A Categorial Type Logic*

Glyn Morrill

Universitat Politècnica de Catalunya
morrill@lsi.upc.edu
http://www.lsi.upc.edu/~morrill

Abstract. In logical categorial grammar (Morrill 2011[23], Moot and Retoré 2012[11]) syntactic structures are categorial proofs and semantic structures are intuitionistic proofs, and the syntax-semantics interface comprises a homomorphism from syntactic proofs to semantic proofs. Thereby, logical categorial grammar embodies in a pure logical form the principles of compositionality, lexicalism, and parsing as deduction. Interest has focused on multimodal versions but the advent of the (dis)placement calculus of Morrill, Valentín and Fadda (2011[21]) suggests that the role of structural rules can be reduced, and this facilitates computational implementation. In this paper we specify a comprehensive formalism of (dis)placement logic for the parser/theorem prover CatLog integrating categorial logic connectives proposed to date and illustrate with a cover grammar of the Montague fragment.

1 Introduction

According to the principle of compositionality of Frege the meaning of an expression is a function of the meanings of its parts and their mode of composition. This is refined in Montague grammar where the syntax-semantics interface comprises a homomorphism from a syntactic algebra to a semantic algebra. In logical categorial grammar (Morrill 2011[23], Moot and Retoré 2012[11]) both syntactic structures and semantic structures are proofs and the Montagovian rendering of Fregean compositionality is further refined to a homomorphism from syntactic (categorial) proofs to semantic (intuitionistic) proofs. Thus we see successive refinements of Frege's principle in theories of the syntax-semantics interface which are expressed first as algebra and then further as algebraic logic. The present paper gathers together and integrates categorial connectives proposed to date to specify a particular formalism according to this design, one implemented in the parser/theorem-prover CatLog (Morrill 2011[15], 2012[16]) and illustrates with a cover grammar of the Montague fragment.

Multimodal categorial grammar (Oehrle and Zhang 1989[25]; Moortgat and Morrill 1991[9]; Moortgat and Oehrle 1994[6]; Morrill 1994[22]; Moortgat 1995[7], 1997[8]; Oehrle 2011[24]) constitutes a methodology rather than a particular categorial calculus, admitting an open class of residuated connective families for multiple modes of composition related by structural rules of interaction and inclusion. On the one hand, since no

* This research was partially supported by an ICREA Acadèmia 2012, by BASMATI MICINN project (TIN2011-27479-C04-03) and by SGR2009-1428 (LARCA).

particular system is identified, the problem of computational implementation is an open-ended one; and on the other hand, the structural rules add to the proof search-space. Moot (1998[10]) and Moot and Retoré (2012[11], Ch. 7) provides a general-purpose implementation Grail. It supports the so-called Weak Sahlqvist structural inclusions and is based on proof-net contraction criteria, with certain contractions according to the structural rules. This seems to constitute the computational scope of the potential of the multimodal framework.

The displacement calculus of Morrill et al. (2011[21]) creates another option. This calculus provides a solution to the problem of discontinuous connectives in categorial grammar initiated in Bach (1981[1], 1984[2]). The calculus addresses a wide range of empirical phenomena, and it does so without the use of structural rules since the rules effecting displacement are *defined*. This opens the possibility of categorial calculus in which the role of structural rules is reduced. To accommodate discontinuity of resources the calculus invokes sorting of types according to their syntactical datatype (number of points of discontinuity), and this requires a novel kind of sequent calculus which we call a hypersequent calculus. In this paper we consider how displacement calculus and existing categorial logic can be integrated in a uniform hypersequent displacement logic, which we call simply placement logic.[1] We observe that this admits a relatively straightforward implementation which we use to illustrate a Montague fragment and we define as a program the goal of implementing increasing fragments of this logic with proof nets.

In the course of the present paper we shall specify the formalism and its calculus. This incorporates connectives introduced over many years addressing numerous linguistic phenomena, but the whole enterprise is characterized by the features of the placement calculus which is extended: sorting for the types and hypersequents for the calculus. In Section 2 we define the semantic representation language; in Section 3 we define the types; in Section 4 we define the calculus. In Section 5 we give a cover grammar of the Montague fragment of Dowty, Wall and Peters (1981[4], Ch. 7). In Section 6 we give analyses of the examples from the second half of that Chapter. We conclude in Section 7.

2 Semantic representation language

Recall the following operations on sets:

(1) a. Functional exponentiation: X^Y = the set of all total functions from Y to X
b. Cartesian product: $X \times Y = \{\langle x, y\rangle |\, x \in X \;\&\; y \in Y\}$
c. Disjoint union: $X \uplus Y = (\{1\} \times X) \cup (\{2\} \times Y)$
d. i-th Cross product, $i \geq 0$: $X^0 = \{0\}$
$X^{1+i} = X \times (X^i)$

The set $\mathcal{T}$ of *semantic types* of the semantic representation language is defined on the basis of a set δ of *basic semantic types* as follows:

[1] The prefix 'dis-' is dropped since reversing the line of reasoning which *dis*places items *places* items.

(2) $\mathcal{T} ::= \delta \mid \top \mid \ \bot \ \mid \mathcal{T}+\mathcal{T} \mid \mathcal{T}\&\mathcal{T} \mid \mathcal{T}\rightarrow\mathcal{T} \mid \mathbf{L}\mathcal{T} \mid \mathcal{T}^+$

A *semantic frame* comprises a family $\{D_\tau\}_{\tau\in\delta}$ of non-empty *basic type domains* and a non-empty set W of worlds. This induces a *type domain* D_τ for each type τ as follows:

(3)
$$\begin{aligned} D_\top &= \{\emptyset\} \\ D_\bot &= \{\} \\ D_{\tau_1+\tau_2} &= D_{\tau_2} \uplus D_{\tau_1} \\ D_{\tau_1\&\tau_2} &= D_{\tau_1} \times D_{\tau_2} \\ D_{\tau_1\rightarrow\tau_2} &= D_{\tau_2}^{D_{\tau_1}} \\ D_{\mathbf{L}\tau} &= D_\tau^W \\ D_{\tau^+} &= \bigcup_{i>0}(D_\tau)^i \end{aligned}$$

The sets Φ_τ of *terms* of type τ for each type τ are defined on the basis of sets C_τ of constants of type τ and enumerably infinite sets V_τ of variables of type τ for each type τ as follows:

(4)

$\Phi_\tau ::= C_\tau$	constants
$\Phi_\tau ::= V_\tau$	variables
$\Phi_\tau ::= \Phi_{\tau_1+\tau_2} \rightarrow V_{\tau_1}.\Phi_\tau;\ V_{\tau_2}.\Phi_\tau$	case statement
$\Phi_{\tau+\tau'} ::= \iota_1\Phi_\tau$	first injection
$\Phi_{\tau'+\tau} ::= \iota_2\Phi_\tau$	second injection
$\Phi_\tau ::= \pi_1\Phi_{\tau\&\tau'}$	first projection
$\Phi_\tau ::= \pi_2\Phi_{\tau'\&\tau}$	second projection
$\Phi_{\tau\&\tau'} ::= (\Phi_\tau, \Phi_{\tau'})$	ordered pair formation
$\Phi_\tau ::= (\Phi_{\tau'\rightarrow\tau}\ \Phi_{\tau'})$	functional application
$\Phi_{\tau\rightarrow\tau'} ::= \lambda V_\tau \Phi_{\tau'}$	functional abstraction
$\Phi_\tau ::= {}^\vee\Phi_{\mathbf{L}\tau}$	extensionalization
$\Phi_{\mathbf{L}\tau} ::= {}^\wedge\Phi_\tau$	intensionalization
$\Phi_{\tau^+} ::= [\Phi_\tau] \mid [\Phi_\tau \vert \Phi_{\tau^+}]$	non-empty list construction

Given a semantic frame, a *valuation* f mapping each constant of type τ into an element of D_τ, an assignment g mapping each variable of type τ into an element of D_τ, and a world $i \in W$, each term ϕ of type τ receives an interpretation $[\phi]^{g,i} \in D_\tau$ as shown in Figure 1.

An occurrence of a variable x in a term is called *free* if and only if it does not fall within any part of the term of the form $x.\cdot$ or $\lambda x\cdot$; otherwise it is *bound* (by the closest $x.$ or λx within the scope of which it falls). The result $\phi\{\psi_1/x_1,\ldots,\psi_n/x_n\}$ of substituting terms $\psi_1,\ldots,\psi_n$ (of types $\tau_1,\ldots,\tau_n$) for variables $x_1,\ldots,x_n$ (of types $\tau_1,\ldots,\tau_n$) respectively in a term ϕ is the result of simultaneously replacing by ψ_i every free occurrence of x_i in ϕ. We say that ψ is *free for x in ϕ* if and only if no variable in ψ becomes bound in $\phi\{\psi/x\}$. We say that a term is *modally closed* if and only if every occurrence of ${}^\vee$ occurs within the scope of an ${}^\wedge$. A modally closed term is denotationally invariant across worlds. We say that a term ψ is *modally free for x in ϕ* if and only if either ψ is modally closed, or no free occurrence of x in ϕ is within the scope of an ${}^\wedge$. The laws of conversion in Figure 2 obtain; we omit the so-called commuting conversions for the case statement.

$$
\begin{aligned}
[a]^{g,i} &= f(a) \text{ for constant } a \in C_\tau \\
[x]^{g,i} &= g(x) \text{ for variable } x \in V_\tau \\
[\phi \to x.\psi; y.\chi]^{g,i} &= \begin{cases} [\psi]^{(g-\{(x,g(x))\})\cup\{(x,\mathbf{snd}([\phi]^{g,i}))\},i} & \text{if } \mathbf{fst}([\phi]^{g,i}) = 1 \\ [\chi]^{(g-\{(y,g(y))\})\cup\{(y,\mathbf{snd}([\phi]^{g,i}))\},i} & \text{if } \mathbf{fst}([\phi]^{g,i}) = 2 \end{cases} \\
[\iota_1\phi]^{g,i} &= \langle 1, [\phi]^{g,i}\rangle \\
[\iota_2\phi]^{g,i} &= \langle 2, [\phi]^{g,i}\rangle \\
[\pi_1\phi]^{g,i} &= \mathbf{fst}([\phi]^{g,i}) \\
[\pi_2\phi]^{g,i} &= \mathbf{snd}([\phi]^{g,i}) \\
[(\phi,\psi)]^{g,i} &= \langle [\phi]^{g,i}, [\psi]^{g,i}\rangle \\
[(\phi\ \psi)]^{g,i} &= [\phi]^{g,i}([\psi]^{g,i}) \\
[\lambda x\phi]^{g,i} &= d \mapsto [\phi]^{(g-\{(x,g(x))\})\cup\{(x,d)\},i} \\
[{}^\vee\phi]^{g,i} &= [\phi]^{g,i}(i) \\
[{}^\wedge\phi]^{g,i} &= j \mapsto [\phi]^{g,j} \\
[[\phi]]^{g,i} &= \langle [\phi]^{g,i}, 0\rangle \\
[[\phi|\psi]]^{g,i} &= \langle [\phi]^{g,i}, [\psi]^{g,i}\rangle
\end{aligned}
$$

Fig. 1. Interpretation of the semantic representation language

$$\phi \to y.\psi; z.\chi = \phi \to x.(\psi\{x/y\}); z.\chi$$
if x is not free in ψ and is free for y in ψ
$$\phi \to y.\psi; z.\chi = \phi \to y.\psi; x.(\chi\{x/z\})$$
if x is not free in χ and is free for z in χ
$$\lambda y\phi = \lambda x(\phi\{x/y\})$$
if x is not free in ϕ and is free for y in ϕ

α-conversion

$$\iota_1\phi \to y.\psi; z.\chi = \psi\{\phi/y\}$$
if ϕ is free for y in ψ and modally free for y in ψ
$$\iota_2\phi \to y.\psi; z.\chi = \chi\{\phi/z\}$$
if ϕ is free for z in χ and modally free for z in χ
$$\pi_1(\phi,\psi) = \phi$$
$$\pi_2(\phi,\psi) = \psi$$
$$(\lambda x\phi\ \psi) = \phi\{\psi/x\}$$
if ψ is free for x in ϕ, and modally free for x in ϕ
$${}^{\vee\wedge}\phi = \phi$$

β-conversion

$$(\pi_1\phi, \pi_2\phi) = \phi$$
$$\lambda x(\phi\ x) = \phi$$
if x is not free in ϕ
$${}^{\wedge\vee}\phi = \phi$$
if ϕ is modally closed

η-conversion

Fig. 2. Semantic conversion laws

3 Syntactic types

The types in (dis)placement calculus and placement logic which extends it are sorted according to the number of points of discontinuity (placeholders) their expressions contain. Each *type predicate letter* will have a sort and an arity which are naturals, and a corresponding semantic type. Assuming ordinary terms to be already given, where P is a type predicate letter of sort i and arity n and $t_1, \ldots, t_n$ are terms, $Pt_1 \ldots t_n$ is an (atomic) type of sort i of the corresponding semantic type. Compound types are formed by connectives given in the following subsections, and the homomorphic semantic type map T associates these with semantic types. In Subsection 3.1 we give relevant details of the multiplicative (dis)placement calculus basis and in Subsection 3.2 we define types for all connectives.

3.1 The placement calculus connectives

Let a *vocabulary* V be a set which includes a distinguished placeholder symbol 1 called the *separator*. For $i \in \mathcal{N}$ we define L_i as the set of strings over V containing i separators:

(5) $L_i = \{s \in V^* | \ |s|_1 = i\}$

V induces the *placement algebra*

$$(\{L_i\}_{i \in \mathcal{N}}, +, \{\times_k\}_{k \in \mathcal{Z}^\pm}, 0, 1)$$

where $+ : L_i, L_j \to L_{i+j}$ is concatenation, and k-th wrapping $\times_k : L_{i+|k|}, L_j \to L_{i+|k|-1+j}$ is defined as replacing by its second operand the $|k|$-th separator in its first operand, counting from the left for positive k and from the right for negative k.[2] 0 is the empty string. Note that 0 is a left and right identity element for $+$ and that 1 is a left and right identity element for $\times$:

(6) $0{+}s = s \quad s = s{+}0$
$1{\times}s = s \quad s = s{\times}1$

Sorted types $\mathcal{F}_i, i \in \mathcal{N}$, are defined and interpreted sort-wise as shown in Figure 3. Where A is a type, let sA denotes its sort. The sorting discipline ensures that $[A] \subseteq L_{sA}$. Note that $\{\backslash, \bullet, /\}$ and $\{\downarrow_k, \odot_k, \uparrow_k\}$ are residuated triples with parents $\bullet$ and $\odot_k$, and that as the canonical extensions of the operations of the placement algebra, I is a left and right identity for $\bullet$ and J is a left and right identity for $\odot_k$.

[2] In the version of Morrill and Valentín (2010[18]) wrapping is only counted from the left, and in the "edge" version of Morrill et al. (2011[21]) there is only leftmost and rightmost wrapping, hence these can be seen as subinstances of the general case given in this paper where $k > 0$ and $k \in \{+1, -1\}$ respectively.

$\mathcal{F}_j ::= \mathcal{F}_i \backslash \mathcal{F}_{i+j}$	$[A\backslash C] = \{s_2 \mid \forall s_1 \in [A], s_1{+}s_2 \in [C]\}$	under
$\mathcal{F}_i ::= \mathcal{F}_{i+j} / \mathcal{F}_j$	$[C/B] = \{s_1 \mid \forall s_2 \in [B], s_1{+}s_2 \in [C]\}$	over
$\mathcal{F}_{i+j} ::= \mathcal{F}_i \bullet \mathcal{F}_j$	$[A \bullet B] = \{s_1{+}s_2 \mid s_1 \in [A] \ \& \ s_2 \in [B]\}$	product
$\mathcal{F}_0 ::= I$	$[I] = \{0\}$	product unit
$\mathcal{F}_j ::= \mathcal{F}_{i+1} \downarrow_k \mathcal{F}_{i+j}$	$[A \downarrow_k C] = \{s_2 \mid \forall s_1 \in [A], s_1 \times_k s_2 \in [C]\}$	infix
$\mathcal{F}_{i+1} ::= \mathcal{F}_{i+j} \uparrow_k \mathcal{F}_j$	$[C \uparrow_k B] = \{s_1 \mid \forall s_2 \in [B], s_1 \times_k s_2 \in [C]\}$	circumfix
$\mathcal{F}_{i+j} ::= \mathcal{F}_{i+1} \odot_k \mathcal{F}_j$	$[A \odot_k B] = \{s_1 \times_k s_2 \mid s_1 \in [A] \ \& \ s_2 \in [B]\}$	wrap
$\mathcal{F}_1 ::= J$	$[J] = \{1\}$	wrap unit

Fig. 3. Types of the placement calculus **D** and their interpretation

3.2 All connectives

We consider type-logical connectives in the context of the placement sorting discipline. The connectives in types may surface as main connectives in either the antecedent or the succedent of sequents and some connectives are restricted with respect to which of these may occur. Hence we define sorted types of each of two polarities: input ($^\bullet$) or antecedent and output ($^\circ$) or succedent; where p is a polarity, $\overline{p}$ is the opposite polarity. The types formed by primitive connectives together with the type map T are defined as shown in Figure 4. The structural modality and Kleene plus are limited to types of

$\mathcal{F}_j^p ::= \mathcal{F}_i^{\overline{p}} \backslash \mathcal{F}_{i+j}^p$	$T(A\backslash C) = T(A) \to T(C)$	
$\mathcal{F}_i^p ::= \mathcal{F}_{i+j}^p / \mathcal{F}_j^{\overline{p}}$	$T(C/B) = T(B) \to T(C)$	
$\mathcal{F}_{i+j}^p ::= \mathcal{F}_i^p \bullet \mathcal{F}_j^p$	$T(A \bullet B) = T(A) \& T(B)$	
$\mathcal{F}_0^p ::= I$	$T(I) = \top$	
$\mathcal{F}_j^p ::= \mathcal{F}_{i+1}^{\overline{p}} \downarrow_k \mathcal{F}_{i+j}^p$	$T(A \downarrow_k C) = T(A) \to T(C)$	
$\mathcal{F}_{i+1}^p ::= \mathcal{F}_{i+j}^p \uparrow_k \mathcal{F}_j^{\overline{p}}$	$T(C \uparrow_k B) = T(B) \to T(C)$	
$\mathcal{F}_{i+j}^p ::= \mathcal{F}_{i+1}^p \odot_k \mathcal{F}_j^p$	$T(A \odot_k B) = T(A) \& T(B)$	
$\mathcal{F}_1^p ::= J$	$T(J) = \top$	
$\mathcal{F}_i^p ::= \mathcal{F}_i^p \& \mathcal{F}_i^p$	$T(A \& B) = T(A) \& T(B)$	additive conjunction [5, 12]
$\mathcal{F}_i^p ::= \mathcal{F}_i^p \oplus \mathcal{F}_i^p$	$T(A \oplus B) = T(A) + T(B)$	additive disjunction [5, 12]
$\mathcal{F}_i^p ::= \mathcal{F}_i^p \sqcap \mathcal{F}_i^p$	$T(A \sqcap B) = T(A) = T(B)$	sem. inert additive conjunction [22]
$\mathcal{F}_i^p ::= \mathcal{F}_i^p \sqcup \mathcal{F}_i^p$	$T(A \sqcup B) = T(A) = T(B)$	sem. inert additive disjunction [22]
$\mathcal{F}_i^p ::= \Box \mathcal{F}_i^p$	$T(\Box A) = \mathbf{L} T(A)$	modality [13]
$\mathcal{F}_i^p ::= \blacksquare \mathcal{F}_i^p$	$T(\blacksquare A) = T(A)$	rigid designator modality
$\mathcal{F}_0^p ::= !\mathcal{F}_0^p$	$T(!A) = T(A)$	structural modality [3]
$\mathcal{F}_i^p ::= \langle\rangle \mathcal{F}_i^p$	$T(\langle\rangle A) = T(A)$	exist. bracket modality [14, 7]
$\mathcal{F}_i^p ::= []^{-1} \mathcal{F}_i^p$	$T([]^{-1} A) = T(A)$	univ. bracket modality [14, 7]
$\mathcal{F}_i^p ::= \forall X \mathcal{F}_i^p$	$T(\forall x A) = T(A)$	1st order univ. qu. [22]
$\mathcal{F}_i^p ::= \exists X \mathcal{F}_i$	$T(\exists x A) = T(A)$	1st order exist. qu. [22]
${\mathcal{F}_0}^\circ ::= {\mathcal{F}_0}^{\circ +}$	$T(A^+) = list(T(A))$	Kleene plus [22]
${\mathcal{F}_i}^\circ ::= \neg {\mathcal{F}_i}^\circ$	$T(\neg A) = \bot$	negation-as-failure [19]

Fig. 4. Primitive connectives

sort 0 because structural operations of contraction and expansion would not preserve other sorts. The Kleene plus and negation-as-failure are restricted to succedent polarity occurrences.

In addition to the primitive connectives we may define derived connectives which do not extend expressivity, but which permit abbreviations. Unary derived connectives are given in Figure 5. Continuous and discontinuous nondeterministic binary derived

$$
\begin{array}{rllll}
\triangleright^{-1}A =_{df} & J\backslash A & \{s|\ 1{+}s \in A\} & T(\triangleright^{-1}A) = T(A) & \text{right projection [20]} \\
\triangleleft^{-1}A =_{df} & A/J & \{s|s{+}1 \in A\} & T(\triangleleft^{-1}A) = T(A) & \text{left projection [20]} \\
\triangleright A =_{df} & J{\bullet}A & \{1{+}s|\ s \in A\} & T(\triangleright A) = T(A) & \text{right injection [20]} \\
\triangleleft A =_{df} & A{\bullet}J & \{s{+}1|\ s \in A\} & T(\triangleleft A) = T(A) & \text{left injection [20]} \\
\check{}^{k}A =_{df} & A{\uparrow_k}I & \{s|\ s{\times_k}0 \in A\} & T(\check{}^{k}A) = T(A) & \text{split [17]} \\
\hat{}^{k}A =_{df} & A{\odot_k}I & \{s{\times_k}0|\ s \in A\} & T(\hat{}^{k}A) = T(A) & \text{bridge [17]}
\end{array}
$$

Fig. 5. Unary derived connectives

connectives are given in Figure 6, where $+(s_1, s_2, s_3)$ if and only if $s_3 = s_1{+}s_2$ or $s_3 = s_2{+}s_1$, and $\times(s_1, s_2, s_3)$ if and only if $s_3 = s_1 \times_1 s_2$ or ... or $s_3 = s_1 \times_n s_2$ where s_1 is of sort n.

$$
\begin{array}{rlllr}
\frac{B}{A} & (A\backslash B) \sqcap (B/A) & \{s|\ \forall s' \in A, s_3, +(s, s', s_3) \Rightarrow s_3 \in B\} & T(\frac{B}{A}) = T(A) \rightarrow T(B) & \text{nondet. division} \\
A \otimes B & (A{\bullet}B) \sqcup (B{\bullet}A) & \{s_3|\ \exists s_1 \in A, s_2 \in B, +(s_1, s_2, s_3)\} & T(A \otimes B) = T(A)\&T(B) & \text{nondet. product} \\
A{\Downarrow}C & (A{\downarrow_1}C) \sqcap \cdots \sqcap (A{\downarrow_{\sigma A}}C) & \{s_2|\ \forall s_1 \in A, s_3, \times(s_1, s_2, s_3) \Rightarrow s_3 \in C\} & T(A{\Downarrow}C) = T(A) \rightarrow T(C) & \text{nondet. infix} \\
C{\Uparrow}B & (C{\uparrow_1}B) \sqcap \cdots \sqcap (C{\uparrow_{\sigma C}}B) & \{s_1|\ \forall s_2 \in B, s_3, \times(s_1, s_2, s_3) \Rightarrow s_3 \in C\} & T(C{\Uparrow}B) = T(B) \rightarrow T(C) & \text{nondet. circumfix} \\
A{\circledcirc}B & (A{\odot_1}B) \sqcup \cdots \sqcup (A{\odot_{\sigma A}}B) & \{s_3|\ \exists s_1 \in A, s_2 \in B, \times(s_1, s_2, s_3)\} & T(A{\circledcirc}B) = T(A)\&T(B) & \text{nondet. wrap}
\end{array}
$$

Fig. 6. Binary nondeterministic derived connectives

4 Calculus

The set $\mathcal{O}$ of *configurations* of hypersequent calculus for our categorial logic is defined as follows, where Λ is the empty string and * is the metalinguistic separator or *hole*:

(7) $\mathcal{O} ::= \Lambda \mid * \mid \mathcal{F}_0 \mid \mathcal{F}_{i+1}\{\underbrace{\mathcal{O} : \ldots : \mathcal{O}}_{i+1\ \mathcal{O}\text{'s}}\} \mid \mathcal{O}, \mathcal{O} \mid [\mathcal{O}]$

The sort of a configuration Γ is the number of holes it contains: $|\Gamma|_*$. Where Δ is a configuration of sort $k{+}i, k > 0$ and Γ is a configuration, $\Delta|_{+k}\Gamma$ ($\Delta|_{-k}\Gamma$) is the configuration resulting from replacing by Γ the k-th hole from the left (right) in Δ. The *figure* $\overrightarrow{A}$ of a type A is defined by:

(8) $\overrightarrow{A} = \begin{cases} A & \text{if } sA = 0 \\ A\{\underbrace{* : \ldots : *}_{sA\ *\text{'s}}\} & \text{if } sA > 0 \end{cases}$

The usual configuration distinguished occurrence notation $\Delta(\Gamma)$ signifies a configuration Δ with a distinguished subconfiguration Γ, i.e. a configuration occurrence Γ with (external) context Δ. In the hypersequent calculus the distinguished *hyperoccurrence* notation $\Delta\langle\Gamma\rangle$ signifies a configuration *hyper*occurrence Γ with external *and internal* context Δ as follows: where Γ is a configuration of sort i and $\Delta_1, \ldots, \Delta_i$ are configurations, the *fold* $\Gamma \otimes \langle\Delta_1, \ldots, \Delta_i\rangle$ is the result of replacing the successive holes in Γ by $\Delta_1, \ldots, \Delta_i$ respectively; the *distinguished hyperoccurrence* notation $\Delta\langle\Gamma\rangle$ represents $\Delta_0(\Gamma \otimes \langle\Delta_1, \ldots, \Delta_i\rangle)$.

A *sequent* $\Gamma \Rightarrow A$ comprises an antecedent configuration Γ of sort i and a succedent type A of sort i. The types which are allowed to enter into the antecedent are the input ($^\bullet$) types and the types which are allowed to enter into the succedent are the output ($^\circ$) types. The hypersequent calculus for the placement categorial logic defined in the previous section has the following identity axiom:

(9) $$\frac{}{\overrightarrow{A} \Rightarrow A}\, id$$

The logical rules for primitive multiplicatives, additives, exponentials,[3] modalities and quantifiers are given in Figures 7, 8, 9, 10 and 11 respectively.

[3] As given, the contraction rules, which are for parastic gaps (Morrill 2011[23], Ch. 5), can be applied only a finite number of times in backward-chaining proof search since they are conditioned on brackets. Alternatively, the contraction rules may be given the form:

$$\frac{\Delta\langle !A, [!A, \Gamma]\rangle \Rightarrow B}{\Delta\langle !A, \Gamma\rangle \Rightarrow B}\, !C \qquad \frac{\Delta\langle [\Gamma, !A], !A\rangle \Rightarrow B}{\Delta\langle \Gamma, !A, \rangle \Rightarrow B}\, !C$$

We think there would still be decidability if there were a bound on the number of brackets it would be appropriate to introduce applying the rules from conclusion to premise, but this needs to be examined in detail.

$$\frac{\Gamma \Rightarrow A \qquad \Delta\langle \overrightarrow{C} \rangle \Rightarrow D}{\Delta\langle \Gamma, \overrightarrow{A\backslash C} \rangle \Rightarrow D}\,\backslash L \qquad \frac{\overrightarrow{A}, \Gamma \Rightarrow C}{\Gamma \Rightarrow A\backslash C}\,\backslash R$$

$$\frac{\Gamma \Rightarrow B \qquad \Delta\langle \overrightarrow{C} \rangle \Rightarrow D}{\Delta\langle \overrightarrow{C/B}, \Gamma \rangle \Rightarrow D}\,/L \qquad \frac{\Gamma, \overrightarrow{B} \Rightarrow C}{\Gamma \Rightarrow C/B}\,/R$$

$$\frac{\Delta\langle \overrightarrow{A}, \overrightarrow{B} \rangle \Rightarrow D}{\Delta\langle \overrightarrow{A\bullet B} \rangle \Rightarrow D}\,\bullet L \qquad \frac{\Gamma_1 \Rightarrow A \qquad \Gamma_2 \Rightarrow B}{\Gamma_1, \Gamma_2 \Rightarrow A\bullet B}\,\bullet R$$

$$\frac{\Delta\langle \Lambda \rangle \Rightarrow A}{\Delta\langle \overrightarrow{I} \rangle \Rightarrow A}\,IL \qquad \frac{}{\Lambda \Rightarrow I}\,IR$$

$$\frac{\Gamma \Rightarrow A \qquad \Delta\langle \overrightarrow{C} \rangle \Rightarrow D}{\Delta\langle \Gamma|_k \overrightarrow{A{\downarrow_k}C} \rangle \Rightarrow D}\,{\downarrow_k}L \qquad \frac{\overrightarrow{A}|_k \Gamma \Rightarrow C}{\Gamma \Rightarrow A{\downarrow_k}C}\,{\downarrow_k}R$$

$$\frac{\Gamma \Rightarrow B \qquad \Delta\langle \overrightarrow{C} \rangle \Rightarrow D}{\Delta\langle \overrightarrow{C{\uparrow_k}B}|_k \Gamma \rangle \Rightarrow D}\,{\uparrow_k}L \qquad \frac{\Gamma|_k \overrightarrow{B} \Rightarrow C}{\Gamma \Rightarrow C{\uparrow_k}B}\,{\uparrow_k}R$$

$$\frac{\Delta\langle \overrightarrow{A}|_k \overrightarrow{B} \rangle \Rightarrow D}{\Delta\langle \overrightarrow{A\odot_k B} \rangle \Rightarrow D}\,\odot_k L \qquad \frac{\Gamma_1 \Rightarrow A \qquad \Gamma_2 \Rightarrow B}{\Gamma_1|_k \Gamma_2 \Rightarrow A\odot_k B}\,\odot_k R$$

$$\frac{\Delta\langle * \rangle \Rightarrow A}{\Delta\langle \overrightarrow{J} \rangle \Rightarrow A}\,JL \qquad \frac{}{* \Rightarrow J}\,JR$$

Fig. 7. Multiplicative rules

$$\frac{\Gamma\langle\overrightarrow{A}\rangle \Rightarrow C}{\Gamma\langle\overrightarrow{A\&B}\rangle \Rightarrow C}\,\&L_1 \qquad \frac{\Gamma\langle\overrightarrow{B}\rangle \Rightarrow C}{\Gamma\langle\overrightarrow{A\&B}\rangle \Rightarrow C}\,\&L_2$$

$$\frac{\Gamma \Rightarrow A \qquad \Gamma \Rightarrow B}{\Gamma \Rightarrow A\&B}\,\&R$$

$$\frac{\Gamma\langle\overrightarrow{A}\rangle \Rightarrow C \qquad \Gamma\langle\overrightarrow{B}\rangle \Rightarrow C}{\Gamma\langle\overrightarrow{A\oplus B}\rangle \Rightarrow C}\,\oplus L$$

$$\frac{\Gamma \Rightarrow A}{\Gamma \Rightarrow A\oplus B}\,\oplus L_1 \qquad \frac{\Gamma \Rightarrow B}{\Gamma \Rightarrow A\oplus B}\,\oplus L_2$$

$$\frac{\Gamma\langle\overrightarrow{A}\rangle \Rightarrow C}{\Gamma\langle\overrightarrow{A\sqcap B}\rangle \Rightarrow C}\,\sqcap L_1 \qquad \frac{\Gamma\langle\overrightarrow{B}\rangle \Rightarrow C}{\Gamma\langle\overrightarrow{A\sqcap B}\rangle \Rightarrow C}\,\sqcap L_2$$

$$\frac{\Gamma \Rightarrow A \qquad \Gamma \Rightarrow B}{\Gamma \Rightarrow A\sqcap B}\,\sqcap R$$

$$\frac{\Gamma\langle\overrightarrow{A}\rangle \Rightarrow C \qquad \Gamma\langle\overrightarrow{B}\rangle \Rightarrow C}{\Gamma\langle\overrightarrow{A\sqcup B}\rangle \Rightarrow C}\,\sqcup L$$

$$\frac{\Gamma \Rightarrow A}{\Gamma \Rightarrow A\sqcup B}\,\sqcup L_1 \qquad \frac{\Gamma \Rightarrow B}{\Gamma \Rightarrow A\sqcup B}\,\sqcup L_2$$

Fig. 8. Additive rules

$$\frac{\Gamma(A) \Rightarrow B}{\Gamma(!A) \Rightarrow B}\,!L \qquad \frac{!A_1, \ldots, !A_n \Rightarrow A}{!A_1, \ldots, !A_n \Rightarrow !A}\,!R$$

$$\frac{\Delta\langle !A, \Gamma\rangle \Rightarrow B}{\Delta\langle \Gamma, !A\rangle \Rightarrow B}\,!P \qquad \frac{\Delta\langle \Gamma, !A\rangle \Rightarrow B}{\Delta\langle !A, \Gamma\rangle \Rightarrow B}\,!P$$

$$\frac{\Delta\langle !A, [!A, \Gamma]\rangle \Rightarrow B}{\Delta\langle !A, [[\Gamma]]\rangle \Rightarrow B}\,!C \qquad \frac{\Delta\langle [\Gamma, !A], !A\rangle \Rightarrow B}{\Delta\langle [[\Gamma]], !A, \rangle \Rightarrow B}\,!C$$

$$\frac{\Gamma \Rightarrow A}{\Gamma \Rightarrow A^+}\,{}^+R \qquad \frac{\Gamma \Rightarrow A \qquad \Delta \Rightarrow A^+}{\Gamma, \Delta \Rightarrow A^+}\,{}^+R$$

Fig. 9. Exponential rules

$$\frac{\Gamma\langle\overrightarrow{A}\rangle \Rightarrow B}{\Gamma\langle\overrightarrow{\Box A}\rangle \Rightarrow B}\,\Box L \qquad \frac{\Box/\blacksquare\Gamma \Rightarrow A}{\Box/\blacksquare\Gamma \Rightarrow \Box A}\,\Box R$$

$$\frac{\Gamma\langle\overrightarrow{A}\rangle \Rightarrow B}{\Gamma\langle\overrightarrow{\blacksquare A}\rangle \Rightarrow B}\,\blacksquare L \qquad \frac{\Box/\blacksquare\Gamma \Rightarrow A}{\Box/\blacksquare\Gamma \Rightarrow \blacksquare A}\,\blacksquare R$$

$$\frac{\Delta\langle\overrightarrow{A}\rangle \Rightarrow B}{\Delta\langle\overrightarrow{[[\,]^{-1}A]}\rangle \Rightarrow B}\,[\,]^{-1}L \qquad \frac{[\Gamma] \Rightarrow A}{\Gamma \Rightarrow [\,]^{-1}A}\,[\,]^{-1}R$$

$$\frac{\Delta\langle[\overrightarrow{A}]\rangle \Rightarrow B}{\Delta\langle\overrightarrow{\langle\rangle A}\rangle \Rightarrow B}\,\langle\rangle L \qquad \frac{\Gamma \Rightarrow A}{[\Gamma] \Rightarrow \langle\rangle A}\,\langle\rangle R$$

Fig. 10. Normal (semantic) and bracket (syntactic) modality rules, where $\Box/\blacksquare\Gamma$ signifies a configuration all the types of which have main connective $\Box$ or $\blacksquare$

$$\frac{\Gamma\langle\overrightarrow{A[t/x]}\rangle \Rightarrow B}{\Gamma\langle\overrightarrow{\forall xA}\rangle \Rightarrow B}\,\forall L \qquad \frac{\Gamma \Rightarrow A[a/x]}{\Gamma \Rightarrow \forall xA}\,\forall R^{\dagger}$$

$$\frac{\Gamma\langle\overrightarrow{A[a/x]}\rangle \Rightarrow B}{\Gamma\langle\overrightarrow{\exists xA}\rangle \Rightarrow B}\,\exists L^{\dagger} \qquad \frac{\Gamma \Rightarrow A[t/x]}{\Gamma \Rightarrow \exists xA}\,\exists R$$

Fig. 11. Quantifier rules, where † indicates that there is no a in the conclusion

The rules for the unary and binary derived connectives are shown in Figures 12 and 13.

$$\frac{\Gamma\langle\overrightarrow{A}\rangle \Rightarrow B}{\Gamma\langle\overrightarrow{\triangleleft^{-1}A}, *\rangle \Rightarrow B}\triangleleft^{-1}L \qquad \frac{\Gamma, * \Rightarrow A}{\Gamma \Rightarrow \triangleleft^{-1}A}\triangleleft^{-1}R$$

$$\frac{\Gamma\langle\overrightarrow{A}, *\rangle \Rightarrow B}{\Gamma\langle\overrightarrow{\triangleleft A}\rangle \Rightarrow B}\triangleleft L \qquad \frac{\Gamma \Rightarrow A}{\Gamma, * \Rightarrow \triangleleft A}\triangleleft R$$

$$\frac{\Gamma\langle\overrightarrow{A}\rangle \Rightarrow B}{\Gamma\langle *, \overrightarrow{\triangleright^{-1}A}\rangle \Rightarrow B}\triangleright^{-1}L \qquad \frac{*, \Gamma \Rightarrow A}{\Gamma \Rightarrow \triangleright^{-1}A}\triangleright^{-1}R$$

$$\frac{\Gamma\langle *, \overrightarrow{A}\rangle \Rightarrow B}{\Gamma\langle\overrightarrow{\triangleright A}\rangle \Rightarrow B}\triangleright L \qquad \frac{\Gamma \Rightarrow A}{*, \Gamma \Rightarrow \triangleright A}\triangleright R$$

$$\frac{\Delta\langle\overrightarrow{B}\rangle \Rightarrow C}{\Delta\langle\overrightarrow{\check{}^{k}B}|_k\Lambda\rangle \Rightarrow C}\check{}^{k}L \qquad \frac{\Delta|_k\Lambda \Rightarrow B}{\Delta \Rightarrow \check{}^{k}B}\check{}^{k}R$$

$$\frac{\Delta\langle\overrightarrow{B}|_k\Lambda\rangle \Rightarrow C}{\Delta\langle\overrightarrow{\hat{}^{k}B}\rangle \Rightarrow C}\hat{}^{k}L \qquad \frac{\Delta \Rightarrow B}{\Delta|_k\Lambda \Rightarrow \hat{}^{k}B}\hat{}^{k}R$$

Fig. 12. Unary derived connective rules

5 Grammar

We give a grammar for the Montague fragment of Dowty, Wall and Peters (1981[4], Ch. 7). We structure atomic types N for name or (referring) nominal and CN for common noun or count noun with feature terms for gender for which there are feature constants m (masculine), f (feminine) and n (neuter) and a denumerably infinit supply of feature variables. Feature variables are understood as being universally quantified outermost in types and thus undergo unification in the usual way. Other atomic types are S for statement or (declarative) sentence and CP for complementizer phrase. All these atomic types are of sort 0. Our lexicon for the Montague fragment is as shown in Figure 14; henceforth we omit the subscript $(+)1$ for first wrap on connectives and abbreviate as $-$ the subscript -1 for last wrap.

$$\dfrac{\Gamma \Rightarrow A \qquad \Delta\langle \overrightarrow{C} \rangle \Rightarrow D}{\Delta\langle \Gamma, \dfrac{\overrightarrow{C}}{A} \rangle \Rightarrow D}\,\text{-}L_1 \qquad \dfrac{\Gamma \Rightarrow A \qquad \Delta\langle \overrightarrow{C} \rangle \Rightarrow D}{\Delta\langle \dfrac{\overrightarrow{C}}{A}, \Gamma \rangle \Rightarrow D}\,\text{-}L_2$$

$$\dfrac{\overrightarrow{A}, \Gamma \Rightarrow C \qquad \Gamma, \overrightarrow{A} \Rightarrow C}{\Gamma \Rightarrow \dfrac{C}{A}}\,\text{-}R$$

$$\dfrac{\Delta\langle \overrightarrow{A}, \overrightarrow{B} \rangle \Rightarrow D \qquad \Delta\langle \overrightarrow{B}, \overrightarrow{A} \rangle \Rightarrow D}{\Delta\langle \overrightarrow{A \otimes B} \rangle \Rightarrow D}\otimes L$$

$$\dfrac{\Gamma_1 \Rightarrow A \qquad \Gamma_2 \Rightarrow B}{\Gamma_1, \Gamma_2 \Rightarrow A \otimes B}\otimes R_1 \qquad \dfrac{\Gamma_1 \Rightarrow B \qquad \Gamma_2 \Rightarrow A}{\Gamma_1, \Gamma_2 \Rightarrow A \otimes B}\otimes R_2$$

$$\dfrac{\Gamma \Rightarrow A \qquad \Delta\langle \overrightarrow{C} \rangle \Rightarrow D}{\Delta\langle \Gamma|_k \overrightarrow{A \Downarrow C} \rangle \Rightarrow D}\Downarrow L \qquad \dfrac{\overrightarrow{A}|_1 \Gamma \Rightarrow C \qquad \cdots \qquad \overrightarrow{A}|_{\sigma A} \Gamma \Rightarrow C}{\Gamma \Rightarrow A \Downarrow C}\Downarrow R$$

$$\dfrac{\Gamma \Rightarrow B \qquad \Delta\langle \overrightarrow{C} \rangle \Rightarrow D}{\Delta\langle \overrightarrow{C \Uparrow B}|_k \Gamma \rangle \Rightarrow D}\Uparrow L \qquad \dfrac{\Gamma|_1 \overrightarrow{B} \Rightarrow C \qquad \cdots \qquad \Gamma|_{\sigma C} \overrightarrow{B} \Rightarrow C}{\Gamma \Rightarrow C \Uparrow B}\Uparrow R$$

$$\dfrac{\Delta\langle \overrightarrow{A}|_1 \overrightarrow{B} \rangle \Rightarrow D \qquad \cdots \qquad \Delta\langle \overrightarrow{A}|_{\sigma A} \overrightarrow{B} \rangle \Rightarrow D}{\Delta\langle \overrightarrow{A \circledcirc B} \rangle \Rightarrow D}\circledcirc L \qquad \dfrac{\Gamma_1 \Rightarrow A \qquad \Gamma_2 \Rightarrow B}{\Gamma_1|_k \Gamma_2 \Rightarrow A \circledcirc B}\circledcirc R$$

Fig. 13. Binary derived connective rules

a : $\Box(((S{\uparrow}\Box NA){\downarrow}S)/CNA) : \hat{}\lambda B\lambda C\exists D[(B\ D) \wedge (C\ \hat{}D)]$
and : $\Box((S\backslash S)/S) : \hat{}\lambda A\lambda B[B \wedge A]$
and : $\Box(((NA\backslash S)\backslash(NA\backslash S))/(NA\backslash S)) : \hat{}\lambda B\lambda C\lambda D[(C\ D) \wedge (B\ D)]$
believes : $\Box((NA\backslash S)/CP) : \mathit{believe}$
bill : $\Box Nm : \hat{}b$
catch : $\Box((NA\backslash S)/NB) : \mathit{catch}$
doesnt : $\Box((NA\backslash S)/(NA\backslash S)) : \hat{}\lambda B\lambda C\neg(B\ C)$
eat : $\Box((NA\backslash S)/NB) : \mathit{eat}$
every : $\Box(((S{\uparrow}NA){\downarrow}S)/CNA) : \hat{}\lambda B\lambda C\forall D[(B\ D) \rightarrow (C\ D)]$
finds : $\Box((NA\backslash S)/NB) : \mathit{finds}$
fish : $\Box CNn : \mathit{fish}$
he : $\Box((\Box S|Nm)/\Box(Nm\backslash S)) : \hat{}\lambda A\lambda B\hat{}(\check{}A\ B)$
her : $\Box(\Box((S{\uparrow}Nf) - (J\bullet(Nf\backslash S))){\downarrow}(\Box S|Nf)) : \hat{}\lambda A\lambda B\hat{}(\check{}A\ B)$
her : $\Box(((((S{\uparrow}Nf) - (J\bullet(Nf\backslash S))){\uparrow}\Box Nf) - (J\bullet((Nf\backslash S){\uparrow}Nf))){\downarrow}_{\_}(S{\uparrow}\Box Nf)) : \hat{}\lambda A\lambda B((A\ B)\ \check{}B)$
in : $\Box(((NA\backslash S)\backslash(NA\backslash S))/NB) : \hat{}\lambda C\lambda D\lambda E((\check{}\mathit{in}\ C)\ (D\ E))$
is : $\Box((NA\backslash S)/NB) : \hat{}\lambda C\lambda D[D = C]$
it : $\Box(\Box(S{\uparrow}Nn){\downarrow}(\Box S|Nn)) : \hat{}\lambda A\lambda B\hat{}(\check{}A\ B)$
it : $\Box(((((S{\uparrow}Nn) - (J\bullet(Nn\backslash S))){\uparrow}\Box Nn) - (J\bullet((Nn\backslash S){\uparrow}Nn))){\downarrow}_{\_}(S{\uparrow}\Box Nn)) : \hat{}\lambda A\lambda B((A\ B)\ \check{}B)$
john : $\Box Nm : \hat{}j$
loses : $\Box((NA\backslash S)/NB) : \mathit{loses}$
loves : $\Box((NA\backslash S)/NB) : \mathit{loves}$
man : $\Box CNm : \mathit{man}$
necessarily : $\Box(S/\Box S) : \hat{}\mathit{nec}$
or : $\Box((S\backslash S)/S) : \hat{}\lambda A\lambda B[B \vee A]$
or : $\Box(((NA\backslash S)\backslash(NA\backslash S))/(NA\backslash S)) : \hat{}\lambda B\lambda C\lambda D[(C\ D) \vee (B\ D)]$
park : $\Box CNn : \mathit{park}$
seeks : $\Box((NA\backslash S)/\Box(((NB\backslash S)/NC)\backslash(NB\backslash S))) : \hat{}\lambda D\lambda E((\mathit{tries}\ \hat{}((\check{}D\ \mathit{find})\ E))\ E)$
she : $\Box((\Box S|Nf)/\Box(Nf\backslash S)) : \hat{}\lambda A\lambda B\hat{}(\check{}A\ B)$
slowly : $\Box(\Box(NA\backslash S)\backslash(NA\backslash S)) : \mathit{slowly}$
such+that : $\Box((CNA\backslash CNA)/(S|NA)) : \hat{}\lambda B\lambda C\lambda D[(C\ D) \wedge (B\ D)]$
talks : $\Box(NA\backslash S) : \mathit{talk}$
that : $\Box(CP/\Box S) : \hat{}\lambda AA$
the : $\Box(NA/CNA) : \mathit{the}$
to : $\Box((NA\backslash S)/(NA\backslash S)) : \hat{}\lambda BB$
tries : $\Box((NA\backslash S)/\Box(NA\backslash S)) : \hat{}\lambda B\lambda C((\check{}\mathit{tries}\ \hat{}(\check{}B\ C))\ C)$
unicorn : $\Box CNn : \mathit{unicorn}$
walk : $\Box(NA\backslash S) : \mathit{walk}$
walks : $\Box(NA\backslash S) : \mathit{walk}$
woman : $\Box CNf : \mathit{woman}$

Fig. 14. The Montague fragment

6 Analyses

We analyse the examples from the second half of Chapter 7 of Dowty, Wall and Peters (1981[4]) — DWP; the example numbers of that source are included within displays. The first examples involve the copula of identity. Minimally:

(10) (7-73) $\mathbf{john{+}is{+}bill} : S$

For this there is the semantically labelled sequent:

(11) $\Box Nm : \hat{}j, \Box((NA\backslash S)/NB) : \hat{}\lambda C\lambda D[D = C], \Box Nm : \hat{}b \ \Rightarrow\ S$

This has the derivation given in Figure 15.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{Nm \Rightarrow Nm}}{\Box Nm \Rightarrow Nm}\Box L \qquad \dfrac{\dfrac{\dfrac{}{Nm \Rightarrow Nm}}{\Box Nm \Rightarrow Nm}\Box L \quad \dfrac{}{S \Rightarrow S}}{\Box Nm, Nm\backslash S \Rightarrow S}\backslash L}{\Box Nm, (Nm\backslash S)/Nm, \Box Nm \Rightarrow S}/L}{\Box Nm, \Box((Nm\backslash S)/Nm), \Box Nm \Rightarrow S}\Box L$$

Fig. 15. Derivation for *John is Bill*

It delivers semantics:

(12) $[j = b]$

More subtly:

(13) (7-76) $\mathbf{john{+}is{+}a{+}man} : S$

Inserting the same lexical entry for the copula, lexical lookup yields the semantically annotated sequent:

(14) $\Box Nm : \hat{}j, \Box((NA\backslash S)/NB) : \hat{}\lambda C\lambda D[D = C], \Box(((S{\uparrow}\Box NE){\downarrow}S)/CNE) : \hat{}\lambda F\lambda G\exists H[(F\ H) \wedge (G\ \hat{}H)], \Box CNm : man \ \Rightarrow\ S$

This has the derivation given in Figure 16. The derivation delivers the semantics:

(15) $\exists C[(\check{}man\ C) \wedge [j = C]]$

This is logically equivalent to $({}^{\vee}man\ j)$, as required. This correct interaction of the copula of identity with an indefinitely quantified complement is a nice prediction of Montague grammar, conserved in type logical grammar, and simplified by the lower type of the copula.

The next example involves an intensional adsentential modifier:

$$
\dfrac{\dfrac{\dfrac{}{CNm \Rightarrow CNm}}{\Box CNm \Rightarrow CNm}\Box L \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{Nm \Rightarrow Nm}}{\Box Nm \Rightarrow Nm}\Box L \quad \dfrac{\dfrac{\dfrac{}{Nm \Rightarrow Nm}}{\Box Nm \Rightarrow Nm}\Box L \quad \dfrac{}{S \Rightarrow S}}{\Box Nm, Nm\backslash S \Rightarrow S}\backslash L}{\Box Nm, (Nm\backslash S)/Nm, \Box Nm \Rightarrow S}/L}{\Box Nm, \Box((Nm\backslash S)/Nm), \Box Nm \Rightarrow S}\Box L}{\Box Nm, \Box((Nm\backslash S)/Nm), 1 \Rightarrow S{\uparrow}\Box Nm}{\uparrow}R \qquad \dfrac{}{S \Rightarrow S}}{\Box Nm, \Box((Nm\backslash S)/Nm), (S{\uparrow}\Box Nm){\downarrow}S \Rightarrow S}{\downarrow}L}{\dfrac{\Box Nm, \Box((Nm\backslash S)/Nm), ((S{\uparrow}\Box Nm){\downarrow}S)/CNm, \Box CNm \Rightarrow S}{\Box Nm, \Box((Nm\backslash S)/Nm), \Box(((S{\uparrow}\Box Nm){\downarrow}S)/CNm), \Box CNm \Rightarrow S}\Box L}/L
$$

Fig. 16. Derivation for *John is a man*

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{Nm \Rightarrow Nm}}{\Box Nm \Rightarrow Nm}\Box L \quad \dfrac{}{S \Rightarrow S}}{\Box Nm, Nm\backslash S \Rightarrow S}\backslash L}{\Box Nm, \Box(Nm\backslash S) \Rightarrow S}\Box L}{\Box Nm, \Box(Nm\backslash S) \Rightarrow \Box S}\Box R \quad \dfrac{}{S \Rightarrow S}}{\dfrac{S/\Box S, \Box Nm, \Box(Nm\backslash S) \Rightarrow S}{\Box(S/\Box S), \Box Nm, \Box(Nm\backslash S) \Rightarrow S}\Box L}/L
$$

Fig. 17. Derivation for *Necessarily John walks*

(16) (7-83) $\mathbf{necessarily{+}john{+}walks} : S$

Lexical lookup yields the following semantically labelled sequent:

(17) $\Box(S/\Box S) : \hat{}nec, \Box Nm : \hat{}j, \Box(NA\backslash S) : walk \ \Rightarrow \ S$

This has the derivation given in Figure 17. The derivation delivers semantics:

(18) $(nec\ \hat{}(\check{}walk\ j))$

The following example involves an adverb:

(19) (7-86) $\mathbf{john{+}walks{+}slowly} : S$

This is also assumed to create an intensional context. Lexical lookup yields:

(20) $\Box Nm : \hat{}j, \Box(NA\backslash S) : walk, \Box(\Box(NB\backslash S)\backslash(NB\backslash S)) : slowly \ \Rightarrow \ S$

This has the derivation given in Figure 18, which delivers semantics (in η-long form):

(21) $((\check{}slowly\ \hat{}\lambda A(\check{}walk\ A))\ j)$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\overline{Nm \Rightarrow Nm} \quad \overline{S \Rightarrow S}}{Nm, Nm\backslash S \Rightarrow S}\backslash L}{Nm, \Box(Nm\backslash S) \Rightarrow S}\Box L}{\Box(Nm\backslash S) \Rightarrow Nm\backslash S}\backslash R}{\Box(Nm\backslash S) \Rightarrow \Box(Nm\backslash S)}\Box R \quad \dfrac{\dfrac{\overline{Nm \Rightarrow Nm}}{\Box Nm \Rightarrow Nm}\Box L \quad \overline{S \Rightarrow S}}{\Box Nm, Nm\backslash S \Rightarrow S}\backslash L}{\Box Nm, \Box(Nm\backslash S), \Box(Nm\backslash S)\backslash(Nm\backslash S) \Rightarrow S}\backslash L}{\Box Nm, \Box(Nm\backslash S), \Box(\Box(Nm\backslash S)\backslash(Nm\backslash S)) \Rightarrow S}\Box L$$

Fig. 18. Derivation for *John walks slowly*

The next example involves an equi control verb:

(22) (7-91) $\mathbf{john{+}tries{+}to{+}walk} : S$

We lexically analyse the equi semantics as a relation of trying between the subject and a proposition of which the subject is agent (something Montague did not do). Lexical lookup yields:

(23) $\Box Nm : \hat{}j, \Box((NA\backslash S)/\Box(NA\backslash S)) : \hat{}\lambda B\lambda C((\check{}tries\ \hat{}(\check{}B\ C))\ C),$
$\Box((ND\backslash S)/(ND\backslash S)) : \hat{}\lambda EE, \Box(NF\backslash S) : walk \ \Rightarrow \ S$

This has the derivation given in Figure 19, which delivers the semantics:

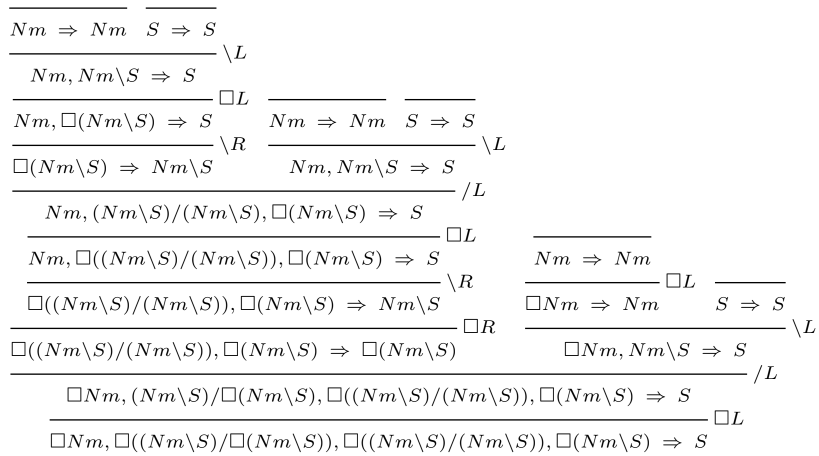

Fig. 19. Derivation for *John tries to walk*

(24) $((\check{~}tries\ \hat{~}(\check{~}walk\ j))\ j)$

I.e. that John tries to bring about the state of affairs that he (John) walks.

The next example involves control, quantification, coordination and also anaphora:

(25) (7-94) $\mathbf{john{+}tries{+}to{+}catch{+}a{+}fish{+}and{+}eat{+}it} : S$

The sentence is ambiguous as to whether *a fish* is wide scope (with existential commitment) or narrow scope (without existential commitment) with respect to *tries*, but in both cases it must be the antecedent of *it*. Lexical lookup inserting a sentential coordinator or the (clause) external anaphora pronoun assignment has no derivation. Lexical lookup inserting the verb phrase coordinator and the internal (clause local) anaphora pronoun assignment yields the semantically labelled sequent:

(26) $\Box Nm : \hat{~}j, \Box((NA\backslash S)/\Box(NA\backslash S)) : \hat{~}\lambda B\lambda C((\check{~}tries\ \hat{~}(\check{~}B\ C))\ C),$
$\Box((ND\backslash S)/(ND\backslash S)) : \hat{~}\lambda EE, \Box((NF\backslash S)/NG) : catch,$
$\Box(((S{\uparrow}\Box NH){\downarrow}S)/CNH) : \hat{~}\lambda I\lambda J\exists K[(I\ K) \wedge (J\ \hat{~}K)], \Box CNn : fish,$
$\Box(((NL\backslash S)\backslash(NL\backslash S))/(NL\backslash S)) : \hat{~}\lambda M\lambda N\lambda O[(N\ O) \wedge (M\ O)],$
$\Box((NP\backslash S)/NQ) : eat,$
$\Box(((((S{\uparrow}Nn) - (J{\bullet}(Nn\backslash S))){\uparrow}\Box Nn) - (J{\bullet}((Nn\backslash S){\uparrow}Nn))){\downarrow_{<}}(S{\uparrow}\Box Nn)) :$
$\hat{~}\lambda R\lambda S((R\ S)\ \check{~}S) \Rightarrow S$

Because we do not have verb form features on S this has one derivation on the pattern *[tries to catch a fish] and [eat it]* in which a finite verb phrase coordinates with a base form verb phrase. This would be excluded as required by adding the features. A wide scope existential derivation delivers semantics with existential commitment as follows; the derivation is too large to fit on a page.

(27) $\exists C[(\check{~}fish\ C) \wedge ((\check{~}tries\ \hat{~}[((\check{~}catch\ C)\ j) \wedge ((\check{~}eat\ C)\ j)])\ j)]$

Also because of the absence of verb form features, there is an existential narrow scope derivation on the pattern of *[to catch a fish] and [eat it]* in which an infinitive verb phrase coordinates with a base form verb phrase. This would also be straightforwardly ruled out by including the relevant features on S. An appropriate existential narrow scope derivation, which is too large to fit on the page, delivers the semantics without existential commitment:

(28) $((\check{}tries\ \hat{}\exists H[(\check{}fish\ H) \wedge [((\check{}catch\ H)\ j) \wedge ((\check{}eat\ H)\ j)]])\ j)$

The next example involves an extensional transitive verb:

(29) (7-98) $\mathbf{john{+}finds{+}a{+}unicorn} : S$

This sentence cannot be true unless a unicorn exists. Our treatment of this is simpler than Montague's because while Montague had to raise the type of extensional verbs to accommodate intensional verbs ("raising to the worst case"), and then use meaning postulates to capture the existential commitment, type logical grammar allows assignment of the lower types which capture it automatically. Lexical lookup yields:

(30) $\Box Nm : \hat{}j, \Box((NA\backslash S)/NB) : \mathit{finds}, \Box(((S{\uparrow}\Box NC){\downarrow}S)/CNC) :$
$\hat{}\lambda D\lambda E\exists F[(D\ F) \wedge (E\ \hat{}F)], \Box CNn : \mathit{unicorn}\ \Rightarrow\ S$

This has the derivation given in Figure 20, which yields the semantics with existential

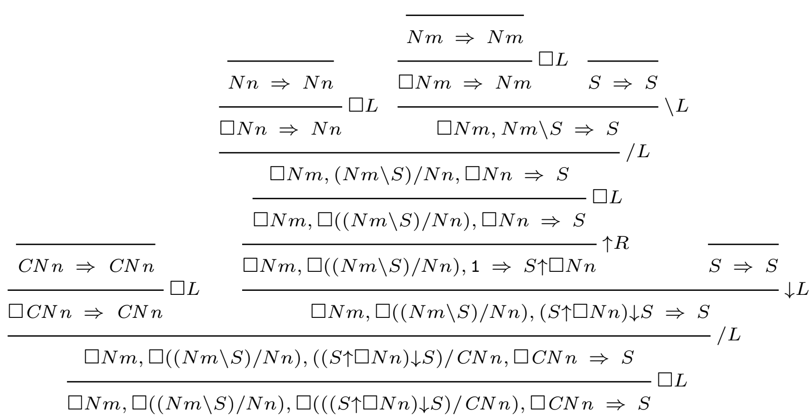

Fig. 20. Derivation for *John finds a unicorn*

commitment:

(31) $\exists C[(\check{}unicorn\ C) \wedge ((\check{}finds\ C)\ j)]$

DWP continue with a donkey sentence, for which of course Montague grammar and our cover grammar make the wrong prediction:

(32) (7-105) $\textbf{every+man+such+that+he+loves+a+woman+loses+her} : S$

There is a dominant reading in which *a woman* which is the donkey anaphora antecedent is understood universally, but Montague semantics obtains only an at best subordinate reading in which *a woman* is quantified existentially at the matrix level. Lexical lookup inserting the external anaphora assignment to *her* yields no derivation. Lexical insertion of the internal anaphora assignment yields:

(33) $\Box(((S{\uparrow}NA){\downarrow}S)/CNA) : \hat{}\lambda B\lambda C\forall D[(B\ D) \to (C\ D)], \Box CNm : man,$
$\Box((CNE\backslash CNE)/(S|NE)) : \hat{}\lambda F\lambda G\lambda H[(G\ H) \wedge (F\ H)],$
$\Box((\Box S|Nm)/\Box(Nm\backslash S)) : \hat{}\lambda I\lambda J\hat{}(\check{}I\ J), \Box((NK\backslash S)/NL) : loves,$
$\Box(((S{\uparrow}\Box NM){\downarrow}S)/CNM) : \hat{}\lambda N\lambda O\exists P[(N\ P) \wedge (O\ \hat{}P)], \Box CNf : woman,$
$\Box((NQ\backslash S)/NR) : loses,$
$\Box(((((S{\uparrow}Nf) - (J{\bullet}(Nf\backslash S))){\uparrow}\Box Nf) - (J{\bullet}((Nf\backslash S){\uparrow}Nf))){\downarrow_{-}}(S{\uparrow}\Box Nf)) :$
$\hat{}\lambda S\lambda T((S\ T)\ \check{}T) \Rightarrow S$

The derivation of this is too large for the page, but it delivers semantics:

(34) $\exists C[(\check{}woman\ C) \wedge \forall K[[(\check{}man\ K) \wedge ((\check{}loves\ C)\ K)] \to ((\check{}loses\ C)\ K)]]$

The assignment of lowest types in type logical grammar also means that existential commitment of a preposition comes without the need for devices such as meaning postulates in Montague grammar:

(35) (7-110) $\textbf{john+walks+in+a+park} : S$

Lexical lookup for this example yields the semantically labelled sequent:

(36) $\Box Nm : \hat{}j, \Box(NA\backslash S) : walk, \Box(((NB\backslash S)\backslash(NB\backslash S))/NC) :$
$\hat{}\lambda D\lambda E\lambda F((\check{}in\ D)\ (E\ F)), \Box(((S{\uparrow}\Box NG){\downarrow}S)/CNG) :$
$\hat{}\lambda H\lambda I\exists J[(H\ J) \wedge (I\ \hat{}J)], \Box CNn : park \Rightarrow S$

This sequent has the proof given in Figure 21, which delivers the semantics (with existential commitment):

(37) $\exists C[(\check{}park\ C) \wedge ((\check{}in\ C)\ (\check{}walk\ j))]$

Finally, DWP analyse the ambiguous example:

(38) (7-116, 7-118) $\textbf{every+man+doesnt+walk} : S$

This has a dominant reading in which the universal has narrow scope with respect to the negation, and a subordinate reading in which the universal has wide scope with respect to the negation. Our grammar generates only the subordinate reading. Lexical lookup yields:

(39) $\Box(((S{\uparrow}NA){\downarrow}S)/CNA) : \hat{}\lambda B\lambda C\forall D[(B\ D) \to (C\ D)], \Box CNm : man,$
$\Box((NE\backslash S)/(NE\backslash S)) : \hat{}\lambda F\lambda G\neg(F\ G), \Box(NH\backslash S) : walk \Rightarrow S$

This has the derivation given in Figure 22, which delivers semantics:

(40) $\forall C[(\check{}man\ C) \to \neg(\check{}walk\ C)]$

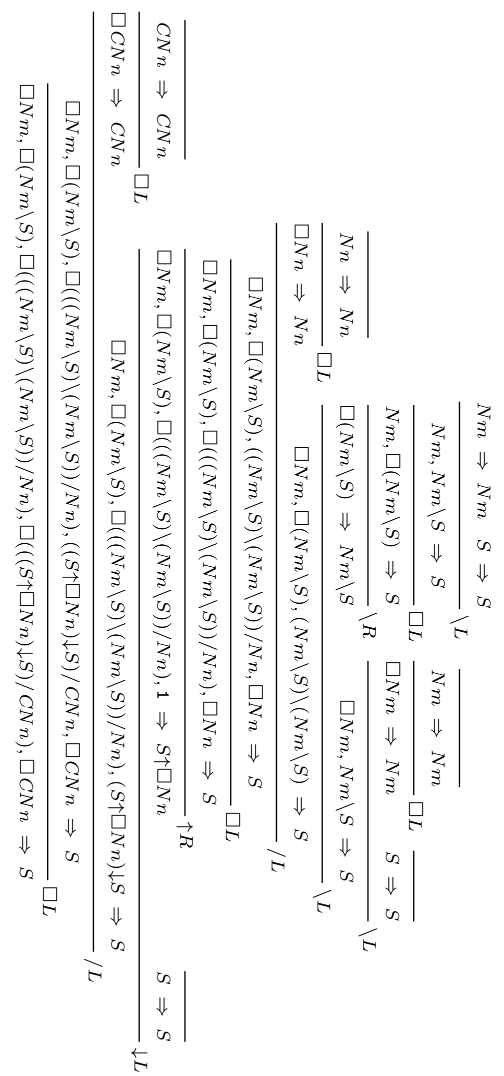

Fig. 21. Derivation for *John walks in a park*

$$\dfrac{Nm \Rightarrow Nm \qquad S \Rightarrow S}{Nm, Nm\backslash S \Rightarrow S}\backslash L$$

$$\dfrac{Nm, Nm\backslash S \Rightarrow S}{Nm, \Box(Nm\backslash S) \Rightarrow S}\Box L$$

$$\dfrac{Nm, \Box(Nm\backslash S) \Rightarrow S}{\Box(Nm\backslash S) \Rightarrow Nm\backslash S}\backslash R \qquad \dfrac{Nm \Rightarrow Nm \qquad S \Rightarrow S}{Nm, Nm\backslash S \Rightarrow S}\backslash L$$

$$\dfrac{\Box(Nm\backslash S) \Rightarrow Nm\backslash S \qquad Nm, Nm\backslash S \Rightarrow S}{Nm, (Nm\backslash S)/(Nm\backslash S), \Box(Nm\backslash S) \Rightarrow S}/L$$

$$\dfrac{Nm, (Nm\backslash S)/(Nm\backslash S), \Box(Nm\backslash S) \Rightarrow S}{Nm, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S}\Box L$$

$$\dfrac{Nm, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S}{1, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S{\uparrow}Nm}{\uparrow}R$$

$$\dfrac{1, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S{\uparrow}Nm \qquad S \Rightarrow S}{(S{\uparrow}Nm){\downarrow}S, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S}{\downarrow}L$$

$$\dfrac{CNm \Rightarrow CNm}{\Box CNm \Rightarrow CNm}\Box L$$

$$\dfrac{\Box CNm \Rightarrow CNm \qquad (S{\uparrow}Nm){\downarrow}S, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S}{((S{\uparrow}Nm){\downarrow}S)/CNm, \Box CNm, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S}/L$$

$$\dfrac{((S{\uparrow}Nm){\downarrow}S)/CNm, \Box CNm, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S}{\Box(((S{\uparrow}Nm){\downarrow}S)/CNm), \Box CNm, \Box((Nm\backslash S)/(Nm\backslash S)), \Box(Nm\backslash S) \Rightarrow S}\Box L$$

Fig. 22. Derivation for *Every man doesn't walk*

7 Conclusion

The negation-as-failure rule is as follows:

(41) $$\frac{\not\vdash \Gamma \Rightarrow A}{\Gamma \Rightarrow \neg A}\neg R$$

The calculus is presented without the Cut rule:

(42) $$\frac{\Gamma \Rightarrow A \qquad \Delta\langle\overrightarrow{A}\rangle \Rightarrow B}{\Delta\langle\Gamma\rangle \Rightarrow B}\, Cut$$

This is because transitivity of inference is unsuitable in the presence of the negation-as-failure (Morrill and Valentín 2010[19]). We believe that the remaining rules enjoy Cut-elimination. Thus, Morrill et al. (2011[21]) appendix proves Cut-elimination for the displacement calculus **D**; Moortgat (1995[7]) proves Cut-elimination for the bracket modalities in ordinary sequent calculus, and the other rules follow patterns in standard logic or linear logic for which there is Cut-elimination. Cut-free backward chaining hypersequent proof search operates in a finite space and so constitutes a terminating procedure for parsing/theorem-proving. Cut-free categorial sequent proof search still suffers from (finite) spurious ambiguity, but this can be treated by normalisation (Morrill 2011[15]). This is the basis of the implementation of the placement logic used for this paper: the parser/theorem prover CatLog of Morrill (2012[16]). Apart from the shorter-term objective of refining the CatLog implementation of the current type formalism in hypersequent calculus, we define as a longer times goal the implementation of the same logic in proof nets.

Bibliography

[1] Emmon Bach. Discontinuous constituents in generalized categorial grammars. In V.A. Burke and J. Pustejovsky, editors, *Proceedings of the 11th Annual Meeting of the North Eastern Linguistics Society, New York*, pages 1–12. GLSA Publications, Department of Linguistics, University of Massachussets at Amherst, Amherst, Massachussets, 1981.

[2] Emmon Bach. Some Generalizations of Categorial Grammars. In Fred Landman and Frank Veltman, editors, *Varieties of Formal Semantics: Proceedings of the Fourth Amsterdam Colloquium*, pages 1–23. Foris, Dordrecht, 1984. Reprinted in Walter J. Savitch, Emmon Bach, William Marsh and Gila Safran-Naveh, editors, 1987, *The Formal Complexity of Natural Language*, D. Reidel, Dordrecht, 251–279.

[3] Guy Barry, Mark Hepple, Neil Leslie, and Glyn Morrill. Proof Figures and Structural Operators for Categorial Grammar. In *Proceedings of the Fifth Conference of the European Chapter of the Association for Computational Linguistics*, Berlin, 1991.

[4] David R. Dowty, Robert E. Wall, and Stanley Peters. *Introduction to Montague Semantics*, volume 11 of *Synthese Language Library*. D. Reidel, Dordrecht, 1981.

[5] J. Lambek. On the Calculus of Syntactic Types. In Roman Jakobson, editor, *Structure of Language and its Mathematical Aspects, Proceedings of the Symposia in Applied Mathematics XII*, pages 166–178. American Mathematical Society, Providence, Rhode Island, 1961.

[6] M. Moortgat and R.T. Oehrle. Adjacency, dependency and order. In P. Dekker and M. Stokhof, editors, *Proceedings of the Ninth Amsterdam Colloquim*, pages 447–466, Amsterdam, 1994. ILLC.

[7] Michael Moortgat. Multimodal linguistic inference. *Journal of Logic, Language and Information*, 5(3, 4):349–385, 1996. Also in *Bulletin of the IGPL*, 3(2,3):371–401, 1995.

[8] Michael Moortgat. Categorial Type Logics. In Johan van Benthem and Alice ter Meulen, editors, *Handbook of Logic and Language*, pages 93–177. Elsevier Science B.V. and the MIT Press, Amsterdam and Cambridge, Massachusetts, 1997.

[9] Michael Moortgat and Glyn Morrill. Heads and phrases: Type calculus for dependency and constituent structure. *Manuscript, Universiteit Utrecht*, 1991.

[10] R. Moot. Grail: An automated proof assistant for categorial grammar logics. In R.C. Backhouse, editor, *Proceedings of the 1998 User Interfaces or Theorem Provers Conference*, 1998.

[11] Richard Moot and Christian Retoré. *The Logic of Categorial Grammars: A Deductive Account of Natural Language Syntax and Semantics*. Springer, Heidelberg, 2012.

[12] G. Morrill. Grammar and Logical Types. In Martin Stockhof and Leen Torenvliet, editors, *Proceedings of the Seventh Amsterdam Colloquium*, pages 429–450, 1990. Also in G. Barry and G. Morrill, editors, *Studies in Categorial Grammar*, Edin-

burgh Working Papers in Cognitive Science, Volume 5, pages 127–148: 1990. Revised version published as Grammar and Logic, *Theoria*, LXII, 3:260–293, 1996.

[13] Glyn Morrill. Intensionality and Boundedness. *Linguistics and Philosophy*, 13(6):699–726, 1990.

[14] Glyn Morrill. Categorial Formalisation of Relativisation: Pied Piping, Islands, and Extraction Sites. Technical Report LSI-92-23-R, Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, 1992.

[15] Glyn Morrill. Logic Programming of the Displacement Calculus. In Sylvain Pogodalla and Jean-Philippe Prost, editors, *Proceedings of Logical Aspects of Computational Linguistics 2011, LACL'11, Montpellier*, number LNAI 6736 in Springer Lecture Notes in AI, pages 175–189, Berlin, 2011. Springer.

[16] Glyn Morrill. CatLog: A Categorial Parser/Theorem-Prover. In *LACL 2012 System Demonstrations*, Logical Aspects of Computational Linguistics 2012, pages 13–16, Nantes, 2012.

[17] Glyn Morrill and Josep-Maria Merenciano. Generalising discontinuity. *Traitement automatique des langues*, 37(2):119–143, 1996.

[18] Glyn Morrill and Oriol Valentín. Displacement Calculus. *Linguistic Analysis*, 36(1–4):167–192, 2010. Special issue Festschrift for Joachim Lambek, http://arxiv.org/abs/1004.4181.

[19] Glyn Morrill and Oriol Valentín. On Anaphora and the Binding Principles in Categorial Grammar. In Anuj Dawar and Ruy de Queiroz, editors, *Proceedings of the 17th International Workshop on Logic, Language, Information and Computation, WoLLIC 2010, Brasília*, number LNAI 6188 in Lecture Notes in Artificial Intelligence, pages 176–190, Berlin, 2010. Springer.

[20] Glyn Morrill, Oriol Valentín, and Mario Fadda. Dutch Grammar and Processing: A Case Study in TLG. In Peter Bosch, David Gabelaia, and Jérôme Lang, editors, *Logic, Language, and Computation: 7th International Tbilisi Symposium, Revised Selected Papers*, number 5422 in Lecture Notes in Artificial Intelligence, pages 272–286, Berlin, 2009. Springer.

[21] Glyn Morrill, Oriol Valentín, and Mario Fadda. The Displacement Calculus. *Journal of Logic, Language and Information*, 20(1):1–48, 2011. Doi 10.1007/s10849-010-9129-2.

[22] Glyn V. Morrill. *Type Logical Grammar: Categorial Logic of Signs*. Kluwer Academic Publishers, Dordrecht, 1994.

[23] Glyn V. Morrill. *Categorial Grammar: Logical Syntax, Semantics, and Processing*. Oxford University Press, New York and Oxford, 2011.

[24] R. T. Oehrle. Multi-Modal Type-Logical Grammar. In R. D. Borsley and K. Börjars, editors, *Non-transformational Syntax: Formal and Explicit Models of Grammar*. Wiley-Blackwell, Oxford, UK, 2011. doi 10.1002/9781444395037.ch6.

[25] Richard T. Oehrle and Shi Zhang. Lambek calculus and preposing of embedded subjects. In *Chicago Linguistics Society*, volume 25, Chicago, 1989.