

# Grammar Logicised

## Relativisation

Glyn Morrill

Received: date / Accepted: date

**Abstract** Many variants of categorial grammar assume an underlying logic which is associative and linear. In relation to left extraction, the former property is challenged by island domains, which involve non-associativity, and the latter property is challenged by parasitic gaps, which involve nonlinearity. We present a version of type logical grammar including ‘structural inhibition’ for non-associativity and ‘structural facilitation’ for nonlinearity and we give an account of relativisation including islands and parasitic gaps and their interaction.

**Keywords** islands · parasitic gaps · type logical categorial grammar · relativisation · structural facilitation · structural inhibition

### 1 Introduction

Today mainstream linguistics is largely informal and computational linguistics is largely statistical. The task of spelling out completely explicit fragments seems either forgotten or taken too lightly. But there is a tradition of logical categorial grammar dating back to Ajdukiewicz (1935[2]) which aims to practice linguistics according to the standards of mathematical logic. For a while (especially around 1985–2000) this aspiration blended well with the method of fragments, a methodology promoting the articulation of formal grammar fragments, such as the Montague fragment, and their combination and integration. Before major results were achieved in such comprehensive explicit integrated analysis, however, the field was overtaken by the aforementioned informal and statistical trends; but a small and committed community has remained. Now we are in a position to present a comprehensive explicit integrated analysis of relativisation as a preliminary example of deep and wide-coverage logical categorial grammar.

‘Categorial grammar’ refers to a family of approaches to syntax and semantics in which grammatical information is lexicalised and in which semantic composition is driven by a structure preserving mapping from syntactic types to semantic types. In its type logical formulation the grammar is purely lexical and the type calculus is universal. There are a number

of monographs, research monographs, reference works and textbooks on type logical grammar: Moortgat (1988[27]; 1997[29]), Morrill (1994[43]; 2011[44]), Carpenter (1997[6]), Jäger (2005[14]), and Moot and Retoré (2012[30]).

Relativisation is an unbounded dependency phenomenon: the distance between a relative pronoun and its extraction site can be indefinitely long:

- (1) a. the man that<sub>*t*<sub>*i*</sub></sub> I know *t*<sub>*i*</sub>  
 b. the man that<sub>*t*<sub>*i*</sub></sub> you know I know *t*<sub>*i*</sub>  
 c. the man that<sub>*t*<sub>*i*</sub></sub> I know you know I know *t*<sub>*i*</sub>  
 ⋮

The treatment of relativisation in categorial grammar by means of assignment of higher-order functors to relative pronouns is well-established since Ades and Steedman (1982[1]) and yields the unboundedness property through associative assembly of the body of a relative clause.

However, although relativisation is unbounded it is not unconstrained. Various ‘islands’ can inhibit or block relativisation: weak islands such as subjects and adverbial phrases, from which extraction is mildly unacceptable, and strong islands such as coordinate structures and relative clauses themselves, from which extraction is completely unacceptable:

- (2) a. ?man who<sub>*t*<sub>*i*</sub></sub> the friend of *t*<sub>*i*</sub> laughed  
 b. ?paper which<sub>*t*<sub>*i*</sub></sub> John laughed before reading *t*<sub>*i*</sub>  
 (3) a. \*man who<sub>*t*<sub>*i*</sub></sub> John laughed and Mary likes *t*<sub>*i*</sub>  
 b. \*man who<sub>*t*<sub>*i*</sub></sub> John likes the woman that loves *t*<sub>*i*</sub>

At least, this is the traditional view of islands. Under certain pragmatic or discourse-oriented conditions the processing of island violations is more acceptable (Lakoff 1986[21]; Deane 1991[8]; Kluender 1992[18], 1998[19]; Kehler 2002[17]; Hofmeister and Sag 2010[13]). For example Kluender provides experimental evidence that there is a gradient effect on the acceptability of violations of the Complex Noun Phrase Constraint (CNPC) whereby increasing referential specificity of the extracted NP makes the sentence more acceptable, and so does decreasing the referential specificity of the extraction path.

Such facts are interpreted by Kubota and Levine (2015[20], Section 4.6.2) to argue for a version of logical categorial grammar that freely overgenerates island constraint ‘violations’ in the syntax. The proposals of the present paper can be made entirely consistent with such a stance: simply erase all brackets. But our position will be more conservative. Taking the step of denying that there are syntactic islands is methodologically forceful in that there will be no undergeneration, but leaves the question of explaining how the processing is impeded in cases where the island effect is robust. For example, without syntactic islands a referentially non-facilitated CNPC violation will be wrongly predicted to be acceptable unless it can be explained how non-syntactic factors conspire to inhibit the application of processing steps.

Retaining syntactic islands, as we do here, leaves the question of explaining how the processing is facilitated in cases where the island effect is not robust. Following Morrill (2011[44]) we do so by proposing that extrasyntactic factors can ‘pay the cost’ of the application of an exceptional island-violating processing step. This does not seem to us entirely satisfactory, but it does seem better than trying to explain how non-syntactic factors conspire to inhibit the application of processing steps, for which we see no path.

Different languages have different island-like properties, for example CNPC violations are more acceptable in Scandinavian languages than in English. Cross-linguistically, therefore, the free island-overgeneration of Kubota and Levine would have to be accompanied by

*language-specific* non-syntactic processing step inhibition. As pointed out by O. Valentín (p.c.) one might as well freely overgenerate word order permutations and say that ungrammaticality is filtered by language-specific non-syntactic factors. Again this is methodologically forceful in that there will be no undergeneration. But we think the cure is worse than the disease because we do not see how non-syntactic factors can conspire to inhibit the application of processing steps.

Hence we shall adopt a conservative, syntactic, treatment of islands, while recognising that such ‘structural inhibition’ represents a challenge to categorial grammar and all approaches to grammar.

Relativisation, furthermore, can also comprise ‘parasitic extraction’ in which a relative pronoun binds more than one extraction site (Taraldsen 1979[50]; Engdahl 1983[9]; Sag 1983[47]). There is a *single* ‘host’ gap which is not in an island, and according to the received wisdom, and according with the terminology ‘parasitic’, this may license a ‘parasitic’ gap in (any number of immediate weak) islands:

- (4) a. \*the slave who<sub>i</sub> John sold  $t_i$   $t_i$   
 b. \*the slave who<sub>i</sub> John sold  $t_i$  to  $t_i$
- (5) a. the man who<sub>i</sub> the friends of  $t_i$  admire  $t_i$   
 b. the paper which<sub>i</sub> John filed  $t_i$  without reading  $t_i$   
 c. the paper which<sub>i</sub> the editor of  $t_i$  filed  $t_i$  without reading  $t_i$

In addition, we observe here that these parasitic gaps may in turn function as host gaps licensing further parasitic gaps in (weak) subislands, and so on recursively:

- (6) a. man who<sub>i</sub> the fact that the friends of  $t_i$  admire  $t_i$  surprises  $t_i$   
 b. man who<sub>i</sub> the fact that the friends of  $t_i$  admire  $t_i$  without praising  $t_i$  offends  $t_i$  without surprising  $t_i$

There are examples in which there appears to be a parasitic gap which is not in an island. The following is example (8a) from Postal (1993[46]):

- (7) man who<sub>i</sub> Mary convinced  $t_i$  that John wanted to visit  $t_i$

And an anonymous referee points out:

- (8) people whom<sub>i</sub> you sent pictures of  $t_i$  to  $t_i$

In respect of such examples we suggest that although there *seems* to be no island, there *could* be one.<sup>1</sup> We give an account along this line of ‘optional islands’ in Section 6.

Parasitic ‘structural facilitation’ represents a further challenge to categorial grammar and all approaches to grammar.

This is the empirical analysis of islands and parasitic gaps and their interaction given a type logical, i.e. purely lexical, categorial account in Morrill (2011[44], Chapter 5). In this paper we give an account of the empirical analysis which improves on that account in the following respects:

- Multimodality and associated multimodal structural postulates are removed.<sup>2</sup>

<sup>1</sup> Tom Roeper, p.c.

<sup>2</sup> Structural postulates increase the derivation search space. Of course ways may be found to manage this, but that would be precisely to absorb the structural properties in the way that is done here (Valentín 2014[52]). *Ceteris paribus*, given the choice between structural postulates or no structural postulates, the latter is to be preferred.

	cont. mult.	disc. mult.	add.	qu.	norm. mod.	brack. mod.	exp.	lim. contr. & weak.
primary	/ \ • I	↑ ↓ ⊖ J	& ∨ ⊕	∧ ∇ ∩	□ ◇	[ ] <sup>-1</sup> ⟨ ⟩	! ?	 W
sem. inactive variants	•—○ ○—• ● ○	• ↓ ○ ↓ ● ○	□ ∩	∇ ∃	■ ◆			
det.	◁ <sup>-1</sup> ▷ <sup>-1</sup>	∨						diff.
synth.	◁ ▷	∧						
nondet.	÷	↑ ↓						
synth.	⊗	⊙						—

**Table 1** Categorical connectives

- The proposal is set in the context of the displacement calculus of Morrill and Valentín (2010[36]) and Morrill, Valentín and Fadda (2011[42]) which is an advance on the discontinuous Lambek calculus of Morrill (2011[44], Chapter 6).
- Nonlinearity (structural facilitation) is formalised by use of a ‘stoup’ (Girard 2011[11]) which reduces the size of the proof search space.
- The rule of contraction generating parasitic gaps is simplified.
- The account integrates other aspects of grammatical analysis such as polymorphism, features, and intensionality.
- The correct interaction of all the grammatical aspects is verified by computer-generation of the analyses.
- Various possible exceptions to the empirical analysis are addressed.

The result is a formal and mathematically principled empirically adequate formalisation of relativisation which is thorough, very high level (concise) and which is computer-verified.

The structure of the paper is as follows. In Section 2 we present and illustrate the sequent calculus for our displacement logic. In Section 3 we present initial examples of analysis. In Section 4 we discuss approaches to relativisation with which we differ, and in Section 5 we present our theoretical analysis of relativisation. In Section 6 we address possible exceptions to our account. We conclude in Section 7. The semantic representation language used here is defined in Appendix A, and a lexicon is given in Appendix B.

## 2 Framework

The formalism used comprises the connectives of Table 1. The heart of the logic is the displacement calculus of Morrill and Valentín (2010[36]) and Morrill, Valentín and Fadda

(2011[42]) 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[51]). Other primary connectives are additives, 1st order quantifiers, normal (i.e. distributive) modalities, bracket (i.e. nondistributive) modalities, exponentials, limited contraction and limited weakening, and difference.

We can draw a clear distinction between the primary connectives and the semantically inactive connectives and the synthetic connectives which 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.<sup>3</sup> Synthetic connectives (Girard 2011[11]) divide into the continuous and discontinuous deterministic (unary) synthetic connectives, and the continuous and discontinuous nondeterministic (binary) synthetic connectives.<sup>4</sup>

## 2.1 Syntactic types

The syntactic types of displacement logic are sorted  $\mathcal{F}_0, \mathcal{F}_1, \mathcal{F}_2, \dots$  according to the number of points of discontinuity  $0, 1, 2, \dots$  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  $t_1, \dots, t_n$  are terms,  $Pt_1 \dots t_n$  is an (atomic) type of sort  $i$  of the corresponding semantic type. Compound types are formed by connectives as indicated in Table 2,<sup>5</sup> and the structure preserving semantic type map  $T$  associates these with semantic types.

## 2.2 Gentzen sequent calculus

(Labelled) natural deduction can proportion a congenial proof format for categorial logic because the compositional term-structure of Curry-Howard semantics follows the structure of natural deduction derivation. However, here we use the Gentzen sequent proof format because:

- Natural deduction does not capture symmetries as satisfactorily as Gentzen sequent calculus. For example, while product right is easy to express in ND, product left is awkward (unnatural); but both are straightforwardly expressed in Gentzen sequent calculus.<sup>6</sup>
- The title of the paper is ‘Grammar logicised’, i.e. there is an emphasis on the thesis that grammar can be reduced to logic. To maintain this it is necessary to pitch the logical aspects as closely as possible to the usual Gentzen format with the associated symmetries and metatheory.

<sup>3</sup> For example, the semantically inactive additive conjunction  $A \sqcap B: \phi$  abbreviates  $A \& B: (\phi, \phi)$ .

<sup>4</sup> For example, the nondeterministic continuous division  $B \div A$  abbreviates  $(A \setminus B) \sqcap (B / A)$ .

<sup>5</sup> We list only connectives drawn from the first two rows of Table 1, and we omit some which are not central here.

<sup>6</sup> Thus, in Gentzen format product left is simply  $\frac{A(A \bullet B) \Rightarrow D}{A(A \bullet B) \Rightarrow D}$  but unlabelled ND requires something like

$$(i) \quad \frac{\begin{array}{c} \vdots \\ A \bullet B \end{array}}{A \quad B}$$

which does not respect the single-conclusion condition.

1.	$\mathcal{F}_i ::= \mathcal{F}_{i+j}/\mathcal{F}_j$	$T(C/B) = T(B) \rightarrow T(C)$	over
2.	$\mathcal{F}_j ::= \mathcal{F}_i \setminus \mathcal{F}_{i+j}$	$T(A \setminus C) = T(A) \rightarrow T(C)$	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.	$\mathcal{F}_{i+1} ::= \mathcal{F}_{i+j} \uparrow_k \mathcal{F}_j, 1 \leq k \leq i+j$	$T(C \uparrow_k B) = T(B) \rightarrow T(C)$	circumfix
6.	$\mathcal{F}_j ::= \mathcal{F}_{i+1} \downarrow_k \mathcal{F}_{i+j}, 1 \leq k \leq i+1$	$T(A \downarrow_k C) = T(A) \rightarrow T(C)$	infix
7.	$\mathcal{F}_{i+j} ::= \mathcal{F}_{i+1} \circ_k \mathcal{F}_j, 1 \leq k \leq i+1$	$T(A \circ_k B) = T(A) \& T(B)$	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.	$\mathcal{F}_i ::= \bigwedge \mathcal{V} \mathcal{F}_i$	$T(\bigwedge vA) = F \rightarrow T(A)$	1st order univ. qu.
12.	$\mathcal{F}_i ::= \bigvee \mathcal{V} \mathcal{F}_i$	$T(\bigvee vA) = F \& T(A)$	1st order exist. qu.
13.	$\mathcal{F}_i ::= \square \mathcal{F}_i$	$T(\square A) = \mathbf{L}T(A)$	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
33.	$\mathcal{F}_i ::= \mathcal{F}_i \sqcap \mathcal{F}_i$	$T(A \sqcap B) = T(A) = T(B)$	sem. inactive additive conjunction
34.	$\mathcal{F}_i ::= \mathcal{F}_i \sqcup \mathcal{F}_i$	$T(A \sqcup B) = T(A) = T(B)$	sem. inactive additive disjunction
35.	$\mathcal{F}_i ::= \forall \mathcal{V} \mathcal{F}_i$	$T(\forall vA) = T(A)$	sem. inactive 1st order univ. qu.
36.	$\mathcal{F}_i ::= \exists \mathcal{V} \mathcal{F}_i$	$T(\exists vA) = T(A)$	sem. inactive 1st order exist. qu.
37.	$\mathcal{F}_i ::= \blacksquare \mathcal{F}_i$	$T(\blacksquare A) = T(A)$	sem. inactive universal modality
38.	$\mathcal{F}_i ::= \blacklozenge \mathcal{F}_i$	$T(\blacklozenge A) = T(A)$	sem. inactive existential modality

**Table 2** Syntactic types

- It enables uniform formulation of all of the rules of inference.
- It dispenses with phonological labels.
- It lends itself more transparently to proof of Cut-elimination and consequent decidability.
- It lends itself more transparently to focalisation (Andreoli 1992[3]); Morrill and Valentín 2015[40]) and consequent efficient computer generation and verification of the analyses.

Crucially, in Gentzen sequent configurations  $(\Gamma, \Delta)$  for displacement calculus a discontinuous type is a mother, rather than a leaf, and dominates its discontinuous components marked off by curly brackets and colons.

In Gentzen sequent antecedents for displacement logic with bracket modalities (structural inhibition) and exponentials (structural facilitation) there is also a bracket constructor for the former and ‘stoups’ for the latter.

*Stoups* (cf. the linear logic of Girard 2011[11]) ( $\zeta$ ) are stores read as multisets for reusable (nonlinear) resources which appear at the left of a configuration marked off by a semicolon (when the stoup is empty the semicolon may be omitted). The stoup of linear logic is for resources which can be contracted (copied) 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 the stoup is used for resources which are also commutative. To anticipate our analysis a little, a hypothetical subtype emitted by a relative pronoun corresponding to a long-distance dependency will enter a stoup, percolate in stoups, maybe contracting to create (parasitic) gaps, and finally permute into a (host) extraction site.

A configuration together with a stoup is a *zone* ( $\mathcal{E}$ ). 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;  $\Lambda$  is the empty configuration; the *separator* 1 marks points of discontinuity):<sup>7</sup>

$$(9) \quad \begin{aligned} \mathcal{S} &::= \emptyset \mid \mathcal{F}_0, \mathcal{S} \\ \mathcal{O} &::= \Lambda \mid \mathcal{T}, \mathcal{O} \\ \mathcal{T} &::= 1 \mid \mathcal{F}_0 \mid \mathcal{F}_{i>0} \{ \underbrace{\mathcal{O} : \dots : \mathcal{O}}_{i\mathcal{O}'s} \} \mid [\mathcal{S}; \mathcal{O}] \end{aligned}$$

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

$$(10) \quad \mathcal{S}; \mathcal{O} \Rightarrow \mathcal{F} \text{ such that } s(\mathcal{O}) = s(\mathcal{F})$$

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

$$(11) \quad \vec{A} = \begin{cases} A & \text{if } s(A) = 0 \\ A\{ \underbrace{1 : \dots : 1}_{s(A) \text{ 1's}} \} & \text{if } s(A) > 0 \end{cases}$$

Where  $\Gamma$  is a configuration of sort  $i$  and  $\Delta_1, \dots, \Delta_i$  are configurations, the *fold*  $\Gamma \otimes \langle \Delta_1 : \dots : \Delta_i \rangle$  is the result of replacing the successive 1's in  $\Gamma$  by  $\Delta_1, \dots, \Delta_i$  respectively. Where  $\Gamma$  is of sort  $i$ , the hyperoccurrence notation  $\Delta(\Gamma)$  abbreviates  $\Delta_0(\Gamma \otimes \langle \Delta_1 : \dots : \Delta_i \rangle)$ , i.e. a context configuration  $\Delta$  (which is externally  $\Delta_0$  and internally  $\Delta_1, \dots, \Delta_i$ ) with a potentially discontinuous distinguished subconfiguration  $\Gamma$ . Where  $\Delta$  is a configuration of sort  $i > 0$  and  $\Gamma$  is a configuration, the  $k$ th *metalinguistic intercalation*  $\Delta|_k \Gamma$ ,  $1 \leq k \leq i$ , is given by:

$$(12) \quad \Delta|_k \Gamma =_{df} \Delta \otimes \langle \underbrace{1 : \dots : 1}_{k-1 \text{ 1's}} : \Gamma : \underbrace{1 : \dots : 1}_{i-k \text{ 1's}} \rangle$$

i.e.  $\Delta|_k \Gamma$  is the configuration resulting from replacing by  $\Gamma$  the  $k$ th separator in  $\Delta$ .

### 2.3 Rules and linguistic applications

A semantically labelled sequent is a sequent in which the antecedent type occurrences  $A_1, \dots, A_n$  are labelled by distinct variables  $x_1, \dots, x_n$  of types  $T(A_1), \dots, T(A_n)$  respectively, and the succedent type  $A$  is labelled by a term of type  $T(A)$  with free variables drawn from  $x_1, \dots, x_n$ . In this section we give the semantically labelled Gentzen sequent rules for some primary connectives, and indicate some linguistic applications.

The continuous multiplicatives of Figure 1, the Lambek connectives, Lambek (1958[24]; 1988[23]), defined in relation to 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*:  $N \setminus S$  for *John sings*:  $S$ , and *loves*:  $(N \setminus S)/N$  for *John loves Mary*:  $S$ . The continuous product  $\bullet$  is exemplified by a ‘small clause’ assignment such as *considers*:  $(N \setminus S)/(N \bullet (CN/CN))$  for say *John considers Mary socialist*:  $S$ .<sup>8</sup>

<sup>7</sup> 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.

<sup>8</sup> But this makes no different empirical predictions from the more standard type of analysis in categorial grammar which simply treats verbs like *consider* as taking a noun phrase and an infinitive. Products are more truly motivated by antecedent occurrences in the (continuous) analysis of past participles of Morrill (2011[44], pp.64–65), or the discontinuous generalisation of this for a past participle such as *loved* in Appendix B here.

$$\begin{array}{l}
1. \quad \frac{\zeta_1; \Gamma \Rightarrow B: \psi \quad \zeta_2; \Delta(\vec{C}: z) \Rightarrow D: \omega}{\zeta_1 \uplus \zeta_2; \Delta(\vec{C}/\vec{B}: x, \Gamma) \Rightarrow D: \omega\{(x \psi)/z\}} /L \quad \frac{\zeta; \Gamma, \vec{B}: y \Rightarrow C: \chi}{\zeta; \Gamma \Rightarrow C/B: \lambda y \chi} /R \\
2. \quad \frac{\zeta_1; \Gamma \Rightarrow A: \phi \quad \zeta_2; \Delta(\vec{C}: z) \Rightarrow D: \omega}{\zeta_1 \uplus \zeta_2; \Delta(\Gamma, \vec{A}\vec{C}: y) \Rightarrow D: \omega\{(y \phi)/z\}} \setminus L \quad \frac{\zeta; \vec{A}: x, \Gamma \Rightarrow C: \chi}{\zeta; \Gamma \Rightarrow A \setminus C: \lambda x \chi} \setminus R \\
3. \quad \frac{\zeta; \Delta(\vec{A}: x, \vec{B}: y) \Rightarrow D: \omega}{\zeta; \Delta(\vec{A}\bullet\vec{B}: z) \Rightarrow D: \omega\{\pi_1 z/x, \pi_2 z/y\}} \bullet L \quad \frac{\zeta_1; \Gamma_1 \Rightarrow A: \phi \quad \zeta_2; \Gamma_2 \Rightarrow B: \psi}{\zeta_1 \uplus \zeta_2; \Gamma_1, \Gamma_2 \Rightarrow A\bullet B: (\phi, \psi)} \bullet R \\
4. \quad \frac{\zeta; \Delta(\Lambda) \Rightarrow A: \phi}{\zeta; \Delta(\vec{T}: x) \Rightarrow A: \phi} IL \quad \frac{}{\emptyset; \Lambda \Rightarrow I: 0} IR
\end{array}$$

Fig. 1 Continuous multiplicatives

$$\begin{array}{l}
5. \quad \frac{\zeta_1; \Gamma \Rightarrow B: \psi \quad \zeta_2; \Delta(\vec{C}: z) \Rightarrow D: \omega}{\zeta_1 \uplus \zeta_2; \Delta(\vec{C}\uparrow_k\vec{B}: x \mid_k \Gamma) \Rightarrow D: \omega\{(x \psi)/z\}} \uparrow_k L \quad \frac{\zeta; \Gamma \mid_k \vec{B}: y \Rightarrow C: \chi}{\zeta; \Gamma \Rightarrow C \uparrow_k B: \lambda y \chi} \uparrow_k R \\
6. \quad \frac{\zeta_1; \Gamma \Rightarrow A: \phi \quad \zeta_2; \Delta(\vec{C}: z) \Rightarrow D: \omega}{\zeta_1 \uplus \zeta_2; \Delta(\Gamma \mid_k \vec{A}\downarrow_k\vec{C}: y) \Rightarrow D: \omega\{(y \phi)/z\}} \downarrow_k L \quad \frac{\zeta; \vec{A}: x \mid_k \Gamma \Rightarrow C: \chi}{\zeta; \Gamma \Rightarrow A \downarrow_k C: \lambda x \chi} \downarrow_k R \\
7. \quad \frac{\zeta; \Delta(\vec{A}: x \mid_k \vec{B}: y) \Rightarrow D: \omega}{\zeta; \Delta(\vec{A}\odot_k\vec{B}: z) \Rightarrow D: \omega\{\pi_1 z/x, \pi_2 z/y\}} \odot_k L \quad \frac{\zeta_1; \Gamma_1 \Rightarrow A: \phi \quad \zeta_2; \Gamma_2 \Rightarrow B: \psi}{\zeta_1 \uplus \zeta_2; \Gamma_1 \mid_k \Gamma_2 \Rightarrow A\odot_k B: (\phi, \psi)} \odot_k R \\
8. \quad \frac{\zeta; \Delta(1) \Rightarrow A: \phi}{\zeta; \Delta(\vec{T}: x) \Rightarrow A: \phi} JL \quad \frac{}{\emptyset; 1 \Rightarrow J: 0} JR
\end{array}$$

Fig. 2 Discontinuous multiplicatives

The discontinuous multiplicatives of Figure 2, the displacement connectives, Morrill and Valentín (2010[36]), Morrill, Valentín and Fadda (2011[42]), are defined in relation to plugging. When the value of the  $k$  subindex indicates the first (leftmost) point of discontinuity it may be omitted. Circumfixation,  $\uparrow$ , is exemplified by a discontinuous idiom assignment *gives+1+the+cold+shoulder*:  $(N \setminus S) \uparrow N$  for *Mary gives John the cold shoulder*:  $S$ , and infixation,  $\downarrow$ , and circumfixation together are exemplified by a quantifier phrase assignment *everyone*:  $(S \uparrow N) \downarrow S$  simulating Montague’s S14 treatment of quantifying in. Circumfixation and discontinuous product,  $\odot$ , are illustrated together with the continuous unit in an assignment to a relative pronoun *that*:  $(CN \setminus CN) / ((S \uparrow N) \odot I)$  allowing both peripheral and medial extraction: *that John likes*:  $CN \setminus CN$  and *that John saw today*:  $CN \setminus CN$ , although we will argue in Section 4 that this strategy is inadequate, and the main point of the present paper is to promote another approach to relativisation.

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[22]), Morrill (1990[31]), Kanazawa (1992[16]), have application to polymorphism. For example the additive conjunction  $\&$  can be used for



$$\begin{array}{c}
9. \quad \frac{\frac{\Xi \langle \vec{A} : x \rangle \Rightarrow C : \chi}{\Xi \langle \vec{A} \& \vec{B} : z \rangle \Rightarrow C : \chi \{ \pi_1 z / x \}} \&L_1 \quad \frac{\Xi \langle \vec{B} : y \rangle \Rightarrow C : \chi}{\Xi \langle \vec{A} \& \vec{B} : z \rangle \Rightarrow C : \chi \{ \pi_2 z / y \}} \&L_2}{\frac{\Xi \Rightarrow A : \phi \quad \Xi \Rightarrow B : \psi}{\Xi \Rightarrow A \& B : (\phi, \psi)} \&R} \\
10. \quad \frac{\frac{\Xi \langle \vec{A} : x \rangle \Rightarrow C : \chi_1 \quad \Xi \langle \vec{B} : y \rangle \Rightarrow C : \chi_2}{\Xi \langle \vec{A} \oplus \vec{B} : z \rangle \Rightarrow C : z \rightarrow x, \chi_1 ; y, \chi_2} \oplus L}{\frac{\Xi \Rightarrow A : \phi}{\Xi \Rightarrow A \oplus B : \iota_1 \phi} \oplus R_1 \quad \frac{\Xi \Rightarrow B : \psi}{\Xi \Rightarrow A \oplus B : \iota_2 \psi} \oplus R_2}
\end{array}$$

Fig. 3 Additives

*rice*:  $N \& CN$  as in *rice grows*:  $S$  and *the rice grows*:  $S$ ,<sup>9</sup> and the additive disjunction  $\oplus$  can be used for *is*:  $(N \setminus S) / (N \oplus (CN / CN))$  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*:  $(N \setminus S) / (N \oplus I)$  for *John eats fish*:  $S$  and *John eats*:  $S$ .<sup>10</sup>

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

$$\begin{array}{c}
11. \quad \frac{\frac{\Xi \langle \vec{A}[t/v] : x \rangle \Rightarrow B : \psi}{\Xi \langle \bigwedge vA : z \rangle \Rightarrow B : \psi \{ (z \ t) / x \}} \wedge L \quad \frac{\Xi \Rightarrow A[a/v] : \phi}{\Xi \Rightarrow \bigwedge vA : \lambda v \phi} \wedge R^\dagger}{12. \quad \frac{\frac{\Xi \langle \vec{A}[a/v] : x \rangle \Rightarrow B : \psi}{\Xi \langle \bigvee vA : z \rangle \Rightarrow B : \psi \{ \pi_2 z / x \}} \vee L^\dagger \quad \frac{\Xi \Rightarrow A[t/v] : \phi}{\Xi \Rightarrow \bigvee vA : (t, \phi)} \vee R}
\end{array}$$

Fig. 4 Quantifiers, where  $\dagger$  indicates that there is no  $a$  in the conclusion

The quantifiers of Figure 4, Morrill (1994[43]), 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*:  $(N \setminus S) / \bigvee tS f(t)$  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.

With respect to the normal modalities of Figure 5, the universal (Morrill 1990[32]) has application to intensionality. For example, for a propositional attitude verb such as *believes* we can assign type  $\Box((N \setminus S) / \Box S)$  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 the subject is not.

<sup>9</sup> 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.

<sup>10</sup> 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 is at best a promissory solution, unless there is true ambiguity.

$$\begin{array}{l}
13. \quad \frac{\frac{\Xi(\vec{A}:x) \Rightarrow B:\psi}{\Xi(\overline{\square}A:z) \Rightarrow B:\psi\{\vee z/x\}} \square L \quad \frac{\boxtimes \Xi \Rightarrow A:\phi}{\boxtimes \Xi \Rightarrow \square A:\wedge \phi} \square R \\
14. \quad \frac{\frac{\boxtimes \Xi(\vec{A}:x) \Rightarrow \diamond B:\psi}{\boxtimes \Xi(\overline{\diamond}A:z) \Rightarrow \diamond B:\psi\{\cup z/x\}} \diamond L \quad \frac{\Xi \Rightarrow A:\phi}{\Xi \Rightarrow \diamond A:\cap \phi} \diamond R
\end{array}$$

**Fig. 5** Normal modalities, where  $\boxtimes/\diamond$  marks a structure all the types of which have main connective a box/diamond

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

$$\begin{array}{l}
15. \quad \frac{\frac{\Xi(\vec{A}:x) \Rightarrow B:\psi}{\Xi(\overline{[\ ]^{-1}A}:x) \Rightarrow B:\psi} [\ ]^{-1}L \quad \frac{[\Xi] \Rightarrow A:\phi}{\Xi \Rightarrow [\ ]^{-1}A:\phi} [\ ]^{-1}R \\
16. \quad \frac{\frac{\Xi(\vec{A}:x) \Rightarrow B:\psi}{\Xi(\overline{\langle \rangle}A:x) \Rightarrow B:\psi} \langle \rangle L \quad \frac{\Xi \Rightarrow A:\phi}{[\Xi] \Rightarrow \langle \rangle A:\phi} \langle \rangle R
\end{array}$$

**Fig. 6** Bracket modalities

The bracket modalities of Figure 6, Morrill (1992[33]) and Moortgat 1995[28]), have application to nonassociativity and syntactical domains such as prosodic phrases and extraction islands. For example, single bracketing for weak islands: *walks*:  $\langle \rangle N \setminus S$  for the subject condition, and *without*:  $[\ ]^{-1}(VP \setminus VP)/VP$  for the adverbial island constraint; and double bracketing for strong islands such as *and*:  $(S \setminus [\ ]^{-1}[\ ]^{-1}S)/S$  for the coordinate structure constraint.

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

$$\begin{array}{l}
17. \quad \frac{\frac{\Xi(\zeta \uplus \{A:x\}; \Gamma_1, \Gamma_2) \Rightarrow B:\psi}{\Xi(\zeta; \Gamma_1, !A:x, \Gamma_2) \Rightarrow B:\psi} !L \quad \frac{\zeta; \Lambda \Rightarrow A:\phi}{\zeta; \Lambda \Rightarrow !A:\phi} !R \\
\frac{\Xi(\zeta; \Gamma_1, A:x, \Gamma_2) \Rightarrow B:\psi}{\Xi(\zeta \uplus \{A:x\}; \Gamma_1, \Gamma_2) \Rightarrow B:\psi} !P \\
\frac{\Xi(\zeta \uplus \{A:x\}; \Gamma_1, [\{A:y\}; \Gamma_2], \Gamma_3) \Rightarrow B:\psi}{\Xi(\zeta \uplus \{A:x\}; \Gamma_1, \Gamma_2, \Gamma_3) \Rightarrow B:\psi\{x/y\}} !C
\end{array}$$

**Fig. 7** Universal exponential

Finally, there is nonlinearity. The universal exponential of Figure 7, Girard (1987[10]), Barry, Hepple, Leslie and Morrill (1991[5]), Morrill (1994[43]), and Morrill and Valentín (2015[39]) has application to 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 possibly host gap). For example:

$$(13) \frac{\frac{\frac{\dots, A, \dots, [\dots, A, \dots], \dots, \dots \Rightarrow D}{\dots, A, \dots, [A; \dots, \dots], \dots, \dots \Rightarrow D} !P}{A; \dots, \dots, [A; \dots, \dots], \dots, \dots \Rightarrow D} !P}{\vdots} !C}{\frac{A; \dots, \dots, \dots, \dots, \dots \Rightarrow B}{\dots, \dots, \dots, \dots, \dots, !A, \dots \Rightarrow B} !L} !L$$

Reading upwards, first the  $!A$  is moved into the stoup by  $!L$  and the exponential modality is removed (being in the stoup *means* that the type is under the associated resource management regime). We assume some derivation steps, indicated by vertical dots, and then an application of contraction  $!C$ . A domain becomes bracketed, *and this domain contains A in its stoup*. This would correspond to a weak island containing a parasitic gap. Finally the ‘host’ and ‘parasitic’ gaps are permuted into position by two applications of  $!P$ .

Using the universal exponential,  $!$ , for which contraction induces island brackets, we can assign a relative pronoun type *that*:  $(CN \setminus CN)/(S/!N)$  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*. See Section 5.

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*.

In relation to the rest of the primary connectives: the existential exponential  $?$  has application to iterated coordination (Morrill 1994[43]; Morrill and Valentín 2015[39]), the limited contraction  $|$  of Jäger (2005[14]) has application to anaphora and the limited weakening  $W$  of Morrill and Valentín (2014[38]) has application to words as types. The remaining, semantically inactive, connectives listed here were introduced as follows. Semantically inactive multiplicatives  $\{\leftarrow, \rightarrow, \circ, \dashv, \bullet, \odot, \uparrow, \downarrow, \updownarrow, \ominus, \otimes\}$ : Morrill and Valentín (2014[38]). Semantically inactive additives  $\{\sqcap, \sqcup\}$ : Morrill (1994[43]). Semantically inactive first-order quantifiers  $\{\forall, \exists\}$ : Morrill (1994[43]). Semantically inactive normal modalities  $\{\blacksquare, \blacklozenge\}$ : Hople (1990[12]), Morrill (1994[43]).<sup>11</sup>

### 3 Initial examples

The first example is as follows:<sup>12</sup>

<sup>11</sup> The synthetic connectives are: left and right projection and injection  $\{\leftarrow^{-1}, \rightarrow^{-1}, \leftarrow, \rightarrow\}$ , Morrill, Fadda and Valentín (2009[41]); split and bridge  $\{\wedge, \vee\}$ , Morrill and Merenciano (1996[35]); continuous and discontinuous nondeterministic multiplicatives  $\{\mp, \otimes, \updownarrow, \oplus\}$ , Morrill, Valentín and Fadda (2011[42]). The difference operator  $-$  of Morrill and Valentín (2014[37]) has application to linguistic exceptions.

<sup>12</sup> The derivations we give have been computer-generated from a lexicon (given in Appendix B) and parser for the categorial logic, available at <http://www.cs.upc.edu/~droman/index.php>. The implementation is a categorial parser/theorem-prover CatLog2 comprising 6000 lines of Prolog using backward chaining proof-search in the Gentzen sequent calculus (Morrill 2011[34]), and the focusing of Andreoli (1992[3]); see Morrill and Valentín (2015[40]). In addition to focusing, the implementation exploits count-invariance (van Benthem 1991[54]; Valentín, Serret and Morrill (2013[53]). In focusing, proofs are built in alternating phases of don’t care nondeterministic invertible/asynchronous rule application and focused noninvertible/synchronous rule application. The boxes in our derivations mark the focused types, which are the active types of synchronous

(14) **[john]+walks** :  $Sf$

Note that in our syntactical form the subject is a bracketed domain, and this will always be the case — implementing that subjects are weak islands. Lookup in our lexicon yields the following semantically labelled sequent:

(15)  $[\blacksquare Nt(s(m)) : j], \square(\langle \exists g Nt(s(g)) \backslash Sf \rangle) : \lambda A (Pres (\sim walk A)) \Rightarrow Sf$

The lexical types are semantically modalised outermost, and this will always be the case — implementing that word meanings are intensions/senses; the modality of the proper name subject is semantically inactive (proper names are rigid designators), while the modality of the tensed verb is semantically active (the interpretation of tensed verbs depends on the temporal reference points). The verb projects a finite sentence (feature  $f$ ) when it combines with a third person singular (bracketed) subject of any gender (the existential quantification); the actual subject is masculine (feature  $m$ ).

The derivation is as follows:

$$(16) \frac{\frac{\frac{Nt(s(m)) \Rightarrow Nt(s(m))}{\blacksquare Nt(s(m)) \Rightarrow Nt(s(m))} \blacksquare L}{\blacksquare Nt(s(m)) \Rightarrow \exists g Nt(s(g))} \exists R}{\frac{[\blacksquare Nt(s(m))] \Rightarrow \langle \exists g Nt(s(g)) \rangle \langle R}{[\blacksquare Nt(s(m))], \langle \exists g Nt(s(g)) \rangle \backslash Sf \Rightarrow Sf} \backslash L}{\frac{[\blacksquare Nt(s(m))], \langle \exists g Nt(s(g)) \rangle \backslash Sf \Rightarrow Sf}{[\blacksquare Nt(s(m))], \square(\langle \exists g Nt(s(g)) \rangle \backslash Sf) \Rightarrow Sf} \square L}$$

The flow of information in the semantic reading of derivations can be illustrated for the case in hand as follows; note that in practice the steps of this information flow are implemented by unification stepwise with derivation. First, variables for the antecedent semantics are added in the endsequent:

(17)  $[\blacksquare Nt(s(m)) : x], \square(\langle \exists g Nt(s(g)) \rangle \backslash Sf) : y \Rightarrow Sf$

Reading bottom-up, at the lowest inference step ( $\square L$ ) the verb semantics is replaced by the extension  $z$  and the subject semantics  $x$  is carried over:

$$(18) \frac{[\blacksquare Nt(s(m)) : x], \langle \exists g Nt(s(g)) \rangle \backslash Sf : z \Rightarrow Sf}{[\blacksquare Nt(s(m)) : x], \square(\langle \exists g Nt(s(g)) \rangle \backslash Sf) : y \Rightarrow Sf} \square L$$

At the second inference we propagate the subject semantics on the argument branch:

$$(19) \frac{\frac{[\blacksquare Nt(s(m)) : x] \Rightarrow \langle \exists g Nt(s(g)) \rangle \quad Sf \Rightarrow Sf}{[\blacksquare Nt(s(m)) : x], \langle \exists g Nt(s(g)) \rangle \backslash Sf : z \Rightarrow Sf} \backslash L}{[\blacksquare Nt(s(m)) : x], \square(\langle \exists g Nt(s(g)) \rangle \backslash Sf) : y \Rightarrow Sf} \square L$$

The next three inferences involve semantically transparent copying of the antecedent semantics:

rule application. All the reader needs to have in mind is that if there is a boxed type in the conclusion of an inference step then it is the active type of that inference step, i.e. the type which is decomposed reading from conclusion to premises.

$$(20) \frac{\frac{\frac{\overline{Nt(s(m)) : x \Rightarrow Nt(s(m))}}{\blacksquare Nt(s(m)) : x \Rightarrow Nt(s(m))} \blacksquare L}{\blacksquare Nt(s(m)) : x \Rightarrow \exists g Nt(s(g))} \exists R}{\blacksquare Nt(s(m)) : x \Rightarrow \langle \exists g Nt(s(g)) \rangle} \langle \rangle R \quad \frac{\overline{Sf \Rightarrow Sf}}{Sf \Rightarrow Sf} \backslash L}{\frac{[\blacksquare Nt(s(m)) : x], \langle \exists g Nt(s(g)) \rangle \backslash Sf : z \Rightarrow Sf}{[\blacksquare Nt(s(m)) : x], \square(\langle \exists g Nt(s(g)) \rangle \backslash Sf) : y \Rightarrow Sf} \square L} \backslash L$$

At the identity axiom the antecedent semantics is copied to the succedent:

$$(21) \frac{\frac{\frac{\overline{Nt(s(m)) : x \Rightarrow Nt(s(m)) : x}}{\blacksquare Nt(s(m)) : x \Rightarrow Nt(s(m))} \blacksquare L}{\blacksquare Nt(s(m)) : x \Rightarrow \exists g Nt(s(g))} \exists R}{\blacksquare Nt(s(m)) : x \Rightarrow \langle \exists g Nt(s(g)) \rangle} \langle \rangle R \quad \frac{\overline{Sf \Rightarrow Sf}}{Sf \Rightarrow Sf} \backslash L}{\frac{[\blacksquare Nt(s(m)) : x], \langle \exists g Nt(s(g)) \rangle \backslash Sf : z \Rightarrow Sf}{[\blacksquare Nt(s(m)) : x], \square(\langle \exists g Nt(s(g)) \rangle \backslash Sf) : y \Rightarrow Sf} \square L} \backslash L$$

In a following phase the succedent semantics is copied from premises to conclusions as far as the root of the argument branch:

$$(22) \frac{\frac{\frac{\overline{Nt(s(m)) : x \Rightarrow Nt(s(m)) : x}}{\blacksquare Nt(s(m)) : x \Rightarrow Nt(s(m)) : x} \blacksquare L}{\blacksquare Nt(s(m)) : x \Rightarrow \exists g Nt(s(g)) : x} \exists R}{\blacksquare Nt(s(m)) : x \Rightarrow \langle \exists g Nt(s(g)) : x \rangle} \langle \rangle R \quad \frac{\overline{Sf \Rightarrow Sf}}{Sf \Rightarrow Sf} \backslash L}{\frac{[\blacksquare Nt(s(m)) : x], \langle \exists g Nt(s(g)) \rangle \backslash Sf : z \Rightarrow Sf}{[\blacksquare Nt(s(m)) : x], \square(\langle \exists g Nt(s(g)) \rangle \backslash Sf) : y \Rightarrow Sf} \square L} \backslash L$$

Now the functor value semantics in the antecedent of the value branch is labelled with a new variable  $w$ :

$$(23) \frac{\frac{\frac{\overline{Nt(s(m)) : x \Rightarrow Nt(s(m)) : x}}{\blacksquare Nt(s(m)) : x \Rightarrow Nt(s(m)) : x} \blacksquare L}{\blacksquare Nt(s(m)) : x \Rightarrow \exists g Nt(s(g)) : x} \exists R}{\blacksquare Nt(s(m)) : x \Rightarrow \langle \exists g Nt(s(g)) : x \rangle} \langle \rangle R \quad \frac{\overline{Sf : w \Rightarrow Sf}}{Sf : w \Rightarrow Sf} \backslash L}{\frac{[\blacksquare Nt(s(m)) : x], \langle \exists g Nt(s(g)) \rangle \backslash Sf : z \Rightarrow Sf}{[\blacksquare Nt(s(m)) : x], \square(\langle \exists g Nt(s(g)) \rangle \backslash Sf) : y \Rightarrow Sf} \square L} \backslash L$$

At the id axiom this semantics is copied from antecedent to succedent:

$$(24) \frac{\frac{\frac{\overline{Nt(s(m)) : x} \Rightarrow Nt(s(m)) : x}{\blacksquare Nt(s(m)) : j} \Rightarrow Nt(s(m)) : j}{\blacksquare Nt(s(m)) : x} \Rightarrow \exists g Nt(s(g)) : x}{\blacksquare Nt(s(m)) : x} \Rightarrow \langle \exists g Nt(s(g)) : x \rangle \exists R}{\frac{\overline{Sf : w} \Rightarrow Sf : w}{\blacksquare Nt(s(m)) : x, \langle \exists g Nt(s(g)) \setminus Sf \rangle : z \Rightarrow Sf} \Rightarrow Sf} \setminus L}{\blacksquare Nt(s(m)) : x, \square(\langle \exists g Nt(s(g)) \setminus Sf \rangle) : y \Rightarrow Sf} \square L$$

In the  $\setminus L$  conclusion succedent the semantics of the major premise is subject to the substitution of  $w$  by the functional application of the functor  $z$  to the argument  $x$ :

$$(25) \frac{\frac{\frac{\overline{Nt(s(m)) : x} \Rightarrow Nt(s(m)) : x}{\blacksquare Nt(s(m)) : x} \Rightarrow Nt(s(m)) : x}{\blacksquare Nt(s(m)) : x} \Rightarrow \exists g Nt(s(g)) : x}{\blacksquare Nt(s(m)) : x} \Rightarrow \langle \exists g Nt(s(g)) : x \rangle \exists R}{\frac{\overline{Sf : w} \Rightarrow Sf : w}{\blacksquare Nt(s(m)) : x, \langle \exists g Nt(s(g)) \setminus Sf \rangle : z \Rightarrow Sf : w\{(z x)/w\} = (z x)} \Rightarrow Sf} \setminus L}{\blacksquare Nt(s(m)) : x, \square(\langle \exists g Nt(s(g)) \setminus Sf \rangle) : y \Rightarrow Sf} \square L$$

And thence to the conclusion of the endsequent:

$$(26) \frac{\frac{\frac{\overline{Nt(s(m)) : x} \Rightarrow Nt(s(m)) : x}{\blacksquare Nt(s(m)) : x} \Rightarrow Nt(s(m)) : x}{\blacksquare Nt(s(m)) : x} \Rightarrow \exists g Nt(s(g)) : x}{\blacksquare Nt(s(m)) : x} \Rightarrow \langle \exists g Nt(s(g)) : x \rangle \exists R}{\frac{\overline{Sf : w} \Rightarrow Sf : w}{\blacksquare Nt(s(m)) : x, \langle \exists g Nt(s(g)) \setminus Sf \rangle : z \Rightarrow Sf : (z x)} \Rightarrow Sf} \setminus L}{\blacksquare Nt(s(m)) : x, \square(\langle \exists g Nt(s(g)) \setminus Sf \rangle) : y \Rightarrow Sf : (z x)\{y/z\} = (\sphericalangle y x)} \square L$$

Now we can substitute in the lexical semantics  $j$  for *John* ( $x$ ) and  $\hat{\lambda}A(\text{Pres } (\sphericalangle \text{walk } A))$  for *walks* ( $y$ ) and evaluate:<sup>13</sup>

$$(27) \begin{aligned} & (\hat{\lambda}A(\text{Pres } (\sphericalangle \text{walk } A)) j) = \\ & (\lambda A(\text{Pres } (\sphericalangle \text{walk } A)) j) = \\ & (\text{Pres } (\sphericalangle \text{walk } j)) \end{aligned}$$

(As we have said, this elucidation is not exactly how CatLog2 extracts semantics; CatLog2 uses unification and instantiation of metavariables to deliver in a single pass the unevaluated semantics of the upwards and downward phases, and then normalises.)

By way of a second example, the following is a simple transitive sentence:

$$(28) [\text{john}] + \text{loves} + \text{mary} : Sf$$

Lexical lookup yields:

$$(29) [\blacksquare Nt(s(m)) : j], \square(\langle \exists g Nt(s(g)) \setminus Sf \rangle / \exists a Na) : \hat{\lambda}A \lambda B (\text{Pres } (\sphericalangle \text{love } A) B), \blacksquare Nt(s(f)) : m \Rightarrow Sf$$

$$\begin{array}{c}
\boxed{Nt(s(m))} \Rightarrow Nt(s(m)) \\
\boxed{\blacksquare Nt(s(m))} \Rightarrow Nt(s(m)) \quad \blacksquare L \\
\boxed{\blacksquare Nt(s(m))} \Rightarrow \exists g Nt(s(g)) \quad \exists R \\
\boxed{Nt(s(f))} \Rightarrow Nt(s(f)) \quad \blacksquare L \\
\boxed{\blacksquare Nt(s(f))} \Rightarrow Nt(s(f)) \quad \blacksquare L \\
\boxed{\blacksquare Nt(s(m))} \Rightarrow \langle \exists g Nt(s(g)) \rangle \quad \langle R \\
\boxed{\blacksquare Nt(s(f))} \Rightarrow \exists a Na \quad \exists R \\
\boxed{\blacksquare Nt(s(m))}, \boxed{\langle \exists g Nt(s(g)) \rangle S f} \Rightarrow S f \quad \backslash L \\
\boxed{\blacksquare Nt(s(m))}, \boxed{\langle \exists g Nt(s(g)) \rangle S f} \Rightarrow S f \quad / L \\
\boxed{\blacksquare Nt(s(m))}, \boxed{\langle \langle \exists g Nt(s(g)) \rangle S f \rangle \exists a Na} \Rightarrow S f \quad \square L \\
\boxed{\blacksquare Nt(s(m))}, \boxed{\square \langle \langle \exists g Nt(s(g)) \rangle S f \rangle \exists a Na}, \boxed{\blacksquare Nt(s(f))} \Rightarrow S f \quad \square L
\end{array}$$

Fig. 8 Derivation for *John loves Mary*

There is the derivation given in Figure 8. Reading upwards from the endsequent, the first inference removes the intensionality modality from the transitive verb, and then over left selects the object to analyse as the argument of the transitive verb; this is done by existential right instantiating the agreement feature to third person singular feminine, followed by (semantically inactive) intensionality modality left. The right hand branch is the same as for example (14) after the first inference. All this delivers semantics:

(30)  $(Pres (\sim love m) j)$

The next example has a subordinate clause:

(31)  $[john]+thinks+[mary]+walks : S f$

Lexical lookup yields the following; note that the propositional attitude verb is polymorphic with respect to a complementised or uncomplementised sentential argument, expressed with a semantically inactive additive disjunction:

(32)  $[\blacksquare Nt(s(m)) : j], \square(\langle \exists g Nt(s(g)) \rangle S f) / (C P that \sqcup \square S f) : \hat{\lambda} A \lambda B (Pres (\sim think A) B),$   
 $[\blacksquare Nt(s(f)) : m], \square(\langle \exists g Nt(s(g)) \rangle S f) : \hat{\lambda} C (Pres (\sim walk C)) \Rightarrow S f$

This has the derivation given in Figure 9. Reading bottom-up, following elimination of the intensionality modality of the propositional attitude verb, over left partitions in such a way as to supply the subordinate clause as the propositional argument. Again, the righthand subtree is the same as for example (14) after the first inference. In the lefthand subtree semantically inactive additive conjunction right selects the modalised uncomplementized sentence type. The succedent modality is removed, this being licensed by the fact that all the antecedent types are modalised, and the remaining derivation is also like that for example (14). The derivation delivers semantics:

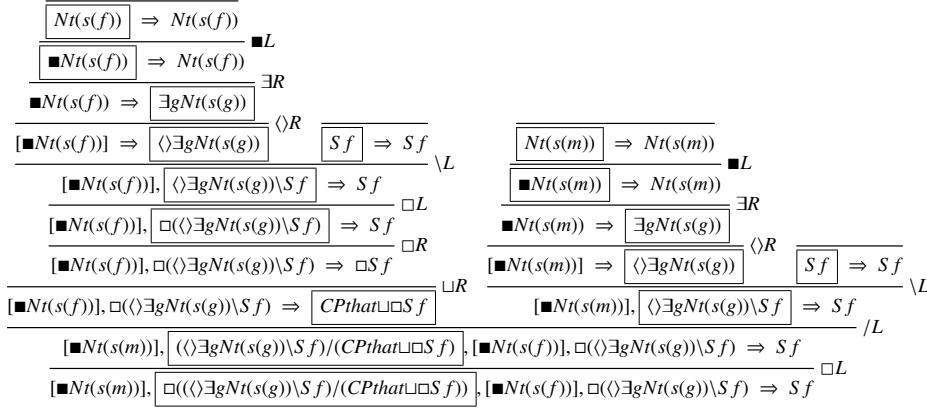
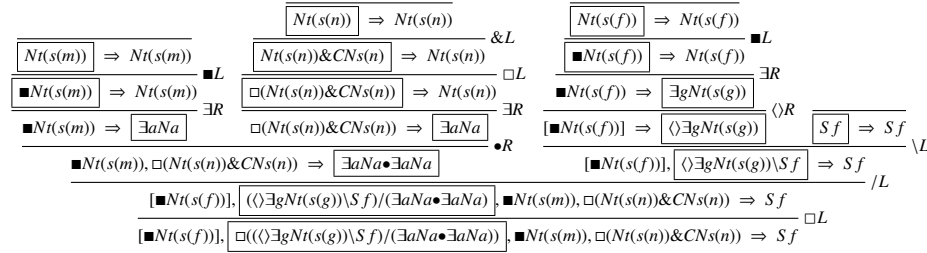
(33)  $(Pres (\sim think (\sim Pres (\sim walk m))) j)$

The following example involves a ditransitive verb:

(34)  $[mary]+buys+john+coffee : S f$

Lexical lookup is as follows; note the use of (continuous) product (multiplicative conjunction) for the ditransitive verb, and the use of additive conjunction for the polymorphism of the mass noun *coffee* which can appear either as a bare nominal or with an article:

<sup>13</sup> Montague's Intensional Logic assigned nonlogical constants of type  $\tau$  a denotation in the intension of  $\tau$  and then interpreted a constant with respect to a world as its extension in that world. By contrast our semantic representation language assigns constants denotations in their own type, so our semantic representations have explicit extensionalizations of intensional constants.

Fig. 9 Derivation for *John thinks Mary walks*Fig. 10 Derivation for *Mary buys John coffee*

$$(35) [\blacksquare Nt(s(f)) : m], \square(\langle \exists g Nt(s(g)) \rangle \backslash S f) / (\exists a N a \bullet \exists a N a) : \hat{\sim} \lambda A \lambda B (Pres ((\sim buy \pi_1 A) \pi_2 A) B), [\blacksquare Nt(s(m)) : j], \square(Nt(s(n)) \& C N s(n)) : \hat{\sim} ((gen \check{coffee}, \check{coffee}) \Rightarrow S f$$

There is the derivation given in Figure 10. After removal of the outer modality of the ditransitive verb, the partitioning of over left selects the two objects as the verb's product argument, partitioned in turn by product right. The indirect object *John* is analysed by existential right and inactive modality left inferences; the direct object *coffee* is analysed by existential right and (active) modality left inferences followed by selection of the bare noun type by additive conjunction left. The rightmost subtree is as usual for an intransitive sentence. This delivers semantics as follows in which a 'generic' operator applies to *coffee*:

$$(36) (Pres ((\sim buy j) (gen \check{coffee}) m))$$

The next example includes a definite article:

$$(37) [\mathbf{the+man}]+\mathbf{walks} : S f$$

We treat the definite article simply as an iota operator which returns the unique individual in the context of discourse satisfying its common noun argument (Carpenter 1997[6]); this unicity is presupposed by the use of the definite. Lexical lookup yields the semantically labelled sequent:

$$(38) [\blacksquare \forall n (Nt(n) / C N n) : \iota, \square C N s(m) : man], \square(\langle \exists g Nt(s(g)) \rangle \backslash S f) : \hat{\sim} \lambda A (Pres (\check{walk} A)) \Rightarrow S f$$



$$\begin{array}{c}
\boxed{CNs(m)} \Rightarrow CNs(m) \\
\boxed{\square CNs(m)} \Rightarrow CNs(m) \quad \square L \\
\boxed{Nt(s(m))} \Rightarrow Nt(s(m)) \\
\boxed{\square CNs(m)} \Rightarrow CNs(m), \boxed{Nt(s(m))} \Rightarrow Nt(s(m)) \quad /L \\
\boxed{Nt(s(m))/CNs(m)}, \square CNs(m) \Rightarrow Nt(s(m)) \quad \forall L \\
\boxed{\forall n(Nt(n)/CNn)}, \square CNs(m) \Rightarrow Nt(s(m)) \quad \forall L \\
\boxed{\blacksquare \forall n(Nt(n)/CNn)}, \square CNs(m) \Rightarrow Nt(s(m)) \quad \blacksquare L \\
\boxed{\blacksquare \forall n(Nt(n)/CNn)}, \square CNs(m) \Rightarrow \boxed{\exists g Nt(s(g))} \quad \exists R \\
\boxed{\blacksquare \forall n(Nt(n)/CNn)}, \square CNs(m) \Rightarrow \boxed{\langle \rangle \exists g Nt(s(g))} \quad \langle \rangle R \\
\boxed{\blacksquare \forall n(Nt(n)/CNn)}, \square CNs(m), \boxed{Sf} \Rightarrow Sf \quad \backslash L \\
\boxed{\blacksquare \forall n(Nt(n)/CNn)}, \square CNs(m), \boxed{\langle \rangle \exists g Nt(s(g)) \backslash Sf} \Rightarrow Sf \quad \square L \\
\boxed{\blacksquare \forall n(Nt(n)/CNn)}, \square CNs(m), \boxed{\langle \rangle \exists g Nt(s(g)) \backslash Sf} \Rightarrow Sf \quad \square L
\end{array}$$

Fig. 11 Derivation for *The man walks*

There is the derivation given in Figure 11. This is like the derivation of an intransitive sentence before, but with the analysis of the definite noun phrase subject at the top left. The derivation delivers semantics:

$$(39) (Pres \checkmark walk (\iota \checkmark man))$$

The next two examples have adverbial and adnominal prepositional modification respectively. We consider the adverbial case first:

$$(40) [\mathbf{john}] + \mathbf{walks} + \mathbf{from} + \mathbf{edinburgh} : Sf$$

Lexical lookup inserts a single value-polymorphic prepositional type, which uses semantically active additive conjunction:

$$(41) [\blacksquare Nt(s(m)) : j], \square(\langle \rangle \exists g Nt(s(g)) \backslash Sf) : \hat{\lambda}A(Pres \checkmark walk A), \square((\forall a \forall f((\langle \rangle Na \backslash Sf) \backslash (\langle \rangle Na \backslash Sf)) \& \forall n(CNn \backslash CNn)) / \exists b Nb) : \hat{\lambda}B(\checkmark fromadv B), (\checkmark fromadn B)), \blacksquare Nt(s(n)) : e \Rightarrow Sf$$

There is the derivation given in Figure 12. After elimination of the outer modality of the preposition, over left selects as the prepositional argument the prepositional object, which is analysed in the leftmost subtree. In the sister subtree additive conjunction left selects the adverbial type for the prepositional phrase and for all left instantiates the subject agreement and verb form features to third person singular masculine, and finite. Following under left, in the middle subtree *walks* is analysed as the intransitive verb second argument of the adverbial preposition; note the analysis of the higher-order type by the under right rule, which lowers the conclusion succedent hypothetical subtype into the premise antecedent. The rightmost subtree is an intransitive sentence case again. All this delivers the semantics:

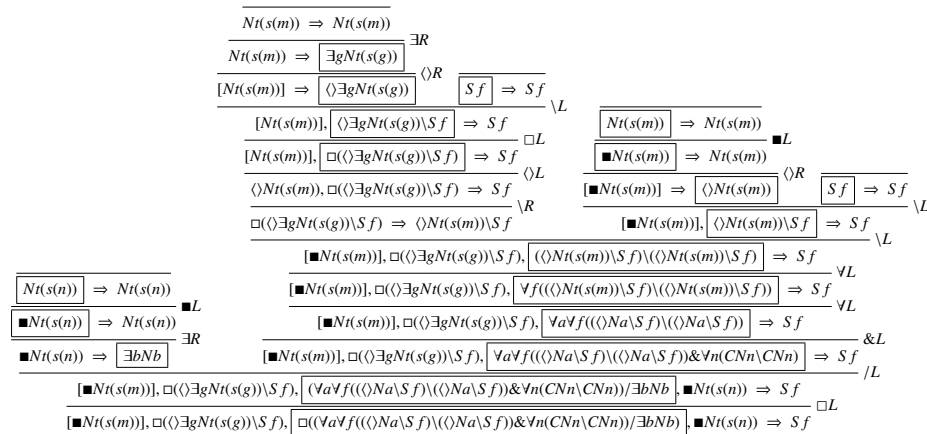
$$(42) (((\checkmark fromadv e) \lambda B(Pres \checkmark walk B))) j$$

The adnominal case is:

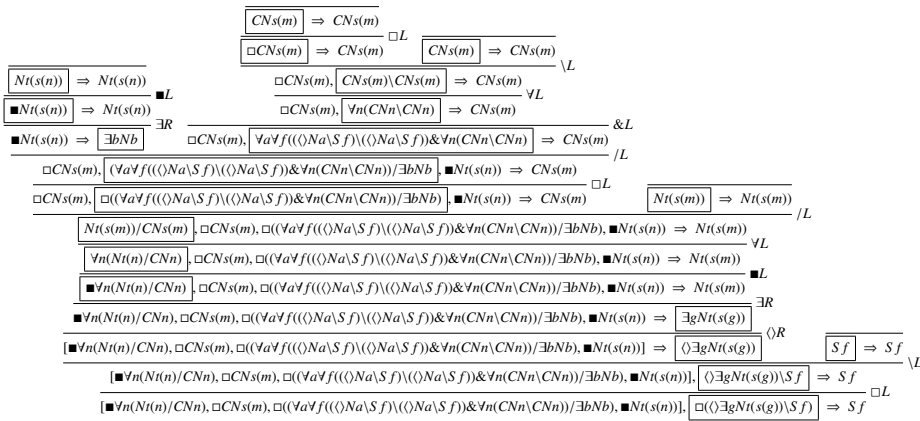
$$(43) [\mathbf{the} + \mathbf{man} + \mathbf{from} + \mathbf{edinburgh}] + \mathbf{walks} : Sf$$

Lexical lookup yields:

$$(44) [\blacksquare \forall n(Nt(n)/CNn) : \iota, \square CNs(m) : man, \square((\forall a \forall f((\langle \rangle Na \backslash Sf) \backslash (\langle \rangle Na \backslash Sf)) \& \forall n(CNn \backslash CNn)) / \exists b Nb) : \hat{\lambda}A(\checkmark fromadv A), (\checkmark fromadn A)), \blacksquare Nt(s(n)) : e], \square(\langle \rangle \exists g Nt(s(g)) \backslash Sf) : \hat{\lambda}B(Pres \checkmark walk B) \Rightarrow Sf$$



**Fig. 12** Derivation for *John walks from Edinburgh*



**Fig. 13** Derivation for *The man from Edinburgh walks*

There is the derivation given in Figure 13. In the first two steps the intransitive verb *walks* is prepared to apply to the complex subject. Bracket right and exists right follow, then (in-)active) modality left and for all left on the determiner, which then applies to the complex common noun. The result of modality left on the preposition applies to the prepositional object and in the major premise additive conjunction left selects the adnominal prepositional type. The semantics delivered is:

$$(45) \textit{ (Pres } (\checkmark \textit{ walk } (\iota ((\checkmark \textit{ from } \textit{adn } e) \checkmark \textit{ man})))) )}$$

The last two initial examples involve the copula with nominal and (intersective) adjectival complementation respectively. We consider first the nominal case:

$$(46) \textit{ [tully]+is+cicero} : S f$$

Lexical lookup inserts a single argument-polymorphic copula type, which uses both semantically active and semantically inactive additive disjunction:<sup>14</sup>

<sup>14</sup> The difference operator (Morrill and Valentín 2014[37]) for linguistic exceptions is also used. It involves negation as failure, which cannot easily be displayed. We do not dwell on this operator here.

$$\begin{array}{c}
\frac{\frac{\frac{Nt(s(A)) \Rightarrow Nt(s(A))}{\forall L} \quad \frac{\forall g Nt(s(g)) \Rightarrow Nt(s(A))}{\blacksquare L} \quad \frac{\blacksquare \forall g Nt(s(g)) \Rightarrow Nt(s(A))}{\exists R} \quad \frac{\blacksquare \forall g Nt(s(g)) \Rightarrow \exists a Na}{\oplus R}}{\blacksquare \forall g Nt(s(g)) \Rightarrow \exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)} \\
\frac{\frac{\frac{Nt(s(m)) \Rightarrow Nt(s(m))}{\blacksquare L} \quad \frac{\blacksquare Nt(s(m)) \Rightarrow Nt(s(m))}{\exists R} \quad \frac{\blacksquare Nt(s(m)) \Rightarrow \exists g Nt(s(g))}{\langle R} \quad \frac{\langle \exists g Nt(s(g)) \rangle \Rightarrow Sf}{\exists R} \quad \frac{Sf \Rightarrow Sf}{\backslash L}}{\blacksquare Nt(s(m)), \langle \exists g Nt(s(g)) \rangle \backslash Sf \Rightarrow Sf} \\
\frac{[\blacksquare Nt(s(m))], \langle \exists g Nt(s(g)) \rangle \backslash Sf / (\exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)) \quad \blacksquare \forall g Nt(s(g)) \Rightarrow Sf}{[\blacksquare Nt(s(m))], \blacksquare \langle \exists g Nt(s(g)) \rangle \backslash Sf / (\exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)) \quad \blacksquare \forall g Nt(s(g)) \Rightarrow Sf} \quad \blacksquare L
\end{array}$$

Fig. 14 Derivation for *Tully is Cicero*

$$\begin{array}{c}
\frac{\frac{CNA \Rightarrow CNA \quad \boxed{CNA} \Rightarrow CNA}{/L} \quad \frac{\boxed{CNA/CNA}, CNA \Rightarrow CNA}{\forall L} \quad \frac{\forall n(CNn/CNn), CNA \Rightarrow CNA}{\square L} \quad \frac{\square \forall n(CNn/CNn), CNA \Rightarrow CNA}{/R} \quad \frac{\square \forall n(CNn/CNn) \Rightarrow CNA/CNA}{\sqcup R} \\
\frac{\frac{\square \forall n(CNn/CNn) \Rightarrow \exists g((CNg/CNg) \sqcup (CNg \setminus CNg))}{\exists R} \quad \frac{\square \forall n(CNn/CNn) \Rightarrow \exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I}{-R} \quad \frac{\square \forall n(CNn/CNn) \Rightarrow \exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)}{\oplus R}}{\square \forall n(CNn/CNn) \Rightarrow \exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)} \\
\frac{[\blacksquare Nt(s(m))], \langle \exists g Nt(s(g)) \rangle \backslash Sf / (\exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)) \quad \square \forall n(CNn/CNn) \Rightarrow Sf}{[\blacksquare Nt(s(m))], \blacksquare \langle \exists g Nt(s(g)) \rangle \backslash Sf / (\exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)) \quad \square \forall n(CNn/CNn) \Rightarrow Sf} \quad \blacksquare L
\end{array}$$

Fig. 15 Derivation for *Tully is humanist*

$$(47) [\blacksquare Nt(s(m)) : t], \blacksquare \langle \exists g Nt(s(g)) \rangle \backslash Sf / (\exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)) : \lambda A \lambda B (Pres (A \rightarrow C.[B = C]; D.((D \lambda E[E = B]) B))), \blacksquare \forall g Nt(s(g)) : cicero \Rightarrow Sf$$

There is the derivation given in Figure 14. After elimination of the outer copula modality the copula is applied to its nominal complement. Additive disjunction right selects the first, nominal, disjunct. The derivation delivers semantics:

$$(48) (Pres [t = c])$$

The (intersective) adjectival case is:

$$(49) [\mathbf{tully}] + \mathbf{is} + \mathbf{humanist} : Sf$$

Lexical lookup yields:

$$(50) [\blacksquare Nt(s(m)) : t], \blacksquare \langle \exists g Nt(s(g)) \rangle \backslash Sf / (\exists a Na \oplus (\exists g((CNg/CNg) \sqcup (CNg \setminus CNg)) - I)) : \lambda A \lambda B (Pres (A \rightarrow C.[B = C]; D.((D \lambda E[E = B]) B))), \square \forall n(CNn/CNn) : \lambda F \lambda G [(F G) \wedge (\sim \mathbf{humanist} G)] \Rightarrow Sf$$

There is the derivation given in Figure 15. After elimination of its outer modality, the copula is applied to its adjectival complement. Semantically active additive disjunction right selects the second disjunct. The difference right rule checks that the antecedent is not empty, but this is not displayed. Exists right substitutes the existentially quantified variable for a metavariable  $A$  and semantically inactive additive disjunction right then selects the adjectival disjunct. The following semantics is delivered:

(51) (*Pres* (*humanist t*))

#### 4 Routes we will not take

Szaboloczi (1983[49]) and Steedman (1987[48]) aim to account for parasitic gaps in combinatory categorial grammar (CCG) by means of the combinator  $\mathbf{S}$  such that  $\mathbf{S} x y z = (x z) (y z)$ , for example positing a combinatory schema:

(52)  $y: Y/Z, x: (Y \setminus X)/Z \Rightarrow \mathbf{S} x y: X/Z$

Such a schema makes no sense from the point of view of the logicisation of grammar pursued here. The rule is not Lambek-valid and any semantics validating it would also validate schemata which overgenerate massively. So much the worse, the proponents of CCG would say, for grammar as logic: grammar is a formal system but not a logic, and one should not care about things like soundness and completeness.

CCG and type logical grammar agree on the task of defining syntax and semantics of the (object) natural language. What is curious about CCG is that at the same time it declines to consider syntax (proof theory) and semantics (model theory) of the (meta-)linguistic formalism. A CCG account of parasitic gaps, which employs just the directional slashes and a minimum of combinatory schemata, must capture the effects of structural inhibition (islands) and structural facilitation (parasiticity) by good fortune in the interaction of the combinatory schemata chosen and the categorial types occurring in grammar. In our approach control of structural inhibition by bracket modalities and control of structural facilitation by exponentials are separated in an analysis recognising the distinct algebraic roles of variation from an associative and linear regime. This type logical approach lets us state our analysis with clarity in the knowledge that whatever the empirical adequacy, metatheoretical facts are known. In CCG the metatheory is not logically investigated.

It is interesting to ask why we treat medial extraction here with  $!$  rather than with  $\uparrow$  as illustrated in Section 2 (cf. also Moortgat 1988[27]; Muskens 2003[45]; Mihaliček and Pollard 2012[26]; Barker and Shan 2015[4]; and Kubota and Levine 2015[20]). The answer is that, on the one hand,  $\uparrow$  as defined does not respect island constraints and, on the other hand,  $\uparrow$  does not extend to parasitic gaps: it is unclear how a single local inference rule can account for unbounded recursive nesting of parasitic gaps in subislands. Our treatment in terms of  $!$  both respects islands, and extends to (unbounded numbers of) parasitic gaps through iteration of contraction.<sup>15</sup>

An option available in both CCG and type logical grammar is to attempt to analyse the non-linearity of parasitic extraction not syntactically but lexically. Thus for example Jansche and Vasishth (2002[15]) propose induction of parasitic gaps in adverbial clauses by a lexicalised gap-duplicating effect in the adverbial head. All contexts allowing parasitic gaps would require a corresponding gap-duplicating lexical ambiguity. The appeal to lexical ambiguity in lexical grammar formalisms is as frequent as it is untenable. Every ambiguity of every item doubles the lexical insertion search space. And in the case in hand there is to our knowledge no independent evidence, such as difference in meaning, for lexical ambiguity underlying parasitic extraction. We continue on the assumption that it is indeed a syntactic phenomenon.

<sup>15</sup> We note that the discontinuity operators serve to account for the pied-piping aspect of relativisation (see e.g. Morrill, Valentín and Fadda 2011[42]), though we do not go into that here.

## 5 Relativisation

Our account of relativisation rests on the lexical projection of islands by argument bracketing ( $\langle \rangle$ ) and value antibracketing ( $[ ]^{-1}$ ), and a single relative pronoun type of overall shape  $R/(\langle \rangle N \sqcap !N) \setminus S$  for both subject and object relativisation. In analysis of the body of relative clauses the higher order succedent argument essentially of form  $\langle \rangle N \sqcap !N$  is lowered into the antecedent according to the deduction theorem; in subject relativisation  $\langle \rangle N$  is selected by conjunction left, and satisfies the (bracketed) subject valency.

In object relativisation  $!N$  is selected. When the  $!L$  rule is applied to  $!N$ , the hypothetical subtype  $N$  moves into the stoup, from where it can move by  $!P$  to any (nonisland) position in its zone, realising nonparasitic extraction.

However, in addition it can be copied by  $!C$  to the stoup of a newly created weak island domain, realising parasitic extraction. The  $N$  in the outer stoup can be copied by  $!C$  repeatedly, capturing that there may be parasitic gaps in any number of local weak islands; at the end of this process it moves by  $!P$  to a host position in its zone. The  $N$  in an inner stoup can also be copied by  $!C$  to the stoup of any number of newly created weak subislands, and so on recursively, capturing that parasitic gaps can also be hosts to further parasitic gaps; finally the stoup contents are copied by  $!P$  to extraction sites in their zone.

In this section we analyse examples illustrating the account of relativisation. The first example is a minimal subject relativisation; note that the relative clause is doubly bracketed, corresponding to the fact that relative clauses are strong islands:<sup>16</sup>

(53)  $\text{man} + [[\text{that} + \text{walks}]] : CNs(m)$

Lexical lookup yields the following, where there is semantically inactive additive conjunction of the hypothetical subtypes  $\langle \rangle N$  for subject relativisation and  $! \blacksquare N$  for object relativisation; the (semantically inactive) modality on the object gap subtype is to permit object relativisation from embedded modal/intensional domains:<sup>17</sup>

(54)  $\square CNs(m) : \text{man}, [[\blacksquare \forall n ([ ]^{-1} [ ]^{-1} (CNn \setminus CNn) / \blacksquare ((\langle \rangle Nt(n) \sqcap ! \blacksquare Nt(n)) \setminus S f)) : \lambda \lambda \lambda B \lambda C [(B C) \wedge (A C)], \square (\langle \rangle \exists g Nt(s(g)) \setminus S f) : \wedge \lambda D (Pres (\sim \text{walk } D))] \Rightarrow CNs(m)$

There is the derivation in Figure 16, which starts with the relative clause doubly bracketed (this will always be the case for relativisation). After elimination of the outer (semantically inactive) modality of the relative pronoun, universal left instantiates it to agree with masculine singular. Then  $/L$  partitions in such a way as to select the intransitive verb body of the relative clause as argument of the relative pronoun. In the righthand, value, subtree two antibracket eliminations cancel the double brackets before the head common noun is modified. In the lefthand, argument, subtree (inactive) box right is enabled since the antecedent is modalised, and under right then lowers the additively conjoined hypothetical subtypes into the antecedent. Observe how in the lefthand subtree  $\sqcap L$  selects the subject relativisation hypothetical subtype  $\langle \rangle Nt(s(m))$ ; the remaining subderivation is the usual intransitive sentence analysis. This delivers the required semantics:

(55)  $\lambda C [(\sim \text{man } C) \wedge (Pres (\sim \text{walk } C))]$

<sup>16</sup> As we will see relative clauses themselves, being doubly bracketed, will not allow parasitic gaps.

<sup>17</sup> The body of the relative clause is marked as a (semantically inactive) modal domain in order to make it a scope island. Thus where, say, *everyone* has a type  $\square((S \uparrow N) \downarrow S)$  the unmodalised hypothetical subtype  $N$  cannot be bound outside the modal domain of the body of a relative clause in which *everyone* occurs. This account of relative clauses as scope islands operates on essentially the same lines as the capture of Principle A by modalities (see Morrill 1990[32]).





Fig. 17 Derivation for *The man that Mary loves walks*

$$(60) \quad [\blacksquare \forall n(Nt(n)/CNn) : \iota, \square CNs(m) : man, [[\blacksquare \forall n([\ ]^{-1}[\ ]^{-1}(CNn \setminus CNn) / \blacksquare((\langle \rangle Nt(n) \square! \blacksquare Nt(n)) \setminus S f)) : \lambda A \lambda B \lambda C[(B C) \wedge (A C)], [\blacksquare Nt(s(m)) : j], \square((\langle \rangle \exists g Nt(s(g)) \setminus S f) / (C P \text{that} \square \square S f)) : \wedge D \lambda E (Pres (\sim \text{think } D) E)), [\blacksquare Nt(s(f)) : m], \square((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \wedge F \lambda G (Pres (\sim \text{love } F) G))]]], \square((\langle \rangle \exists g Nt(s(g)) \setminus S f) : \wedge H (Pres (\sim \text{walk } H)) \Rightarrow S f]$$

There is the derivation given in Figure 18. Inference up as far as  $\textcircled{1}$  brings us to analysis of the complex common noun phrase in the lefthand subtree. The following preparation of the relative pronoun and double bracket cancellation of its value are as usual. After modality right and under right on the relative pronoun higher-order argument,  $\square L$  selects the object relativisation hypothetical subtype and  $!L$  moves this into the stoup. In the stoup it percolates to the subordinate clause, (observe how  $\square R$  selects the uncomplementised sentential argument type of the propositional attitude verb) and there  $!P$  moves it into position to satisfy the embedded clause object valency. This delivers the correct semantics:

$$(61) \quad (Pres (\sim \text{walk } (\iota \lambda D[(\sim \text{man } D) \wedge (Pres ((\sim \text{think } (\sim (Pres ((\sim \text{love } D) m))) j))]))))$$

There follows an example of medial object relativisation (the gap is in a non-peripheral position left of the adverb):

$$(62) \quad \text{man} + [[\text{that} + [\text{mary}] + \text{likes} + \text{today}]] : CNs(m)$$

Appropriate lexical lookup yields:

$$(63) \quad \square CNs(m) : man, [[\blacksquare \forall n([\ ]^{-1}[\ ]^{-1}(CNn \setminus CNn) / \blacksquare((\langle \rangle Nt(n) \square! \blacksquare Nt(n)) \setminus S f)) : \lambda A \lambda B \lambda C[(B C) \wedge (A C)], [\blacksquare Nt(s(f)) : m], \square((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \wedge D \lambda E (Pres (\sim \text{like } D) E)), \square \forall a \forall f((\langle \rangle Na \setminus S f) \setminus (\langle \rangle Na \setminus S f)) : \wedge F \lambda G (\sim \text{today } (F G))] \Rightarrow CNs(m)$$

There is the derivation in Figure 19. Analysis of the complex common noun phrase begins at the lefthand subtree  $\textcircled{1}$ . After modality right and conditionalisation of the conjoined hypothetical subtype, additive conjunction left applies to this latter to select the object relativisation subtype, which then moves into the stoup. After preparation of the adverb the stoup contents pass into its argument subbranch. Note how the object relativisation hypothetical gap subtype percolates in the stoup to satisfy the transitive verb object valency. The semantics delivered is:

$$(64) \quad \lambda C[(\sim \text{man } C) \wedge (\sim \text{today } (Pres ((\sim \text{like } C) m)))]$$

As we remarked at the beginning of Section 3 subjects are weak islands (the Subject Condition of Chomsky 1973[7]); accordingly in our CatLog2 fragment there is no derivation of simple relativisation from a subject such as:

$$(65) \quad \text{man} + [[\text{that} + [\text{the friends of}] + \text{walk}]] : CNs(m)$$

This is because *walk* projects brackets around its subject, but the permutation of the  $!$  hypothetical gap subtype issued by the relative pronoun is limited to its zone and cannot penetrate a bracketed subzone. Roughly, the derivation blocks at  $*$  in:

$$(66) \quad \frac{\frac{[N/CN, CN/PP, PP/N, N], N \setminus S \Rightarrow S}{N; [N/CN, CN/PP, PP/N], N \setminus S \Rightarrow S} *!P}{!N, [N/CN, CN/PP, PP/N], N \setminus S \Rightarrow S} !L}{[N/CN, CN/PP, PP/N], N \setminus S \Rightarrow !N \setminus S} \setminus R$$

However, a weak island ‘parasitic’ gap can be licensed by a host gap:





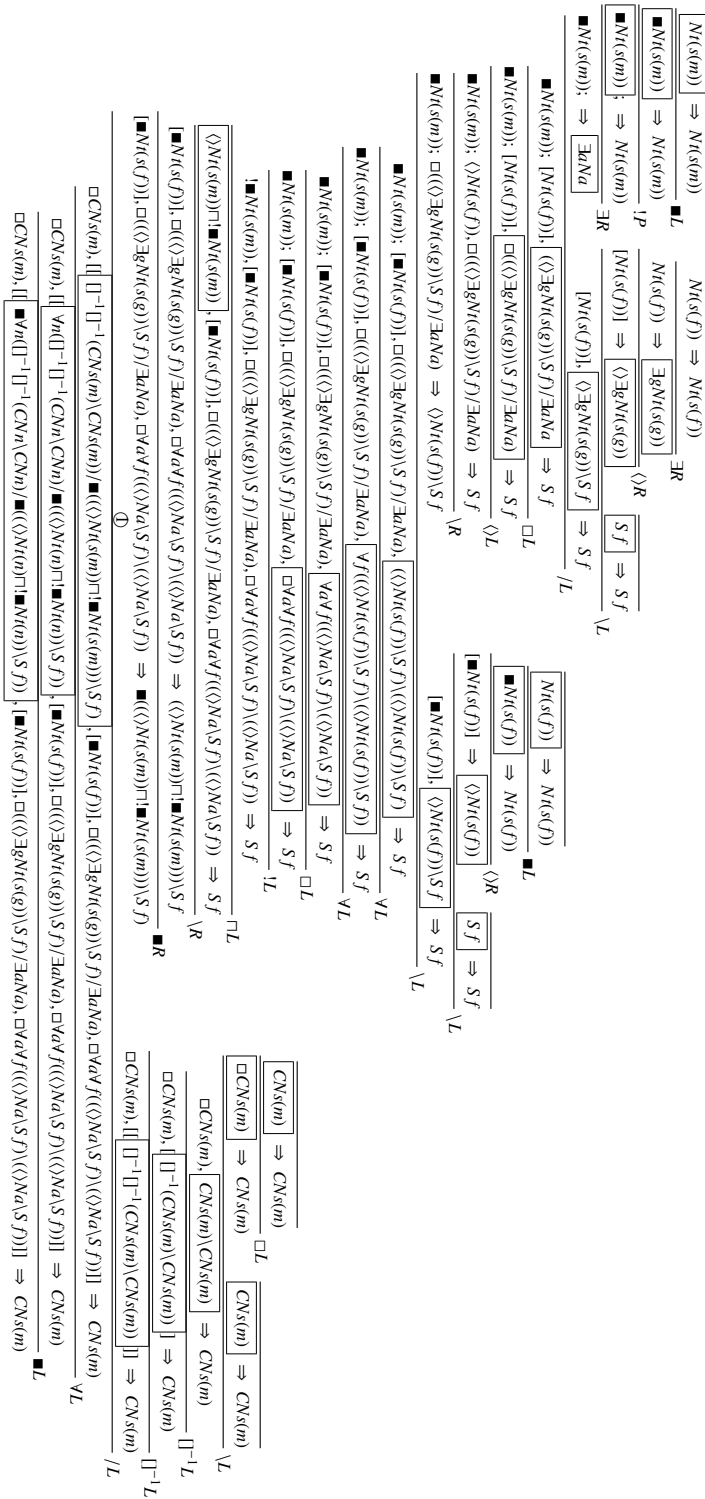


Fig. 19 Derivation of medial relativisation: *man that Mary likes today*

(67) **man+[[that+the+friends+of+admire]]** :  $CNs(m)$

Lexical lookup yields:<sup>18</sup>

(68)  $\square CNs(m) : man, [[\blacksquare \forall n([\ ]^{-1}[\ ]^{-1}(CNn \setminus CNn) / \blacksquare((\langle \rangle Nt(n) \sqcap ! \blacksquare Nt(n)) \setminus S f)) :$   
 $\lambda A \lambda B \lambda C[(B C) \wedge (A C)], \blacksquare \forall n(Nt(n) / CNn) : \iota, \square(CNp / PPof) : friends,$   
 $\square((\forall n(CNn \setminus CNn) / \blacksquare \exists b Nb) \& (PPof / \exists a Na)) : \hat{\sim} of, \lambda DD),$   
 $\square((\langle \rangle (\exists a Na - \exists g Nt(s(g))) \setminus S f) / \exists a Na) : \hat{\sim} \lambda E \lambda F(Pres((\sim admire E) F))] \Rightarrow CNs(m)$

There is the derivation given in Figure 20, where the use of contraction  $!C$ , involving brackets and stoups, corresponds to generating the parasitic gap. The object relativisation hypothetical subtype moves into the stoup at depth seven in the lefthand subtree (before this the analysis is standard). Contraction then applies copying the gap type into the stoup of a newly created bracketed domain around the subordinate subject. Applications of  $!P$  then move the stoup contents into the object position of *admire* (host) and *of* (parasitic). This delivers the following semantics in which the gap variable is multiply bound:

(69)  $\lambda C[(\sim man C) \wedge (Pres((\sim admire C) (\iota(\sim friends C))))]$

Parasitic extraction from strong islands such as coordinate structures is not acceptable:

(70) \* $that_i$  Mary showed [[John and the friends of  $t_i$ ] to  $t_i$

This is successfully blocked because strong islands are doubly bracketed. Although contraction could apply twice to introduce two bracketings, a copy of the hypothetical gap subtype would remain trapped in the stoup at the intermediate level of bracketing, blocking overall derivation. Likewise, as we remarked in footnote 16, parasitic extraction is not possible from relative clauses themselves, for the same reason: a superfluous gap subtype would remain in between the double brackets required for the strong island.

A parasitic gap can also appear in an adverbial weak island:

(71) **paper+[[that+[john]+filed+without+reading]]** :  $CNs(n)$

Lexical lookup for this example yields:

(72)  $\square CNs(n) : paper, [[\blacksquare \forall n([\ ]^{-1}[\ ]^{-1}(CNn \setminus CNn) / \blacksquare((\langle \rangle Nt(n) \sqcap ! \blacksquare Nt(n)) \setminus S f)) :$   
 $\lambda A \lambda B \lambda C[(B C) \wedge (A C)], [\blacksquare Nt(s(m)) : j], \square((\langle \rangle \exists g Nt(s(g))) \setminus S f) / \exists a Na) :$   
 $\hat{\sim} \lambda D \lambda E(Past((\sim file D) E)), \blacksquare \forall a \forall f([\ ]^{-1}((\langle \rangle Na \setminus S f) \setminus (\langle \rangle Na \setminus S f)) / (\langle \rangle Na \setminus S psp)) :$   
 $\lambda F \lambda G \lambda H[(G H) \wedge \neg(F H)], \square((\langle \rangle \exists a Na \setminus S psp) / \exists a Na) : \hat{\sim} \lambda I \lambda J((\sim read I) J)] \Rightarrow CNs(n)$

There is the derivation given in Figure 21. This time at depth eight contraction copies the host stoup gap into the stoup of a newly created bracketed domain around the subordinate adverbial phrase. This delivers semantics:

(73)  $\lambda C[(\sim paper C) \wedge [(Past((\sim file C) j)) \wedge \neg((\sim read C) j)]]$

In our final relativisation example the host gap licences two parasitic gaps, in the subject noun phrase and in an adverbial phrase:

(74) **paper+[[that+the+editor+of+filed+without+reading]]** :  $CNs(n)$

Lexical lookup yields:

<sup>18</sup> We gloss over the use of ‘difference’ here to mark non-third person singular; its use depends on *absence* of derivability (negation as failure) which of course cannot easily be displayed.



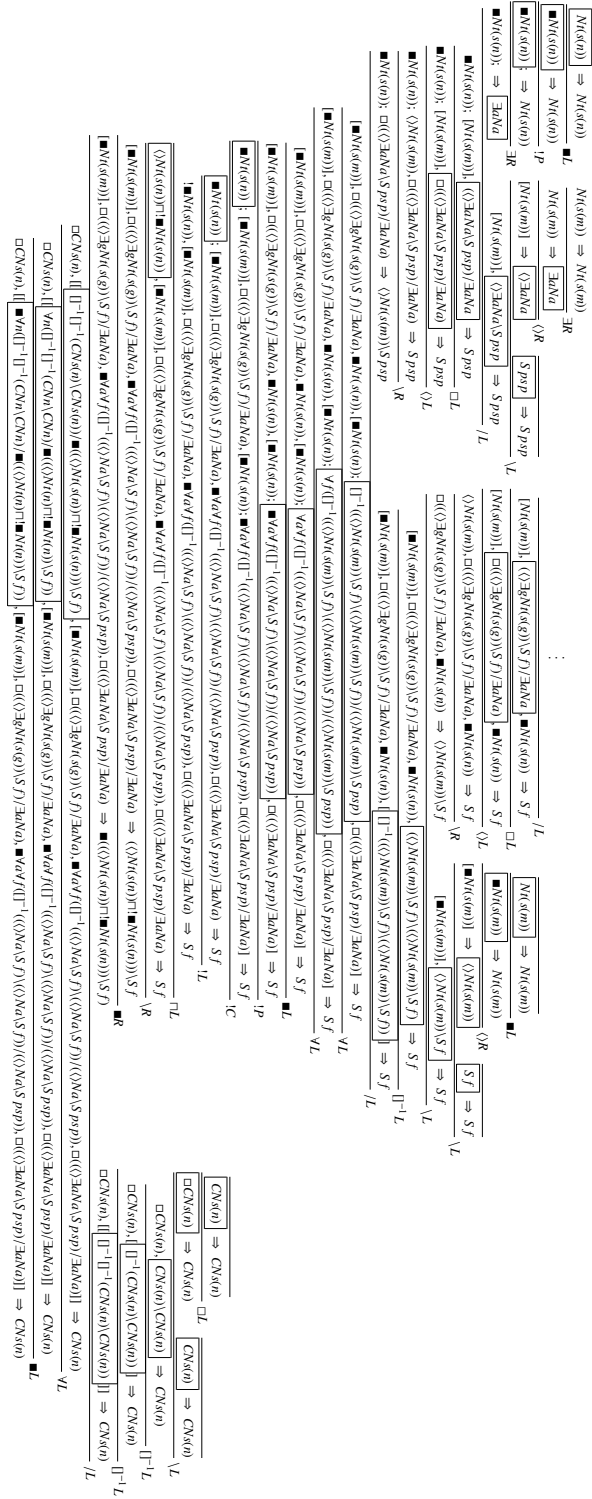


Fig. 21 Derivation for paper that John filed without reading

$$\begin{aligned}
(75) \quad & \Box CNs(n) : \text{paper}, [[\Box \forall n([\Box^{-1} \Box^{-1} (CNn \setminus CNn) / \Box((\langle \rangle Nt(n) \Box ! \Box Nt(n)) \setminus S f)) : \\
& \lambda \lambda \lambda B \lambda C[(B C) \wedge (A C)], \Box \forall n(Nt(n) / CNn) : \iota, \Box (\forall g CNs(g) / PPof) : \text{editor}, \\
& \Box ((\forall n(CNn \setminus CNn) / \Box \exists b Nb) \& (PPof / \exists a Na)) : \sim \text{of}, \lambda DD), \\
& \Box ((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \wedge E \lambda F(\text{Past}(\sim \text{file } E) F), \\
& \Box \forall a \forall f([\Box^{-1}((\langle \rangle Na \setminus S f) \setminus (\langle \rangle Na \setminus S f)) / (\langle \rangle Na \setminus S psp)) : \lambda G \lambda H \lambda I[(H I) \wedge \neg(G I)], \\
& \Box ((\langle \rangle \exists a Na \setminus S psp) / \exists a Na) : \wedge \lambda \lambda K(\sim \text{read } J) K]] \Rightarrow CNs(n)
\end{aligned}$$

There is the derivation fragmented into Figures 22 and 23. There are two applications of contraction, at depth nine and ten, projecting brackets around the subordinate subject and adverbial phrase and giving rise to two parasitic gaps. This delivers the correct semantics:

$$(76) \quad \lambda C[(\sim \text{paper } C) \wedge [(\text{Past}(\sim \text{file } C) (\iota \sim \text{editor } C))] \wedge \neg(\sim \text{read } C) (\iota \sim \text{editor } C)]]$$

## 6 Possible exceptions

In this section we address three kinds of possible exceptions to the account given here, along the lines anticipated in the introduction.

First, there are examples in which there appears to be a parasitic gap which is not in an island. The following is example (8a) from Postal (1993[46]):

$$(77) \quad \text{man who}_i \text{ Mary convinced } t_i \text{ that John wanted to visit } t_i$$

And an anonymous referee points out:

$$(78) \quad \text{people whom}_i \text{ you sent pictures of } t_i \text{ to } t_i$$

In respect of such examples we have suggested that although there *seems* to be no island, there *could* be one. This is effected as follows for (77). Instead of a type of the form  $((N \setminus S) / CP) / N$  for *convince* we assume  $((N \setminus S) / CP) / (N \sqcup \langle \rangle N)$  where the semantically inactive additive disjunction disjunct  $N$  will be selected ordinarily, and  $\langle \rangle N$  when there is parasitic extraction, as in (77). Similarly for (78) we assume for *picture* type  $CN / (PP \sqcup \langle \rangle PP)$  where the second disjunct projects the brackets of a weak island.<sup>19</sup> Thus in examples such as the following the semantically inactive additive disjunction inference for *convince* of type  $((N \setminus S) / CP) / (N \sqcup \langle \rangle N)$  will select  $N$ :

$$\begin{aligned}
(79) \quad & \text{a. man who}_i \text{ Mary convinced } t_i \text{ that John wanted to visit Suzy} \\
& \text{b. man who}_i \text{ Mary convinced the friends of } t_i \text{ that John wanted to visit Suzy}
\end{aligned}$$

But for (77) the semantically inactive additive disjunction inference for *convince* of type  $((N \setminus S) / CP) / (N \sqcup \langle \rangle N)$  will select  $\langle \rangle N$ . Similarly for the picture noun case (78). This account is a bit heavy-handed, but it enables the facts to be captured. Note that any account not attesting to island-specificity of parasitic gaps must at the same time find a constraint for examples such as (4).

In relation to island violations we cannot undertake to offer a full account of how non-syntactic factors may enable exceptional processing steps, but we do suggest how the weak island violation is combinatorially possible without changing the grammar. This is to assume a variant  $\%!P$  of  $!P$  as follows applicable under the right processing conditions (Morrill 2011[44], Section 5.4):

$$(80) \quad \frac{\Xi(\zeta; \Gamma_1, [\{A: x\}; \Gamma_2], \Gamma_3) \Rightarrow B: \psi}{\Xi(\zeta \uplus \{A: x\}; \Gamma_1, \Gamma_2, \Gamma_3) \Rightarrow B: \psi} \%!P$$

<sup>19</sup> The argument pattern  $X \sqcup \langle \rangle X$  is a general mechanism for an argument optional island  $X$ . Likewise the dual value pattern  $X \sqcap [\Box^{-1} X]$  is a general mechanism for a value optional island  $X$ .



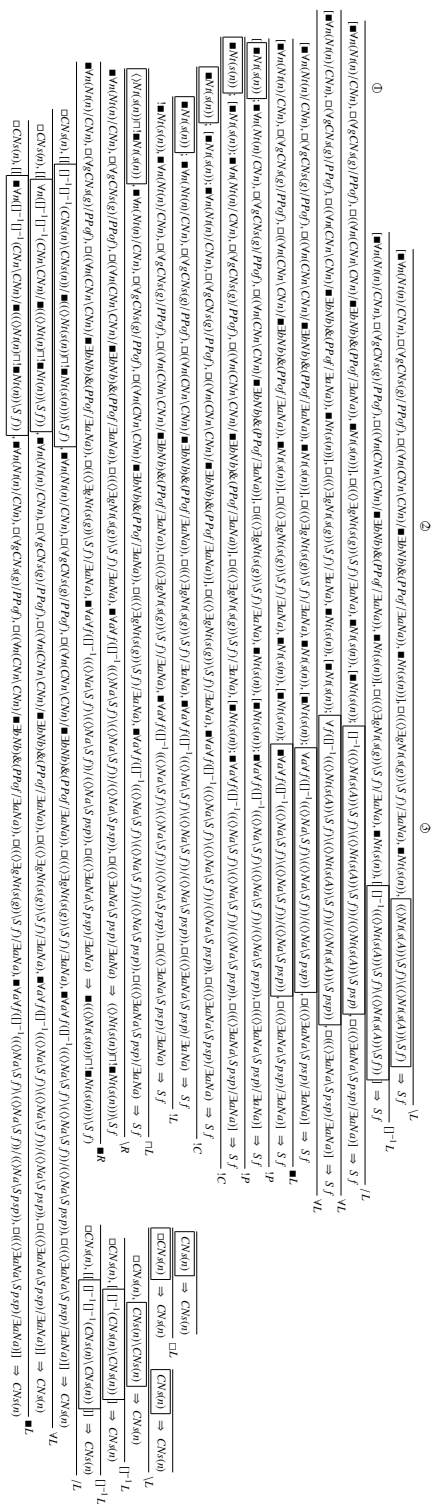


Fig. 23 Main derivation for paper that the editor of filed without reading



This allows extraction from weak islands, thus for example:

$$(81) \frac{\frac{\dots, [\dots, A, \dots], \dots \Rightarrow B}{\dots, [A; \dots, \dots], \dots \Rightarrow B} !P}{\frac{A; \dots, \dots, \dots, \dots \Rightarrow B}{!A, \dots, \dots, \dots, \dots \Rightarrow B} \%!P} !L$$

Thirdly, Levine and Hukari (2006[25]) cite an apparent example of ‘symbiotic’ extraction without a host gap:

(82) people that<sub>i</sub> fans of  $t_i$  gather from every continent just to listen to  $t_i$

Our response to the second issue predicts a possibility such as (82). An analysis can begin with the hypothetical subtype in the stoup emitting a parasitic gap by  $!C$  in, say, the subject in the usual way. Then  $\%!P$  can move the hypothetical subtype in the outer stoup into the stoup of the adjunct island. The two gap types in the two island stoups are then moved into their positions by  $!P$ ’s.

## 7 Conclusion

Our type logical categorial grammar incorporating nonassociativity, nonlinearity, and their interaction is, we suggest, mathematically interesting, technically robust, and at least as empirically adequate and computationally advanced as other proposals.

**Acknowledgements** Research partially supported by an ICREA Acadèmia 2012, SGR2014-890 (MACDA) of the Generalitat de Catalunya and MINECO project APCOM (TIN2014-57226-P). Many thanks to *Linguistics and Philosophy* editing and reviewing which has greatly improved the paper, and to Oriol Valentín for insight and support. All errors are my own.

## Appendix A: Semantic Representation Language

Semantic types

Recall the following operations on sets:

- (83) a. Functional exponentiation:  $X^Y$  = the set of all total functions from  $Y$  to  $X$   
 b. Cartesian product:  $X \times Y = \{\langle x, y \rangle \mid x \in X \ \& \ y \in Y\}$   
 c. Disjoint union:  $X \uplus Y = (\{1\} \times X) \cup (\{2\} \times Y)$

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

$$(84) \mathcal{T} ::= \delta \mid \top \mid \mathcal{T} + \mathcal{T} \mid \mathcal{T} \& \mathcal{T} \mid \mathcal{T} \rightarrow \mathcal{T} \mid \mathbf{MT} \mid \mathbf{LT}$$

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 non-empty *type domain*  $D_\tau$  for each type  $\tau$  as follows:

$$(85) \begin{array}{ll} D_\top = \{\emptyset\} & \text{singleton set} \\ D_{\tau_1 + \tau_2} = D_{\tau_1} \uplus D_{\tau_2} & \text{disjoint union} \\ D_{\tau_1 \& \tau_2} = D_{\tau_1} \times D_{\tau_2} & \text{Cartesian product} \\ D_{\tau_1 \rightarrow \tau_2} = D_{\tau_2}^{D_{\tau_1}} & \text{functional exponentiation} \\ D_{\mathbf{M}\tau} = W \times D_\tau & \text{Cartesian product} \\ D_{\mathbf{L}\tau} = D_\tau^W & \text{functional exponentiation} \end{array}$$

## Semantic terms

The sets  $\Phi_\tau$  of *terms* of type  $\tau$  for each semantic type  $\tau$  are defined on the basis of sets  $C_\tau$  of constants of type  $\tau$  and denumerably infinite sets  $V_\tau$  of variables of type  $\tau$  for each type  $\tau$  as follows:

(86)	$\Phi_\tau ::= C_\tau$	constants
	$\Phi_\tau ::= V_\tau$	variables
	$\Phi_\tau ::= 0$	dummy
	$\Phi_\tau ::= \Phi_{\tau_1+\tau_2} \rightarrow V_{\tau_1} \cdot \Phi_\tau; V_{\tau_2} \cdot \Phi_\tau$	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
	$\Phi_\tau ::= \bigvee \Phi_{\mathbf{L}\tau}$	extensionalisation
	$\Phi_{\mathbf{L}\tau} ::= \bigwedge \Phi_\tau$	intensionalisation
	$\Phi_\tau ::= \bigcup \Phi_{\mathbf{M}\tau}$	projection
	$\Phi_{\mathbf{M}\tau} ::= \bigcap \Phi_\tau$	injection

Given a semantic frame, a *valuation*  $f$  mapping each constant of type  $\tau$  into an element of  $D_\tau$ , an assignment  $g$  mapping each variable of type  $\tau$  into an element of  $D_\tau$ , and a world  $i \in W$ , each term  $\phi$  of type  $\tau$  receives an interpretation  $[\phi]^{g,i} \in D_\tau$  as shown in Figure 24; the *update*  $g[x := d]$  is  $(g - \{(x, g(x))\}) \cup \{(x, d)\}$ , i.e. the function which sends  $x$  to  $d$  and agrees with  $g$  elsewhere.

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

**Fig. 24** Interpretation of the semantic representation language

In  $x.\phi$ ,  $\lambda x \phi$  or  $\bigwedge \phi$ ,  $\phi$  is the *scope* of  $x$ ,  $\lambda x$  or  $\bigwedge$ . An occurrence of a variable  $x$  in a term is called *free* if and only if it does not fall within the scope of any  $x$ . or  $\lambda x$ ; otherwise it is *bound* (by the closest  $x$ . or  $\lambda x$  within the scope of which it falls). The result  $\phi\{\psi_1/x_1, \dots, \psi_n/x_n\}$  of substituting terms  $\psi_1, \dots, \psi_n$  for variables  $x_1, \dots, x_n$  of the same types respectively in a

$$\begin{aligned}
& \phi \rightarrow y.\psi; z.\chi = \phi \rightarrow x.(\psi\{x/y\}); z.\chi \\
& \text{if } x \text{ is not free in } \psi \text{ and is free for } y \text{ in } \psi \\
& \phi \rightarrow y.\psi; z.\chi = \phi \rightarrow y.\psi; x.(\chi\{x/z\}) \\
& \text{if } x \text{ is not free in } \chi \text{ and is free for } z \text{ in } \chi \\
& \quad \lambda y\phi = \lambda x(\phi\{x/y\}) \\
& \text{if } x \text{ is not free in } \phi \text{ and is free for } y \text{ in } \phi \\
& \quad \alpha\text{-conversion} \\
\\
& \iota_1\phi \rightarrow y.\psi; z.\chi = \psi\{\phi/y\} \\
& \text{if } \phi \text{ is free for } y \text{ in } \psi \text{ and modally free for } y \text{ in } \psi \\
& \iota_2\phi \rightarrow y.\psi; z.\chi = \chi\{\phi/z\} \\
& \text{if } \phi \text{ is free for } z \text{ in } \chi \text{ and modally free for } z \text{ in } \chi \\
& \quad \pi_1(\phi, \psi) = \phi \\
& \quad \pi_2(\phi, \psi) = \psi \\
& \quad (\lambda x\phi\psi) = \phi\{\psi/x\} \\
& \text{if } \psi \text{ is free for } x \text{ in } \phi, \text{ and modally free for } x \text{ in } \phi \\
& \quad \vee^\wedge\phi = \phi \\
& \quad \cup^\cap\phi = \phi \\
& \quad \beta\text{-conversion} \\
\\
& (\pi_1\phi, \pi_2\phi) = \phi \\
& \lambda x(\phi x) = \phi \\
& \text{if } x \text{ is not free in } \phi \\
& \quad \wedge^\vee\phi = \phi \\
& \text{if } \phi \text{ is modally closed} \\
& \quad \cap^\cup\phi = \phi \\
& \quad \eta\text{-conversion}
\end{aligned}$$

Fig. 25 Semantic conversion laws

term  $\phi$  is the result of simultaneously replacing by  $\psi_i$  every free occurrence of  $x_i$  in  $\phi$ . We say that  $\psi$  is *free for  $x$  in  $\phi$*  if and only if no variable in  $\psi$  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  $\psi$  is *modally free for  $x$  in  $\phi$*  if and only if either  $\psi$  is modally closed, or no free occurrence of  $x$  in  $\phi$  is within the scope of an  $\wedge$ . The laws of conversion in Figure 25 obtain. The so-called commuting conversions with respect to normalisation for the case statement are omitted.

## Appendix B: A lexicon

**a** :  $\blacksquare\forall g(\forall f((S f^\dagger \blacksquare Nt(s(g)))^\perp S f)/CNs(g)) : \lambda A\lambda B\exists C[(A C) \wedge (B C)]$   
**admire** :  $\square((\langle \rangle \exists a Na - \exists g Nt(s(g))) \setminus S f) / \exists a Na : \lambda A\lambda B(Pres (\ulcorner admire A B))$   
**And** :  $\blacksquare\forall f(S f/S f) : \lambda AA$   
**and** :  $\blacksquare\forall f(\langle \rangle \blacksquare S f \setminus \square^{-1} \square^{-1} S f) / \blacksquare S f : (\Phi^{n+} 0 \text{ and})$   
**and** :  $\blacksquare\forall a\forall f(\langle \rangle \blacksquare (\langle \rangle Na \setminus S f) \setminus \square^{-1} \square^{-1} (\langle \rangle Na \setminus S f)) / \blacksquare (\langle \rangle Na \setminus S f) : (\Phi^{n+} (s 0) \text{ and})$   
**and** :  $\blacksquare\forall a\forall f(\langle \rangle \blacksquare (S f / !Na) \setminus \square^{-1} \square^{-1} (S f / !Na)) / \blacksquare (S f / !Na) : (\Phi^{n+} (s 0) \text{ and})$   
**and** :  $\blacksquare\forall f(\langle \rangle \blacksquare (S f / \exists a Na) \setminus \square^{-1} \square^{-1} (S f / \exists a Na)) / \blacksquare (S f / \exists a Na) : (\Phi^{n+} (s 0) \text{ and})$   
**and** :  $\blacksquare\forall w\forall a\forall b\forall f(\langle \rangle \blacksquare ((S f^\dagger ((\langle \rangle Na \setminus S f) \ominus Ww) / Nb)) \uparrow_2 Ww) \setminus \square^{-1} \square^{-1} ((S f^\dagger ((\langle \rangle Na \setminus S f) \ominus Ww) / Nb)) \uparrow_2 Ww) / \sim \blacksquare ((S f^\dagger ((\langle \rangle Na \setminus S f) \ominus Ww) / Nb)) \uparrow_2 Ww) : \lambda A\lambda B\lambda C[(B C) \wedge (A C)]$   
**and** :  $\blacksquare\forall f\forall a(\langle \rangle \blacksquare ((\langle \rangle Na \setminus S f) / \exists b Nb) \setminus \square^{-1} \square^{-1} ((\langle \rangle Na \setminus S f) / \exists b Nb)) / \blacksquare ((\langle \rangle Na \setminus S f) / \exists b Nb) : (\Phi^{n+} (s (s 0)) \text{ and})$   
**and** :  $\blacksquare\forall f\forall a(\langle \rangle \blacksquare ((\langle \rangle Na \setminus S f) / \exists b Nb) \setminus (\langle \rangle Na \setminus S f) \setminus \square^{-1} \square^{-1} ((\langle \rangle Na \setminus S f) / \exists b Nb) \setminus (\langle \rangle Na \setminus S f)) / \blacksquare ((\langle \rangle Na \setminus S f) / \exists b Nb) \setminus (\langle \rangle Na \setminus S f) : (\Phi^{n+} (s (s 0)) \text{ and})$   
**and** :  $\blacksquare\forall f\forall a(\langle \rangle \blacksquare ((\langle \rangle Na \setminus S f) / (\exists b Nb \oplus \exists g((CNg/CNg) \sqcup (CNg \setminus CNg)))) \setminus (\langle \rangle Na \setminus S f) \setminus \square^{-1} \square^{-1} ((\langle \rangle Na \setminus S f) / (\exists b Nb \oplus \exists g((CNg/CNg) \sqcup (CNg \setminus CNg)))) \setminus (\langle \rangle Na \setminus S f)) / \blacksquare ((\langle \rangle Na \setminus S f) / (\exists b Nb \oplus \exists g((CNg/CNg) \sqcup (CNg \setminus CNg)))) \setminus (\langle \rangle Na \setminus S f) : \lambda A\lambda B\lambda C\lambda D[(B C D) \wedge ((A C) D)]$   
**and** :  $\blacksquare\forall a\forall b\forall f(\langle \rangle \blacksquare ((\langle \rangle Na \setminus S f) / (\exists c Nc \oplus CPb)) \setminus (\langle \rangle Na \setminus S f) \setminus \square^{-1} \square^{-1} ((\langle \rangle Na \setminus S f) / (\exists c Nc \oplus CPb)) \setminus (\langle \rangle Na \setminus S f)) / \blacksquare ((\langle \rangle Na \setminus S f) / (\exists c Nc \oplus CPb)) \setminus (\langle \rangle Na \setminus S f) : (\Phi^{n+} (s (s 0)) \text{ and})$

$\blacksquare(((\langle \rangle Na \setminus S f) / (\exists c Nc \bullet CPb)) \setminus (\langle \rangle Na \setminus S f)) : (\Phi^{n+} (s (s 0)) \text{ and})$   
**and** :  $\blacksquare \forall a \forall b \forall f ((\langle \rangle Na \setminus S f) / PPb) \setminus [ ]^{-1} [ ]^{-1} ((\langle \rangle Na \setminus S f) / PPb) / \blacksquare(((\langle \rangle Na \setminus S f) / PPb)) : (\Phi^{n+} (s (s 0)) \text{ and})$   
**and** :  $\blacksquare \forall a \forall b \forall f ((\langle \rangle Na \setminus S f) / (\exists c Nc \bullet PPb)) \setminus (\langle \rangle Na \setminus S f) \setminus [ ]^{-1} [ ]^{-1} ((\langle \rangle Na \setminus S f) / (\exists c Nc \bullet PPb)) \setminus (\langle \rangle Na \setminus S f)) / \blacksquare(((\langle \rangle Na \setminus S f) / (\exists c Nc \bullet PPb)) \setminus (\langle \rangle Na \setminus S f)) : (\Phi^{n+} (s (s 0)) \text{ and})$   
**and** :  $\blacksquare \forall a \forall b \forall f ((\langle \rangle Na \setminus S f) / !Nb) \setminus [ ]^{-1} [ ]^{-1} ((\langle \rangle Na \setminus S f) / !Nb) / \blacksquare(((\langle \rangle Na \setminus S f) / !Nb)) : (\Phi^{n+} (s (s 0)) \text{ and})$   
**ate** :  $\square(((\langle \rangle \exists a Na \setminus S f) / \exists a Na) : \lambda A \lambda B (Past (\textit{eat} A) B))$   
**bagels** :  $\square(Nt(p(n)) \& CNp(n)) : \sim((gen \textit{bagels}), \textit{bagels})$   
**barn** :  $\square CNs(n) : barn$   
**be** :  $\square(((\langle \rangle W[there] \rightarrow S b) / \exists a Na) : \lambda A (\textit{be} A))$   
**before** :  $\blacksquare(\forall a \forall f ((\langle \rangle Na \setminus S f) \setminus (\langle \rangle Na \setminus S f)) / S f) : \lambda A \lambda B \lambda C ((before A) (B C))$   
**beginning** :  $\square CNs(n) : beginning$   
**believes** :  $\square(((\langle \rangle \exists g Nt(s(g)) \setminus S f) / (CPthat \sqcup \square S f)) : \lambda A \lambda B (Pres (\textit{believe} A) B))$   
**bill** :  $\blacksquare Nt(s(m)) : b$   
**book** :  $\square CNs(n) : book$   
**bought** :  $\square(((\langle \rangle \exists a Na \setminus S f) / (\exists a Na \bullet \exists a Na)) : \lambda A \lambda B (Past ((\textit{buy} \pi_1 A) \pi_2 A) B))$   
**bought** :  $\square(((\langle \rangle \exists a Na \setminus S f) / \exists a Na) : \lambda A \lambda B (Past (\textit{buy} A) B))$   
**by** :  $\blacksquare \forall a ((\langle \rangle Na \setminus S -) \setminus (\langle \rangle Na \setminus S -)) / Na) : \lambda A \lambda B \lambda C [(C = A) \wedge (B C)]$   
**by** :  $\square(\forall n (CNn \setminus CNn) / \exists a Na) : \lambda A \lambda B (\textit{by} A) B$   
**buys** :  $\square(((\langle \rangle \exists g Nt(s(g)) \setminus S f) / (\exists a Na \bullet \exists a Na)) : \lambda A \lambda B (Pres ((\textit{buy} \pi_1 A) \pi_2 A) B))$   
**calls** :  $\square(((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a ((W[up] \bullet Na) \sqcup (Na \bullet W[up]))) : \lambda A \lambda B (\textit{phone} A) B)$   
**catch** :  $\square(((\langle \rangle \exists a Na \setminus S b) / \exists a Na) : \lambda A \lambda B (\textit{catch} A) B)$   
**cezanne** :  $\blacksquare Nt(s(m)) : c$   
**cd** :  $\square CNs(n) : cd$   
**charles** :  $\blacksquare Nt(s(m)) : c$   
**cicero** :  $\blacksquare \forall g Nt(s(g)) : c$   
**clark** :  $\blacksquare \forall g Nt(s(g)) : c$   
**coffee** :  $\square(Nt(s(n)) \& CNs(n)) : \sim((gen \textit{coffee}), \textit{coffee})$   
**created** :  $\square(((\langle \rangle \exists a Na \setminus S f) / \exists a Na) : \lambda A \lambda B (Past (\textit{create} A) B))$   
**darkness** :  $\square(CNs(n) \& Nt(s(n))) : \sim(\textit{darkness}, (gen \textit{darkness}))$   
**deep** :  $\square CNs(n) : deep$   
**did** :  $\blacksquare \forall a \forall g \forall b \forall h (((\langle \rangle Na \setminus S g) \uparrow (\langle \rangle Nb \setminus S h) / (\exists c \langle \rangle Nc \setminus S f)) \setminus ((\langle \rangle Na \setminus S g) \uparrow (\langle \rangle Nb \setminus S h))) : \lambda A \lambda B ((A B) B)$   
**did+too** :  $\square(((\langle \rangle Na \setminus S B) \uparrow (\langle \rangle Nc \setminus S D)) / (\langle \rangle Ne \setminus S F)) \setminus ((\langle \rangle Ng \setminus S H) \uparrow (Nt \setminus S J)) : \lambda K \lambda L ((K L) L)$   
**doesnt** :  $\blacksquare \forall g \forall a ((S g \uparrow ((\langle \rangle Na \setminus S f) / (\langle \rangle Na \setminus S b)) \downarrow S g) : \lambda A \neg (A \lambda B \lambda C (B C))$   
**dog** :  $\square CNs(n) : dog$   
**donuts** :  $\square(Nt(p(n)) \& CNp(n)) : \sim((gen \textit{donuts}), \textit{donuts})$   
**earth** :  $\square CNs(n) : earth$   
**eat** :  $\square(((\langle \rangle \exists a Na \setminus S b) / \exists a Na) : \lambda A \lambda B (\textit{eat} A) B)$   
**edinburgh** :  $\blacksquare Nt(s(n)) : e$   
**editor** :  $\square(\forall g CNs(g) / PPof) : editor$   
**every** :  $\blacksquare \forall g (\forall f ((S f \uparrow Nt(s(g))) \downarrow S f) / CNs(g)) : \lambda A \lambda B \forall C [(A C) \rightarrow (B C)]$   
**everyone** :  $\square \forall f ((S f \uparrow \forall g Nt(s(g))) \downarrow S f) : \lambda A \forall B [(\textit{person} B) \rightarrow (A B)]$   
**face** :  $\square CNs(n) : face$   
**fell** :  $\square(\exists a \langle \rangle Na \setminus S f) : \lambda A (Past (\textit{fall} A))$   
**filed** :  $\square(((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \lambda A \lambda B (Past (\textit{file} A) B))$   
**finds** :  $\square(((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \lambda A \lambda B (Pres (\textit{find} A) B))$   
**fish** :  $\square CNs(n) : fish$   
**for** :  $\blacksquare (PPfor / \exists a Na) : \lambda AA$   
**form** :  $\square(CNs(n) \& Nt(s(n))) : \sim(\textit{form}, (gen \textit{form}))$   
**friends** :  $\square(CNp / PPof) : friends$   
**from** :  $\square(\forall a \forall f ((\langle \rangle Na \setminus S f) \setminus (\langle \rangle Na \setminus S f)) \setminus \forall n (CNn \setminus CNn) / \exists b Nb) : \lambda A ((\textit{fromadv} A), (\textit{fromadn} A))$   
**gave** :  $\square(((\langle \rangle \exists a Na \setminus S f) / (\exists b Nb \bullet Ppto)) : \lambda A \lambda B (Past ((\textit{give} \pi_2 A) \pi_1 A) B))$   
**gave** :  $\square(((\langle \rangle \exists g Nt(s(g)) \setminus S f) / (\exists a Na \bullet W[the, cold, shoulder])) : \lambda A \lambda B (Past (\textit{shun} A) B))$   
**gave** :  $\square(((\langle \rangle \exists a Na \setminus S f) / \exists a Na) / \exists a Na) : \lambda A \lambda B \lambda C (Past ((\textit{give} A) B) C))$   
**girl** :  $\square CNs(f) : girl$   
**gives** :  $\square(((\langle \rangle \exists g Nt(s(g)) \setminus S f) / (\exists a Na \bullet W[the, cold, shoulder])) : \lambda A \lambda B (Pres (\textit{shun} A) B))$   
**God** :  $\blacksquare Nt(s(m)) : God$   
**good** :  $\square \forall n (CNn / CNn) : good$   
**has** :  $\square(((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \lambda A \lambda B (Pres (\textit{have} A) B))$   
**he** :  $\blacksquare [ ]^{-1} \forall g ((\blacksquare S g \setminus \blacksquare Nt(s(m))) / (\langle \rangle Nt(s(m)) \setminus S g)) : \lambda AA$   
**heaven** :  $\square CNs(n) : heaven$

**her** :  $\blacksquare \forall g \forall a (((\langle \rangle Na \setminus S g) \uparrow \blacksquare Nt(s(f))) \downarrow (\blacksquare (\langle \rangle Na \setminus S g) \blacksquare Nt(s(f)))) : \lambda AA$   
**himself** :  $\blacksquare \forall f (((\langle \rangle Nt(s(m)) \setminus S f) \uparrow Nt(s(m))) \downarrow (\langle \rangle Nt(s(m)) \setminus S f)) : \lambda A \lambda B ((A B) B)$   
**horse** :  $\square CNs(n) : horse$   
**humanist** :  $\square \forall n (CNn / CNn) : \sim \lambda A \lambda B [(A B) \wedge (\sim humanist B)]$   
**in** :  $\square (\forall a \forall f ((\langle \rangle Na \setminus S f) \setminus (\langle \rangle Na \setminus S f)) / \exists a Na) : \sim \lambda A \lambda B \lambda C ((\sim in A) (B C))$   
**it** :  $\blacksquare W[it] : 0$   
**it** :  $\blacksquare \forall f \forall a (((\langle \rangle Na \setminus S f) \uparrow \blacksquare Nt(s(n))) \downarrow (\blacksquare (\langle \rangle Na \setminus S f) \blacksquare Nt(s(n)))) : \lambda AA$   
**it** :  $\blacksquare []^{-1} \forall f ((\blacksquare S f \blacksquare Nt(s(n))) / (\langle \rangle Nt(s(n)) \setminus S f)) : \lambda AA$   
**jogs** :  $\square (\langle \rangle \exists g Nt(s(g)) \setminus S f) : \sim \lambda A (Pres (\sim jog A))$   
**john** :  $\blacksquare Nt(s(m)) : j$   
**laughs** :  $\square (\langle \rangle \exists g Nt(s(g)) \setminus S f) : \sim \lambda A (Pres (\sim laugh A))$   
**left** :  $\square (\langle \rangle \exists g Nt(s(g)) \setminus S f) : \sim \lambda A (Pres (\sim leave A))$   
**let** :  $\square (S im / S b) : let$   
**light** :  $\square (CNs(n) \& Nt(s(n))) : \sim (\sim light, (gen \sim light))$   
**likes** :  $\square ((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \sim \lambda A \lambda B (Pres ((\sim like A) B))$   
**logic** :  $\square (Nt(s(n)) \& CNs(n)) : \sim ((gen \sim logic), \sim logic)$   
**london** :  $\blacksquare Nt(s(n)) : l$   
**loses** :  $\square ((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \sim \lambda A \lambda B (Pres ((\sim lose A) B))$   
**love** :  $\square ((\langle \rangle \exists a Na \setminus S b) / \exists a Na) : \sim \lambda A \lambda B ((\sim love A) B)$   
**loved** :  $\square \forall a \forall b (((\langle \rangle Na \setminus S -) \uparrow Nb) \circ ((\langle \rangle Na \setminus S -) \uparrow Nb) \downarrow \forall g (CNg \setminus CNg)) : \sim (\sim love, \lambda A \lambda B \lambda C [(B C) \wedge \exists D ((A C) D)])$   
**loves** :  $\square ((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \sim \lambda A \lambda B (Pres ((\sim love A) B))$   
**man** :  $\square CNs(m) : man$   
**mary** :  $\blacksquare Nt(s(f)) : m$   
**met** :  $\square ((\langle \rangle \exists a Na \setminus S f) / \exists a Na) : \sim \lambda A \lambda B (Past ((\sim meet A) B))$   
**more** :  $\blacksquare \forall h \forall g \forall f (S f \uparrow ((S h \uparrow Nt(p(g))) \downarrow S h) / CNp(g)) \downarrow S f / (CPthan \uparrow \blacksquare ((S h \uparrow Nt(p(g))) \downarrow S h) / CNp(g))) : \lambda A \lambda B [\lambda C (A \lambda D \lambda E [(D C) \wedge (E C)])] > \lambda F (B \lambda G \lambda H [(G F) \wedge (H F)])]$   
**mountain** :  $\square CNs(n) : mountain$   
**moved** :  $\square (\langle \rangle \exists a Na \setminus S f) : \sim \lambda A (Past (\sim move A))$   
**necessarily** :  $\blacksquare (S A / \square S A) : Nec$   
**of** :  $\square ((\forall n (CNn / CNn) / \blacksquare \exists b Nb) \& (PPof / \exists a Na)) : \sim (\sim of, \lambda AA)$   
**or** :  $\blacksquare \forall f ((\blacksquare S f \blacksquare []^{-1} []^{-1} S f) / \blacksquare S f) : (\Phi^{n+} 0 or)$   
**or** :  $\blacksquare \forall a \forall f ((\langle \rangle Na \setminus S f) \setminus []^{-1} []^{-1} (\langle \rangle Na \setminus S f)) / \blacksquare ((\langle \rangle Na \setminus S f)) : (\Phi^{n+} (s 0) or)$   
**or** :  $\blacksquare \forall f ((\blacksquare S f / (\langle \rangle \exists g Nt(s(g)) \setminus S f)) \setminus []^{-1} []^{-1} (S f / (\langle \rangle \exists g Nt(s(g)) \setminus S f))) / \blacksquare (S f / (\langle \rangle \exists g Nt(s(g)) \setminus S f)) : (\Phi^{n+} (s 0) or)$   
**or** :  $\blacksquare \forall a \forall f ((\langle \rangle Na \setminus S f) / \exists b Nb) \setminus []^{-1} []^{-1} (((\langle \rangle Na \setminus S f) / \exists b Nb) / \exists b Nb) / \blacksquare (((\langle \rangle Na \setminus S f) / \exists b Nb) / \exists b Nb) : (\Phi^{n+} (s (s 0))) or)$   
**painting** :  $\square (CNs(n) / PPof) : \sim \lambda A ((\sim of A) \sim painting)$   
**paper** :  $\square CNs(n) : paper$   
**park** :  $\square CNs(n) : park$   
**past** :  $\square \forall a \forall f (((\langle \rangle Na \setminus S f) \setminus (\langle \rangle Na \setminus S f)) / \exists b Nb) : \sim \lambda A \lambda B \lambda C ((\sim past A) (B C))$   
**perseverance** :  $\square (Nt(s(n)) \& CNs(n)) : \sim ((gen \sim perseverance), \sim perseverance)$   
**peter** :  $\blacksquare Nt(s(m)) : p$   
**phonetics** :  $\square (Nt(s(n)) \& CNs(n)) : \sim ((gen \sim phonetics), \sim phonetics)$   
**praises** :  $\square ((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \sim \lambda A \lambda B (Pres ((\sim praise A) B))$   
**raced** :  $\square (\langle \rangle \exists a Na \setminus S f) : \sim \lambda A (Past (\sim race A))$   
**raced** :  $\square \forall a \forall b (((\langle \rangle Na \setminus S -) \uparrow Nb) \circ ((\langle \rangle Na \setminus S -) \uparrow Nb) \downarrow \forall g (CNg \setminus CNg)) : \sim (\sim race2, \lambda A \lambda B \lambda C [(B C) \wedge \exists D ((A C) D)])$   
**rains** :  $\square (\langle \rangle W[it] \rightarrow S f) : \sim (Pres \sim itrains)$   
**reading** :  $\square ((\langle \rangle \exists a Na \setminus S psp) / \exists a Na) : \sim \lambda A \lambda B ((\sim read A) B)$   
**robin** :  $\blacksquare \forall g Nt(s(g)) : r$   
**said** :  $\square ((\langle \rangle \exists a Na \setminus S f) / S im) : \sim \lambda A \lambda B (Past ((\sim say A) B))$   
**saw** :  $\square ((\langle \rangle \exists a Na \setminus S f) / (\exists a Na \oplus CPthat)) : \sim \lambda A \lambda B (Past ((A \rightarrow C. (\sim see C); D. (\sim see D)) B))$   
**seeks** :  $\square ((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \square \forall a \forall f (((Na \setminus S f) / \exists b Nb) \setminus (Na \setminus S f))) : \sim \lambda A \lambda B ((\sim tries (\sim A \sim find B)) B)$   
**sees** :  $\square ((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \sim \lambda A \lambda B (Pres ((\sim see A) B))$   
**sent** :  $\square ((\langle \rangle \exists a Na \setminus S f) / (\exists b Nb \bullet Ppto)) : \sim \lambda A \lambda B (Past (((\sim sent \pi_2 A) \pi_1 A) B))$   
**sent** :  $\square (((\langle \rangle \exists a Na \setminus S f) / \exists a Na) / \exists a Na) : \sim \lambda A \lambda B \lambda C (Past (((\sim send A) B) C))$   
**she** :  $\blacksquare []^{-1} \forall g ((\blacksquare S g \blacksquare Nt(s(f))) / (\langle \rangle Nt(s(f)) \setminus S g)) : \lambda AA$   
**sings** :  $\square (\langle \rangle \exists g Nt(s(g)) \setminus S f) : \sim \lambda A (Pres (\sim sing A))$   
**slept** :  $\square (\langle \rangle \exists g Nt(s(g)) \setminus S f) : \sim \lambda A (Past (\sim sleep A))$   
**slowly** :  $\square \forall a \forall f ((\langle \rangle Na \setminus S f) \setminus (\langle \rangle \square Na \setminus S f)) : \sim \lambda A \lambda B ((\sim slowly \sim A \sim B))$   
**sneezed** :  $\square (\langle \rangle \exists g Nt(s(g)) \setminus S f) : \sim \lambda A (Past (\sim sneeze A))$   
**sold** :  $\square ((\langle \rangle \exists a Na \setminus S f) / (\exists b Nb \bullet PPfor)) : \sim \lambda A \lambda B (Past (((\sim sell \pi_2 A) \pi_1 A) B))$

**someone** :  $\square \forall f((S f \uparrow \blacksquare \forall g Nt(g)) \downarrow S f) : \hat{\lambda} A \exists B[(\sim person B) \wedge (A B)]$   
**Spirit** :  $\square CNs(m) : Spirit$   
**studies** :  $\square((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \exists a Na) : \hat{\lambda} A \lambda B(Pres(\sim study A B))$   
**such+that** :  $\blacksquare \forall n((CNn \setminus CNn) / (S f \blacksquare Nt(n))) : \lambda A \lambda B \lambda C[(B C) \wedge (A C)]$   
**suzy** :  $\blacksquare Nt(s(f)) : s$   
**talks** :  $\square(\langle \rangle \exists g Nt(s(g)) \setminus S f) : \hat{\lambda} A(Pres(\sim talk A))$   
**tall** :  $\square \forall g(CNg / CNg) : tall$   
**tenmilliondollars** :  $\square Nt(s(n)) : tenmilliondollars$   
**than** :  $\blacksquare (Cpthan / \square S f) : \lambda AA$   
**that** :  $\blacksquare (Cpthat / \square S f) : \lambda AA$   
**that** :  $\blacksquare \forall n([\ ]^{-1} [\ ]^{-1} (CNn \setminus CNn) / \blacksquare((\langle \rangle Nt(n) \uparrow \blacksquare Nt(n)) \setminus S f)) : \lambda A \lambda B \lambda C[(B C) \wedge (A C)]$   
**the** :  $\blacksquare \forall n(Nt(n) / CNn) : \iota$   
**the+cold+shoulder** :  $\blacksquare W[the, cold, shoulder] : 0$   
**there** :  $\blacksquare W[there] : 0$   
**thinks** :  $\square((\langle \rangle \exists g Nt(s(g)) \setminus S f) / (Cpthat \sqcup \square S f)) : \hat{\lambda} A \lambda B(Pres(\sim think A B))$   
**to** :  $\blacksquare((PPro / \exists a Na) \uparrow \forall n((\langle \rangle Nn \setminus Si) / (\langle \rangle Nn \setminus Sb))) : \lambda AA$   
**today** :  $\square \forall a \forall f((\langle \rangle Na \setminus S f) / (\langle \rangle Na \setminus S f)) : \hat{\lambda} A \lambda B(\sim today(A B))$   
**tries** :  $\square((\langle \rangle \exists g Nt(s(g)) \setminus S f) / \square(\langle \rangle \exists g Nt(s(g)) \setminus Si)) : \hat{\lambda} A \lambda B(\sim tries(\sim A B) B)$   
**tully** :  $\blacksquare Nt(s(m)) : t$   
**unicorn** :  $\square CNs(n) : unicorn$   
**up** :  $\blacksquare W[up] : 0$   
**upon** :  $\square((\forall b \forall f((\langle \rangle Nb \setminus S f) / (\langle \rangle Nb \setminus S f)) \& \forall g(CNg \setminus CNg) / \exists a Na) : \hat{\lambda} A(\sim uponadv A), (\sim uponadn A))$   
**void** :  $\square \forall g(CNg / CNg) : void$   
**walk** :  $\square(\langle \rangle (\exists a Na - \exists g Nt(s(g))) \setminus S f) : \hat{\lambda} A(Pres(\sim walk A))$   
**walk** :  $\square(\langle \rangle \exists a Na \setminus S b) : \hat{\lambda} A(\sim walk A)$   
**walks** :  $\square(\langle \rangle \exists g Nt(s(g)) \setminus S f) : \hat{\lambda} A(Pres(\sim walk A))$   
**was** :  $\blacksquare((\langle \rangle \exists g Nt(s(g)) \setminus S f) / (\exists a Na \oplus (\exists g(CNg / CNg) \sqcup (CNg \setminus CNg) - I))) : \lambda A \lambda B(Past(A \rightarrow C.[B = C]; D.((D \lambda E[E = B] B)))$   
**was** :  $\square((\langle \rangle W[there] \rightarrow S f) / \exists a Na) : \hat{\lambda} A(Past(\sim be A))$   
**waters** :  $\square CNp(n) : waters$   
**which** :  $\blacksquare \forall n \forall m((Nt(n) \uparrow Nt(m)) \downarrow ([\ ]^{-1} [\ ]^{-1} (CNm \setminus CNm) / \blacksquare((\langle \rangle Nt(n) \uparrow \blacksquare Nt(n)) \setminus S f))) : \lambda A \lambda B \lambda C \lambda D[(C D) \wedge (B(A D))]$   
**who** :  $\blacksquare \forall h \forall n([\ ]^{-1} [\ ]^{-1} (Nt(n) \setminus ((S h \uparrow Nt(n)) \downarrow S h)) / \blacksquare((\langle \rangle Nt(n) \uparrow \blacksquare Nt(n)) \setminus S f)) : \lambda A \lambda B \lambda C[(A B) \wedge (C B)]$   
**will** :  $\blacksquare \forall a((\langle \rangle Na \setminus S f) / (\langle \rangle Na \setminus S b)) : \lambda A \lambda B(Fut(A B))$   
**without** :  $\square(\forall g(CNg \setminus CNg) / \exists a Na) : \hat{\lambda} A \lambda B \lambda C[(B C) \wedge \neg(\sim with A C)]$   
**without** :  $\blacksquare \forall a \forall f([\ ]^{-1} ((\langle \rangle Na \setminus S f) / (\langle \rangle Na \setminus S f)) / (\langle \rangle Na \setminus S psp)) : \lambda A \lambda B \lambda C[(B C) \wedge \neg(A C)]$   
**woman** :  $\square CNs(f) : woman$   
**yesterday** :  $\square \forall a \forall f((\langle \rangle Na \setminus S f) / (\langle \rangle Na \setminus S f)) : \hat{\lambda} A \lambda B(\sim yesterday(A B))$

## References

1. Antony E. Ades and Mark J. Steedman. On the order of words. *Linguistics and Philosophy*, 4:517–558, 1982.
2. 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.
3. J. M. Andreoli. Logic programming with focusing in linear logic. *Journal of Logic and Computation*, 2(3):297–347, 1992.
4. Chris Barker and Chung-chieh Shan. *Continuations and Natural Language*. Oxford University Press, New York and Oxford, 2015.
5. Guy Barry, Mark Hepple, Neil Leslie, and Glyn Morrill. Proof Figures and Structural Operators for Categorical Grammar. In *Proceedings of the Fifth Conference of the European Chapter of the Association for Computational Linguistics*, Berlin, 1991.
6. Bob Carpenter. *Type-Logical Semantics*. MIT Press, Cambridge, MA, 1997.
7. N. Chomsky. Conditions on transformations. In S. Anderson and P. Kiparsky, editors, *A Festschrift for Morris Halle*, pages 232–286. Holt, Rinehart and Winston, New York, 1973.
8. Paul Deane. Limits to attention: A cognitive theory of island phenomena. *Cognitive Linguistics*, 2(1):1–63, 1991.
9. E. Engdahl. Parasitic gaps. *Linguistics and Philosophy*, 6:5–34, 1983.

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. Philip Hofmeister and Ivan A. Sag. Cognitive constraints and island effects. *Language*, 86(2):366–415, 2010.
14. Gerhard Jäger. *Anaphora and Type Logical Grammar*, volume 24 of *Trends in Logic – Studia Logica Library*. Springer, Dordrecht, 2005.
15. Martin Jansche and Shrvan Vasishth. Review of Mark Steedman, The Syntactic Process. *Journal of Linguistics*, 38:684–690, 2002.
16. M. Kanazawa. The Lambek calculus enriched with additional connectives. *Journal of Logic, Language and Information*, 1:141–171, 1992.
17. Andrew Kehler. *Coherence, Reference and the Theory of Grammar*. CSLI Publications, Stanford, California, 2002.
18. Robert Kluender. Deriving island constraints from principles of predication. In H. Goodluck and M. Rochemont, editors, *Island Constraints: Theory, Acquisition, and Processing*, pages 223–258. Kluwer, Dordrecht, 1992.
19. Robert Kluender. On the distinction between strong and weak islands: A processing perspective. In P. Culicover and L. McNally, editors, *The Limits of Syntax*, volume 29 of *Syntax and Semantics*, pages 241–279. Academic Press, San Diego, 1998.
20. Yusuke Kubota and Robert Levine. Against ellipsis: Arguments for the direct licensing of ‘non-canonical’ coordinations. *Linguistics and Philosophy*, 2015. to appear.
21. George Lakoff. Frame semantic control of the Coordinate Structure Constraint. In A. M. Farley, P. T. Farley, and K.-E. McCullough, editors, *CLS 22 PART 2: Papers from the Parasession on Pragmatics and Grammatical Theory, 152*. Chicago Linguistics Society, Chicago, 1986.
22. 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.
23. J. Lambek. Categorical and Categorical Grammars. In Richard T. Oehrle, Emmon Bach, and Deidre Wheeler, editors, *Categorical Grammars and Natural Language Structures*, volume 32 of *Studies in Linguistics and Philosophy*, pages 297–317. D. Reidel, Dordrecht, 1988.
24. Joachim Lambek. The mathematics of sentence structure. *American Mathematical Monthly*, 65:154–170, 1958.
25. Robert D. Levine and Thomas E. Hukari. *The Unity of Unbounded Dependency Constructions*. CSLI Publications, Stanford, California, 2006.
26. Vedrana Mihalíček and Carl Pollard. Distinguishing phenogrammar from tectogrammar simplifies the analysis of interrogatives. In P. de Groot and M.-J. Nederhof, editors, *Formal Grammar 2010/2011*, pages 130–145. Heidelberg, 2012. Springer.
27. Michael Moortgat. *Categorical Investigations: Logical and Linguistic Aspects of the Lambek Calculus*. Foris, Dordrecht, 1988. PhD thesis, Universiteit van Amsterdam.
28. 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.
29. Michael Moortgat. Categorical 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.
30. Richard Moot and Christian Retoré. *The Logic of Categorical Grammars: A Deductive Account of Natural Language Syntax and Semantics*. Springer, Heidelberg, 2012.
31. G. Morrill. Grammar and Logical Types. In Martin Stockhof and Leen Torenvliet, editors, *Proceedings of the Seventh Amsterdam Colloquium*, pages 429–450, 1990.
32. Glyn Morrill. Intensionality and Boundedness. *Linguistics and Philosophy*, 13(6):699–726, 1990.
33. Glyn Morrill. Categorical 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.
34. 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.
35. Glyn Morrill and Josep-Maria Merenciano. Generalising discontinuity. *Traitement automatique des langues*, 37(2):119–143, 1996.
36. 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>.

37. 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>.
38. 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.
39. 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).
40. 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.
41. 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.
42. 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.
43. Glyn V. Morrill. *Type Logical Grammar: Categorical Logic of Signs*. Kluwer Academic Publishers, Dordrecht, 1994.
44. Glyn V. Morrill. *Categorical Grammar: Logical Syntax, Semantics, and Processing*. Oxford University Press, New York and Oxford, 2011.
45. Reinhard Muskens. Language, lambdas and logic. In Geert-Jan M. Kruijff and Richard T. Oehrlé, editors, *Resource-Sensitivity in Binding and Anaphora*, volume 80 of *Studies in Linguistics and Philosophy*, pages 23–54. Kluwer Academic Press, Dordrecht, 2003.
46. Paul M. Postal. Parasitic Gaps and Across-the-Board Phenomenon. *Linguistic Inquiry*, 24(4):735–754, 1993.
47. I.A. Sag. On parasitic gaps. *Linguistics and Philosophy*, 6:35–45, 1983.
48. Mark Steedman. Combinatory Grammars and Parasitic Gaps. *Natural Language and Linguistic Theory*, 5:403–439, 1987.
49. Anna Szabolcsi. ECP in categorial grammar. *Manuscript*, 1983, Max Planck Institute, Nijmegen.
50. T. Taraldsen. The theoretical interpretation of a class of marked extractions. In A. Belletti, L. Brandi, and L. Rizzi, editors, *Theory of Markedness in Generative Grammar*. Scuole Normal Superiore de Pisa, Pisa, 1979.
51. Oriol Valentín. *Theory of Discontinuous Lambek Calculus*. PhD thesis, Universitat Autònoma de Barcelona, Barcelona, 2012.
52. Oriol Valentín. The Hidden Structural Rules of the Discontinuous Lambek Calculus. In Claudia Casadio, Bob Coecke, Michael Moortgat, 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 402–420. Springer, Berlin, 2014.
53. 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.
54. 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.