{\bot, \top\}\}$ is the set of *option types.*

Iterative types. $\mathbf{I} = \{\mathbf{u}^{(\omega)} \mid \mathbf{u} \in \mathbf{P} \setminus \{\bot, \top\}\}$ is the set of *iterative types.*

Basic types. $\mathbf{B} = \mathbf{P} \cup \mathbf{O} \cup \mathbf{I}$ is the set of *basic types.* The type instance order $\prec$ is naturally extended to $\mathbf{B} \setminus \{\bot, \top\} : \mathbf{u^{(0)}} \prec \mathbf{v^{(0)}}$ and $\mathbf{u}^{(\omega)} \prec \mathbf{v}^{(\omega)}$ for all $\mathbf{u} \prec \mathbf{v}$ in $\mathbf{B} \setminus \{\bot, \top\}$.

Being combined with nominators, the qualifiers give new nominators. Being combined with qualifiers, the circumscriptors form new qualifiers, and being combined with sententiators, they form new sententiators. From this follows the particularity, that all words have functional meaning types with an iterative subtype. For instance, common nouns have type $(\mathbf{q}^{(\omega)} \rightarrow \mathbf{n})$. They are interpreted by functions from lists of qualifier type meanings to nominator type meanings. Adjectives have type $(\mathbf{c}^{(\omega)} \rightarrow \mathbf{q})$. They are interpreted by functions from lists of circumscriptor type meanings to qualifier type meanings. Adverbs have type $(\mathbf{c}^{(\omega)} \rightarrow \mathbf{c})$. Their meanings are functions from lists of circumscriptor type meanings to circumscriptor type meanings. Non-functional meanings are of only two kinds: $\emptyset^{\mathbf{u^{(0)}}}$ (*empty meaning* of option type $\mathbf{u^{(0)}}$)

and $\mathbf{nil}^{\mathbf{u}^{(\omega)}}$ (*empty list* of type $\mathbf{u}^{(\omega)}$). This is why, we distinguish between iterative type and non-iterative type function arguments, the latter being named *actants*. For instance, the meaning of the majority of transitive verbs has two actants and the value of type $\mathbf{s}$. Meanwhile, many words with meanings of other value types also have actants. For instance, the nominator, which is the meaning of the noun '$PART$', has two nominator type actors (*what*) and (*of-what*). So it has the functional type $(\mathbf{q}^{(\omega)}\mathbf{nn} \rightarrow \mathbf{n})$.

Notation. The expression $(\mathbf{u}_1\mathbf{u}_2...\mathbf{u}_k \rightarrow \mathbf{v})$ denotes the type $(\mathbf{u}_1 \rightarrow (\mathbf{u}_2 \rightarrow ... (\mathbf{u}_k \rightarrow \mathbf{v})...))$.

Meanings with actants are the first primitive meaning structures. We call them *situations*.

Situation types. There are four families of *situation types*:
$\mathbf{T}^{\mathbf{s}} = \{(\mathbf{c}^{(\omega)}\mathbf{u}_1...\mathbf{u}_r \rightarrow \mathbf{s}) \mid \mathbf{u}_i \in \mathbf{P} \cup \mathbf{O}, r \geq 0\}$,
$\mathbf{T}^{\mathbf{n}} = \{(\mathbf{q}^{(\omega)}\mathbf{u}_1...\mathbf{u}_r \rightarrow \mathbf{n}) \mid \mathbf{u}_i \in \mathbf{P} \cup \mathbf{O}, r \geq 0\}$,
$\mathbf{T}^{\mathbf{q}} = \{(\mathbf{c}^{(\omega)}\mathbf{u}_1...\mathbf{u}_r \rightarrow \mathbf{q}) \mid \mathbf{u}_i \in \mathbf{P} \cup \mathbf{O}, r \geq 0\}$,
$\mathbf{T}^{\mathbf{c}} = \{(\mathbf{c}^{(\omega)}\mathbf{u}_1...\mathbf{u}_r \rightarrow \mathbf{c}) \mid \mathbf{u}_i \in \mathbf{P} \cup \mathbf{O}, r \geq 0\}$.
$\mathbf{T} = \mathbf{B} \cup \mathbf{T}^{\mathbf{s}} \cup \mathbf{T}^{\mathbf{n}} \cup \mathbf{T}^{\mathbf{q}} \cup \mathbf{T}^{\mathbf{c}}$ is the set of *types*.

Situations are defined in the dictionary by *situation profiles* consisting of the situation's key, of its type, of the basic lexeme and of roles of its actants. For instance, the situation, which is the meaning of the verb lexeme 'PUT_1', has profile

$$\mathbf{sit}(\mathbf{put_1}^{(\mathbf{c}^{(\omega)}\mathbf{n_a}\mathbf{nc} \rightarrow \mathbf{s})}, \text{'}PUT_1\text{'}(\mathbf{AGT}(1), \mathbf{PAT}(2), \mathbf{LOC}(3)))$$

identified by the key $\mathbf{put_1}$, stating that this situation has three obligatory actants: the first actant of type $\mathbf{n_a}$ has the role **AGT**, the second of type $\mathbf{n}$ has the role **PAT**, the third of type $\mathbf{c}$ has the role **LOC**. Besides them, it has the standard iterative circumscriptor-type argument. This situation has value type $\mathbf{s}$.

In order to be used in complex meanings, the dictionary situations must be abstracted, i.e. transformed into Lambda terms of the corresponding types. For instance, the profile of $\mathbf{put_1}$ induces the Lambda term[2]:

$$\lambda Y^{\mathbf{c}^{(\omega)}} X_1^{\mathbf{n_a}} X_2^{\mathbf{n}} X_3^{\mathbf{c}}.\ \mathbf{put}_1(Y, X_1, X_2, X_3).$$

5.3 Discourse Plans

Building complex meanings from abstracted primitive situations meets with the following three obstacles. First is that the type of the actant of the host situation which should be developed using a new situation may conflict with the value type of the latter. For instance, in order to

[2]We abbreviate $\lambda X_1.\ (...(\lambda X_n.\ X_n.\ t)...)$ by $\lambda X_1...X_n.\ t$.

use a sententiator-type meaning as a qualifier-type argument of a nominator (cf. relative clauses modifying names), the type transformation ($\mathbf{s} \Rightarrow \mathbf{q}$) should be effected somehow. The second problem is more complex and is often due to the intended communicative organization of the complex meaning. It arises when the new situation should be a semantic derivative of some "clue" situation, as it is the case of the voices of verbs. In order to convert the original (i.e. available in the dictionary) situation into its derivative necessary to express a specific communicative structure, some semantic analogues are needed of the voice diatheses (Mel'čuk and Holodovich 1970). The third problem has been the subject of long debate in the literature (see (van Benthem and ter Meulen 1997)). This is the problem of using references to a meaning unit outside of the scope of its definition. For instance, in the example of Geach:

If a farmer has a donkey, he beats it

the anaphorical pronouns corresponding to the actants of situation *beat* reference the corresponding actants of situation *have*, whereas their scopes are disjoint. More complex are implicit tense relations, as in the following Kamp's sentence:

A child was born that will become ruler of the world.

So on the one hand, there should be a structure specifying the intended composition of primitive meanings, and on the other hand, there must be some means of realization of the meaning specifications by complex meanings. The specifications, we call them *discourse plans (DPs)*, are defined by the grammar in Fig. 1.

A DP can be seen as a sequence of *plan points* and also as a hierarchy of *sub-plans*. These two orders induce the corresponding mixed order of sub-plans. Each plan point introduces into DP a new primitive, or a new aggregate, or a situation's derivative defined by a situation converter. They all can be referenced. Long term references are kept in global context Γ, which is never updated. Short-term references added at some points of DP to local context ρ can be used and deleted from ρ later in DP. The scope and visibility of local references in the DP is defined in terms of the mixed order on sub-plans.

Primitives are of four kinds:

(p_1) Empty lists of qualifiers and circumscriptors $\mathbf{nil}^{\mathbf{q}^{(\omega)}}$, $\mathbf{nil}^{\mathbf{c}^{(\omega)}}$ and null values of option types: $\emptyset^{\mathbf{u}^{(\mathbf{0})}}$ (both omitted in examples).

(p_2) Actant-less lexical units, for instance, common nouns such as 'CAT'$^{(\mathbf{q}^{(\omega)} \to \mathbf{n_a})}$.

(p_3) Embedded relations, e.g. tense relations.

(p_4) Context access operators:

DP	::=	$\mathbf{SP}^+$	(discourse plan)
SP	::=	**Prim** |	(sub-plan)
		$\mathbf{A_{op}}\{\mathbf{AComps}\}$ |	(aggregate)
		Key Conv Mode {**DOs**}	(situation)
DOs	::=	**DO** | **DO**, **DOs**	(development operators)
DO	::=	**Arg** ⇐ **SP**	(development operator)
AComps	::=	**SP**, **SP** | **SP**, **AComps**	(aggregate components)
Prim			(primitive)
Conv			(converter)
Mode			(intonation marker)
Key			(situation identifier)
Arg			(situation argument)
op			(aggregation operator)

FIGURE 1 DP syntax

$(\Uparrow x)^{\mathbf{u}}$ co-referential with an element of type **u** in the local context;
$(\Downarrow_x^{\mathbf{u}}\ LEX)^{\mathbf{u}}$ creating and adding to the local context a new reference x of type **u** to the lexical unit LEX;
$(\Uparrow^{\mathbf{g}}\ N)^{\mathbf{u}}$ accessing the global context element of type **u** identified by N;
$(\Uparrow_x^{\mathbf{g}}\ N)^{\mathbf{u}}$ creating and adding to the local context a reference x to the global context element identified by N (x has the type of N).

Converters are fundamental means of planning. A converter **conv** used in a development operator **Arg** ⇐ **sub−plan**, in which

$$\mathbf{sub\!-\!plan} = \mathbf{key\ \ conv\ \ mode}\ \{\mathbf{sub\!-\!plans}\},$$

determines a semantic derivative **der**(**key**, **conv**) of situation **key** to be composed in (i.e. to unfold) the argument *Arg* of the host situation. So it is a second order operator applied to situation's meaning. For any two types $\mathbf{u}, \mathbf{v} \in \mathbf{T}$, $(\mathbf{u} \Rightarrow \mathbf{v})$ is a *converter type*. Below, we outline three kinds of converters: *TR-Converters, abstractors* and *direct diatheses*.

TR-converters relate to situations and to primitives their inherent attributes determined by roles and constrained by types. Intuitively, the role **R** of a **uR**-converter applied to a DP element E determines an inherent attribute of the meaning of E. The type **u** of this converter is one of the possible types of values of this attribute. For instance, each situation with **s**-type value has a semantic tense attribute **TNS**, and each nominator expressing a physical body has a location attribute **LOC**. Respectively, the tense TR-converter has the form $(\ )^{\mathbf{u}\ \mathbf{TNS}}$, where $\mathbf{u}$ is a tense circumscriptor type (e.g. $\mathbf{t_{ntr}} \prec \mathbf{t}$: neutral (gnomical) tense, $\mathbf{t_{pnt}} \prec \mathbf{t}$: pointwise tense, $\mathbf{t_{int}} \prec \mathbf{t}$: interval tense), and the location TR-converter has the form $(\ )^{\mathbf{v}\ \mathbf{LOC}}$, where $\mathbf{v}$ is a location circumscriptor type. For example, let ρ contain a reference s_2 to the situation:

$$\mathbf{sit}(\mathbf{be_3}^{(\mathbf{c}^{(\omega)}\mathbf{nn} \rightarrow \mathbf{s})}, \text{`}BE_3\text{'}(\mathbf{EL}(1), \mathbf{CLS}(2))).$$

Then in the DP fragment:

$[t]$ $\quad \mathbf{t_{prg}} \Leftarrow \Downarrow_{t_1}^{\mathbf{t_{prg}}} (\Uparrow s_2)^{\mathbf{t_{prg}}\ \mathbf{TNS}},$

$[t+1]$ $\quad \mathbf{t} \Leftarrow ((\Uparrow^{\mathbf{g}}\ \mathbf{now}) \in t_1)^{\mathbf{t}}, \quad \%\ (\Uparrow^{\mathbf{g}}\ \mathbf{now})$: the moment of speech

TR-converter $(\Uparrow s_2)^{\mathbf{t_{prg}}\ \mathbf{TNS}}$ defines the tense attribute of situation $\mathbf{be_3}$ with value type $\mathbf{t_{prg}}$. At point $[t]$, the operator $\Downarrow_{t_1}^{\mathbf{t_{prg}}} (\Uparrow s_2)^{\mathbf{t_{prg}}\ \mathbf{TNS}}$, adds to ρ the reference t_1 to this attribute. Besides this, at this point, $(\Uparrow s_2)^{\mathbf{t_{prg}}\ \mathbf{TNS}}$ becomes a circumscriptor of $\mathbf{be_3}$ of type $\mathbf{t_{prg}}$. At the next point, the reference t_1 is used to introduce a new circumscriptor of $\mathbf{be_3}$, which states the semantic present tense.

Abstractor ($\mathbf{abs}^x$) has the approximate meaning "*such x that*". Abstractors are used at the plan points of the form:

$$\mathbf{q} \Leftarrow \mathbf{key_1}^{(\bar{\mathbf{u}} \rightarrow \mathbf{v})}\ \mathbf{abs}^x\ \{\mathbf{sub{-}plans}\},$$

where the situation $\mathbf{key}_1$ develops a qualifier type argument of its host situation $\mathbf{key}_2$. So $\mathbf{abs}^x$ describes type conversions ($\mathbf{v} \ \Rightarrow\ \mathbf{q}$).

Direct diathesis ($\mathbf{dth}$) determines how the type and the argument structure of the introduced situation should be adapted to the developed argument type and role and to the new positions and roles of its actants. For instance, applied to situation:

$\mathbf{sit}(\mathbf{bear_2}^{(\mathbf{c}^{(\omega)}\mathbf{n_a n_a} \rightarrow \mathbf{s})}, \text{`}GIVE\_BIRTH\text{'}(\mathbf{AGT}(1)^{\mathbf{n_a}}, \mathbf{PAT}(2)^{\mathbf{n_a}})),$

the diathesis $\mathbf{dth_1}(\mathbf{bear_2}) = (\mathbf{PAT}(2)^{\mathbf{n_a}}, \mathbf{AGT}(1)^{\mathbf{n_a}^{(\mathbf{0})}})^{\mathbf{s}\ \mathbf{EVT}}$ determines the semantic derivative: $\mathbf{der}(\mathbf{bear_2}, \mathbf{dth_1}) =$

$\mathbf{sit}(\mathbf{be\_born}^{(\mathbf{c}^{(\omega)}\mathbf{n_a n_a^{(0)}} \rightarrow \mathbf{s})}, \text{`}GIVE\_BIRTH\text{'}(\mathbf{PAT}(2), \mathbf{AGT}(1))).$

Applied to situation:

$\mathbf{sit}(\mathbf{rule_1}^{(\mathbf{c}^{(\omega)}\mathbf{n_a n} \rightarrow \mathbf{s})}, \text{`}RULE_1\text{'}(\mathbf{AGT}(1)^{\mathbf{n_a}}, \mathbf{PAT}(2)^{\mathbf{n}})),$

the diathesis $\mathbf{dth_2}(\mathbf{rule_1}) = (\mathbf{PAT}(2)^{\mathbf{n}^{(\mathbf{0})}})^{\mathbf{n_a}\ \mathbf{AGT}}$ determines:

$\mathbf{der}(\mathbf{dth_2}, \mathbf{rule_1}) = \mathbf{sit}(\mathbf{ruler\_of}^{(\mathbf{q}^{(\omega)}\mathbf{n} \rightarrow \mathbf{n_a})}, \text{`}RULE_1\text{'}(\mathbf{PAT}(2))).$

In general, a direct diathesis

$$\mathbf{dth}(\mathbf{key}) = (\mathbf{R'_1}(j_1)^{\mathbf{u}'_1}, ..., \mathbf{R'_k}(j_k)^{\mathbf{u}'_k})^{\mathbf{v}'\ \mathbf{R}}$$

applied to a situation $\mathbf{key}$ with profile

$$\mathbf{sit}(\mathbf{key}^{(\mathbf{u}^{(\omega)}\mathbf{u}_1 \ldots \mathbf{u}_n \rightarrow \mathbf{v})}, \text{`}LEXEME\text{'}(\mathbf{R_1}(1), ..., \mathbf{R_n}(n)))$$

determines a derivative with profile

$$\mathbf{sit}(\mathbf{key'}^{(\mathbf{u'}^{(\omega)}\mathbf{u}'_1...\mathbf{u}'_k \to \mathbf{v'})}, \text{‘}LEXEME\text{’}(\mathbf{R'_1}(j_1), ..., \mathbf{R'_k}(j_k))).$$

This derivative is introduced in sub-plan of type $\mathbf{v'}$ for role $\mathbf{R}$ by development operators:

$[t_1]$ $\quad \mathbf{Arg} \Leftarrow \mathbf{key}^{(\mathbf{u}^{(\omega)}\mathbf{u}_1...\mathbf{u}_n \to \mathbf{v})}$ **dth** $\{$

...

$[t_2]$ $\quad (j_1)^{\mathbf{u}'_1} \Leftarrow \{sub{-}plan_1\}^{\mathbf{u'_1 R'_1}},$

...

$[t_3]$ $\quad (j_k)^{\mathbf{u}'_k} \Leftarrow \{sub{-}plan_k\}^{\mathbf{u'_k R'_k}} \quad \}$

where each $\{sub{-}plan_i\}^{\mathbf{u'_i}\ \mathbf{R'_i}}$ is a sub-plan of type $\mathbf{u'_i}$ for role $\mathbf{R'_i}$.

Co-reference and scope. ρ is a bounded resource memory. Adding to ρ a local reference may cause deletion of some other references. Reference deletion may occur not only for the reason of bounded memory size, but also because this memory cannot keep at the same plan point two “similar” “independent” references. We propose the following notion of reference similarity. When a local reference x is created in expression $\Downarrow_x^{\mathbf{u}}$ **key Conv** $\{Args\}$, it is assigned the *format* $fm(x) = (\mathbf{u}, \mathbf{R})$, where $\mathbf{u}$ is the type of x and $\mathbf{R}$ is the role assigned to the sub-plan **key Conv** $\{Args\}$ by the development operator in which it is used. If the role is not assigned, this will be the most general *undefined role* $\Re$. A reference x is *visible* at point $[i]$ of DP if $x \in \rho$ at this point. Visibility is determined by the following rules ($\leq_c$ is the infix order of sub-plans):

$(\mathbf{v_1})$ There is a constant κ limiting the number of visible references in ρ Adding to ρ $(\kappa + 1)$-th reference causes **deletion** from ρ of the earliest[3] reference.

$(\mathbf{v_2})$ Expressions

$$\Downarrow_x^{\mathbf{u}} \mathbf{key\ Conv}\ \{\mathbf{Args}\} \text{ and } (\Uparrow_x^{\mathbf{g}} \mathbf{key}^{\mathbf{u}})$$

add the reference x [4] to ρ and at the same time **remove** from ρ every similar local reference y, on which x does not c-depend: $y \not\leq_c x$.

$(\mathbf{v_3})$ Local context access/update operators are executed in the order of the plan points where they are used.

$(\mathbf{v_4})$ In operators $\mathbf{q} \Leftarrow \mathbf{SP}$, $\quad \mathbf{c} \Leftarrow \mathbf{SP}$ developing in DP π a qualifier/circumscriptor argument of a situation (of a primitive) S, every reference $\Uparrow x$ used in $\mathbf{SP}$ is c-dependent only on those elements in π on which S c-depends (in particular, on S itself).

$(\mathbf{v_5})$ A reference to a situation argument is removed together with the references to all this argument's qualifiers/circumscriptors.

[3] I.e. with the least DP point number.

[4] Never used at the preceding plan points.

Communicative structure. Semantic diatheses are related with communicative structure. Namely, to become the *theme* of a situation S, its actant should be promoted to the first actant position in a derivative of S. Somewhat simplifying, the *rheme* of S will be the set of all other *locally referenced arguments* in the derivative. Intuitively, the rheme consists of those arguments related with the theme through S, which can be passed as parameters to situations introduced later in the DP. Another communicative structure component, which can also entail a diathesis, is the (main) focus. In this paper, somewhat simplifying the real facts, we mark by the focus $\odot$ a unique DP point. The DP element introduced at the focalized point must be locally referenced.

Let us plan the meaning of the Kamp's sentence using the two diatheses above.

Discourse plan π:

[1] $\Downarrow^{\mathbf{s}}_{s_1}\ \mathbf{bear_2}^{(\mathbf{c}^{(\omega)}\mathbf{n_a n_a} \to \mathbf{s})}\ \mathbf{dth_1}\ \{$

[2] $\quad \mathbf{t_{pnt}} \Leftarrow \Downarrow^{\mathbf{t_{pnt}}}_{t_1}\ (\Uparrow s_1)^{\mathbf{t_{pnt}}}\ \mathbf{TNS},$

[3] $\quad \mathbf{t} \Leftarrow (\Uparrow t_1 \triangleleft (\Uparrow^{\mathbf{g}}\ \mathbf{now}))^{\mathbf{t}},$ % before now

[4]$\odot$ $\quad (2)^{\mathbf{n_a}} \Leftarrow \Downarrow^{\mathbf{n_a}}_{n_1}\ \text{'}CHILD\text{'}^{(\mathbf{q}^{(\omega)} \to \mathbf{n_a})}\{$

[5] $\quad\quad \mathbf{q} \Leftarrow \Downarrow^{\mathbf{q}}_{q_1} \Downarrow^{\mathbf{s}}_{s_2}\ \mathbf{be_3}^{(\mathbf{c}^{(\omega)}\mathbf{nn} \to \mathbf{s})}\ \mathbf{abs}^{n_1}\ \{$

[6] $\quad\quad\quad \mathbf{t_{pnt}} \Leftarrow \Downarrow_{t_2}(\Uparrow s_2)^{\mathbf{t_{pnt}}}\ \mathbf{TNS},$

[7] $\quad\quad\quad \mathbf{t} \Leftarrow (\Uparrow t_1 \triangleleft \Uparrow t_2)^{\mathbf{t}},$ % t_2 after t_1

[8] $\quad\quad\quad (1)^{\mathbf{n_a}} \Leftarrow (\Uparrow n_1)^{\mathbf{n_a}},$

[9] $\quad\quad\quad (2)^{\mathbf{n}} \Leftarrow \Downarrow^{\mathbf{n}}_{n_2}\mathbf{rule}^{(\mathbf{c}^{(\omega)}\mathbf{n_a n} \to \mathbf{s})}\ \mathbf{dth_2}\ \{$

[10] $\quad\quad\quad\quad (2)^{\mathbf{n}} \Leftarrow \text{'}WORLD_1\text{'}^{(\mathbf{q}^{(\omega)} \to \mathbf{n})}$

$\quad \}\ \ \}\ \ \}\ \ \}$

This DP is consistent in the following sense:

Definition 1 *A DP is* s(cope)-consistent *if all local references used in this plan are visible at the points, where they are used* [5]*.*

Moreover, π is *realizable*, in the sense that all the planned diatheses have the corresponding semantic derivatives in English (see above).

5.4 Meanings

Meanings are Lambda terms derived from realizable DPs by means of the following simple translation τ defined by induction on sub-plans S.

[5] There is a real-time algorithm checking this property.

Definition 2 ($\mathbf{m_1}$) *$\tau(S) = S$ if S is a primitive sub-plan.*
($\mathbf{m_2}$) $\tau(\odot S) = \odot\tau(S)$.
($\mathbf{m_3}$) $\tau((S_1, S_2)) = (\tau(S_1), \tau(S_2))$.
($\mathbf{m_4}$) $\tau(\mathbf{A_{op}}\{Comps\}) = \mathbf{A_{op}}\{\tau(Comps)\}$.
($\mathbf{m_5}$) *Let S be a sub-plan of the form:*

$$\begin{array}{lll} [t_0] & \mathbf{key\ Converter\ Mode}\ \{ & \\ [t_1] & \quad \mathbf{u}_1 \Leftarrow\ S_{0,1}, & \\ & ... & \\ [t_l] & \quad \mathbf{u}_l \Leftarrow\ S_{0,l}, & \\ [t_m] & \quad (j_1)^{\mathbf{v}_1} \Leftarrow\ S_1, & \\ & ... & \\ [t_k] & \quad (j_k)^{\mathbf{v}_k} \Leftarrow\ S_k\ \ \} & \end{array}$$

and let

$$\mathbf{der}(\mathbf{key}, \mathbf{Converter}) = \mathbf{key_{der}}^{(\mathbf{u}^{(\omega)}\mathbf{v}_1...\mathbf{v}_n\ \rightarrow\ \mathbf{v})},\ \mathbf{u}_m \preceq \mathbf{u}\ (1 \leq m \leq l).$$

be the derivative determined in the dictionary for **key** *by the* **Converter**. *Then, using the abstract situation*

$$absder(\mathbf{key}, \mathbf{Converter}) = \lambda X_0^{\mathbf{u}^{(\omega)}} X_1^{\mathbf{v}_1}...X_k^{\mathbf{v}_k}.\ \mathbf{key_{der}}(X_0, X_1, ..., X_k),$$

the translation of S is defined as :

$$\tau(S) = \mathbf{sit\ Mode}\ (absder(\mathbf{key}, \mathbf{Converter}) \\ [\tau(S_{k+1,1}), ..., \tau(S_{k+1,l})], \mathbf{act}_1(\tau(S_1)), ..., \mathbf{act}_k(\tau(S_k))).$$

The meaning derived from a realizable DP π *is the normal form term M such that $\tau(\pi) \twoheadrightarrow M$ (see below the point* **Reducibility***).*

Proposition 1 *1. If π is a realizable DP, then there exists a Lambda term $\tau(\pi)$. Moreover, for each sub-plan S of π of a primitive type* $\mathbf{u}$*, $\tau(S)$ also has type* $\mathbf{u}$.
2. If π is s-consistent, then the meaning derivable from π is also s-consistent.

The meaning of the sentence of Kamp derivable from π has the form:

(1) $\Downarrow_{s_1}^{\mathbf{s}}\ \mathbf{sit}(\mathbf{be\_born}^{(\mathbf{c}^{(\omega)}\mathbf{n_a}\mathbf{n_a^{(0)}}\ \rightarrow\ \mathbf{s})}$
(2) $[(\Downarrow_{t_1}^{\mathbf{t_{pnt}}}\ \Uparrow s_1)^{\mathbf{t_{pnt}}},$
(3) $(\Uparrow t_1 \triangleleft (\Uparrow^{\mathbf{g}}\ \mathbf{now}))^{\mathbf{t}}]^{\mathbf{c}^{(\omega)}},$
(4)$\odot$ $\mathbf{act}_2(\Downarrow_{n_1}^{\mathbf{n_a}}\ `CHILD'^{(\mathbf{q}^{(\omega)}\ \rightarrow\ \mathbf{n_a})}($
(5) $[\Downarrow_{q_1}^{\mathbf{q}}\Downarrow_{s_2}^{\mathbf{s}}\ \mathbf{sit}(\mathbf{der}(\mathbf{be_3}, \mathbf{abs}^{n_1})$
(6) $[(\Downarrow_{t_2}^{\mathbf{t_{pnt}}}\Uparrow s_2)^{\mathbf{t_{pnt}}},$

(7) $(\Uparrow t_1 \lhd \Uparrow t_2)^{\mathbf{t}}]^{\mathbf{c}^{(\omega)}}$,
(8) $\mathbf{act}_1((\Uparrow n_1)^{\mathbf{n_a}})$,
(9) $\mathbf{act}_2(\Downarrow^{\mathbf{n}}_{n_2}$ $\mathbf{sit}(\mathbf{ruler\_of}^{(\mathbf{q}^{(\omega)}\mathbf{n} \to \mathbf{n_a})}$
(10) $\mathbf{act}_2(\text{'}WORLD_1\text{'}^{(\mathbf{q}^{(\omega)} \to \mathbf{n})})$
$))\quad)]^{\mathbf{q}^{(\omega)}}\quad))\quad)$

Language of linguistic meanings $\mathcal{MT}^{\mathbf{lt}}$ (for space reasons, we don't include aggregates, modes, roles and foci).
Variables. We suppose that there are disjoint countable sets $\mathbf{V}^{\mathbf{u}}$ of *object variables* of type $\mathbf{u}$, $\mathbf{u} \in (\mathbf{B} \setminus \{\top\})$, and $\mathbf{R}^{\mathbf{u}}$ of *context variables* of type $\mathbf{u}$, for $\mathbf{u} \in \mathbf{T}$. We use upper-case latin indexed letters for the former and lower-case latin indexed letters for the latter.
Constants. For each type $\mathbf{u} \in \mathbf{T}$, $\mathbf{C}^{\mathbf{u}}$ is a countable set of *constants* of type $\mathbf{u}$. For each primitive type $\mathbf{u} \in (\mathbf{P} \setminus \{\bot, \top\})$, $\mathbf{C}^{\mathbf{u}^{(\mathbf{0})}} = \{\emptyset^{\mathbf{u}^{(\mathbf{0})}}\}$ ($\emptyset^{\mathbf{u}^{(\mathbf{0})}}$ is the *empty element* of type $\mathbf{u}^{(\mathbf{0})}$) and $\mathbf{C}^{\mathbf{u}^{(\omega)}} = \{\mathbf{nil}^{\mathbf{u}^{(\omega)}}\}$ ($\mathbf{nil}^{\mathbf{u}^{(\omega)}}$ is the *empty list* of type $\mathbf{u}^{(\omega)}$).
Converters. For each converter type $\mathbf{v} = (\mathbf{u}_1 \;\Rightarrow\; \mathbf{u}_2), \mathbf{u}_1, \mathbf{u}_2 \in \mathbf{T}$, $\mathbf{\Omega}^{\mathbf{v}}$ is a countable set of *converter constants* of converter type $\mathbf{v}$.
Context operators. $\mathbf{\Phi}$ is a countable set of *context operator names.*
Terms. The set $\mathcal{MT}^{\mathbf{lt}} =_{df} \bigcup\limits_{\mathbf{u} \in \mathbf{T}} \mathcal{T}^{\mathbf{u}}$ of typed terms is the least set verifying the following conditions:
($\mathbf{t_0}$) If $t = X \in \mathbf{V}^{\mathbf{u}}$, then $t \in \mathcal{T}^{\mathbf{u}}$ and $FV(t) = \{X\}$.
($\mathbf{t_1}$) If $\mathbf{k} \in \mathbf{C}^{\mathbf{u}}$, then $\mathbf{k} \in \mathcal{T}^{\mathbf{u}}$ and $FV(\mathbf{k}) = \emptyset$.
($\mathbf{t_2}$) If $\phi \in \mathbf{\Phi}$ and $x \in \mathbf{R}^{\mathbf{u}}$, then $t = \phi_x \in \mathcal{T}^{\mathbf{u}}$ and $FV(t) = \emptyset$.
($\mathbf{t_3}$) If $\mathbf{u}^{(0)} \in \mathbf{O}$, then $\mathcal{T}^{\mathbf{u}} \cup \{\emptyset^{\mathbf{u}^{(\mathbf{0})}}\} \subseteq \mathcal{T}^{\mathbf{u}^{(\mathbf{0})}}$ and $FV(\emptyset^{\mathbf{u}^{(\mathbf{0})}}) = \emptyset$.
($\mathbf{t_4}$) If $\mathbf{u} \in \mathbf{P}$ and $\mathbf{u}_1, ..., \mathbf{u}_k \preceq \mathbf{u}$, $k > 0$, then:
 (*i*) $\mathbf{nil}^{\mathbf{u}^{(\omega)}} \in \mathcal{T}^{\mathbf{u}^{(\omega)}}$, $FV(\mathbf{nil}^{\mathbf{u}^{(\omega)}}) = \emptyset$, and
 (*ii*) $t = [t_1, ..., t_k] \in \mathcal{T}^{\mathbf{u}^{(\omega)}}$ and $FV(t) = \bigcup\limits_{i=1}^{k} FV(t_i)$ for any $t_1 \in \mathcal{T}^{\mathbf{u}_1}, ..., t_k \in \mathcal{T}^{\mathbf{u}_k}$, such that $FV(t_i) \cap FV(t_j) = \emptyset$, $1 \leq i \neq j \leq k$.
($\mathbf{t_5}$) If $\gamma \in \mathbf{\Omega}^{(\mathbf{u} \Rightarrow \mathbf{v})}$ and $t \in \mathcal{T}^{\mathbf{u}}$, then $t_1 = \gamma\{t\} \in \mathcal{T}^{\mathbf{v}}$ and $FV(t_1) = FV(t)$.
($\mathbf{t_6}$) If $t_0 \in \mathcal{T}^{(\mathbf{u} \to \mathbf{v})}$, $t_1 \in \mathcal{T}^{\mathbf{u}_1}$ for some $\mathbf{u}_1 \preceq \mathbf{u}$, and $FV(t_0) \cap FV(t_1) = \emptyset$, then $t = (t_0\ t_1) \in \mathcal{T}^{\mathbf{v}}$ and $FV(t) = FV(t_0) \cup FV(t_1)$.
($\mathbf{t_7}$) If $t_0 \in \mathcal{T}^{\mathbf{v}}$, $(\mathbf{u} \to \mathbf{v}) \in \mathbf{T} \setminus \mathbf{B}$, and $X \in \mathbf{V}^{\mathbf{u}} \cap FV(t_0)$, then
$$t = \lambda X.\ t_0 \in \mathcal{T}^{(\mathbf{u} \to \mathbf{v})} \text{ and } FV(t) = FV(t_0) \setminus \{X\}. \ \square$$
Let $t_0 \in \mathcal{T}^{\mathbf{u}}$ be a subterm of a term $t = C[t_0] \in \mathcal{T}^{\mathbf{v}}$ identified by the *typed context* $C[\ ]^{\mathbf{u}}$. Given some other term $t_1 \in \mathcal{T}^{\mathbf{u}}$, if $C[t_1]$, i.e. the result of replacement of t_0 by t_1 in $C[\ ]^{\mathbf{u}}$, does not violate the convention of free variables in the definition of terms then, clearly, $C[t_1] \in \mathcal{T}^{\mathbf{v}}$. We say that a binary relation R on terms is *closed under typed contexts*, if
$$t_1\ R\ t_2 \text{ and } C[t_1] \in \mathcal{T}^{\mathbf{v}} \text{ implies } C[t_2] \in \mathcal{T}^{\mathbf{v}} \text{ and } C[t_1]\ R\ C[t_2].$$

Reducibility. The *immediate reducibility* of terms being the relation

$$(\lambda X^{\mathbf{u}}.\ t_1)\ t_2 \rightarrow_\beta t_1[t_2/X]$$

between terms $t = (\lambda X^{\mathbf{u}}.\ t_1)\ t_2$ and $t_0 = t_1[t_2/X]$, $t, t_0 \in \mathcal{T}^{\mathbf{v}}$, such that $\lambda X^{\mathbf{u}}.\ t_1 \in \mathcal{T}^{(\mathbf{u} \rightarrow \mathbf{v})}$, $t_2 \in \mathcal{T}^{\mathbf{u}_1}$ for some $\mathbf{u}_1 \preceq \mathbf{u}$, and $X \in \mathbf{V}^{\mathbf{u}} \cap FV(t_1)$, the *reducibility relation* $\twoheadrightarrow$ is the least preorder containing $\rightarrow_\beta$ and closed under typed contexts $C[\,]^{\mathbf{u}}$ and renaming of bound variables. A term t_0 is a *normal form* of a term t if $t \twoheadrightarrow t_0$ and t_0 is $\twoheadrightarrow$-minimal.

By the classical Church-Rosser theorem (see (Barendregt 1981)), the reducibility $\twoheadrightarrow$ is confluent and terminal. Then it is not difficult to prove that each term $t \in \mathcal{T}^{\mathbf{u}}$ has a unique normal form $t_0 : t \twoheadrightarrow t_0$ and $t_0 \in \mathcal{T}^{\mathbf{u}}$.

5.5 Discussion

The basic difference between the classical logical type system and that of the linguistic meanings is in the way the adnominal/adverbial modifiers are treated. The classical way (which can be traced back to Aristotle) is to interpret them as properties or, better to say, as conjunctive constraints to the intention of the modified unit: $|(mod\ U)| = |mod| \wedge |U|$. This is why, they obtain functional recursive types $(T/T), (T\backslash T)$. Through the Curry-Howard isomorphism, the meanings in the conventional system are isomorphic to derivations in formal systems, in which implication elimination corresponds to the function application and implication introduction corresponds to Lambda-abstraction. This leads directly to the Lambek calculus and its generalizations (see Buszkowski 1997). The iterative and non-recursive type system of linguistic meanings syntactically separates the linguistic semantics stricto sensu (argument structure, inheritance, semantic dependencies, co-reference, communicative structure) from the property / relation based factual semantics. At the same time, it can serve as a formal interface between the two semantics:

$$|(Nominator^{(\mathbf{q}^{(\omega)} \rightarrow \mathbf{n})}\ [m_1, \ldots, m_k]^{\mathbf{q}^{(\omega)}})|_{fact} = \mathbf{modify}(|Nominator|_{fact}, [|m_1|_{fact}, \ldots, |m_k|_{fact}]),$$

where:

$$\begin{array}{lcl} \mathbf{modify}(N, \mathbf{nil}) & = & N \\ \mathbf{modify}(N, [M|R]) & = & M \ \wedge\ \mathbf{modify}(N, R). \end{array}$$

On the other hand, the linguistic meaning types are directly translated into the corresponding semantic dependencies. This translation can be described in the form of the archetype dependency grammar in Fig. 2, in which the underlined nonterminals correspond to the heads. The dependencies defined by this grammar have the same orientation as the corresponding surface syntax dependencies. Due to this direct translation, the linguistic meaning can be naturally linked with the surface

s	$\rightarrow$	$\underline{\textbf{verb}} \oplus \{\textbf{arg} \oplus\}^i\ \textbf{c} \quad (1 \leq i \leq 5)$
arg	$\rightarrow$	**n | q | c**
n	$\rightarrow$	$\textbf{q} \oplus \underline{\textbf{n}}$ | **name**
q	$\rightarrow$	$\textbf{c} \oplus \underline{\textbf{q}}$ | **adj**
c	$\rightarrow$	$\textbf{c} \oplus \underline{\textbf{c}}$ | **adv**

FIGURE 2 Archetype grammar ($\oplus$ stands for the standard word order, which is a language dependent parameter).

dependency syntax. Such direct translation is impossible for classical logical types because the dependencies corresponding to them are oriented from modifiers to modified units [6].

One more aspect relating the linguistic meaning with some existing theories of linguistic semantics is its *dynamicity* (cf. Muskens et al. (1997)). There is however a great difference between the way the planned linguistic meaning transforms local contexts and the way the grammatical sentences of DRT Kamp (1981) transform boxes. The former only establishes anaphorical bindings and communicative structure relations between meaning units in disjoint sub-plans, whereas the latter construct and updates models. So the DRT-like dynamic systems are destined for knowledge acquisition through linguistic form. As such, they can, for instance, be a means of the global context control, which we don't take into account. Another difference between the two models is in the nature of the dynamism. In the planned linguistic meaning model, the local context control is resource sensitive, whereas in the factual dynamic systems there are no dynamic memory size limits.

We think that it is more appropriate from the linguistic point of view to define logical semantics (be it type logical, or game-theoretic, or DRT, etc.) on top of the linguistic meanings than to do it directly from sentences or from their syntactic structures.

APPENDIX: Set Theoretic Semantics of $\mathcal{MT}^{\mathbf{lt}}$

Domains. For each primitive type $\mathbf{u} \in \mathbf{P}$, let a domain $D^{\mathbf{u}}$ be chosen so that the following conditions were met:

($\mathbf{d_1}$) $D^{\perp} = \{\varepsilon\}$ for some object ε.
($\mathbf{d_2}$) If $\mathbf{u} \prec \mathbf{v}$, then $D^{\mathbf{u}} \subset D^{\mathbf{v}}$.
($\mathbf{d_3}$) $D^{\top} = \bigcup\limits_{\mathbf{u} \prec \top} D^{\mathbf{u}}$.

For each option type $\mathbf{u}^{(\mathbf{0})}$, a *null element* $\epsilon^{\mathbf{u}^{(\mathbf{0})}}$ of type $\mathbf{u}^{(\mathbf{0})}$ is selected

[6]The natural dependencies can be simulated through proofs in multimodal extensions of Lambek calculus Moortgat 1997.

and for each iterative type $\mathbf{u}^{(\omega)}$, a special object $[\,]^{\mathbf{u}^{(\omega)}}$ is selected called *empty list of type* $\mathbf{u}^{(\omega)}$, and the domains of option and iterative types are defined by:

($\mathbf{d_4}$) If $\mathbf{u}^{(\mathbf{0})} \in \mathbf{O}$, then $D^{\mathbf{u}^{(\mathbf{0})}} = D^{\mathbf{u}} \cup \{\epsilon^{\mathbf{u}^{(\mathbf{0})}}\}$.

($\mathbf{d_5}$) If $\mathbf{u}^{(\omega)} \in \mathbf{I}$, then $D^{\mathbf{u}^{(\omega)}} = \mathbf{list}(D^{\mathbf{u}})$, where $\mathbf{list}(D^{\mathbf{u}})$ is the set of all finite lists of objects in $D^{\mathbf{u}}$ with $[\,]^{\mathbf{u}^{(\omega)}}$ being the empty list. Somewhat more precisely, $\mathbf{list}(D^{\mathbf{u}})$ is the least set L containing $[\,]^{\mathbf{u}^{(\omega)}}$ and containing the pair $[e|l]$ for any $e \in L$ and $l \in L$.

($\mathbf{d_6}$) For any $\mathbf{u}, \mathbf{v} \in \mathbf{B} \setminus \{\bot, \top\}$, if $\mathbf{u} \prec \mathbf{v}$, then $\epsilon^{\mathbf{u}^{(\mathbf{0})}} = \epsilon^{\mathbf{v}^{(\mathbf{0})}}$ and $[\,]^{\mathbf{u}^{(\omega)}} = [\,]^{\mathbf{v}^{(\omega)}}$.

($\mathbf{d_7}$) If $(\mathbf{u} \;\rightarrow\; \mathbf{v}) \in \mathbf{T} \setminus \mathbf{B}$ and domains $D^{\mathbf{u}}$, $D^{\mathbf{v}}$ are defined, then the domain $D^{(\mathbf{u} \,\rightarrow\, \mathbf{v})}$ is defined as the set $(D^{\mathbf{v}})^{(D^{\mathbf{u}})}$ of all total functions from $D^{\mathbf{u}}$ to $D^{\mathbf{v}}$.

Interpretations. An *interpretation* $\iota = (\sigma, \tau, \omega, \Xi)$ consisting of *variables assignment* σ, *constants assignment* τ, *converters assignment* ω, and *contexts assignment* Ξ, is defined as follows:

($\mathbf{i_1}$) σ is a total function from $\bigcup\limits_{\mathbf{u} \in (\mathbf{B} \setminus \{\top\})} \mathbf{V}^{\mathbf{u}}$ to $D^{\top}$ such that $\sigma(X) \in D^{\mathbf{u}}$ for $X \in \mathbf{V}^{\mathbf{u}}$.

($\mathbf{i_2}$) τ is a total function from constants to objects of corresponding types (i.e. $\tau(c) \in D^{\mathbf{v}}$ for $c \in \mathbf{C}^{\mathbf{v}}$).

($\mathbf{i_3}$) ω is a total function from converter constants to second order operators transforming $\mathbf{T}$-type functions to $\mathbf{T}$-type functions such that for each $\gamma \in \Omega^{(\mathbf{u} \,\Rightarrow\, \mathbf{v})}$, $\omega(\gamma) \in (D^{\mathbf{v}})^{(D^{\mathbf{u}})}$.

($\mathbf{i_4}$) Ξ is a total function from terms, context operators and typed context variables into objects in $D^{\top}$: $\Xi(t, \phi, x) \in D^{\mathbf{u}}$ for any term t, context operator $\phi \in \mathbf{\Phi}$ and context variable $x \in \mathbf{R}^{\mathbf{u}}$.

For an interpretation $\iota = (\sigma, \tau, \omega, \Xi)$, an object variable $X \in \mathbf{V}^{\mathbf{u}}$ and an object $d \in D^{\mathbf{u}}$, we denote by $\iota\{X := d\}$ the interpretation $(\sigma_1, \tau, \omega, \Xi)$, in which $\sigma_1(Y) = \sigma(Y)$ for all $Y \neq X$ and
$\sigma_1(X) = d$.

Term values. For an *interpretation* $\iota = (\sigma, \tau, \omega, \Xi)$ and a term $t_0 \in \mathcal{T}^{\mathbf{u}}$, the *value* of t_0 in interpretation ι is denoted $|t_0\,\|^{\mathbf{u},\iota}$ and defined by $|t_0\,\|^{\mathbf{u},\iota} =_{df} |t_0\,\|^{t_0,\mathbf{u},\iota}$, where the value $|t\,\|^{t_0,\mathbf{u},\iota}$ of sub-terms t of term t_0 is defined recursively as follows:

($\mathbf{s_0}$) $|t\,\|^{t_0,\mathbf{u},\iota} = \sigma(X)$ for $t = X \in \mathbf{V}^{\mathbf{u}}$.

($\mathbf{s_1}$) $|t\,\|^{t_0,\mathbf{u},\iota} = \tau(\mathbf{k})$ for $t = \mathbf{k} \in C^{\mathbf{u}}$.

($\mathbf{s_2}$) Let $t = \phi_x$ for some $\phi \in \mathbf{\Phi}$ and $x \in \mathbf{R}^{\mathbf{u}}$. Then $|t\,\|^{t_0,\mathbf{u},\iota} = \Xi(t_0, \phi, x)$.

($\mathbf{s_3}$) Let $\mathbf{u} = \mathbf{v}^{(\mathbf{0})} \in \mathbf{O}$. Then $|t\,\|^{t_0,\mathbf{u},\iota} = \varepsilon^{\mathbf{v}^{(\mathbf{0})}}$ if $t = \emptyset^{\mathbf{v}^{(\mathbf{0})}}$, and $|t\,\|^{t_0,\mathbf{u},\iota} = |t\,\|^{t_0,\mathbf{v},\iota}$ if $t \neq \emptyset^{\mathbf{v}^{(\mathbf{0})}}$.

($\mathbf{s_4}$) Let $\mathbf{u} = \mathbf{v}^{(\omega)}$ and $\mathbf{v}^{(\omega)} \in \mathbf{I}$. Then (*i*) $|t\|^{t_0,\mathbf{u},\iota} = [\,]^{\mathbf{v}^{(\omega)}}$, if $t = \mathbf{nil}^{\mathbf{v}^{(\omega)}}$,

(*ii*) $|t\|^{t_0,\mathbf{u},\iota} = [|t_1\|^{t_0,\mathbf{v}_1,\iota}, ..., |t_k\|^{t_0,\mathbf{v}_k,\iota}]$, if $t = [t_1, ..., t_k]$ and $t_i \in \mathcal{T}^{\mathbf{v}_i}$, for some $\mathbf{v}_i \prec \mathbf{v}$ and all $1 \le i \le k$.

(s5) $|t\|^{t_0,\mathbf{u},\iota} = (\omega(\gamma))(|t_1\|^{t_0,\overline{\mathbf{u}},\iota})$, if $t = \gamma\{t_1\}$.

(s6) Let $t = (t_1\ t_2)$, where $t_1 \in \mathcal{T}^{(\mathbf{u}_1 \to \mathbf{u})}$ and $t_2 \in \mathcal{T}^{\mathbf{v}_1}$ for some $\mathbf{v}_1 \preceq \mathbf{u}_1$. Then

$$|t\|^{t_0,\mathbf{u},\iota} = |t_1\|^{t_0,(\mathbf{u}_1 \to \mathbf{v}),\iota}(|t_2\|^{t_0,\mathbf{v}_1,\iota}).$$

(s7) Let $\mathbf{u} = (\mathbf{u}_1 \to \mathbf{v})$, $t = \lambda X.\ t_1 \in \mathcal{T}^{(\mathbf{u}_1 \to \mathbf{v})}$ for some $t_1 \in \mathcal{T}^{\mathbf{v}}$ and $X \in \mathbf{V}^{\mathbf{u}_1} \cap FV(t_1)$. Then $|t\|^{t_0,\mathbf{u},\iota} = f$, where f is the function defined by:

$$f(d) = |t_1\|^{t_0,\mathbf{u},\iota\{X:=d\}} \text{ for each } d \in D^{\mathbf{u}_1}.\ \square$$

It is not difficult to prove that this definition is correct.

Proposition 2 $|t\|^{\mathbf{u},\iota} \in D^{\mathbf{u}}$ *for any type* $\mathbf{u}$*, any term* $t \in \mathcal{T}^{\mathbf{u}}$ *and any interpretation* ι.

Besides this, term values are invariant with respect to reducibility.

Proposition 3 *If* $t_1 \twoheadrightarrow t_2$, *then* $|t_1\|^{\mathbf{u},\iota} = |t_2\|^{\mathbf{u},\iota}$ *for any type* $\mathbf{u}$ *and interpretation* ι.

References

Barendregt, H. P. 1981. *The Lambda Calculus. Its Syntax and Semantics*. Amsterdam: North Holland Publishing Co.

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

Buszkowski, W. 1997. Mathematical linguistics and proof theory. In van Benthem and ter Meulen (1997), chap. 12, pages 683–736.

Dikovsky, A. 2003. Discourse plan → meaning. In *Proceedings of the First International Conference on Meaning-Text Theory (MTT2003)*, pages 299–308. Paris, France.

Gamut, L. T. F., ed. 1991. *Intensional Logic and Logical Grammar. Vol. 2*. Chicago: University of Chicago Press.

Kamp, H. 1981. A theory of truth and semantic representation. In J. Groenendijk, T. Janssen, and M. Stokhoff, eds., *Formal Methods in the Study of Language*. Dordrecht: Foris.

Mel'čuk, I. 1997. Vers une linguistique ≪Sens–Texte≫ . Leçon inagurale au Collège de France .

Mel'čuk, I. and A. A. Holodovich. 1970. K teorii grammaticheskogo zaloga [Towards a theory of grammatical voice]. *Narody Afriki i Azii [Nations of Africa and Asia]* (4).

Moortgat, M. 1997. Categorial type logics. In van Benthem and ter Meulen (1997), chap. 2, pages 93–177.

Muskens, R., J. van Benthem, and A. Visser. 1997. Dynamics. In van Benthem and ter Meulen (1997), chap. 10, pages 587–648.

Sowa, J., ed. 1991. *Principles of Semantic Networks*. San Mateo: Morgan Kaufmann.

van Benthem, J. and A. ter Meulen, eds. 1997. *Handbook of Logic and Language*. Amsterdam: North-Holland Elsevier, MIT Press.