Parsing Logical Grammar: CatLog3*

Glyn Morrill
Department of Computer Science
Universitat Politècnica de Catalunya
Barcelona, Spain.

morrill@cs.upc.edu

Abstract

CatLog3 is a Prolog parser/theorem-prover for (type) logical (categorial) grammar. In such logical grammar, grammar is *reduced* to logic: a string of words is grammatical if and only if an associated logical statement is a theorem. CalLog3 implements a logic extending displacement calculus, a sublinear fragment including as primitive connectives the continuous (Lambek) and discontinuous wrapping connectives of the displacement calculus, additives, 1st order quantifiers, normal modalities, bracket modalities and subexponentials. In this paper we survey how CatLog3 is implemented on the principles of Andreoli's focusing and a generalisation of van Benthem's count-invariance.

1 Introduction

The linguistics that has descended from formal grammar as popularised by Chomsky (1957[6]) has renegaded on formalisation, and discrete computational grammar in the genre of the 1980s has given way to statistical NLP. However, there is a venerable older line of linguistics practicing grammar according to the standards of mathematical logic. The seminal paper in this line is Lambek (1958[19]) which defines a syntactic calculus and proves Cut-elimination for it, but the tradition dates back at least to Bar-Hillel (1953[3]) and Ajdukiewicz (1935[1]). The Lambek calculus is a calculus of concatentation which is free of structural rules. The displacement calculus of Morrill et al. (2011[46]) generalises Lambek calculus with intercalation, containing both continuous and discontinuous connective families, while remaining free of structural rules, and preserving Cut-elimination and its good corollaries: the subformula property, decidability, the finite reading property, and the focusing property. There are several monographs and reference articles on this type logical approach: Moortgat (1988[22]; 1997[24])), Morrill (1994[47];

*This work was partially supported by an ICREA Academia 2012, and MINECO project APCOM (TIN2014-57226-P). Thanks to Oriol Valentín for discussions in relation to this work.

2010[31]; 2011[48]; 2012[34]), Carpenter (1997[5]), Jäger (2005[13]), and Moot and Retoré (2012[27]). The CatLog program series comprises implementations in Prolog of type logical parser/theorem-provers starting from the basis of logic programming of displacement calculus theorem-proving (Morrill 2011[32]):

- CatLog1 (Morrill 2012[33] was based on the method of uniform proof (Miller et al. 1991[20]), and the method of count-invariance for multiplicatives (van Benthem 1991[51]).
- CatLog2 was based on Andreoli's focusing (Andreoli 1992[2]) , and count-invariance for multiplicatives, additives and bracket modalities (Valentín et al. 2013[50]).
- CatLog3 is based on focalisation and count-invariance for multiplicatives, additives, bracket modalities and exponentials (Kuznetov et al. 2017[16]).

In this paper we survey the methods on which the implementation of CatLog3 is based. In Section 2 we describe the primitive connectives of the logical fragment for which parsing/theorem-proving is implemented. In Sections 3 and 4 we discuss focusing and count-invariance respectively. In Section 5 we illustrate in relation to the *Montague Test* (Morrill and Valentín 2016[43]): the task of providing a computational grammar of Montague's (1973[21]) fragment.

2 Displacement logic

The formalism used comprises the connectives of Table 1. The heart of the logic is the displacement calculus of Morrill and Valentín (2010[38]) and Morrill, Valentín and Fadda (2011[46]) made up of twin continuous and discontinuous residuated families of connectives having a pure Gentzen sequent calculus —without labels and free of structural rules— and enjoying Cut-elimination (Valentín 2012[49]). Other primary connectives include additives, 1st order quantifiers, normal (i.e. distributive) modalities, bracket (i.e. nondistributive) modalities, and exponentials.[1] We can draw a clear distinction between the primary connectives, the semantically inactive connectives, and the synthetic connectives; the latter two are abbreviatory and are there for convenience, and to simplify derivation. There are semantically inactive variants of the continuous and discontinuous multiplicatives, and semantically inactive variants of the additives, 1st order quantifiers, and normal modalities.[2] Synthetic connectives (Girard 2011[11]) divide into the continuous

[1]Once Cut-elimination is established, the only challenge to decidability comes from nonlinearity: the contraction rule of the universal exponential, and the infinitary left rule of the existential exponential. In this connection, linguistically the existential left rule is not required; and Morrill and Valentín (2015[41]) introduced displacement logics $\mathbf{Db!_b?}$ and $\mathbf{Db!?}$ with a relevant modality ! ,with and without bracket conditioning for contaction. Kanovich et al. (2016[15]) prove the undecidability of $\mathbf{Db}$!? and in unpublished work announce the undecidability of $\mathbf{Db!_b?}$. But Morrill and Valentín (2015[41]) prove the decidability of a linguistically sufficient 'bracket non-negative' special case of $\mathbf{Db!_b?}$.

[2]For example, the semantically inactive additive conjunction $A \sqcap B\colon \phi$ abbreviates $A \& B\colon (\phi, \phi)$.

	cont. mult.	disc. mult.	add.	qu.	norm. mod.	brack. mod.	exp.	lim. contr. & weak.
primary	$/$ $\backslash$ $\bullet$ I	$\uparrow$ $\downarrow$ $\odot$ J	$\&$ $\oplus$	$\bigwedge$ $\bigvee$	$\Box$ $\Diamond$	$[\,]^{-1}$ $\langle\rangle$	$!$ $?$	$\mid$ W
sem. inactive variants	$\bullet\!\!-$ $-\!\!\circ$ $\circ\!\!-$ $-\!\!\bullet$ ◐ ◑	⫯ ⫰ ⫯ ⫰ ◓ ◒	$\sqcap$ $\sqcup$	∀ $\exists$	$\blacksquare$ $\blacklozenge$			
det. synth.	$\triangleleft^{-1}$ $\triangleright^{-1}$ $\triangleleft$ $\triangleright$	$\check{}$ $\hat{}$						diff.
nondet. synth.	$\div$ $\times$	$\Uparrow$ $\Downarrow$ $\circledcirc$						$-$

Table 1: Categorial connectives

1.	$\mathcal{F}_i$	::=	$\mathcal{F}_{i+j}/\mathcal{F}_j$	$T(C/B)$	=	$T(B)\to T(C)$	over
2.	$\mathcal{F}_j$	::=	$\mathcal{F}_i\backslash\mathcal{F}_{i+j}$	$T(A\backslash C)$	=	$T(A)\to T(C)$	under
3.	$\mathcal{F}_{i+j}$	::=	$\mathcal{F}_i\bullet\mathcal{F}_j$	$T(A\bullet B)$	=	$T(A)\&T(B)$	continuous product
4.	$\mathcal{F}_0$	::=	I	$T(I)$	=	$\top$	continuous unit
5.	$\mathcal{F}_{i+1}$	::=	$\mathcal{F}_{i+j}\uparrow_k\mathcal{F}_j, 1\leq k\leq i+j$	$T(C\uparrow_k B)$	=	$T(B)\to T(C)$	circumfix
6.	$\mathcal{F}_j$	::=	$\mathcal{F}_{i+1}\downarrow_k\mathcal{F}_{i+j}, 1\leq k\leq i+1$	$T(A\downarrow_k C)$	=	$T(A)\to T(C)$	infix
7.	$\mathcal{F}_{i+j}$	::=	$\mathcal{F}_{i+1}\odot_k\mathcal{F}_j, 1\leq k\leq i+1$	$T(A\odot_k B)$	=	$T(A)\&T(B)$	discontinuous product
8.	$\mathcal{F}_1$	::=	J	$T(J)$	=	$\top$	discontinuous unit
9.	$\mathcal{F}_i$	::=	$\mathcal{F}_i\&\mathcal{F}_i$	$T(A\&B)$	=	$T(A)\&T(B)$	additive conjunction
10.	$\mathcal{F}_i$	::=	$\mathcal{F}_i\oplus\mathcal{F}_i$	$T(A\oplus B)$	=	$T(A)+T(B)$	additive disjunction
11.	$\mathcal{F}_i$	::=	$\bigwedge V\mathcal{F}_i$	$T(\bigwedge vA)$	=	$F\to T(A)$	1st order univ. qu.
12.	$\mathcal{F}_i$	::=	$\bigvee V\mathcal{F}_i$	$T(\bigvee vA)$	=	$F\&T(A)$	1st order exist. qu.
13.	$\mathcal{F}_i$	::=	$\Box\mathcal{F}_i$	$T(\Box A)$	=	$\mathbf{L}T(A)$	universal modality
14.	$\mathcal{F}_i$	::=	$\Diamond\mathcal{F}_i$	$T(\Diamond A)$	=	$\mathbf{M}T(A)$	existential modality
15.	$\mathcal{F}_i$	::=	$[\,]^{-1}\mathcal{F}_i$	$T([\,]^{-1}A)$	=	$T(A)$	univ. bracket modality
16.	$\mathcal{F}_i$	::=	$\langle\rangle\mathcal{F}_i$	$T(\langle\rangle A)$	=	$T(A)$	exist. bracket modality
17.	$\mathcal{F}_0$	::=	$!\mathcal{F}_0$	$T(!A)$	=	$T(A)$	universal exponential
18.	$\mathcal{F}_0^\circ$	::=	$?\mathcal{F}_0^\circ$	$T(?A)$	=	$T(A)^+$	existential exponential

Table 2: Syntactic types of $\mathbf{DA1S4b!_b?}$

and discontinuous deterministic (unary) synthetic connectives, and the continuous and discontinuous nondeterministic (binary) synthetic connectives.[3]

2.1 Syntactic types

The syntactic types of displacement logic are sorted $\mathcal{F}_0, \mathcal{F}_1, \mathcal{F}_2, \ldots$ according to the number of points of discontinuity $0, 1, 2, \ldots$ their expressions contain. Each type predicate letter has 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 of $\mathbf{DA1S4b!_b?}$ are formed as illustrated in Table 2, and the structure preserving semantic type map T associates these with semantic types.

2.2 Gentzen sequent calculus

We use a Gentzen sequent presentation standard from Gentzen (1934[9]) and Lambek (1958[19]). In Gentzen sequent antecedents for displacement logic with bracket modalities (structural inhibition) and exponentials (structural facilitation) there are also bracket constructors and 'stoups'.

Stoups (cf. the linear logic of Girard 2011[11]) (ζ) are stores read as multisets for re-usable (nonlinear) resources which appear at the left of a configuration marked off by a semicolon (when the stoup is empty the semicolon may be ommited). The stoup of linear logic is for resources which can be contracted (copied)

[3]For example, the nondeterministic continuous division $B\div A$ abbreviates $(A\backslash B)\sqcap(B/A)$.

or weakened (deleted). By contrast, our stoup is for a linguistically motivated variant of contraction, and does not allow weakening. Furthermore, whereas linear logic is commutative, our logic is in general noncommutative and here the stoup is used for resources which are also commutative. A configuration together with a stoup is a *zone* (Ξ). The bracket constructor applies not to a configuration alone but to a configuration with a stoup, i.e a zone: reusable resources are specific to their domain. Stoups $\mathcal{S}$ and configurations $\mathcal{O}$ are defined by ($\emptyset$ is the empty stoup; Λ is the empty configuration; the *separator* 1 marks points of discontinuity):[4]

$$(1)\quad \begin{array}{lcl} \mathcal{S} & ::= & \emptyset \mid \mathcal{F}_0, \mathcal{S} \\ \mathcal{O} & ::= & \Lambda \mid \mathcal{T}, \mathcal{O} \\ \mathcal{T} & ::= & 1 \mid \mathcal{F}_0 \mid \mathcal{F}_{i>0}\{\underbrace{\mathcal{O} : \ldots : \mathcal{O}}_{i\ \mathcal{O}'\text{s}}\} \mid [\mathcal{S}; \mathcal{O}] \end{array}$$

For a type A, its sort $s(A)$ is the i such that $A \in \mathcal{F}_i$. For a configuration Γ, its sort $s(\Gamma)$ is $|\Gamma|_1$, i.e. the number of points of discontinuity 1 which it contains. Sequents are of the form:

$$(2)\quad \mathcal{S}; \mathcal{O} \Rightarrow \mathcal{F} \text{ such that } s(\mathcal{O}) = s(\mathcal{F})$$

The figure $\overrightarrow{A}$ of a type A is defined by:

$$(3)\quad \overrightarrow{A} = \begin{cases} A & \text{if } s(A) = 0 \\ A\{\underbrace{1 : \ldots : 1}_{s(A)\ 1'\text{s}}\} & \text{if } s(A) > 0 \end{cases}$$

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 1's in Γ by $\Delta_1, \ldots, \Delta_i$ respectively. Where Γ is of sort i, the hyperoccurrence notation $\Delta\langle\Gamma\rangle$ abbreviates $\Delta_0(\Gamma \otimes \langle \Delta_1 : \ldots : \Delta_i \rangle)$, i.e. a context configuration Δ (which is externally Δ_0 and internally $\Delta_1, \ldots, \Delta_i$) with a potentially discontinuous distinguished subconfiguration Γ (continuous if $i = 0$, discontinuous if $i > 0$). Where Δ is a configuration of sort $i > 0$ and Γ is a configuration, the kth *metalinguistic intercalation* $\Delta \,|_k\, \Gamma$, $1 \leq k \leq i$, is given by:

$$(4)\quad \Delta \,|_k\, \Gamma =_{df} \Delta \otimes \langle \underbrace{1 : \ldots : 1}_{k-1\ 1\text{'s}} : \Gamma : \underbrace{1 : \ldots : 1}_{i-k\ 1\text{'s}} \rangle$$

i.e. $\Delta \,|_k\, \Gamma$ is the configuration resulting from replacing by Γ the kth separator in Δ.

2.3 Rules and linguistic applications

A semantically labelled sequent is a sequent in which the antecedent type occurrences $A_1, \ldots, A_n$ are labelled by distinct variables $x_1, \ldots, x_n$ which are of types

[4] Note that only types of sort 0 can go into the stoup; reusable types of other sorts would not preserve the sequent antecedent-succedent sort equality under contraction or expansion: $0 + 0 = 0$, but $i + i \neq i$ for $i > 0$.

$T(A_1), \ldots, T(A_n)$ respectively, and the succedent type A is labelled by a term of type $T(A)$ with free variables drawn from $x_1, \ldots, x_n$. In this section we give the semantically labelled Gentzen sequent rules for the connectives of **DA1S4b!$_\mathbf{b}$?**, and indicate some linguistic applications.

1. $$\frac{\zeta_1; \Gamma \Rightarrow B\colon \psi \qquad \Xi(\zeta_2; \Delta\langle \overrightarrow{C}\colon z\rangle) \Rightarrow D\colon \omega}{\Xi(\zeta_1 \uplus \zeta_2; \Delta\langle \overrightarrow{C/B}\colon x, \Gamma\rangle) \Rightarrow D\colon \omega\{(x\ \psi)/z\}}\ /L \qquad \frac{\zeta; \Gamma, \overrightarrow{B}\colon y \Rightarrow C\colon \chi}{\zeta; \Gamma \Rightarrow C/B\colon \lambda y \chi}\ /R$$

2. $$\frac{\zeta_1; \Gamma \Rightarrow A\colon \phi \qquad \Xi(\zeta_2; \Delta\langle \overrightarrow{C}\colon z\rangle) \Rightarrow D\colon \omega}{\Xi(\zeta_1 \uplus \zeta_2; \Delta\langle \Gamma, \overrightarrow{A\backslash C}\colon y\rangle) \Rightarrow D\colon \omega\{(y\ \phi)/z\}}\ \backslash L \qquad \frac{\zeta; \overrightarrow{A}\colon x, \Gamma \Rightarrow C\colon \chi}{\zeta; \Gamma \Rightarrow A\backslash C\colon \lambda x \chi}\ \backslash R$$

3. $$\frac{\Xi\langle \overrightarrow{A}\colon x, \overrightarrow{B}\colon y\rangle \Rightarrow D\colon \omega}{\Xi\langle \overrightarrow{A \bullet B}\colon z\rangle \Rightarrow D\colon \omega\{\pi_1 z/x, \pi_2 z/y\}}\ \bullet L \qquad \frac{\zeta_1; \Delta \Rightarrow A\colon \phi \qquad \zeta_2; \Gamma \Rightarrow B\colon \psi}{\zeta_1 \uplus \zeta_2; \Delta, \Gamma \Rightarrow A \bullet B\colon (\phi, \psi)}\ \bullet R$$

4. $$\frac{\Xi\langle \Lambda\rangle \Rightarrow A\colon \phi}{\Xi\langle \overrightarrow{I}\colon x\rangle \Rightarrow A\colon \phi}\ IL \qquad \frac{}{\emptyset; \Lambda \Rightarrow I\colon 0}\ IR$$

Figure 1: Lambek multiplicatives

The continuous multiplicatives, the Lambek connectives of Lambek (1958[19]; 1988[18]), Figure 1, defined in relation to concatenation/appending, are the basic means of categorial categorization and subcategorization. Note that here and throughout the active types in antecedents are figures (vectorial) whereas those in succedents are not; intuitively this is because antecedents are structured but succedents are not. The directional divisions over, /, and under, \, are exemplified by assignments such as *the*: N/CN for *the man*: N and *sings*: $N\backslash S$ for *John sings*: S, and *loves*: $(N\backslash S)/N$ for *John loves Mary*: S.

The discontinuous multiplicatives of Figure 2, the displacement connectives, Morrill and Valentín (2010[38]), Morrill et al. (2011[46]), are defined in relation to intercalation/plugging. When the value of the k subindex indicates the first (leftmost) point of discontinuity it may be omitted, i.e. it defaults to 1. Circumfixation, $\uparrow$, is exemplified by a discontinuous particle verb assignment such as *calls*+1+*up*: $(N\backslash S)\uparrow N$ for *Mary calls John up*: S, and infixation, $\downarrow$, and circumfixation together are exemplified by a quantifier phrase assignment of the form *everyone*: $(S\uparrow N)\downarrow S$ simulating Montague's S14 treatment of quantifying in; see Section 5.

In relation to the multiplicative rules, notice how the stoup is distributed reading bottom-up from conclusions to premise: it is partitioned between the two premises in the case of binary rules, copied to the premise in the case of unary rules, and empty in the case of nullary rules (axioms).

The additives of Figure 3, Lambek (1961[17]), Morrill (1990[28]), Kanazawa (1992[14]), have application to polymorphism. For example the additive conjunc-

5. $$\frac{\zeta_1; \Gamma \Rightarrow B\colon \psi \qquad \Xi(\zeta_2; \Delta\langle \overrightarrow{C}\colon z\rangle) \Rightarrow D\colon \omega}{\Xi(\zeta_1 \uplus \zeta_2; \Delta\langle \overrightarrow{C{\uparrow_k}B}\colon x\,|_k\,\Gamma\rangle) \Rightarrow D\colon \omega\{(x\,\psi)/z\}}\uparrow_k L \qquad \frac{\zeta; \Gamma\,|_k\,\overrightarrow{B}\colon y \Rightarrow C\colon \chi}{\zeta; \Gamma \Rightarrow C{\uparrow_k}B\colon \lambda y\chi}\uparrow_k R$$

6. $$\frac{\zeta_1; \Gamma \Rightarrow A\colon \phi \qquad \Xi(\zeta_2; \Delta\langle \overrightarrow{C}\colon z\rangle) \Rightarrow D\colon \omega}{\Xi(\zeta_1 \uplus \zeta_2; \Delta\langle \Gamma\,|_k\,\overrightarrow{A{\downarrow_k}C}\colon y\rangle) \Rightarrow D\colon \omega\{(y\,\phi)/z\}}\downarrow_k L \qquad \frac{\zeta; \overrightarrow{A}\colon x\,|_k\,\Gamma \Rightarrow C\colon \chi}{\zeta; \Gamma \Rightarrow A{\downarrow_k}C\colon \lambda x\chi}\downarrow_k R$$

7. $$\frac{\Xi\langle \overrightarrow{A}\colon x\,|_k\,\overrightarrow{B}\colon y\rangle \Rightarrow D\colon \omega}{\Xi\langle \overrightarrow{A{\odot_k}B}\colon z\rangle \Rightarrow D\colon \omega\{\pi_1 z/x, \pi_2 z/y\}}\odot_k L \qquad \frac{\zeta_1; \Delta \Rightarrow A\colon \phi \qquad \zeta_2; \Gamma \Rightarrow B\colon \psi}{\zeta_1 \uplus \zeta_2; \Delta\,|_k\,\Gamma \Rightarrow A{\odot_k}B\colon (\phi, \psi)}\odot_k R$$

8. $$\frac{\Xi\langle 1\rangle \Rightarrow A\colon \phi}{\Xi\langle \overrightarrow{J}\colon x\rangle \Rightarrow A\colon \phi}JL \qquad \frac{}{\emptyset; 1 \Rightarrow J\colon 0}JR$$

Figure 2: Displacement multiplicatives

9. $$\frac{\Xi\langle \overrightarrow{A}\colon x\rangle \Rightarrow C\colon \chi}{\Xi\langle \overrightarrow{A\&B}\colon z\rangle \Rightarrow C\colon \chi\{\pi_1 z/x\}}\&L_1 \qquad \frac{\Xi\langle \overrightarrow{B}\colon y\rangle \Rightarrow C\colon \chi}{\Xi\langle \overrightarrow{A\&B}\colon z\rangle \Rightarrow C\colon \chi\{\pi_2 z/y\}}\&L_2$$

$$\frac{\Xi \Rightarrow A\colon \phi \qquad \Xi \Rightarrow B\colon \psi}{\Xi \Rightarrow A\&B\colon (\phi, \psi)}\&R$$

10. $$\frac{\Xi\langle \overrightarrow{A}\colon x\rangle \Rightarrow C\colon \chi_1 \qquad \Xi\langle \overrightarrow{B}\colon y\rangle \Rightarrow C\colon \chi_2}{\Xi\langle \overrightarrow{A\oplus B}\colon z\rangle \Rightarrow C\colon z \to x.\chi_1; y.\chi_2}\oplus L$$

$$\frac{\Xi \Rightarrow A\colon \phi}{\Xi \Rightarrow A\oplus B\colon \iota_1\phi}\oplus R_1 \qquad \frac{\Xi \Rightarrow B\colon \psi}{\Xi \Rightarrow A\oplus B\colon \iota_2\psi}\oplus R_2$$

Figure 3: Additives

tion & can be used for *rice*: $N\&CN$ as in *rice grows*: S and *the rice grows*: S,[5] and the additive disjunction $\oplus$ can be used for *is*: $(N\backslash S)/(N\oplus (CN/CN))$ as in *Tully is Cicero*: S and *Tully is humanist*: S. The additive disjunction can be used together with the continuous unit to express the optionality of a complement as in *eats*: $(N\backslash S)/(N\oplus I)$ for *John eats fish*: S and *John eats*: S.[6]

Notice how the stoup is identical in conclusions and premises of additive rules.

[5]Note the computational advantage of this approach over assuming an empty determiner: if empty operators were allowed they could occur any number of times in any positions.

[6]Note the advantage of this over simply listing intransitive and transitive lexical entries: empirically the latter does not capture the generalisation that in both cases the verb *eats* combines with a subject to the left, and computationally every lexical ambiguity doubles the lexical insertion search space. Appeal to lexical ambiguity constitutes resignation from the capture of generalisations and is at best a promissory solution, unless there is true ambiguity.

$$11. \quad \frac{\Xi\langle\overrightarrow{A[t/v]}\colon x\rangle \Rightarrow B\colon \psi}{\Xi\langle\overrightarrow{\bigwedge vA}\colon z\rangle \Rightarrow B\colon \psi\{(z\ t)/x\}} \bigwedge L \qquad \frac{\Xi \Rightarrow A[a/v]\colon \phi}{\Xi \Rightarrow \bigwedge vA\colon \lambda v\phi} \bigwedge R^{\dagger}$$

$$12. \quad \frac{\Xi\langle\overrightarrow{A[a/v]}\colon x\rangle \Rightarrow B\colon \psi}{\Xi\langle\overrightarrow{\bigvee vA}\colon z\rangle \Rightarrow B\colon \psi\{\pi_2 z/x\}} \bigvee L^{\dagger} \qquad \frac{\Xi \Rightarrow A[t/v]\colon \phi}{\Xi \Rightarrow \bigvee vA\colon (t, \phi)} \bigvee R$$

Figure 4: Quantifiers, where † indicates that there is no a in the conclusion

The quantifiers of Figure 4, Morrill (1994[47]), have application to features. For example, singular and plural number in *sheep*: $\bigwedge nCNn$ for *the sheep grazes*: S and *the sheep graze*: S. And for a past, present or future tense finite sentence complement we can have *said*: $(N\backslash S)/\bigvee tSf(t)$ in *John said Mary walked*: S, *John said Mary walks*: S and *John said Mary will walk*: S.

Notice how the stoup is identical in conclusion and premise in each quantifier rule.

$$13. \quad \frac{\Xi\langle\overrightarrow{A}\colon x\rangle \Rightarrow B\colon \psi}{\Xi\langle\overrightarrow{\Box A}\colon z\rangle \Rightarrow B\colon \psi\{^{\vee}z/x\}} \Box L \qquad \frac{\boxtimes\Xi \Rightarrow A\colon \phi}{\boxtimes\Xi \Rightarrow \Box A\colon {}^{\wedge}\phi} \Box R$$

$$14. \quad \frac{\boxtimes\Xi\langle\overrightarrow{A}\colon x\rangle \Rightarrow \diamond\!\!\!+ B\colon \psi}{\boxtimes\Xi\langle\overrightarrow{\Diamond A}\colon z\rangle \Rightarrow \diamond\!\!\!+ B\colon \psi\{^{\cup}z/x\}} \Diamond L \qquad \frac{\Xi \Rightarrow A\colon \phi}{\Xi \Rightarrow \Diamond A\colon {}^{\cap}\phi} \Diamond R$$

Figure 5: Normal modalities, where $\boxtimes/\diamond\!\!\!+$ marks a structure all the types of which have main connective a box/diamond

With respect to the (**S4**) normal modalities of Figure 5, the universal (Morrill 1990[29]) has application to intensionality. For example, for a propositional attitude verb such as *believes* we can assign type $\Box((N\backslash S)/\Box S)$ with a modality outermost since the word has a sense, and a modality on the first argument but not the second, since the sentential complement is an intensional domain, but not the subject. The modalities are in the categorial type, distinctly from, but in relation to, the logical interpretation of the propositional attitude verb. The $\Box$ Right rule is semantically interpreted by intensionalisation $^{\wedge}$ and the $\Box$ Left rule is semantically interpreted by extensionalisation $^{\vee}$ in such a way that the Curry-Howard correspondence for the modality yields the law of down-up cancellation (Dowty et al. 1981[7]): $^{\vee\wedge}\phi = \phi$.

Notice how the stoup is identical in conclusion and premise in each normal modality rule.

The bracket modalities of Figure 6, Morrill (1992[30]) and Moortgat 1995[23]), have application to nonassociativity and syntactical domains such as extraction

15. $$\frac{\Xi\langle\overrightarrow{A}: x\rangle \Rightarrow B: \psi}{\Xi\langle[\overrightarrow{[\,]^{-1}A}: x]\rangle \Rightarrow B: \psi}\,[\,]^{-1}L \qquad \frac{[\Xi] \Rightarrow A: \phi}{\Xi \Rightarrow [\,]^{-1}A: \phi}\,[\,]^{-1}R$$

16. $$\frac{\Xi\langle[\overrightarrow{A}: x]\rangle \Rightarrow B: \psi}{\Xi\langle\overrightarrow{\langle\rangle A}: x\rangle \Rightarrow B: \psi}\,\langle\rangle L \qquad \frac{\Xi \Rightarrow A: \phi}{[\Xi] \Rightarrow \langle\rangle A: \phi}\,\langle\rangle R$$

Figure 6: Bracket modalities

islands and prosodic phrases. For example, single bracketing for weak islands: *walks*: $\langle\rangle N\backslash S$ for the subject condition, and *without*: $[\,]^{-1}(VP\backslash VP)/VP$ for the adverbial island constraint; and double bracketing for strong islands of the kind *and*: $(S\backslash[\,]^{-1}[\,]^{-1}S)/S$ for the coordinate structure constraint.

Notice how the stoup is identical in conclusions and premises of bracket modality rules.

17. $$\frac{\Xi(\zeta \uplus \{A: x\}; \Gamma_1, \Gamma_2) \Rightarrow B: \psi}{\Xi(\zeta; \Gamma_1, !A: x, \Gamma_2) \Rightarrow B: \psi}\,!L \qquad \frac{\zeta; \Lambda \Rightarrow A: \phi}{\zeta; \Lambda \Rightarrow !A: \phi}\,!R$$

$$\frac{\Xi(\zeta; \Gamma_1, A: x, \Gamma_2) \Rightarrow B: \psi}{\Xi(\zeta \uplus \{A: x\}; \Gamma_1, \Gamma_2) \Rightarrow B: \psi}\,!P$$

$$\frac{\Xi(\zeta_1 \uplus \zeta_2 \uplus \{A: x\}; \Gamma_1, [\zeta_2 \uplus \{A: y\}; \Gamma_2], \Gamma_3) \Rightarrow B: \psi}{\Xi(\zeta_1 \uplus \zeta_2 \uplus \{A: x\}; \Gamma_1, \Gamma_2, \Gamma_3) \Rightarrow B: \psi\{x/y\}}\,!C$$

18. $$\frac{\Xi(A: x_1) \Rightarrow B: \psi([x_1]) \qquad \Xi(A: x_1, A: x_2) \Rightarrow B: \psi([x_1, x_2]) \qquad \cdots}{\Xi(?A: x) \Rightarrow B: \psi(x)}\,?L$$

$$\frac{\Xi \Rightarrow A: \phi}{\Xi \Rightarrow ?A: [\phi]}\,?R \qquad \frac{\zeta; \Gamma \Rightarrow A: \phi \qquad \zeta'; \Delta \Rightarrow ?A: \psi}{\zeta \uplus \zeta'; \Gamma, \Delta \Rightarrow ?A: [\phi|\psi]}\,?M$$

Figure 7: Exponentials

Finally, there is nonlinearity. The universal exponential of Figure 7, Girard (1987[10]), Barry, Hepple, Leslie and Morrill (1991[4]), Morrill (1994[47]), Morrill and Valentín (2015[41]), and Morrill (2017[35]), has application to extraction including parasitic extraction. In the formulation here $!L$ moves the operand of a universal exponential (e.g. the hypothetical subtype of relativisation) into the stoup, where it will percolate as commented for the above rules. From there it can be copied into the stoup of a newly-created bracketed domain by the contraction rule $!C$ (producing a parasitic gap), and it can be moved into any position in the matrix configuration of its zone by $!P$ (producing a nonparasitic or host gap).

Using the universal exponential, !, for which contraction induces island brackets, we can assign a relative pronoun type *that*: $(CN\backslash CN)/(S/!N)$ allowing parasitic extraction such as *paper that John filed without reading*: CN, where parasitic gaps can appear only in (weak) islands, but can be iterated in subislands, for example, *man who the fact that the friends of admire without praising surprises*. Crucially, in the linguistic formulation ! does not have weakening, i.e. deletion, since, e.g., the body of a relative clause *must* contain a gap: **man who John loves Mary*.

The existential exponential ? has application to iterated coordination (Morrill 1994[47]; Morrill and Valentín 2015[41]) and (unboundedly iterated) *respectively* (Morrill and Valentín 2016[44]). Using the existential exponential, ?, we can assign a coordinator type *and*: $(?N\backslash N)/N$ allowing iterated coordination as in *John, Bill, Mary and Suzy*: N, or *and*: $(?(S/N)\backslash(S/N))/(S/N)$ for *John likes Mary dislikes, and Bill hates, London* (iterated right node raising), and so on.

In relation to the rest of the primary connectives: the limited contraction | of Jäger (2005[13]) has application to anaphora and the limited weakening W of Morrill and Valentín (2014[40]) has application to words as types. The remaining, semantically inactive, connectives listed here were introduced as follows. Semantically inactive multiplicatives {•−, −∘, ∘−, −•, ◐, ◑, ⫯, ⫰, ⫯, ⫰, ◓, ◒}: Morrill and Valentín (2014[40]). Semantically inactive additives {$\sqcap$, $\sqcup$}: Morrill (1994[47]). Semantically inactive first-order quantifiers {$\forall$, $\exists$}: Morrill (1994[47]). Semantically inactive normal modalities {$\blacksquare$, $\blacklozenge$}: Hepple (1990[12]), Morrill (1994[47]). The rules for semantically inactive variants are the same as those for the semantically active versions syntactically, but have the same label on premises and conclusions semantically.[7]

3 Focusing

Spurious ambiguity is the phenomenon whereby distinct derivations in grammar may assign the same structural reading, resulting in redundancy in the parse search space and inefficiency in parsing. Understanding the problem depends on identifying the essential mathematical structure of derivations. This is trivial in the case of context free grammar, where the parse structures are ordered trees; in the case of type logical categorial grammar, the parse structures are proof nets. However, with respect to multiplicatives intrinsic proof nets have not yet been given for displacement calculus (but see Morrill and Fadda (2008[36], Fadda 2010[8], and Moot 2014[25], 2016[26]) In this context CatLog3 approaches spurious ambiguity by means of Andreoli's (1982[2]) proof-theoretic technique of focalisation, which engenders a substantial reduction of spurious ambiguity.

[7]The synthetic connectives are: left and right projection and injection {$\triangleleft^{-1}$, $\triangleright^{-1}$, $\triangleleft$, $\triangleright$}, Morrill, Fadda and Valentín (2009[45]); split and bridge {ˇ, ˆ}, Morrill and Merenciano (1996[37]); continuous and discontinuous nondeterministic multiplicatives {$\div$, $\times$, $\Uparrow$, $\Downarrow$, $\circledcirc$}, Morrill, Valentín and Fadda (2011[46]). The difference operator − of Morrill and Valentín (2014[39]) has application to linguistic exceptions.

$$\frac{\overrightarrow{A}\colon x, \Gamma \Rightarrow C\colon \chi \quad \neg\ \mathbf{foc}}{\Gamma \Rightarrow A\backslash C\colon \lambda x\chi \quad \neg\ \mathbf{foc} \wedge\ \mathbf{rev}}\backslash R \qquad \frac{\Gamma, \overrightarrow{B}\colon y \Rightarrow C\colon \chi \quad \neg\ \mathbf{foc}}{\Gamma \Rightarrow C/B\colon \lambda y\chi \quad \neg\ \mathbf{foc} \wedge\ \mathbf{rev}}/R$$

$$\frac{\Delta\langle \overrightarrow{A}\colon x, \overrightarrow{B}\colon y\rangle \Rightarrow D\colon \omega \quad \neg\ \mathbf{foc}}{\Delta\langle \overrightarrow{A\bullet B}\colon z\rangle \Rightarrow D\colon \omega\{\pi_1 z/x, \pi_2 z/y\} \quad \neg\ \mathbf{foc} \wedge\ \mathbf{rev}}\bullet L$$

$$\frac{\Delta\langle \Lambda\rangle \Rightarrow A\colon \phi \quad \neg\ \mathbf{foc}}{\Delta\langle \overrightarrow{I}\colon x\rangle \Rightarrow A\colon \phi \quad \neg\ \mathbf{foc} \wedge\ \mathbf{rev}} IL$$

$$\frac{\overrightarrow{A}\colon x|_k \Gamma \Rightarrow C\colon \chi \quad \neg\ \mathbf{foc}}{\Gamma \Rightarrow A{\downarrow_k}C\colon \lambda x\chi \quad \neg\ \mathbf{foc} \wedge\ \mathbf{rev}}{\downarrow_k}R \qquad \frac{\Gamma|_k \overrightarrow{B}\colon y \Rightarrow C\colon \chi \quad \neg\ \mathbf{foc}}{\Gamma \Rightarrow C{\uparrow_k}B\colon \lambda y\chi \quad \neg\ \mathbf{foc} \wedge\ \mathbf{rev}}{\uparrow_k}R$$

$$\frac{\Delta\langle \overrightarrow{A}\colon x|_k \overrightarrow{B}\colon y\rangle \Rightarrow D\colon \omega \quad \neg\ \mathbf{foc}}{\Delta\langle \overrightarrow{A\odot_k B}\colon z\rangle \Rightarrow D\colon \omega\{\pi_1 z/x, \pi_2 z/y\} \quad \neg\ \mathbf{foc} \wedge\ \mathbf{rev}}\odot_k L$$

$$\frac{\Delta\langle 1\rangle \Rightarrow A\colon \phi \quad \neg\ \mathbf{foc}}{\Delta\langle \overrightarrow{J}\colon x\rangle \Rightarrow A\colon \phi \quad \neg\ \mathbf{foc} \wedge\ \mathbf{rev}} JL$$

Figure 8: Reversible multiplicative rules

In focalisation, *situated* (in the antecedent of a sequent, input, $^\bullet$/ in the succedent of a sequent, output, $^\circ$) non-atomic types are classified as of *reversible/negative* or *irreversible/positive polarity* according as their associated rule is reversible or not. There are alternating phases of don't-care nondeterministic negative rule application, and positive rule application locking on to *focalised* formulas. Given a sequent with no occurrences of negative formulas, one chooses a positive formula as principal formula (which is boxed; we say it is focalised) and applies proof search to its subformulas while these remain positive. When one finds a negative formula or a literal, invertible rules are applied in a don't care nondeterminitic fashion until no longer possible, when another positive formula is chosen, and so on. CatLog3 can be set to focus all atoms in the input (as in the example at the end) or in the output, i.e. it implements uniform *bias*.

A sequent is either unfocused and as before, or else focused and has exactly one type boxed. This is the focused type. The focalised logical rules for displacement calculus are given in Figures 8–12. Sequents are accompanied by *judgements*: focalised or not focalised and reversible or not reversible.[8] The completeness of this focalisation, together with additives, is proved in Morrill and Valentín (2015[42]). The completeness of focalisation for other connectives of CatLog3 is a topic of ongoing research.

[8]This idea is due to Oriol Valentín.

$$\frac{\Gamma \Rightarrow \boxed{P}\colon \phi \quad \textbf{foc} \wedge \neg\, \textbf{rev} \qquad \Delta\langle \overrightarrow{\boxed{Q}}\colon z\rangle \Rightarrow D\colon \omega \quad \textbf{foc} \wedge \neg\, \textbf{rev}}{\Delta\langle \Gamma, \overrightarrow{\boxed{P\backslash Q}}\colon y\rangle \Rightarrow D\colon \omega\{(y\ \phi)/z\} \quad \textbf{foc} \wedge \neg\, \textbf{rev}}\ \backslash L$$

$$\frac{\Gamma \Rightarrow \boxed{P_1}\colon \phi \quad \textbf{foc} \wedge \neg\, \textbf{rev} \qquad \Delta\langle \overrightarrow{P_2}\colon z\rangle \Rightarrow D\colon \omega \quad \neg\, \textbf{foc} \wedge ?P_2\ \textbf{rev}}{\Delta\langle \Gamma, \overrightarrow{\boxed{P_1\backslash P_2}}\colon y\rangle \Rightarrow D\colon \omega\{(y\ \phi)/z\} \quad \textbf{foc} \wedge \neg\, \textbf{rev}}\ \backslash L$$

$$\frac{\Gamma \Rightarrow Q_1\colon \phi \quad \neg\, \textbf{foc} \wedge ?Q_1\ \textbf{rev} \qquad \Delta\langle \overrightarrow{\boxed{Q_2}}\colon z\rangle \Rightarrow D\colon \omega \quad \textbf{foc} \wedge \neg\, \textbf{rev}}{\Delta\langle \Gamma, \overrightarrow{\boxed{Q_1\backslash Q_2}}\colon y\rangle \Rightarrow D\colon \omega\{(y\ \phi)/z\} \quad \textbf{foc} \wedge \neg\, \textbf{rev}}\ \backslash L$$

$$\frac{\Gamma \Rightarrow Q\colon \phi \quad \neg\, \textbf{foc} \wedge ?Q\ \textbf{rev} \qquad \Delta\langle \overrightarrow{P}\colon z\rangle \Rightarrow D\colon \omega \quad \neg\, \textbf{foc} \wedge ?P\ \textbf{rev}}{\Delta\langle \Gamma, \overrightarrow{\boxed{Q\backslash P}}\colon y\rangle \Rightarrow D\colon \omega\{(y\ \phi)/z\} \quad \textbf{foc} \wedge \neg\, \textbf{rev}}\ \backslash L$$

$$\frac{\Gamma \Rightarrow \boxed{P}\colon \psi \quad \textbf{foc} \wedge \neg\, \textbf{rev} \qquad \Delta\langle \overrightarrow{\boxed{Q}}\colon z\rangle \Rightarrow D\colon \omega \quad \textbf{foc} \wedge \neg\, \textbf{rev}}{\Delta\langle \overrightarrow{\boxed{Q/P}}\colon x, \Gamma\rangle \Rightarrow D\colon \omega\{(x\ \psi)/z\} \quad \textbf{foc} \wedge \neg\, \textbf{rev}}\ /L$$

$$\frac{\Gamma \Rightarrow Q_1\colon \psi \quad \neg\, \textbf{foc} \wedge ?Q_1\ \textbf{rev} \qquad \Delta\langle \overrightarrow{\boxed{Q_2}}\colon z\rangle \Rightarrow D\colon \omega \quad \textbf{foc} \wedge \neg\, \textbf{rev}}{\Delta\langle \overrightarrow{\boxed{Q_2/Q_1}}\colon x, \Gamma\rangle \Rightarrow D\colon \omega\{(x\ \psi)/z\} \quad \textbf{foc} \wedge \neg\, \textbf{rev}}\ /L$$

$$\frac{\Gamma \Rightarrow \boxed{P_1}\colon \psi \quad \textbf{foc} \wedge \neg\, \textbf{rev} \qquad \Delta\langle \overrightarrow{P_2}\colon z\rangle \Rightarrow D\colon \omega \quad \neg\, \textbf{foc} \wedge ?P_2\ \textbf{rev}}{\Delta\langle \overrightarrow{\boxed{P_2/P_1}}\colon x, \Gamma\rangle \Rightarrow D\colon \omega\{(x\ \psi)/z\} \quad \textbf{foc} \wedge \neg\, \textbf{rev}}\ /L$$

$$\frac{\Gamma \Rightarrow Q\colon \psi \quad \neg\, \textbf{foc} \wedge ?Q\ \textbf{rev} \qquad \Delta\langle \overrightarrow{P}\colon z\rangle \Rightarrow D\colon \omega \quad \neg\, \textbf{foc} \wedge ?P\ \textbf{rev}}{\Delta\langle \overrightarrow{\boxed{P/Q}}\colon x, \Gamma\rangle \Rightarrow D\colon \omega\{(x\ \psi)/z\} \quad \textbf{foc} \wedge \neg\, \textbf{rev}}\ /L$$

Figure 9: Left irreversible continuous multiplicative rules

$$\frac{\Gamma \Rightarrow \boxed{P}\colon \phi \quad \mathbf{foc} \wedge \neg\, \mathbf{rev} \qquad \Delta\langle \overrightarrow{\boxed{Q}}\colon z\rangle \Rightarrow D\colon \omega \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}{\Delta\langle \Gamma |_k \overrightarrow{\boxed{P{\downarrow_k}Q}}\colon y\rangle \Rightarrow D\colon \omega\{(y\ \phi)/z\} \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}\ {\downarrow_k}L$$

$$\frac{\Gamma \Rightarrow \boxed{P_1}\colon \phi \quad \mathbf{foc} \wedge \neg\, \mathbf{rev} \qquad \Delta\langle \overrightarrow{P_2}\colon z\rangle \Rightarrow D\colon \omega \quad \neg\, \mathbf{foc} \wedge ?P_2\ \mathbf{rev}}{\Delta\langle \Gamma |_k \overrightarrow{\boxed{P_1{\downarrow_k}P_2}}\colon y\rangle \Rightarrow D\colon \omega\{(y\ \phi)/z\} \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}\ {\downarrow_k}L$$

$$\frac{\Gamma \Rightarrow Q_1\colon \phi \quad \neg\, \mathbf{foc} \wedge ?Q_1\ \mathbf{rev} \qquad \Delta\langle \overrightarrow{\boxed{Q_2}}\colon z\rangle \Rightarrow D\colon \omega \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}{\Delta\langle \Gamma |_k \overrightarrow{\boxed{Q_1{\downarrow_k}Q_2}}\colon y\rangle \Rightarrow D\colon \omega\{(y\ \phi)/z\} \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}\ {\downarrow_k}L$$

$$\frac{\Gamma \Rightarrow Q\colon \phi \quad \neg\, \mathbf{foc} \wedge ?Q\ \mathbf{rev} \qquad \Delta\langle \overrightarrow{P}\colon z\rangle \Rightarrow D\colon \omega \quad \neg\, \mathbf{foc} \wedge ?P\ \mathbf{rev}}{\Delta\langle \Gamma |_k \overrightarrow{\boxed{Q{\downarrow_k}P}}\colon y\rangle \Rightarrow D\colon \omega\{(y\ \phi)/z\} \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}\ {\downarrow_k}L$$

$$\frac{\Gamma \Rightarrow \boxed{P}\colon \psi \quad \mathbf{foc} \wedge \neg\, \mathbf{rev} \qquad \Delta\langle \overrightarrow{\boxed{Q}}\colon z\rangle \Rightarrow D\colon \omega \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}{\Delta\langle \overrightarrow{\boxed{Q{\uparrow_k}P}}\colon x |_k \Gamma\rangle \Rightarrow D\colon \omega\{(x\ \psi)/z\} \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}\ {\uparrow_k}L$$

$$\frac{\Gamma \Rightarrow Q_1\colon \psi \quad \neg\, \mathbf{foc} \wedge ?Q_1\ \mathbf{rev} \qquad \Delta\langle \overrightarrow{\boxed{Q_2}}\colon z\rangle \Rightarrow D\colon \omega \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}{\Delta\langle \overrightarrow{\boxed{Q_2{\uparrow_k}Q_1}}\colon x |_k \Gamma\rangle \Rightarrow D\colon \omega\{(x\ \psi)/z\} \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}\ {\uparrow_k}L$$

$$\frac{\Gamma \Rightarrow \boxed{P_1}\colon \psi \quad \mathbf{foc} \wedge \neg\, \mathbf{rev} \qquad \Delta\langle \overrightarrow{P_2}\colon z\rangle \Rightarrow D\colon \omega \quad \neg\, \mathbf{foc} \wedge ?P_2\ \mathbf{rev}}{\Delta\langle \overrightarrow{\boxed{P_2{\uparrow_k}P_1}}\colon x |_k \Gamma\rangle \Rightarrow D\colon \omega\{(x\ \psi)/z\} \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}\ {\uparrow_k}L$$

$$\frac{\Gamma \Rightarrow Q\colon \psi \quad \neg\, \mathbf{foc} \wedge ?Q\ \mathbf{rev} \qquad \Delta\langle \overrightarrow{P}\colon z\rangle \Rightarrow D\colon \omega \quad \neg\, \mathbf{foc} \wedge ?P\ \mathbf{rev}}{\Delta\langle \overrightarrow{\boxed{P{\uparrow_k}Q}}\colon x |_k \Gamma\rangle \Rightarrow D\colon \omega\{(x\ \psi)/z\} \quad \mathbf{foc} \wedge \neg\, \mathbf{rev}}\ {\uparrow_k}L$$

Figure 10: Left irreversible discontinuous multiplicative rules

$$\frac{\Delta \Rightarrow \boxed{P_1}\colon \phi \quad \textbf{foc} \wedge \neg\, \textbf{rev} \qquad \Gamma \Rightarrow \boxed{P_2}\colon \psi \quad \textbf{foc} \wedge \neg\, \textbf{rev}}{\Delta, \Gamma \Rightarrow \boxed{P_1 \bullet P_2}\colon (\phi, \psi) \quad \textbf{foc} \wedge \neg\, \textbf{rev}} \bullet R$$

$$\frac{\Delta \Rightarrow \boxed{P}\colon \phi \quad \textbf{foc} \wedge \neg\, \textbf{rev} \qquad \Gamma \Rightarrow Q\colon \psi \quad \neg\, \textbf{foc} \wedge ?Q\ \textbf{rev}}{\Delta, \Gamma \Rightarrow \boxed{P \bullet Q}\colon (\phi, \psi) \quad \textbf{foc} \wedge \neg\, \textbf{rev}} \bullet R$$

$$\frac{\Delta \Rightarrow N\colon \phi \quad \neg\, \textbf{foc} \wedge ?N\ \textbf{rev} \qquad \Gamma \Rightarrow \boxed{P}\colon \psi \quad \textbf{foc} \wedge \neg\, \textbf{rev}}{\Delta, \Gamma \Rightarrow \boxed{N \bullet P}\colon (\phi, \psi) \quad \textbf{foc} \wedge \neg\, \textbf{rev}} \bullet R$$

$$\frac{\Delta \Rightarrow N_1\colon \phi \neg\, \textbf{foc} \wedge ?N_1\ \textbf{rev} \qquad \Gamma \Rightarrow N_2\colon \psi \quad \textbf{foc} \wedge ?N_2\ \textbf{rev}}{\Delta, \Gamma \Rightarrow \boxed{N_1 \bullet N_2}\colon (\phi, \psi) \quad \textbf{foc} \wedge \neg\, \textbf{rev}} \bullet R$$

$$\frac{}{\Lambda \Rightarrow \boxed{I}\colon 0\ \textbf{foc} \wedge \neg\, \textbf{rev}} IR$$

Figure 11: Right irreversible continuous multiplicative rules

$$\frac{\Delta \Rightarrow \boxed{P_1}\colon \phi \quad \textbf{foc} \wedge \neg\, \textbf{rev} \qquad \Gamma \Rightarrow \boxed{P_2}\colon \psi \quad \textbf{foc} \wedge \neg\, \textbf{rev}}{\Delta \,|_k\, \Gamma \Rightarrow \boxed{P_1 \odot_k P_2}\colon (\phi, \psi) \quad \textbf{foc} \wedge \neg\, \textbf{rev}} \odot_k R$$

$$\frac{\Delta \Rightarrow \boxed{P}\colon \phi \quad \textbf{foc} \wedge \neg\, \textbf{rev} \qquad \Gamma \Rightarrow Q\colon \psi \quad \neg\, \textbf{foc} \wedge ?N\ \textbf{rev}}{\Delta \,|_k\, \Gamma \Rightarrow \boxed{P \odot_k Q}\colon (\phi, \psi) \quad \textbf{foc} \wedge \neg\, \textbf{rev}} \odot_k R$$

$$\frac{\Delta \Rightarrow Q\colon \phi \quad \neg\, \textbf{foc} \wedge ?N\ \textbf{rev} \qquad \Gamma \Rightarrow \boxed{P}\colon \psi \quad \textbf{foc} \wedge \neg\, \textbf{rev}}{\Delta \,|_k\, \Gamma \Rightarrow \boxed{Q \odot_k P}\colon (\phi, \psi) \quad \textbf{foc} \wedge \neg\, \textbf{rev}} \odot_k R$$

$$\frac{\Delta \Rightarrow Q_1\colon \phi \quad \neg\, \textbf{foc} \wedge ?Q_1\ \textbf{rev} \qquad \Gamma \Rightarrow Q_2\colon \psi \quad \neg\, \textbf{foc} \wedge ?Q_2\ \textbf{rev}}{\Delta \,|_k\, \Gamma \Rightarrow \boxed{Q_1 \odot_k Q_2}\colon (\phi, \psi) \quad \textbf{foc} \wedge \neg\, \textbf{rev}} \odot_k R$$

$$\frac{}{1 \Rightarrow \boxed{J}\colon 0\ \textbf{foc} \wedge \neg\, \textbf{rev}} JR$$

Figure 12: Right irreversible discontinuous multiplicative rules

4 Count-invariance

We define infinitary count invariance for categorial logic extending count invariance for multiplicatives (van Benthem 1991[51]) and additives and bracket modalities (Valentín et al. 2013[50]) to include exponentials. This affords effective pruning of proof search in categorial parsing/theorem-proving.

Count invariance for multiplicatives in (sub)linear logic is introduced in van Benthem (1991[51]). This involves simply checking the number of positive and negative occurrences of each atom in a sequent. Thus where $\#(\Sigma)$ is a count of the sequent Σ we have:

(5) $\vdash \Sigma \Longrightarrow \#(\Sigma) = 0$

I.e. the numbers of positive and negative occurrences of each atom must exactly balance. This provides a necessary, but of course not sufficient, criterion for theoremhood, and it can be checked rapidly. It can be used as a filter in proof search: if backward chaining proof search generates a goal which does not satisfy the count invariant, the goal can be safely made to fail immediately. This notion of count for multiplicatives was included in the categorial parser/theorem-prover CatLog1 (Morrill 2012[33]).

In Valentín et al. (2013[50]) the idea is extended to additives (and bracket modalities). Instead of a single count for each atom of a sequent Σ we have a minimum count $\#_{\mathrm{min}}(\Sigma)$ and a maximum count $\#_{\mathrm{max}}(\Sigma)$ and for a sequent to be a theorem it must satisfy two inequations:

(6) $\vdash \Sigma \Longrightarrow \#_{\mathrm{min}}(\Sigma) \leq 0 \leq \#_{\mathrm{max}}(\Sigma)$

I.e. the count functions $\#_{\mathrm{min}}$ and $\#_{\mathrm{max}}$ define an interval which must include the point of balance 0; for the multiplicatives, $\#_{\mathrm{min}} = \#_{\mathrm{max}} = \#$ and (6) reduces to the special case (5). This count-invariance is included in the categorial parser/theorem-prover CatLog2. Here we describe the count-invariance of CatLog3 which includes further infinitary count functions for exponentials (Kuznetsov et al. 2017[16]).

We consider terms built over the constants 0, 1, $\bot$ (minus infinity, $-\infty$), and $\top$ (plus infinity, $+\infty$) by operations plus (+), minus (−), minimum (min) and maximum (max), and infinitary step functions X and Y thus; $i, j \in \mathcal{Z}$ and $n \in \mathcal{Z}^+$:

$+$	j	$\bot$	$\top$
i	$i+j$	$\bot$	$\top$
$\bot$	$\bot$	$\bot$	$*$
$\top$	$\top$	$*$	$\top$

$-$	j	$\bot$	$\top$
i	$i-j$	$\top$	$\bot$
$\bot$	$\bot$	$*$	$\bot$
$\top$	$\top$	$\top$	$*$

min	j	$\bot$	$\top$
i	$\frac{\lvert i+j\rvert-\lvert i-j\rvert}{2}$	$\bot$	i
$\bot$	$\bot$	$\bot$	$\bot$
$\top$	j	$\bot$	$\top$

max	j	$\bot$	$\top$
i	$\frac{\lvert i+j\rvert+\lvert i-j\rvert}{2}$	i	$\top$
$\bot$	j	$\bot$	$\top$
$\top$	$\top$	$\top$	$\top$

$$X(i) = \begin{cases} \top & \text{if } i > 0 \\ i & \text{if } i \leq 0 \end{cases} \qquad Y(i) = \begin{cases} i & \text{if } i \geq 0 \\ \bot & \text{if } i < 0 \end{cases}$$

Where $\mathcal{P}$ is the set of primitive types, $P \in \mathcal{P}$, $Q \in \mathcal{P} \cup \{[]\}$, $p \in \{{}^{\bullet}, {}^{\circ}\}$, and $\overline{{}^{\bullet}} = {}^{\circ}$ and $\overline{{}^{\circ}} = {}^{\bullet}$ we define the count functions for $\mathbf{DA1S4b!_b?}$ as shown in Figure 13.

For zones, stoups, tree terms and configurations, counts are as follows:

$$\begin{array}{rcl}
\#_{m,Q}(\mathcal{S};\mathcal{O}) & = & \#_{m,Q}(\mathcal{S}) + \#_{m,Q}(\mathcal{O}) \\
\#_{m,Q}(\emptyset) & = & 0 \\
\#_{m,Q}(\mathcal{F},\mathcal{S}) & = & \#_{m,Q}(\mathcal{F}) + \#_{m,Q}(\mathcal{S}) \\
\#_{m,Q}(\Lambda) & = & 0 \\
\#_{m,Q}(\mathcal{T},\mathcal{O}) & = & \#_{m,Q}(\mathcal{T}) + \#_{m,Q}(\mathcal{O}) \\
\#_{m,Q}(1) & = & 0 \\
\#_{m,Q}(\mathcal{F}) & = & \#^{\bullet}_{m,Q}(\mathcal{F}) \\
\#_{m,Q}(\mathcal{F}\{\mathcal{O}_1 : \ldots : \mathcal{O}_i\}) & = & \#^{\bullet}_{m,Q}(\mathcal{F}) + \Sigma^{i}_{n=1}\#_{m,Q}(\mathcal{O}_n) \\
\#_{m,[]}([\mathcal{Z}]) & = & \#_{m,[]}(\mathcal{Z}) + 1 \\
\#_{m,P}([\mathcal{Z}]) & = & \#_{m,P}(\mathcal{Z})
\end{array}$$

The count-invariance theorem is:

(7) **Theorem.**
$\vdash \Xi \Rightarrow A \Longrightarrow \forall Q \in \mathcal{P} \cup \{[]\}, \#_{\min,Q}(\Xi \Rightarrow A) \leq 0 \leq \#_{\max,Q}(\Xi \Rightarrow A)$
where, $\#_{m,Q}(\Xi \Rightarrow A) = \#^{\circ}_{m,Q}(A) - \#_{\overline{m},Q}(\Xi)$.

Relativisation including medial and parasitic extraction is obtained by assigning a relative pronoun a type $(CN\backslash CN)/(!N\backslash S)$ whereby the body of a relative clause is analysed as $!N\backslash S$. By way of example of count-invariance, we show how it discards $N, N\backslash S \Rightarrow\ !N\backslash S$ corresponding to the ungrammaticality of a relative clause without a gap: **paper that John walks*. We have the max N-count: $\#_{\max,N}(N, N\backslash S \Rightarrow\ !N\backslash S) = \#^{\circ}_{\max,N}(!N\backslash S) - \#^{\bullet}_{\min,N}(N, N\backslash S) = \#^{\circ}_{\max.N}(S) - \#^{\bullet}_{\min,N}(!N) - \#^{\bullet}_{\min,N}(N) - \#^{\bullet}_{\min,N}(N\backslash S) = 0 - Y(\#^{\bullet}_{\min,N}(N)) - 1 - \#^{\bullet}_{\min,N}(S) + \#^{\circ}_{\min,N}(N) = -Y(1) - 1 - 0 + 1 = -1 - 1 + 1 = -1 \ngeq 0$ which means that the count-invariance is not satisfied.

Iterated sentential coordination is obtained by assigning a coordinator the type $(?S\backslash S)/S$. By way of a second example we show how count-invariance discards $N, N, N\backslash S \Rightarrow ?S$ corresponding to the ungrammaticality of unequilibrated coordination: **John Mary walks and Suzy talks*. For this example maximum N-count is: $\#_{\max,N}(N, N, N\backslash S \Rightarrow ?S) = \#^{\circ}_{\max,N}(?S) - \#^{\bullet}{}_{\min,N}(N, N, N\backslash S) = X(\#^{\circ}_{\max,N}(S)) - \#^{\bullet}_{\min,N}(N) - \#^{\bullet}_{\min,N}(N) - \#^{\bullet}_{\min,N}(N\backslash S) = X(0) - 1 - 1 - \#^{\bullet}_{\min,N}(S) + \#^{\bullet}_{\max,N}(N) = 0 - 2 - 0 + 1 = -1 \ngeq 0$ which means that the count-invariance is not satisfied.

$$
\begin{array}{rcll}
\#^p_{m,Q}(P) & = & \begin{array}{ll} 1 & \text{if } Q = P \\ 0 & \text{if } Q \neq P \end{array} & \\
\#^p_{m,Q}(A\backslash C) & = & \#^p_{m,Q}(A{\downarrow_k}C) & = \#^p_{m,Q}(C) - \#^{\overline{p}}_{\overline{m},Q}(A) \\
\#^p_{m,Q}(C/B) & = & \#^p_{m,Q}(C{\uparrow_k}B) & = \#^p_{m,Q}(C) - \#^{\overline{p}}_{\overline{m},Q}(B) \\
\#^p_{m,Q}(A\bullet B) & = & \#^p_{m,Q}(A\odot_k B) & = \#^p_{m,Q}(A) + \#^p_{m,Q}(B) \\
\#^p_{m,Q}(I) & = & \#^p_{m,Q}(J) & = 0 \\
\#^{\circ}_{m,Q}(A\&B) & = & \overline{m}(\#^{\circ}_{m,Q}(A), \#^{\circ}_{m,Q}(B)) & \\
\#^{\bullet}_{m,Q}(A\&B) & = & m(\#^{\bullet}_{m,Q}(A), \#^{\bullet}_{m,Q}(B)) & \\
\#^{\circ}_{m,Q}(A\oplus B) & = & m(\#^{\circ}_{m,Q}(A), \#^{\circ}_{m,Q}(B)) & \\
\#^{\bullet}_{m,Q}(A\oplus B) & = & \overline{m}(\#^{\bullet}_{m,Q}(A), \#^{\bullet}_{m,Q}(B)) & \\
\#^p_{m,Q}(\bigwedge xA) & = & \#^p_{m,Q}(\bigvee xA) & = \#^p_{m,Q}(A) \\
\#^p_{m,Q}(\Box A) & = & \#^p_{m,Q}(\Diamond A) & = \#^p_{m,Q}(A) \\
\#^p_{m,P}([\,]^{-1}A) & = & \#^p_{m,P}(A) & \\
\#^p_{m,[]}([\,]^{-1}A) & = & \#^p_{m,[]}(A) - 1 & \\
\#^p_{m,P}(\langle\rangle A) & = & \#^p_{m,P}(A) & \\
\#^p_{m,[]}(\langle\rangle A) & = & \#^p_{m,[]}(A) + 1 & \\
\#^{\bullet}_{\min,Q}(!A) & = & Y(\#^{\bullet}_{\min,Q}(A)) & \\
\#^{\bullet}_{\max,P}(!A) & = & X(\#^{\bullet}_{\max,P}(A)) & \\
\#^{\bullet}_{\max,[]}(!A) & = & \top & \\
\#^{\circ}_{m,Q}(!A) & = & \#^{\circ}_{m,Q}(A) & \\
\#^{\circ}_{\max,Q}(?A) & = & X(\#^{\circ}_{\max,Q}(A)) & \\
\#^{\circ}_{\min,Q}(?A) & = & Y(\#^{\circ}_{\min,Q}(A)) & \\
\#^{\bullet}_{m,Q}(?A) & = & \#^{\bullet}_{m,Q}(A) &
\end{array}
$$

Figure 13: Count function

```
str(dwp('(7-7)'), [b([john]), walks], s(f)).
str(dwp('(7-16)'), [b([every, man]), talks], s(f)).
str(dwp('(7-19)'), [b([the, fish]), walks], s(f)).
str(dwp('(7-32)'), [b([every, man]), b([b([walks, or, talks])])], s(f)).
str(dwp('(7-34)'), [b([b([b([every, man]), walks, or, b([every, man]), talks])])], s(f)).
str(dwp('(7-39)'), [b([b([b([a, woman]), walks, and, b([she]), talks])])], s(f)).
str(dwp('(7-43, 45)'), [b([john]), believes, that, b([a, fish]), walks], s(f)).
str(dwp('(7-48, 49, 52)'), [b([every, man]), believes, that, b([a, fish]), walks], s(f)).
str(dwp('(7-57)'), [b([every, fish, such, that, b([it]), walks]), talks], s(f)).
str(dwp('(7-60, 62)'), [b([john]), seeks, a, unicorn], s(f)).
str(dwp('(7-73)'), [b([john]), is, bill], s(f)).
str(dwp('(7-76)'), [b([john]), is, a, man], s(f)).
str(dwp('(7-83)'), [necessarily, b([john]), walks], s(f)).
str(dwp('(7-86)'), [b([john]), walks, slowly], s(f)).
str(dwp('(7-91)'), [b([john]), tries, to, walk], s(f)).
str(dwp('(7-94)'), [b([john]), tries, to, b([b([catch, a, fish, and, eat, it])])], s(f)).
str(dwp('(7-98)'), [b([john]), finds, a, unicorn], s(f)).
str(dwp('(7-105)'), [b([every, man, such, that, b([he]), loves, a, woman]), loses, her], s(f)).
str(dwp('(7-110)'), [b([john]), walks, in, a, park], s(f)).
str(dwp('(7-116, 118)'), [b([every, man]), doesnt, walk], s(f)).
```

Figure 14: Montague sentences

5 Illustration

Morrill and Valentín (2016[43]) defines as the *Montague Test* the task of providing a computational grammar of the PTQ fragment of Montague (1973[21]), and shows how CatLog fulfils this task. We are not aware of any other system which has passed the Montague Test. The example sentences of Chapter 7 of Dowty et al. (1981[7]) are given in Figure 14; the lexicon is given in Figure 15.

The CatLog3 LaTeX output for the (ambiguous) last sentence is as follows:

(dwp((7-116, 118))) [**every**+**man**]+**doesnt**+**walk** : Sf

$[\blacksquare\forall g(\forall f((Sf{\uparrow}Nt(s(g))){\downarrow}Sf)/CNs(g)) : \lambda A\lambda B\forall C[(A\ C) \to (B\ C)],$
$\square CNs(m) : man], \blacksquare\forall g\forall a((Sg{\uparrow}((\langle\rangle Na\backslash Sf)/(\langle\rangle Na\backslash Sb))){\downarrow}Sg) : \lambda D\neg(D\ \lambda E\lambda F(E\ F)),$
$\square(\langle\rangle(\exists aNa{-}\exists gNt(s(g)))\backslash Sf) : \hat{}\lambda G(Pres\ (\check{}walk\ G)) \ \Rightarrow\ Sf$

$[\blacksquare\forall g(\forall f((Sf{\uparrow}Nt(s(g))){\downarrow}Sf)/CNs(g)) : \lambda A\lambda B\forall C[(A\ C) \to (B\ C)],$
$\square CNs(m) : man], \blacksquare\forall g\forall a((Sg{\uparrow}((\langle\rangle Na\backslash Sf)/(\langle\rangle Na\backslash Sb))){\downarrow}Sg) : \lambda D\neg(D\ \lambda E\lambda F(E\ F)),$
$\square(\langle\rangle\exists aNa\backslash Sb) : \hat{}\lambda G(\check{}walk\ G) \ \Rightarrow\ Sf$

a : $\blacksquare\forall g(\forall f((S f{\uparrow}\blacksquare Nt(s(g))){\downarrow}S f)/CNs(g))$: $\lambda A\lambda B\exists C[(A\ C) \wedge (B\ C)]$

and : $\blacksquare\forall f((\blacksquare ?S f\backslash[]^{-1}[]^{-1}S f)/\blacksquare S f)$: $(\Phi^{n+}\ 0\ and)$

and : $\blacksquare\forall a\forall f((\blacksquare ?(\langle\rangle Na\backslash S f)\backslash[]^{-1}[]^{-1}(\langle\rangle Na\backslash S f))/\blacksquare(\langle\rangle Na\backslash S f))$: $(\Phi^{n+}\ (s\ 0)\ and)$

believes : $\Box((\langle\rangle\exists gNt(s(g))\backslash S f)/(CPthat\sqcup\Box S f))$: ${}^\wedge\lambda A\lambda B(Pres\ (({}^\vee believe\ A)\ B))$

bill : $\blacksquare Nt(s(m))$: b

catch : $\Box((\langle\rangle\exists aNa\backslash S b)/\exists aNa)$: ${}^\wedge\lambda A\lambda B(({}^\vee catch\ A)\ B)$

doesnt : $\blacksquare\forall g\forall a((S g{\uparrow}((\langle\rangle Na\backslash S f)/(\langle\rangle Na\backslash S b))){\downarrow}S g)$: $\lambda A\neg(A\ \lambda B\lambda C(B\ C))$

eat : $\Box((\langle\rangle\exists aNa\backslash S b)/\exists aNa)$: ${}^\wedge\lambda A\lambda B(({}^\vee eat\ A)\ B)$

every : $\blacksquare\forall g(\forall f((S f{\uparrow}Nt(s(g))){\downarrow}S f)/CNs(g))$: $\lambda A\lambda B\forall C[(A\ C) \rightarrow (B\ C)]$

finds : $\Box((\langle\rangle\exists gNt(s(g))\backslash S f)/\exists aNa)$: ${}^\wedge\lambda A\lambda B(Pres\ (({}^\vee find\ A)\ B))$

fish : $\Box CNs(n)$: $fish$

he : $\blacksquare[]^{-1}\forall g((\blacksquare S g|\blacksquare Nt(s(m)))/(\langle\rangle Nt(s(m))\backslash S g))$: λAA

her : $\blacksquare\forall g\forall a(((\langle\rangle Na\backslash S g){\uparrow}\blacksquare Nt(s(f))){\downarrow}(\blacksquare(\langle\rangle Na\backslash S g)|\blacksquare Nt(s(f))))$: λAA

in : $\Box(\forall a\forall f((\langle\rangle Na\backslash S f)\backslash(\langle\rangle Na\backslash S f))/\exists aNa)$: ${}^\wedge\lambda A\lambda B\lambda C(({}^\vee in\ A)\ (B\ C))$

is : $\blacksquare((\langle\rangle\exists gNt(s(g))\backslash S f)/(\exists aNa\oplus(\exists g((CNg/CNg)\sqcup(CNg\backslash CNg))-I)))$:
$\lambda A\lambda B(Pres\ (A \rightarrow C.[B = C]; D.((D\ \lambda E[E = B])\ B)))$

it : $\blacksquare\forall f\forall a(((\langle\rangle Na\backslash S f){\uparrow}\blacksquare Nt(s(n))){\downarrow}(\blacksquare(\langle\rangle Na\backslash S f)|\blacksquare Nt(s(n))))$: λAA

it : $\blacksquare[]^{-1}\forall f((\blacksquare S f|\blacksquare Nt(s(n)))/(\langle\rangle Nt(s(n))\backslash S f))$: λAA

john : $\blacksquare Nt(s(m))$: j

loses : $\Box((\langle\rangle\exists gNt(s(g))\backslash S f)/\exists aNa)$: ${}^\wedge\lambda A\lambda B(Pres\ (({}^\vee lose\ A)\ B))$

loves : $\Box((\langle\rangle\exists gNt(s(g))\backslash S f)/\exists aNa)$: ${}^\wedge\lambda A\lambda B(Pres\ (({}^\vee love\ A)\ B))$

man : $\Box CNs(m)$: man

necessarily : $\blacksquare(S A/\Box S A)$: Nec

or : $\blacksquare\forall f((\blacksquare ?S f\backslash[]^{-1}[]^{-1}S f)/\blacksquare S f)$: $(\Phi^{n+}\ 0\ or)$

or : $\blacksquare\forall a\forall f((\blacksquare ?(\langle\rangle Na\backslash S f)\backslash[]^{-1}[]^{-1}(\langle\rangle Na\backslash S f))/\blacksquare(\langle\rangle Na\backslash S f))$: $(\Phi^{n+}\ (s\ 0)\ or)$

or : $\blacksquare\forall f((\blacksquare ?(S f/(\langle\rangle\exists gNt(s(g))\backslash S f))\backslash[]^{-1}[]^{-1}(S f/(\langle\rangle\exists gNt(s(g))\backslash S f)))/\blacksquare(S f/(\langle\rangle\exists gNt(s(g))\backslash S f)))$:
$(\Phi^{n+}\ (s\ 0)\ or)$

park : $\Box CNs(n)$: $park$

seeks : $\Box((\langle\rangle\exists gNt(s(g))\backslash S f)/\Box\forall a\forall f(((Na\backslash S f)/\exists bNb)\backslash(Na\backslash S f)))$:
${}^\wedge\lambda A\lambda B(({}^\vee tries\ {}^\wedge(({}^\vee A\ {}^\vee find)\ B))\ B)$

she : $\blacksquare[]^{-1}\forall g((\blacksquare S g|\blacksquare Nt(s(f)))/(\langle\rangle Nt(s(f))\backslash S g))$: λAA

slowly : $\Box\forall a\forall f(\Box(\langle\rangle Na\backslash S f)\backslash(\langle\rangle\Box Na\backslash S f))$: ${}^\wedge\lambda A\lambda B({}^\vee slowly\ {}^\wedge({}^\vee A\ {}^\vee B))$

such+that : $\blacksquare\forall n((CNn\backslash CNn)/(S f|\blacksquare Nt(n)))$: $\lambda A\lambda B\lambda C[(B\ C) \wedge (A\ C)]$

talks : $\Box(\langle\rangle\exists gNt(s(g))\backslash S f)$: ${}^\wedge\lambda A(Pres\ ({}^\vee talk\ A))$

that : $\blacksquare(CPthat/\Box S f)$: λAA

the : $\blacksquare\forall n(Nt(n)/CNn)$: ι

to : $\blacksquare((PPto/\exists aNa)\sqcap\forall n((\langle\rangle Nn\backslash S i)/(\langle\rangle Nn\backslash S b)))$: λAA

tries : $\Box((\langle\rangle\exists gNt(s(g))\backslash S f)/\Box(\langle\rangle\exists gNt(s(g))\backslash S i))$: ${}^\wedge\lambda A\lambda B(({}^\vee tries\ {}^\wedge({}^\vee A\ B))\ B)$

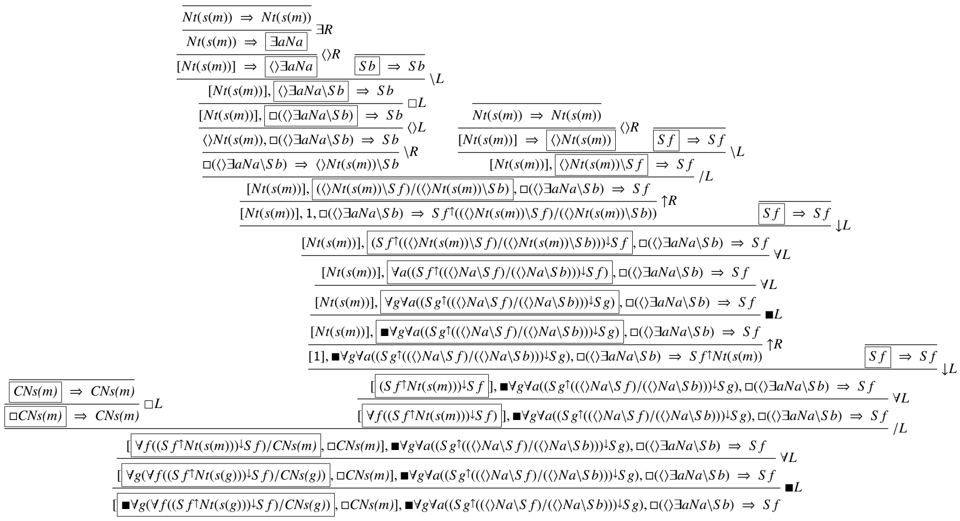
unicorn : $\Box CNs(n)$: $unicorn$

walk : $\Box(\langle\rangle\exists aNa\backslash S b)$: ${}^\wedge\lambda A({}^\vee walk\ A)$

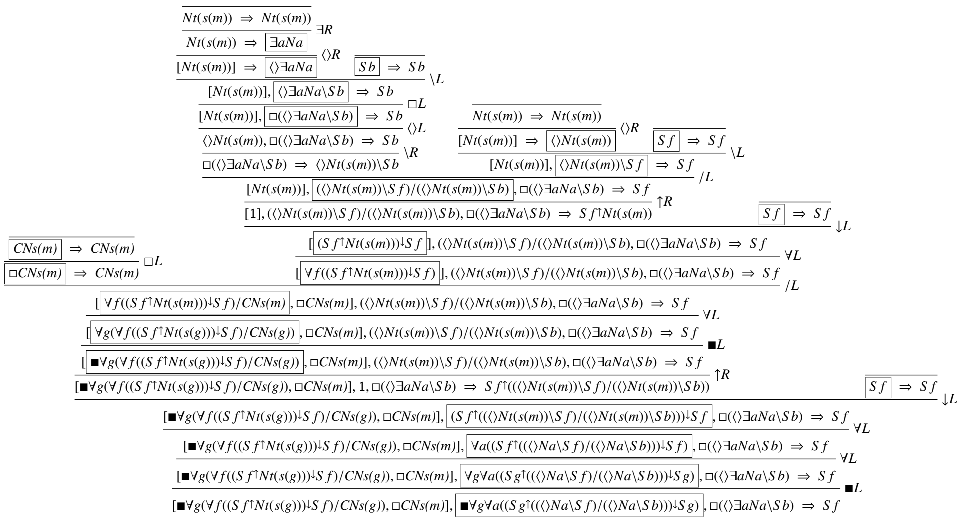
walks : $\Box(\langle\rangle\exists gNt(s(g))\backslash S f)$: ${}^\wedge\lambda A(Pres\ ({}^\vee walk\ A))$

woman : $\Box CNs(f)$: $woman$

Figure 15: Montague lexicon



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



$\neg\forall G[(\check{}man\ G) \rightarrow (\check{}walk\ G)]$

References

[1] Kazimierz Ajdukiewicz. Die syntaktische Konnexität. *Studia Philosophica*, 1:1–27, 1935. Translated in Storrs McCall, editor, 1967, *Polish Logic: 1920–1939*, Oxford University Press, Oxford, 207–231.

[2] J. M. Andreoli. Logic programming with focusing in linear logic. *Journal of Logic and Computation*, 2(3):297–347, 1992.

[3] Yehoshua Bar-Hillel. A quasi-arithmetical notation for syntactic description. *Language*, 29:47–58, 1953.

[4] 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*, pages 198–203, Berlin, 1991.

[5] Bob Carpenter. *Type-Logical Semantics*. MIT Press, Cambridge, MA, 1997.

[6] Noam Chomsky. *Syntactic Structures*. Mouton, The Hague, 1957.

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

[8] Mario Fadda. *Geometry of Grammar: Exercises in Lambek Style*. PhD thesis, Universitat Politècnica de Catalunya, Barcelona, 2010.

[9] G. Gentzen. Untersuchungen über das logische Schliessen. *Mathematische Zeitschrift*, 39:176–210 and 405–431, 1934. Translated in M.E. Szabo, editor, 1969, *The Collected Papers of Gerhard Gentzen*, North-Holland, Amsterdam, 68–131.

[10] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[11] Jean-Yves Girard. *The Blind Spot*. European Mathematical Society, Zürich, 2011.

[12] Mark Hepple. *The Grammar and Processing of Order and Dependency*. PhD thesis, University of Edinburgh, 1990.

[13] Gerhard Jäger. *Anaphora and Type Logical Grammar*, volume 24 of *Trends in Logic – Studia Logica Library*. Springer, Dordrecht, 2005.

[14] M. Kanazawa. The Lambek calculus enriched with additional connectives. *Journal of Logic, Language and Information*, 1:141–171, 1992.

[15] Max Kanovich, Stepan Kuznetsov, and Andre Scedrov. Undecidability of the Lambek calculus with a relevant modality. In Annie Foret, Glyn Morrill, Reinhard Muskens, Rainer Osswald, and Sylvain Pogodalla, editors, *Formal Grammar 2015: Revised Selected Papers. Formal Grammar 2016: Proceedings*, volume 9804, pages 240–246, Berlin, 2016. Springer.

[16] Stepan Kuznetsov, Glyn Morrill, and Oriol Valentín. Count-invariance including exponentials. In Makoto Kanazawa, editor, *Mathematics of Language*, London, 2017.

[17] 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.

[18] J. Lambek. Categorial and Categorical Grammars. In Richard T. Oehrle, Emmon Bach, and Deidre Wheeler, editors, *Categorial Grammars and Natural Language Structures*, volume 32 of *Studies in Linguistics and Philosophy*, pages 297–317. D. Reidel, Dordrecht, 1988.

[19] Joachim Lambek. The mathematics of sentence structure. *American Mathematical Monthly*, 65:154–170, 1958.

[20] Dale Miller, Gopalan Nadathur, Frank Pfenning, and Andre Scedrov. Uniform proofs as a foundation for logic programming. *Annals of Pure and Applied Logic*, 51(1-2):125–157, 1991.

[21] Richard Montague. The Proper Treatment of Quantification in Ordinary English. In J. Hintikka, J.M.E. Moravcsik, and P. Suppes, editors, *Approaches to Natural Language: Proceedings of the 1970 Stanford Workshop on Grammar and Semantics*, pages 189–224. D. Reidel, Dordrecht, 1973. Reprinted in R.H. Thomason, editor, 1974, *Formal Philosophy: Selected Papers of Richard Montague*, Yale University Press, New Haven, 247–270.

[22] Michael Moortgat. *Categorial Investigations: Logical and Linguistic Aspects of the Lambek Calculus*. Foris, Dordrecht, 1988. PhD thesis, Universiteit van Amsterdam.

[23] 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.

[24] 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.

[25] Richard Moot. Extended Lambek Calculi and First-Order Linear Logic. In Michael Moortgat Claudia Casadio, Bob Coeke and Philip Scott, editors, *Categories and Types in Logic, Language and Physics: Essays Dedicated to Jim Lambek on the Occasion of His 90th Birthday*, volume 8222 of *LNCS, FoLLI Publications in Logic, Language and Information*, pages 297–330. Springer, Berlin, 2014.

[26] Richard Moot. Proof nets for the displacement calculus. In Annie Foret, Glyn Morrill, Reinhard Muskens, Rainer Osswald, and Sylvain Pogodalla, editors, *Formal Grammar: 20th and 21st International Conferences*, volume 9804 of

LNCS, FoLLI Publications in Logic, Language and Information, pages 273–289, Berlin, 2016. Springer.

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

[28] Glyn Morrill. Grammar and Logical Types. In Martin Stockhof and Leen Torenvliet, editors, *Proceedings of the Seventh Amsterdam Colloquium*, pages 429–450, Amsterdam, 1990. Universiteit van Amsterdam.

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

[30] 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.

[31] Glyn Morrill. Categorial grammar. In Bernd Heine and Heiko Narrog, editors, *The Oxford Handbook of Linguistic Analysis*, pages 67–86. Oxford University Press, Oxford, 2010.

[32] 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.

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

[34] Glyn Morrill. Logical grammar. In Ruth Kempson, Tim Fernando, and Nicholas Asher, editors, *The Handbook of Philosophy of Linguistics*, pages 63–92. Elsevier, Oxford and Amsterdam, 2012.

[35] Glyn Morrill. Grammar logicised: relativisation. *Linguistics and Philosophy*, 40(2):119–163, 2017.

[36] Glyn Morrill and Mario Fadda. Proof Nets for Basic Discontinuous Lambek Calculus. *Logic and Computation*, 18(2):239–256, 2008.

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

[38] Glyn Morrill and Oriol Valentín. Displacement Calculus. *Linguistic Analysis*, 36(1–4):167–192, 2010. Special issue Festschrift for Joachim Lambek.

[39] Glyn Morrill and Oriol Valentín. Displacement Logic for Anaphora. *Journal of Computing and System Science*, 80:390–409, 2014. http://dx.doi.org/10.1016/j.jcss.2013.05.006.

[40] Glyn Morrill and Oriol Valentín. Semantically Inactive Multiplicatives and Words as Types. In Nicholas Asher and Sergei Soloviev, editors, *Proceedings of Logical Aspects of Computational Linguistics, LACL'14, Toulouse*, number 8535 in LNCS, FoLLI Publications on Logic, Language and Information, pages 149–162, Berlin, 2014. Springer.

[41] Glyn Morrill and Oriol Valentín. Computational Coverage of TLG: Nonlinearity. In M. Kanazawa, L.S. Moss, and V. de Paiva, editors, *Proceedings of NLCS'15. Third Workshop on Natural Language and Computer Science*, volume 32 of *EPiC*, pages 51–63, Kyoto, 2015. Workshop affiliated with Automata, Languages and Programming (ICALP) and Logic in Computer Science (LICS).

[42] Glyn Morrill and Oriol Valentín. Multiplicative-Additive Focusing for Parsing as Deduction. In I. Cervesato and C. Schürmann, editors, *First International Workshop on Focusing, workshop affiliated with LPAR 2015*, number 197 in EPTCS, pages 29–54, Suva, Fiji, 2015.

[43] Glyn Morrill and Oriol Valentín. Computational coverage of Type Logical Grammar: The Montague Test. In C. Piñón, editor, *Empirical Issues in Syntax and Semantics*, volume 11, pages 141–170. Colloque de Syntaxe et Sémantique à Paris (CSSP), Paris, 2016.

[44] Glyn Morrill and Oriol Valentín. On the Logic of Expansion in Natural Language. In Maxime Amblard, Phillipe de Groote, Sylvain Pogodalla, and Christian Retoré, editors, *Proceedings of Logical Aspects of Computational Linguistics, LACL'16, Nancy*, volume 10054 of *LNCS, FoLLI Publications on Logic, Language and Information*, pages 228–246, Berlin, 2016. Springer.

[45] 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.

[46] Glyn Morrill, Oriol Valentín, and Mario Fadda. The Displacement Calculus. *Journal of Logic, Language and Information*, 20(1):1–48, 2011.

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

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

[49] Oriol Valentín. *Theory of Discontinuous Lambek Calculus*. PhD thesis, Universitat Autònoma de Barcelona, Barcelona, 2012.

[50] Oriol Valentín, Daniel Serret, and Glyn Morrill. A Count Invariant for Lambek Calculus with Additives and Bracket Modalities. In Glyn Morrill and Mark-Jan Nederhof, editors, *Proceedings of Formal Grammar 2012 and 2013*, volume 8036 of *Springer LNCS, FoLLI Publications in Logic, Language and Information*, pages 263–276, Berlin, 2013. Springer.

[51] J. van Benthem. *Language in Action: Categories, Lambdas, and Dynamic Logic*. Number 130 in Studies in Logic and the Foundations of Mathematics. North-Holland, Amsterdam, 1991. Revised student edition printed in 1995 by the MIT Press.