Copy Raising and Formal Variation[*]

Ash Asudeh

Oxford University & Carleton University

`ash.asudeh@ling-phil.ox.ac.uk`

November 24, 2011

University of Edinburgh

1 Introduction

- Generative linguistics has been tasked since its inception with capturing underlying commonalities in distinct grammars — call it *language universals*.

- The study of differences between grammars — call it *language variation* — has an even longer tradition.

- The study of variation was brought back into the fold of mainstream formal linguistics some thirty years ago with the advent of Principles & Parameters Theory (Chomsky 1981), henceforth P&P.

- Language variation from a P&P perspective (broadly construed), continues to be pursued with the context of the Minimalist Program (Chomsky 1995), where the locus of variation is lexical features.

- Within this context, attention has recently also focused on variation between closely related languages and dialects. This often goes under the rubric of *syntactic microvariation* (see, e.g., Barbiers 2008), *microparametric syntax* (see, e.g., Black and Montapanyane 1996, Cinque and Kayne 2005, or *microcomparative syntax* (see, e.g., Cinque and Kayne 2005 or the NORMS project).

- Given the standard psychological assumptions of generative linguistics — in particular, that grammars are properties of individuals — there is of course no coherent distinction between general language variation and microvariation.

- Microvariation is nevertheless interesting for at least two reasons:

  1. It potentially strengthens the connection between theoretical linguistics and sociolinguistics.
  2. It minimises factors that may confound the point of variation of interest.

- This talk concerns some points of variation in a corner of English and how to capture this variation in a formally satisfactory way.

- The phenomenon of interest is English *copy raising* (Rogers 1973, Postal 1974, Horn 1981, Heycock 1994, Gisborne 1996, 2010, Potsdam and Runner 2001, Matushansky 2002, Asudeh 2002, 2004, 2012, Asudeh and Toivonen 2006, 2007, 2012, Landau 2009b,a):

  (1)   Harry seems/appears like/as if/as though <u>he</u> fell.

---

2 Main Claims

- Copy raising is a kind of resumption, but not involving an unbounded dependency.

- There are four distinct dialects or grammar-types for English copy raising.

- The distinctions between these dialects can be captured formally, and hopefully insightfully, in a theory that pairs a lexicalist syntactic theory (Lexical-Functional Grammar) with a flexible theory of semantic composition (Glue Semantics).

3 Overview

1. *Introduction*

2. *Main Claims*

3. *Overview*

4. Data and Generalizations

5. Background

   (a) Lexical-Functional Grammar
   (b) Glue Semantics

6. The Resource Management Theory of Resumption

7. Analysis: Dialect C

8. A Formal Lexicalist Account of Copy Raising Variation

9. Conclusion

4 Data and Generalizations

- True copy raising in English is limited to the verbs *seem* and *appear* with complements introduced by *like*, *as if* or *as though*, containing a pronominal 'copy' of the subject of the copy raising verb. This was shown in (1) above, repeated here:

  (1)   Harry seems/appears like/as if/as though <u>he</u> fell.

- The similar subcategorizations of the perception verbs *look*, *sound*, *smell*, *taste* and *feel* are typically grouped with copy raising, but I will keep these aside, for reasons that will become clearer shortly.

- In this talk, I want to focus on variation in copy raising with respect to three factors:

  1. Position of the copy pronoun
  2. Obligatoriness of the copy pronoun
  3. Expletive alternants of copy raising verbs

- English speakers can be divided into four dialects, according to the results of questionnaire studies of 110 native speakers reported by Asudeh and Toivonen (2012).

- The division is based on patterns of grammaticality judgements, on a forced three-point scale, for the following kinds of sentences, presented without context, mixed with grammatical and ungrammatical fillers:

  (2)   Alfred seems like he hurt Thora.

  (3)   Alfred seems like Madeline claimed that he hurt Thora.

  (4)   Alfred seems like Thora hurt him.

  (5)   Alfred seems like Thora's hurt.

  Dialect A: These speakers do not accept *seem/appear* with a non-expletive matrix subject and a *like*-complement and would reject all of (2)–(5).

  Dialect B: These speakers accept copy raising only if the copy pronoun is the highest subject in the complement introduced by *like/as if/as though*; they would reject (3)–(5).

  Dialect C: These speakers accept copy raising no matter where the copy pronoun occurs in the *like*-complement; they would reject only (5).

  Dialect D: These speakers accept *seem/appear* with a non-expletive matrix subject and a *like*-complement but do not require a copy pronoun in the complement.

- The copy raising verbs *seem* and *appear* can take an expletive subject and a *like*-complement, as in (6). This is a key piece of evidence that copy raising can take a non-thematic subject and therefore really is a kind of raising (Rogers 1974, Horn 1981, Potsdam and Runner 2001).

  (6)   a.   It seems like Harry fell.
             b.   It appears as if Alfred hurt Harry.

- Even Dialect A speakers, who reject copy raising, accept examples like these.

- Table 2 summarizes the grammaticality patterns for the four dialects by sentence type.

	% of speakers (n = 110)	Description
Dialect A	6.35%	No copy raising subcategorization with non-expletive matrix subject
Dialect B	45.1%	True copy raising I — copy pronoun must be highest subject in complement of *like/as*
Dialect C	42.2%	True copy raising II — copy pronoun not necessarily highest subject
Dialect D	6.35%	Copy raising subcategorization with non-expletive matrix subject and no copy pronoun in complement

Table 1: Variation for English copy raising

	Dialect			
Example	**A**	**B**	**C**	**D**
It seems like Harry fell.	✓	✓	✓	✓
Alfred seems like he hurt Thora.	*	✓	✓	✓
Alfred seems like Madeline claimed that he hurt Thora.	*	*	✓	✓
Alfred seems like Thora hurt him.	*	*	✓	✓
Alfred seems like Thora's hurt.	*	*	*	✓

Table 2: Grammaticality patterns for English copy raising

- Copy raising also shows interesting variation with respect to expletives.

- A number of speakers also allow an expletive *there* to be the subject of copy raising, as in (7), even though the copy raising verb cannot otherwise take a *there* expletive subject, as shown in (8) and (9).

  (7) a. % There seems like there's moisture in the engine.

       b. % There seem like there are two garden gnomes missing.

  (8) a.   It seems like Harry's jumping.

       b. * There seems like Harry's jumping.

  (9) a.   It seems like it's raining.

       b. * There seems like it's raining.

- These examples show that a matrix *there* expletive subject is licensed only by virtue of the embedded *there* expletive. This is further underscored by the fact that, at least for some speakers, the matrix expletive has the agreement features of the embedded expletive, as shown in (7).

- Raising is a local operation: we would expect that the verb could raise only the subject of the *like*-complement; otherwise we would have to give up the locality of raising.

- This leads to the following surprising generalization:

  (10)   *Like* and *as* have raising alternants.

- This means that *like* or *as*, the head of the *like*-complement, raises the expletive subject from its complement, and then the expletive is raised one step further by the copy raising verb, which we know independently can raise the subject of its predicative complement.

- Thus, we have double raising, but each step is completely local.

5 Background

5.1 Lexical-Functional Grammar

- LFG is a declarative, constraint-based linguistic theory (Kaplan and Bresnan 1982).

- The motivation behind LFG is to have a theory that contributes in three ways to our understanding of language:

  1. Theory, including language universals and typology
  2. Psycholinguistics, including language acquisition
  3. Computational linguistics, including automatic parsing and generation, machine translation, and language modelling

5.1.1 The Correspondence Architecture

- The grammatical architecture of LFG posits that different kinds of linguistic information are modelled by distinct data structures, all of which are present simultaneously.

- Structures are related by functions, called correspondence or projection functions., which map elements of one structure to elements of another.

- This architecture is a generalization of the architecture of Kaplan and Bresnan (1982) and is called the *Parallel Projection Architecture* or *Correspondence Architecture* (Kaplan 1987, 1989, Halvorsen and Kaplan 1988, Asudeh 2006, 2012, Asudeh and Toivonen 2009).

- Syntax: constituent structure (c-structure) and functional structure (f-structure).

- C-structure is represented by phrase structure trees:

  1. Word order
  2. Dominance
  3. Constituency
  4. Syntactic categories

- F-structure is represented by feature structures (also known as attribute value matrices):

  1. Grammatical functions, such as SUBJECT and OBJECT
  2. Case
  3. Agreement
  4. Tense and aspect
  5. Local dependencies (e.g., control and raising)
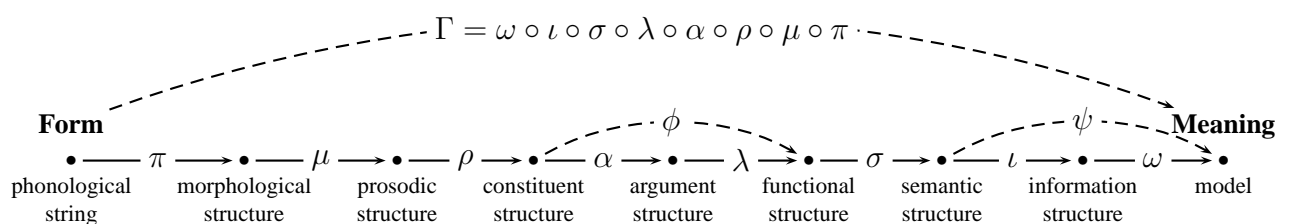 6. Unbounded dependencies (e.g., question formation, relative clause formation)

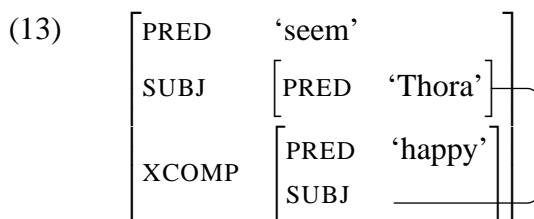Figure 1: The Correspondence Architecture, pipeline version (Asudeh 2012)

5.1.2   Raising

- Raising is represented as functional equality between a grammatical function of the raising verb and the subject of its open complement (Bresnan 1982), which is a predicative or infinitival complement.

- An open complement is normally represented as the grammatical function XCOMP and lacks a subject of its own.

- The XCOMP must have its subject specified by the predicate that selects the XCOMP, through a local functional equality called a *functional control* equation (Bresnan 1982).
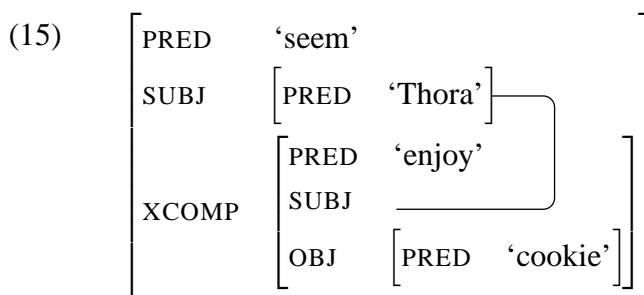
  (11) $(\uparrow \text{SUBJ}) = (\uparrow \text{XCOMP SUBJ})$

- The matrix and subordinate subjects are identified at f-structure and share a single, token-identical value.

  (12)   Thora seems happy.

  (13)
  $$\begin{bmatrix} \text{PRED} & \text{`seem'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`Thora'} \end{bmatrix} \\ \text{XCOMP} & \begin{bmatrix} \text{PRED} & \text{`happy'} \\ \text{SUBJ} & \underline{\quad\quad} \end{bmatrix} \end{bmatrix}$$

  (14)   Thora seems to enjoy cookies.

  (15)
  $$\begin{bmatrix} \text{PRED} & \text{`seem'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`Thora'} \end{bmatrix} \\ \text{XCOMP} & \begin{bmatrix} \text{PRED} & \text{`enjoy'} \\ \text{SUBJ} & \underline{\quad\quad} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{`cookie'} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

- Raising is thus a lexically controlled local dependency and involves simultaneous instantiation of two grammatical functions to a single f-structure value. There is no movement involved in raising and the target of raising in the complement is not represented in c-structure (Dalrymple 2001, Asudeh 2005a, Asudeh and Toivonen 2009).

5.1.3 Templates

- An LFG template is a label for a functional description — a set of equations and constraints that describes linguistic structures, such as the functional descriptions that describe f-structures (Dalrymple et al. 2004, Asudeh et al. 2008, Crouch et al. 2011).

- Template invocation is denoted by the prefix @ in a functional description.

- The semantics of template invocation is substitution. Any occurrence of a template in a lexical entry or rule can be equivalently replaced by the grammatical description that the template is associated with.

- Templates are therefore purely abbreviatory devices, but can nevertheless capture linguistic generalizations, since associating a grammatical description with a template treats the description as a natural class.

- Thus, a grammar with templates is extensionally equivalent to the same grammar with all templates replaced by their associated grammatical descriptions, but the first grammar might express generalizations that the second grammar does not.

- Templates can also encode information hierarchically, since template definitions may refer to other templates. This is reminiscent of the type hierarchies of HPSG (Pollard and Sag 1987, 1994) and Sign-Based Construction Grammar (SBCG; Michaelis 2010, Sag 2010).

- However, template hierarchies represent inclusion, rather than inheritance. If template B is a sub-template of template A, then the description that A labels is included in the description that B labels.

- Illustration: English agreement

  (16) $3\text{SG} =$ $(\uparrow \text{SUBJ PERS}) = 3$
        $(\uparrow \text{SUBJ NUM}) = \text{SG}$

  (17) *smiles* $(\uparrow \text{PRED}) = \text{`smile}\langle\text{SUBJ}\rangle\text{'}$
        @3SG

  (18) *smile* $(\uparrow \text{PRED}) = \text{`laugh}\langle\text{SUBJ}\rangle\text{'}$
        $\neg$@3SG

  (19)
  $$3\text{SG}$$
  smiles   *smile*

- Templates can also take arguments.

  (20) $\text{INTRANS}(P) =$ $(\uparrow \text{PRED}) = \text{`P}\langle\text{SUBJ}\rangle\text{'}$

- Given the substitutional semantics of templates, the following two lexical entries are equivalent:

  (21) *smiles* @INTRANS(smile)
        @3SG

  (22) *smiles* $(\uparrow \text{PRED}) = \text{`smile}\langle\text{SUBJ}\rangle\text{'}$
        $(\uparrow \text{SUBJ PERS}) = 3$
        $(\uparrow \text{SUBJ NUM}) = \text{SG}$

5.1.4 Examples

(23)

CP
DP C′
who
C IP
did DP I′
you
VP
V′
V CP
say C′
IP
I′
VP
V′
V DP
injured himself

ϕ

$$\begin{bmatrix} \text{PRED} & \text{'say}\langle\text{SUBJ,COMP}\rangle\text{'} \\ \text{UDF} & \begin{bmatrix} \text{PRED} & \text{'pro'} \\ \text{PRONTYPE} & \text{WH} \end{bmatrix} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{'pro'} \\ \text{PERSON} & 2 \end{bmatrix} \\ \text{COMP} & \begin{bmatrix} \text{PRED} & \text{'injure}\langle\text{SUBJ,OBJ}\rangle\text{'} \\ \text{SUBJ} & \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{'pro'} \\ \text{PRONTYPE} & \text{REFL} \\ \text{PERSON} & 3 \\ \text{NUMBER} & \text{SING} \\ \text{GENDER} & \text{MASC} \end{bmatrix} \\ \text{TENSE} & \text{PAST} \\ \text{MOOD} & \text{DECLARATIVE} \end{bmatrix} \\ \text{TENSE} & \text{PAST} \\ \text{MOOD} & \text{INTERROGATIVE} \end{bmatrix}$$

(24)

IP
I′
I⁰ VP
Joi-(n) V′
DP
vettä

$$\begin{bmatrix} \text{PRED} & \text{'drink}\langle\text{SUBJ,OBJ}\rangle\text{'} \\ \text{TENSE} & \text{PAST} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{'pro'} \\ \text{PERSON} & 1 \\ \text{NUMBER} & \text{SG} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{'water'} \\ \text{PERSON} & 3 \\ \text{NUMBER} & \text{SG} \end{bmatrix} \end{bmatrix}$$

IP
DP I′
I VP
V′
V⁰ DP
drank water

5.2 Glue Semantics

- Glue Semantics (Dalrymple 1999, 2001, Asudeh 2004, 2005b, 2012, Lev 2007, Kokkonidis 2008) is a theory of semantic composition and the syntax–semantics interface.

- Glue *meaning constructors* are obtained from lexical items instantiated in particular syntactic structures.

  (25) $\mathcal{M} : G$

  $\mathcal{M}$ is a term from some representation of meaning, a *meaning language*, and G is a term of the Glue logic that sticks meanings together, i.e. performs composition. The colon is an uninterpreted pairing symbol.

- Linear logic (Girard 1987) serves as the Glue logic (Dalrymple et al. 1993, 1999a,b).

- The meaning constructors are used as premises in a (linear logic) proof that consumes the lexical premises to produce a sentential meaning.

- A successful Glue proof for a sentence terminates in a meaning constructor of type t:

  (26) $\Gamma \vdash \mathcal{M} : G_t$

- Alternative derivations from the same set of premises $\rightarrow$ semantic ambiguity (e.g., scope)

- Linear logic is a *resource logic*: each premise in valid linear logic proof must be used exactly once.

- As discussed in detail by Dalrymple et al. (1999a), Glue Semantics is essentially a type-logical theory and is thus related to type-logical approaches to Categorial Grammar (Morrill 1994, Moortgat 1997, Carpenter 1997, Jäger 2005).

- The key difference between Glue and Categorial Grammar concerns grammatical architecture, particularly the conception of the syntax–semantics interface (Asudeh 2004, 2005b, 2006). Glue Semantics posits a strict separation between syntax and semantics, such that there is a syntax that is separate from the syntax of semantic composition. Categorial Grmamar rejects the separation of syntax from semantic composition.

- I assume a small, rather weak fragment of linear logic, multiplicative intuitionistic linear logic (MILL; Asudeh 2004, 2005b).

- Three proof rules of this fragment are of particular interest here: elimination for $\otimes$ (multiplicative conjunction) and introduction and elimination for linear implication $\multimap$.

Application : Impl. Elim.   Abstraction : Impl. Intro.   Pairwise substitution : Conj. Elim.

$$\cfrac{a : A \qquad f : A \multimap B}{f(a) : B} \,{\multimap}_{\mathcal{E}} \qquad \cfrac{\begin{array}{c}[x : A]^1 \\ \vdots \\ f : B\end{array}}{\lambda x.f : A \multimap B}\,{\multimap}_{\mathcal{I},1} \qquad \cfrac{a : A \otimes B \qquad \begin{array}{c}[x : A]^1 \ [y : B]^2 \\ \vdots \\ f : C\end{array}}{let\ a\ be\ x \times y\ in\ f : C}\,{\otimes}_{\mathcal{E},1,2}$$

Figure 2: Linear logic proof rules with Curry-Howard correspondence

(27)   Bo chortled.

(28) $\cfrac{bo : b \qquad chortle : b \multimap c}{chortle(bo) : c}\,{\multimap}_{\mathcal{E}}$

- Anaphora in Glue Semantics are typically treated as functions on their antecedents (Dalrymple et al. 1999c, Dalrymple 2001). This is a kind of a variable-free treatment of anaphora, which has also been adopted in certain Categorial Grammar analyses (Jacobson 1999, Jäger 2005, among others), although the two variable-free traditions developed separately.

- A variable-free treatment of anaphora is quite natural in Glue, because the commutative linear logic allows anaphora to combine directly with their antecedents, in opposition to the kind of intervening operations that are necessary for variable-free anaphoric resolution in non-commutative Categorial Grammar.

- The meaning constructor for a pronominal has the following general form, where $\uparrow$ is the f-structure of the pronoun and $\uparrow_\sigma$ is its σ-projection in sem-structure:

(29) $\quad \lambda z.z \times z : (\uparrow_\sigma \text{ANTECEDENT}) \multimap [(\uparrow_\sigma \text{ANTECEDENT}) \otimes \uparrow_\sigma]$

- The pronoun's type is therefore $\langle \sigma, \langle \sigma, \tau \rangle \rangle$, where σ is the type of the antecedent and τ is the type of the pronoun. I here assume that both σ and τ are type e (individuals).

(30) $\quad$ Bo fooled himself.

(31)

$$
\cfrac{
  \cfrac{
    \text{\textbf{Bo}} \quad \text{\textbf{himself}}
  }{
    bo : b \quad \lambda z.z \times z : b \multimap (b \otimes p)
  }
  \quad
  \cfrac{
    bo \times bo : b \otimes p
  }{}{}^{\multimap_\mathcal{E}}
  \qquad
  \cfrac{
    \cfrac{
      [x:b]^1 \quad \cfrac{\text{\textbf{fooled}}}{\lambda u \lambda v.fool(u,v) : b \multimap p \multimap f}
    }{
      \lambda v.fool(x,v) : p \multimap f
    }^{\multimap_\mathcal{E}} \quad [y:p]^2
  }{
    fool(x,y) : f
  }^{\multimap_\mathcal{E}}
}{
  \cfrac{
    let\ bo \times bo\ be\ x \times y\ in\ fool(x,y) : f
  }{
    fool(bo,bo) : f
  }^{\Rightarrow_\beta}
}^{\otimes_{\mathcal{E},1,2}}
$$

6 The Resource Management Theory of Resumption

- The Resource Management Theory of Resumption (RMTR) is based on the following two claims.

  1. **The Resource Sensitivity Hypothesis (RSH)**:
     Natural language is resource-sensitive.

  2. **McCloskey's Generalization**:
     Resumptive pronouns are ordinary pronouns.

- RSH stems from the resource-logical perspective on semantic composition in Glue Semantics (Dalrymple 1999, 2001), which uses the resource logic *linear logic* (Girard 1987) to assemble meanings.

- The upshot of RSH is that compositional semantics is constrained by resource accounting, such that component meanings cannot go unused or be reused.

- For example, in the following sentence, the adverb *slowly* contributes a single lexical meaning resource which cannot be used twice to derive the unavailable meaning that the plummeting was also slow.

  (32) $\quad$ John rolled over the edge slowly and plummeted to the ground.

- The Resource Sensitivity Hypothesis paves the way to substantial simplification, since the following independent principles can be reduced to resource sensitivity (Asudeh 2012: 110–123):

    1. Bounded Closure

    2. Completeness and Coherence

    3. The Theta Criterion

    4. The Projection Principle

    5. No Vacuous Quantification

    6. The Inclusiveness Condition

    7. Full Interpretation

- Not only does RSH set the ground for eliminating these principles from our theories, it also gives us a deeper understanding of the principles, since they are reduced to the basic combinatoric logic of language.

- Consider resumptive pronouns in unbounded dependencies, as in (33), and copy pronouns in copy arising, as in (34):

    (33) an ghirseach ar   ghoid na síogaí í   (Irish)
          the girl   COMP.PAST stole the fairies her
          'the girl that the fairies stole (her) away'
          (McCloskey 2002: 189, (9b))

    (34) Thora seems like she's enjoying the movie.

- From the perspective of RSH, both the resumptive pronoun in (33) and the copy pronoun in (34) are equally *surplus resources*.

- The logic behind the Resource Management Theory of Resumption is as follows. If a resumptive (unbounded dependency resumptive or copy pronoun) is an ordinary pronoun, then it constitutes a surplus resource, because it would otherwise saturate either the scope of an operator, as in (33), or the argument position of the matrix subject, as in (34). If Resource Sensitivity is to be maintained, then there must be an additional consumer of the pronominal resource present.

- The consumer of the resumptive or copy pronoun is a further resource that consumes a pronominal resource. These resources are called *manager resources*, because they manage an otherwise unconsumable pronominal resource.

- A language with copy raising has such manager resources lexically associated with copy raising verbs.

- Manager resources have the following general compositional schema, where P is some pronoun that the lexical contributor of the manager resource can access and A is the antecedent or binder of P:

  (35) $(A \multimap A \otimes P) \multimap (A \multimap A)$

- The general template for manager resources:

  (36) $\mathrm{MR}_2(g, f) = \lambda P \lambda y.y : [g_\sigma \ \multimap \ (g_\sigma \ \otimes \ f_\sigma)] \ \multimap \ [g_\sigma \ \multimap \ g_\sigma]$

- F-structure f is the f-structure of the pronoun to be consumed and f-structure g is the f-structure of the UDF, in the case of unbounded dependency resumptives, or of the SUBJ, in the case of copy raising resumptives.

- In order to avoid some clutter, we define a monadic template for copy raising as follows, are defined in terms of MR_2 as follows, where MR_s is a manager resource parametrized such that the antecedent of the surplus pronoun is the local SUBJECT :

  (37) $\mathrm{MR}_s(f) = \mathrm{MR}_2((\uparrow \text{SUBJ}), f)$
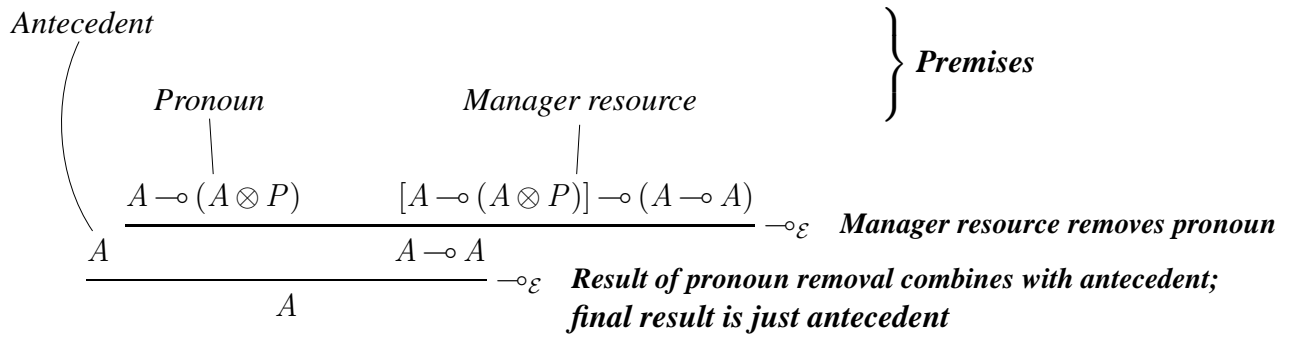

Figure 3: A manager resource in action (binder of lower type)
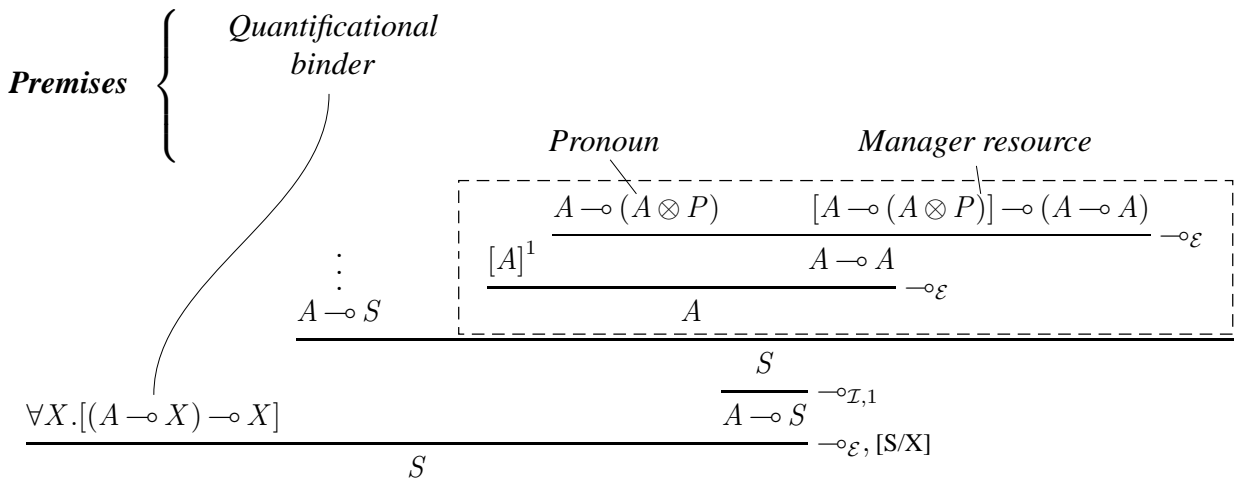

Figure 4: A manager resource in action (quantificational binder)

7   Analysis: Dialect C

- I assume an event semantics for copy raising (for more details, see Asudeh and Toivonen 2012).

- Events facilitate the analysis of the semantics of the expressions that introduce the complement to copy raising, *like/as if/as though*.

    (38)   Alfred seems like he hurt Thora.

    (39)   Alfred seems like he knows the answer.

- Copy raising involves a comparison between a state of something seeming the be the case and an eventuality in the *like*-complement, where I follow Bach (1981) in adopting the term *eventuality* as a cover term for events and states.

- A target event semantics for (38) is:

    (40) $\exists s.seem(s, \exists e.[(s \sim e) \land hurt(e, a, t)])$

- I adopt variables $s, s', s'', \ldots$, for states and variables $e, e', e'', \ldots$, for events. I adopt an eventuality metavariable, e, over state or event variables. All bound instances of e in a formula must be resolved to the same state or event variable; Glue proofs below will involve an implicit resolution of this kind, marked as, e.g., [e/ε].

- Here is an example of an event semantics analysis of a simpler sentence. The proof assumes that there is a generally available optional operation of existential closure of an eventuality variable, indicated by $\exists\varepsilon$; this is a standard assumption in event semantics, which can be implemented in various ways (Asudeh 2012).

    (41)   Alfred consoled Thora.

    (42)   1. $a : a$ Lex. **Alfred**
              2. $\lambda y \lambda x \lambda e.console(e, x, y) : t \multimap a \multimap e \multimap c$ Lex. **consoled**
              3. $t : t$ Lex. **Thora**

    (43)
$$
\cfrac{
  \cfrac{
    \textbf{Thora} \quad\quad \textbf{consoled}
    \atop
    t : t \quad\quad \lambda y \lambda x \lambda e.console(e, x, y) : t \multimap a \multimap e \multimap c
  }{
    \lambda x \lambda e.console(e, x, t) : a \multimap e \multimap c
  }
  \quad
  {\textbf{Alfred} \atop a : a}
}{
  \cfrac{
    \lambda e.console(e, a, t) : e \multimap c
  }{
    \exists e.console(e, a, t) : c
  } \; \exists\varepsilon
}
$$

- The interpretation in (40) introduces a similarity operator, $\sim$, which is defined as follows:

    (44)   For any two eventualities α and β, $\alpha \sim \beta$ is true if and only if there is a property P such that $P(\alpha)$ is true and $P(\beta)$ is true.

    In other words, for two eventualities to be considered similar, they must share some property.

- The similarity operator provides the basis for the following lexical entry for *like*, the head of the copy raising verb's complement:

    (45)   *like*: P^0 ($\uparrow$ PRED) $=$ 'like'
                        $\lambda P \lambda s.\exists\varepsilon.[(s \sim \varepsilon) \land P(\varepsilon)] :$
                        $[(\uparrow \text{COMP}_\sigma \text{ EVENT}) \multimap (\uparrow \text{COMP})_\sigma] \multimap$
                        $((\text{XCOMP} \uparrow)_\sigma \text{ EVENT}) \multimap \uparrow_\sigma$

- The lexical entry for the *seems like/as though/as if* subcategorization of *seem* in Dialect C is given in (46). It uses two templates, which are defined in (47) and (48).

(46) $seem_{like}$:   V $(\uparrow \text{PRED}) = $ 'seem'
                        @RAISING
                        $(@\text{CR}(\uparrow \text{GF}^{+}))$

                        $\lambda P \lambda s.seem(s, P(s)) :$
                        $[(\uparrow_\sigma \text{EVENT}) \multimap (\uparrow \text{XCOMP})_\sigma] \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$

(47)   RAISING $=$ $\{ \ (\uparrow \text{SUBJ EXPLETIVE}) =_c \text{IT} \ \wedge \ \neg \ (\uparrow \text{XCOMP}) \ \ | $
                            $(\uparrow \text{SUBJ}) = (\uparrow \text{XCOMP SUBJ}) \ \}$

(48)   CR$(f) = $ $\%\text{Copy} = f$
                   $(\uparrow \text{SUBJ})_\sigma = (\%\text{Copy}_\sigma \text{ ANTECEDENT})$
                   $@\text{MR}_s(\%\text{Copy})$
                   $@\text{RELABEL}_s(\%\text{Copy})$

                   $\lambda x \lambda P.P(x) : (\uparrow \text{SUBJ})_\sigma \multimap [(\uparrow \text{SUBJ}) \multimap \uparrow_\sigma] \multimap \uparrow_\sigma$

- The meaning constructor in the lexical entry for the copy raising verb, (46), treats the meaning of the verb as a property of states that results from combining with the property of eventualities that is the meaning of its complement (the *like*-complement).

- The c-structure and f-structure for (38) are shown in (49) and (50). This f-structure instantiates the lexically contributed meaning constructors, shown in (51). Figure 5 shows the proof constructed from these premises. Figure 6 also shows the meaning language side of the meaning constructors. The result of the proof is the target interpretation in (40): $\exists s.seem(s, \exists e.[(s \sim e) \wedge hurt(e, a,$

(49)

```
                        IP
          ┌──────────────┴──────────────┐
    (↑ SUBJ) = ↓                    ↑ = ↓
        DP                            I′
        △                             │
      Alfred                        ↑ = ↓
                                      VP
                          ┌───────────┴───────────┐
                      ↑ = ↓                  (↑ XCOMP) = ↓
                       V⁰                         PP
                        │                          │
                      seems                      ↑ = ↓
                                                   P′
                                    ┌──────────────┴──────────────┐
                                ↑ = ↓                      (↑ COMP) = ↓
                                 P⁰                              IP
                                  │               ┌──────────────┴──────────────┐
                                 like       (↑ SUBJ) = ↓                    ↑ = ↓
                                                 DP                           I′
                                                 △                            │
                                                 he                         ↑ = ↓
                                                                             VP
                                                                ┌─────────────┴─────────────┐
                                                            ↑ = ↓                    (↑ OBJ) = ↓
                                                             V                            DP
                                                             │                            △
                                                           hurt                         Thora
```

(50)

$$
\begin{bmatrix}
\text{PRED} & \text{‘seem’} \\
\text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{‘Alfred’} \end{bmatrix} \\
\text{XCOMP} & \begin{bmatrix}
\text{PRED} & \text{‘like’} \\
\text{SUBJ} & \quad \\
\text{COMP} & \begin{bmatrix}
\text{PRED} & \text{‘hurt’} \\
\text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{‘pro’} \\ \text{PERS} & 3 \\ \text{NUM} & \text{SG} \\ \text{GEND} & \text{MASC} \end{bmatrix} \\
\text{OBJ} & \begin{bmatrix} \text{PRED} & \text{‘Thora’} \end{bmatrix}
\end{bmatrix}
\end{bmatrix}
\end{bmatrix}
$$
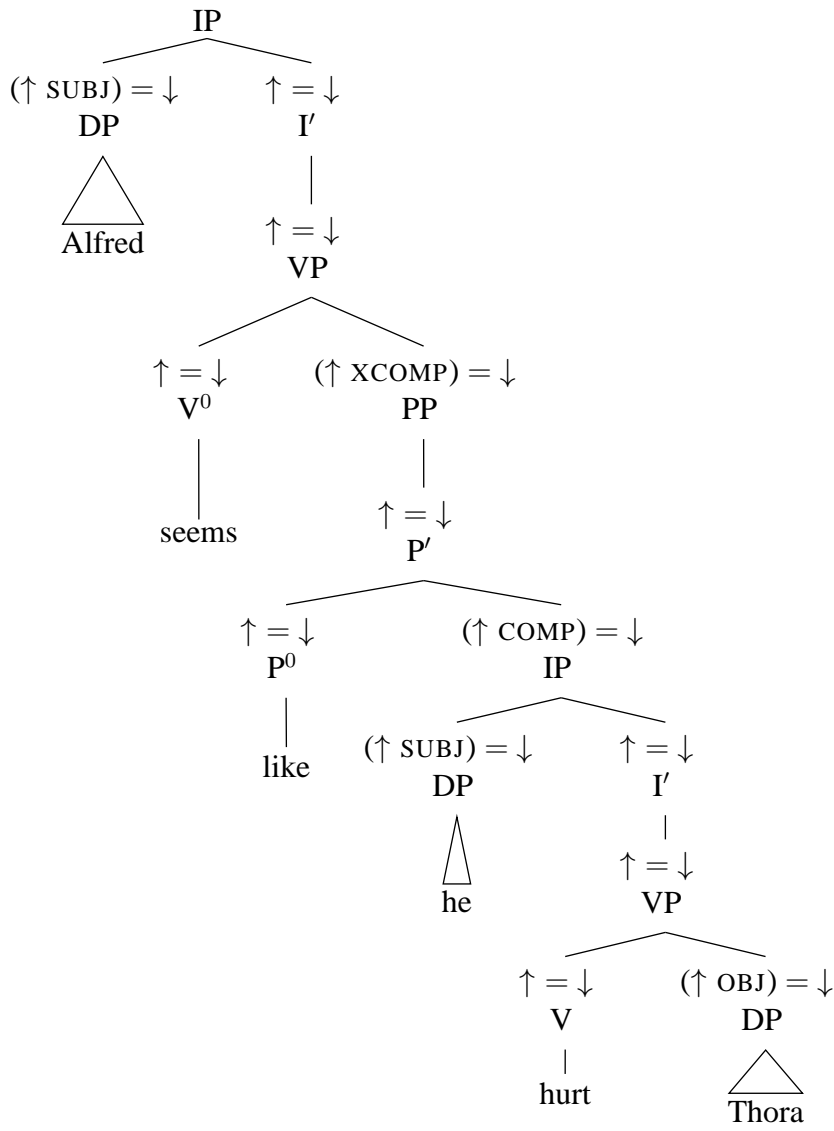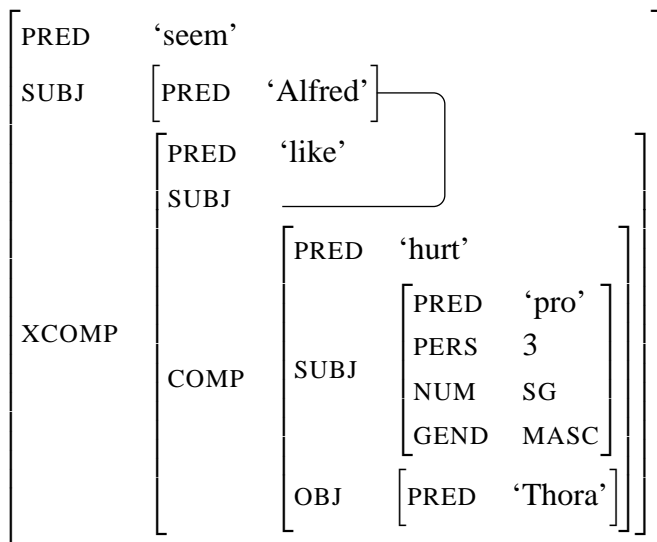
(51)   1. $a : a$ Lex. **Alfred**

2. $\lambda P \lambda s.seem(s, P(s)) :$ Lex. **seems**
$(e1 \multimap l) \multimap (e1 \multimap s)$

3. $\lambda x \lambda P.P(x) :$ Lex. **seems (CR)**
$a \multimap (a \multimap s) \multimap s$

4. $\lambda P \lambda y.y :$ Lex. **seems (CR: MR)**
$[a \multimap (a \otimes p)] \multimap (a \multimap a)$

5. $\lambda P.P :$ Lex. **seems (CR: RELABEL)**
$(p \multimap s) \multimap (a \multimap s)$

6. $\lambda P \lambda s'.\exists e.[(s' \sim e) \wedge P(e)] :$ Lex. **like**
$(e2 \multimap h) \multimap e1 \multimap l$

7. $\lambda z.z \times z :$ Lex. **he**
$a \multimap (a \otimes p)$

8. $\lambda y \lambda x \lambda e'.hurt(e', x, y) :$ Lex. **hurt**
$t \multimap p \multimap e2 \multimap h$

9. $t : t$ Lex. **Thora**

Copy pronoun (resumptive) licenser (seem$_{like}$ @CR):
Result of dependency relabelling

(38)   Alfred seems like he hurt Thora.

seems (CR: MR)
$[a \multimap (a \otimes p)] \multimap (a \multimap a)$ **he**
$a \multimap (a \otimes p)$ **Alfred**
a

$(a \multimap a)$
a

seems (CR)
$a \multimap (a \multimap s) \multimap s$

$(a \multimap s) \multimap s$

hurt   **Thora**
$t \multimap p \multimap e2 \multimap h$ t $[p]^1$

$p \multimap e2 \multimap h$
$e2 \multimap h$ **like**
$(e2 \multimap h) \multimap e1 \multimap l$

$e1 \multimap l$ **seems**
$(e1 \multimap l) \multimap (e1 \multimap s)$

$e1 \multimap s$
s $\exists\varepsilon$
$p \multimap s$ $\multimap_{\mathcal{I},1}$

seems (CR: RELABEL)
$(p \multimap s) \multimap (a \multimap s)$

$\lfloor a \multimap s \rfloor$

$\exists s.seem(s, \exists e.[(s \sim e) \land hurt(e, a, t)]) : s$
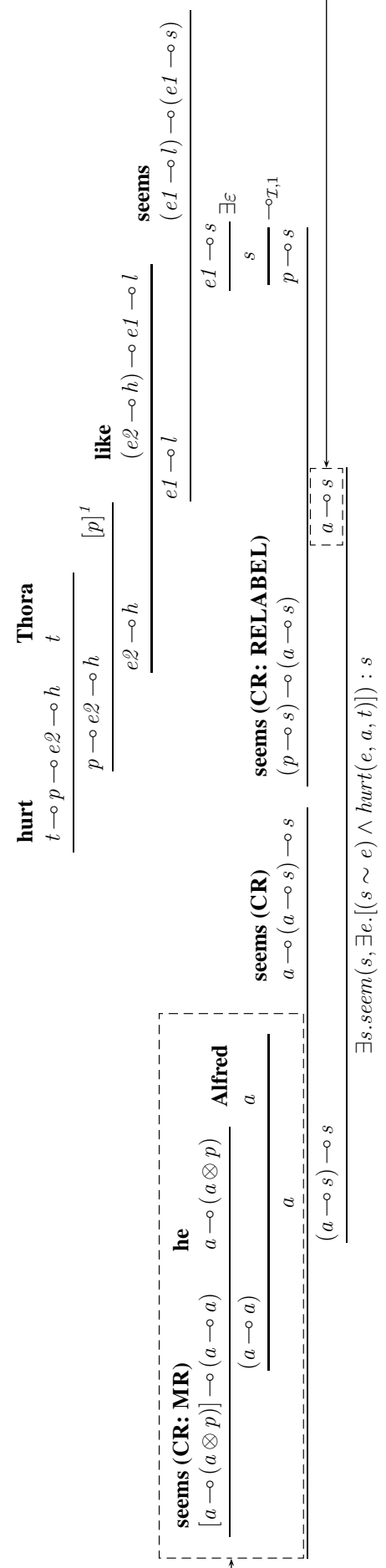
Figure 5: Proof for an English copy raising example

Copy pronoun (resumptive) licenser (seem$_{like}$ @CR):
Manager resource (top left) consumes pronoun (top right), leaving antecedent resource

(38) Alfred seems like he hurt Thora.

hurt
$\lambda y \lambda x \lambda e'.hurt(e', x, y) :$
$t \multimap p \multimap e2 \multimap h$

Thora
$t :$
t

$[z : p]^1$

$\lambda x \lambda e'.hurt(e', x, t) :$
$p \multimap e2 \multimap h$

$\lambda e'.hurt(e', z, t) :$
$e2 \multimap h$

like
$\lambda P \lambda s'.\exists \varepsilon.[(s' \sim \varepsilon) \land P(\varepsilon)] :$
$(e2 \multimap h) \multimap e1 \multimap l$

$\lambda s'.\exists e.[(s' \sim e) \land hurt(e, z, t)] :$ [e/ε]
$e1 \multimap l$

seems
$\lambda P \lambda s.seem(s, P(s)) :$
$(e1 \multimap l) \multimap (e1 \multimap s)$

$\lambda s.seem(s, \exists e.[(s \sim e) \land hurt(e, z, t)]) :$
$e1 \multimap s$

$\exists s[seem(s, \exists e.[(s \sim e) \land hurt(e, z, t)])] :$ ∃ε
s

$\lambda z.\exists s[seem(s, \exists e.[(s \sim e) \land hurt(e, z, t)])] :$ $\multimap_{I,1}$
$p \multimap s$

seems (CR)
$\lambda x \lambda P.P(x) :$
$a \multimap (a \multimap s) \multimap s$

seems (CR: RELABEL)
$\lambda P.P :$
$(p \multimap s) \multimap (a \multimap s)$

$\lambda z.\exists s[seem(s, \exists e.[(s \sim e) \land hurt(e, z, t)])] :$
$a \multimap s$

seems (CR: MR)
$\lambda P \lambda y.y :$
$[a \multimap (a \otimes p)] \multimap (a \multimap a)$

he
$\lambda z.z \times z :$
$a \multimap (a \otimes p)$

$\lambda y.y :$
$(a \multimap a)$

Alfred
$a : a$

$a : a$

$\lambda P.P(a) :$
$(a \multimap s) \multimap s$

$\exists s.seem(s, \exists e.[(s \sim e) \land hurt(e, a, t)]) : s$

Figure 6: Proof for an English copy raising example, with meaning language

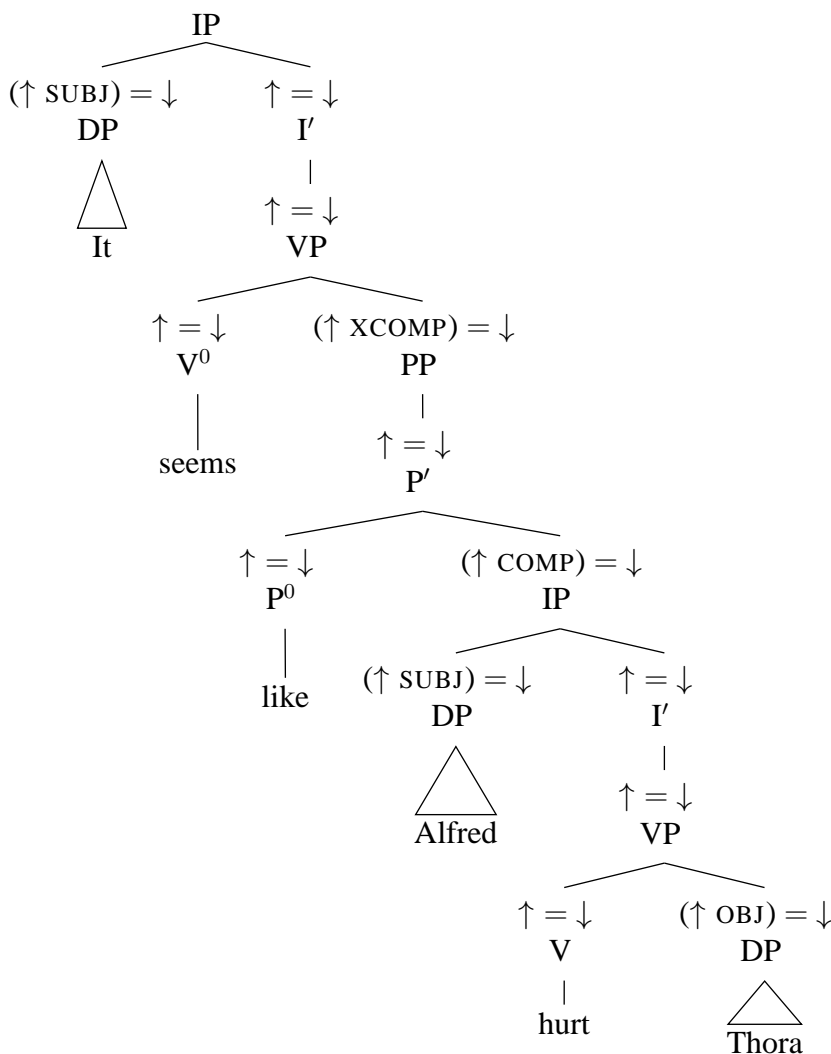- Next we turn to an expletive alternant, as in (52):

  (52)   It seems like Alfred hurt Thora.

- If we take seriously the claim that the expletive variant, (52), has the same interpretation as the non-expletive variant (38).

- This is typically considered a key property of copy raising, then the target interpretation is the same as (40), repeated here:
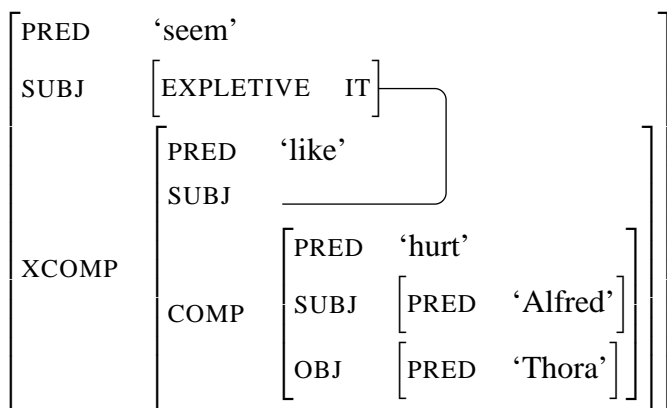
  (53) $\exists s.seem(s, \exists e.[(s \sim e) \wedge hurt(e, a, t)])$

- The only real distinction here is that the denotation of *Alfred*, a, composes directly with the denotation of *hurt*, rather than being threaded through a manager resource and copy pronoun, as in the copy raising alternant.

- The key to the expletive interpretation is the optionality of the copy raising template, CR, in the lexical entry for the copy raising verb, (46).

  - In the expletive variant, this template cannot be satisfied, since the manager resource cannot be satisfied, because the expletive does not contribute a resource for the manager resource to consume.

  - The lexically contributed meaning constructors for (52) are therefore the same as in (51), with the exception of the three premises that are contributed by the CR template and the premise contributed by the pronoun *he*.

- The c-structure and f-structure for (52) are shown in (54) and (55). The lexically contributed meaning constructors are shown in (56). Figure 7 shows the proof constructed from these premises. Figure 8 also shows the meaning language side of the meaning constructors. The result of the proof is the target interpretation in (53): $\exists s.seem(s, \exists e.[(s \sim e) \wedge hurt(e, a, t)])$.

(54)

(55)

$$\begin{bmatrix} \text{PRED} & \text{`seem'} \\ \text{SUBJ} & \begin{bmatrix} \text{EXPLETIVE} & \text{IT} \end{bmatrix} \\ \text{XCOMP} & \begin{bmatrix} \text{PRED} & \text{`like'} \\ \text{SUBJ} & \\ \text{COMP} & \begin{bmatrix} \text{PRED} & \text{`hurt'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`Alfred'} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{`Thora'} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

(56)   1. $(e1 \multimap l) \multimap (e1 \multimap s)$ Lex. **seems**
         2. $(e2 \multimap h) \multimap e1 \multimap l$ Lex. **like**
         3. a Lex. **Alfred**
         4. $t \multimap a \multimap e2 \multimap h$ Lex. **hurt**
         5. t Lex. **Thora**

$$\cfrac{\cfrac{\cfrac{\text{\textbf{hurt}}}{t \multimap a \multimap e2 \multimap h} \quad \cfrac{\text{\textbf{Thora}}}{t}}{a \multimap e2 \multimap h} \quad \cfrac{\text{\textbf{Alfred}}}{a}}{\cfrac{\cfrac{e2 \multimap h \quad \cfrac{\text{\textbf{like}}}{(e2 \multimap h) \multimap e1 \multimap l}}{e1 \multimap l} \quad \cfrac{\text{\textbf{seems}}}{(e1 \multimap l) \multimap (e1 \multimap s)}}{\cfrac{e1 \multimap s}{\exists s.seem(s, \exists e.[(s \sim e) \wedge hurt(e, a, t)]) : s} \; \exists \varepsilon}}$$

Figure 7: Proof for an expletive alternant example of English copy raising

$$\cfrac{\cfrac{\cfrac{\begin{array}{c}\text{\textbf{hurt}} \\ \lambda y \lambda x \lambda e'.hurt(e', x, y) : \\ t \multimap a \multimap e2 \multimap h\end{array} \quad \begin{array}{c}\text{\textbf{Thora}} \\ t : \\ t\end{array}}{\begin{array}{c}\lambda x \lambda e'.hurt(e', x, t) : \\ t \multimap a \multimap e2 \multimap h \quad a \multimap e2 \multimap h\end{array} \quad \begin{array}{c}\text{\textbf{Alfred}} \\ a : \\ a\end{array}}{\begin{array}{c}\lambda e'.hurt(e', a, t) : \\ e2 \multimap h\end{array} \quad \cfrac{\begin{array}{c}\text{\textbf{like}} \\ \lambda P \lambda s'.\exists \varepsilon.[(s' \sim \varepsilon) \wedge P(\varepsilon)] : \\ (e2 \multimap h) \multimap e1 \multimap l\end{array}}{}}{\cfrac{\cfrac{\begin{array}{c}\lambda s'.\exists e.[(s' \sim e) \wedge hurt(e, a, t)] : \\ e1 \multimap l\end{array} \; {\scriptstyle [e/\varepsilon]} \quad \begin{array}{c}\text{\textbf{seems}} \\ \lambda P \lambda s.seem(s, P(s)) : \\ (e1 \multimap l) \multimap (e1 \multimap s)\end{array}}{\begin{array}{c}\lambda s.seem(s, \exists e.[(s \sim e) \wedge hurt(e, a, t)]) : \\ e1 \multimap s\end{array}}}{\exists s.seem(s, \exists e.[(s \sim e) \wedge hurt(e, a, t)]) : s} \; \exists \varepsilon}$$

Figure 8: Proof for an expletive alternant example of English copy raising, with meaning language

8 A Formal Lexicalist Account of Copy Raising Variation

8.1 Dialect B

- Dialect B is like Dialect C in having copy raising, except that the copy pronoun must be the subject of the complement of *like/as*:

  (57) **Dialect B: Restricted Copy Raising**

      a. Alfred seems like he hurt Thora.

      b. *Alfred seems like Thora hurt him.

- The distinction between the two dialects is captured by restricting the argument of the copy raising template in the lexical entry of the copy raising verb as follows:

  (58)   *seem$_{like}$*: V   ⋮

                       (@CR(↑ XCOMP COMP SUBJ))

                              ⋮

- The path is restricted such that the copy pronoun must be the SUBJ of the COMP of the copy raising verb's XCOMP. This is the highest overt subject, the subject of the complement of *like/as*.

- As indicated by the ellipses, the lexical entry is otherwise the same as the lexical entry for Dialect C in (46).

- The CR template is again optional, which correctly allows Dialect B grammars to generate expletive-subject copy raising alternants, like *It seems like Alfred hurt Thora*, in the same way as in Dialect C.

8.2 Dialect A

- Dialect A is even more strict than Dialect B. It does not have copy raising:

  (59) **Dialect A: No Copy Raising**

      a. *Alfred seems like he hurt Thora.

      b. *Alfred seems like Thora hurt him.

- This dialect is accounted for by deleting the CR template from the lexical entry for the copy raising verb.

- The lexical entries for Dialect A's copy raising verbs are just like those for Dialect C and Dialect B, except that there is no CR line.

- This correctly allows Dialect A grammars to generate expletive-subject copy raising alternants in the same way as in Dialect C, since these are generated when the optional CR template is absent.

8.3   Dialect D

- Dialect D was the most permissive dialect, not requiring a copy pronoun at all with a copy raising verb:

  (60)   **Dialect D: No Copy Pronoun Required**
      a.   Alfred seems like Harry's hurt.
      b.   Alfred seems like Isak hurt Thora.

- This dialect arguably has a modified interpretation for the copy raising subcategorization, such that the copy raising verb is in fact a control verb.

- Like Dialect A, this dialect lacks the CR template entirely. But unlike Dialects A—C, the putative copy raising verb is instead ambiguous, such that one reading is associated with a thematic subject, while the other reading is associated with the same raising meaning constructor as in (46) above.

- This ambiguity is somewhat unappealing, but is required to allow for the expletive alternation.

- The meaning constructors are numbered in (61), for ease of subsequent reference.

  (61) $seem_{like}$:   V $(\uparrow \text{PRED}) = $ 'seem'
                            @RAISING

   $\{\ \lambda x \lambda P \lambda s.seem(s, x, P(s)) :$ ①
     $(\uparrow \text{SUBJ})_\sigma \multimap [(\uparrow_\sigma \text{EVENT}) \multimap (\uparrow \text{XCOMP})_\sigma] \multimap$
     $(\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$

   $\lambda x \lambda P.P(x) :$ ②
     $(\uparrow \text{SUBJ})_\sigma \multimap [(\uparrow \text{SUBJ}) \multimap \uparrow_\sigma] \multimap \uparrow_\sigma$ |

   $\lambda P \lambda s.seem(s, P(s)) :$ ③
     $[(\uparrow_\sigma \text{EVENT}) \multimap (\uparrow \text{XCOMP})_\sigma] \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma\ \}$

- The first and second meaning constructors constitute one option for interpretation of the 'copy raising' verb in Dialect D, while the third meaning constructor constitutes the other.

  – The third meaning constructor in (61) is a raising meaning constructor, just as in (46).

  – The second meaning constructor is the meaning constructor associated with the CR template in the other dialects.

  – The first meaning constructor is the same as the meaning constructor for copy raising in the other dialects, except that the verb takes the matrix subject as an argument.

  – Nevertheless, the verb invokes the RAISING template; i.e. it has the syntax of a raising verb, even though the matrix subject is an argument.

  – This means that the relationship between the copy raising verb and its subject, in Dialect D, is like that of a subject control verb with that of its subject, rather than like that of a raising verb.

  – This is, in effect, predicted by the standard LFG theory of functional control (Bresnan 1982), which posits the same syntactic relation of equality for raising and obligatory control, and LFG's grammatical architecture, which allows mismatches between levels of grammar (Kaplan and Bresnan 1982, Kaplan 1987, 1989, Asudeh 2006, 2012).

- Figure 9 shows a proof with meaning language for the Dialect D example (60b). Dialect D expletive examples are analyzed exactly like the expletive examples in the other dialects, as previously illustrated in Figures 7 and 8.

(60b)   Alfred seems like Isak hurt Thora.

hurt
$\lambda y \lambda x \lambda e'.hurt(e', x, y)$:
$t \multimap i \multimap e2 \multimap h$

Thora
t :
t

Isak
i :
i

$\lambda x \lambda e'.hurt(e', x, t)$:
$i \multimap e2 \multimap h$

$\lambda e'.hurt(e', i, t)$:
$e2 \multimap h$

like
$\lambda P \lambda s'.\exists \varepsilon.[(s' \sim \varepsilon) \wedge P(\varepsilon)]$:
$(e2 \multimap h) \multimap e1 \multimap l$

$\lambda s'.\exists e.[(s' \sim e) \wedge hurt(e, i, t)]$:
$e1 \multimap l$ $[e/\varepsilon]$

seems (Option 1)
$\lambda x \lambda P \lambda s.seem(s, x, P(s))$:
$a \multimap (e1 \multimap l) \multimap (e1 \multimap s)$ $[z : a]^1$

$\lambda P \lambda s.seem(s, z, P(s))$:
$(e1 \multimap l) \multimap (e1 \multimap s)$

$\lambda s.seem(s, z, \exists e.[(s \sim e) \wedge hurt(e, i, t)])$:
$e1 \multimap s$

$\exists s[seem(s, z, \exists e.[(s \sim e) \wedge hurt(e, i, t)])]$:
s $\exists \varepsilon$

$\lambda z.\exists s[seem(s, z, \exists e.[(s \sim e) \wedge hurt(e, i, t)])]$:
$a \multimap s$ $\multimap_{\mathcal{I},1}$

Alfred
$a : a$

seems (Option 1)
$\lambda x \lambda P.P(x)$:
$a \multimap (a \multimap s) \multimap s$

$\lambda P.P(a)$:
$(a \multimap s) \multimap s$

$\exists s.seem(s, a, \exists e.[(s \sim e) \wedge hurt(e, i, t)])$: s

Figure 9: Proof for English Dialect D 'copy raising', with meaning language
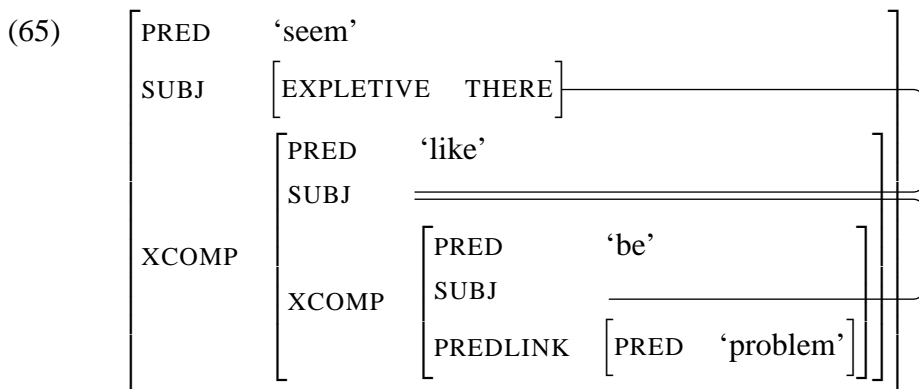
8.4   Expletives

- The last aspect of dialectal variation to be captured concerns the nature of expletive subjects, which seems to cross-cut the four dialects. All speakers accept sentences like (62), and no speakers accept sentences like (63), but some speakers also accept sentences like (64):

  (62)   It seemed like there was a problem.

  (63)   *There seemed like it rained.

  (64)   %There seemed like there was a problem.

- The existing lexical entries given so far generate (62) and correctly fail to generate (63), due to the RAISING template.

- As discussed above, the contrast between sentences like (63) and (64) indicates that the matrix expletive *there* is licensed only if there is an embedded expletive *there* in the *like*-complement.

- However, the f-structure projected by *like/as* itself has a SUBJ.

- In order to preserve the locality of raising, the lower *there* must be equated with this intervening subject position and the intervening subject position must then be equated with the matrix subject position.

- This is shown in the following abbreviated f-structure for sentence (64).

  (65)
  $$
  \begin{bmatrix}
  \text{PRED} & \text{`seem'} \\
  \text{SUBJ} & \begin{bmatrix} \text{EXPLETIVE} & \text{THERE} \end{bmatrix} \\
  \text{XCOMP} & \begin{bmatrix}
  \text{PRED} & \text{`like'} \\
  \text{SUBJ} & \\
  \text{XCOMP} & \begin{bmatrix}
  \text{PRED} & \text{`be'} \\
  \text{SUBJ} & \\
  \text{PREDLINK} & \begin{bmatrix} \text{PRED} & \text{`problem'} \end{bmatrix}
  \end{bmatrix}
  \end{bmatrix}
  \end{bmatrix}
  $$

- In sum, in order for the locality of raising to be preserved, *like* and *as* must have raising alternants.

- In dialects that allow (64), the lexical entry for *like/as* thus optionally calls the RAISING template:

  (66)   *like*: P^0 $(\uparrow \text{PRED}) = $ `like'

  (@RAISING)

  $\lambda P \lambda s. \exists \varepsilon. [(s \sim \varepsilon) \wedge P(\varepsilon)] :$
  $[(\uparrow \text{COMP}_\sigma \text{ EVENT}) \multimap (\uparrow \text{COMP})_\sigma] \multimap$
  $((\text{XCOMP} \uparrow)_\sigma \text{ EVENT}) \multimap \uparrow_\sigma$

- The RAISING is repeated here:

  (67)   RAISING = { $(\uparrow \text{SUBJ EXPLETIVE}) =_c \text{IT} \wedge \neg (\uparrow \text{XCOMP})$ |
  $(\uparrow \text{SUBJ}) = (\uparrow \text{XCOMP SUBJ})$ }

- The optional RAISING template can only be selected if both the subject of the copy raising verb and the subject of *like* are expletives.

9 Conclusion

- I have provided an analysis of copy raising that treats it as a kind of resumption.

- The unification of copy raising with resumption in unbounded dependencies is based on the Resource Management Theory of Resumption, which is in turn based on the Resource Sensitivity Hypothesis.

- Within the context of this analysis of copy raising, I have shown how we can take variation in grammars of English copy raising seriously.

- We can account for the variation through differences in lexical entries, but where the formalization of these featural differences is well-grounded.

- The flexible theory of semantic composition provided by Glue Semantics is an important part of this lexical factorization, since it allows us to split the relevant bits of meaning up such that they can be associated with different lexical entries as needed.

- An obvious direction for future work is to investigate whether there are limits on this kind of formal variation, but in order to investigate such limits we first need to formalize variation properly, beyond general appeals to 'features' or 'parameters'.

References

Asudeh, Ash. 2002. Richard III. In Mary Andronis, Erin Debenport, Anne Pycha, and Keiko Yoshimura, eds., *CLS 38: The Main Session*, vol. 1, 31–46. Chicago, IL: Chicago Linguistic Society.

—. 2004. Resumption as Resource Management. Ph.D. thesis, Stanford University.

—. 2005a. Control and Semantic Resource Sensitivity. *Journal of Linguistics* 41(3): 465–511.

—. 2005b. Relational Nouns, Pronouns, and Resumption. *Linguistics and Philosophy* 28(4): 375–446.

—. 2006. Direct Compositionality and the Architecture of LFG. In Miriam Butt, Mary Dalrymple, and Tracy Holloway King, eds., *Intelligent Linguistic Architectures: Variations on Themes by Ronald M. Kaplan*, 363–387. Stanford, CA: CSLI Publications.

—. 2012. *The Logic of Pronominal Resumption*. Oxford: Oxford University Press. In press.

Asudeh, Ash, Mary Dalrymple, and Ida Toivonen. 2008. Constructions with Lexical Integrity: Templates as the Lexicon-Syntax Interface. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of the LFG08 Conference*, 68–88. Stanford, CA: CSLI Publications.

Asudeh, Ash, and Ida Toivonen. 2006. Expletives and the Syntax and Semantics of Copy Raising. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of the LFG06 Conference*, 13–29. Stanford, CA: CSLI Publications.

—. 2007. Copy Raising and Its Consequences for Perception Reports. In Annie Zaenen, Jane Simpson, Tracy Holloway King, Jane Grimshaw, Joan Maling, and Christopher Manning, eds., *Architectures, Rules, and Preferences: Variations on Themes by Joan W. Bresnan*, 49–67. Stanford, CA: CSLI Publications.

—. 2009. Lexical-Functional Grammar. In Heine and Narrog 2009, 425–458.

—. 2012. Copy Raising and Perception. *Natural Language and Linguistic Theory*. Forthcoming.

Bach, Emmon. 1981. On Time, Tense and Aspect: An Essay on English Metaphysics. In Peter Cole, ed., *Radical Pragmatics*, 62–81. New York: Academic Press.

Barbiers, Sjef, ed. 2008. *Microvariation in Syntactic Doubling*, vol. 36 of *Syntax and Semantics*. Bingley: Emerald Group Publishing.

Black, James R., and Virginia Montapanyane, eds. 1996. *Microparametric Syntax and Dialect Variation*. Amsterdam: John Benjamins.

Bresnan, Joan. 1982. Control and Complementation. *Linguistic Inquiry* 13(3): 343–434.

Butt, Miriam, and Tracy Holloway King, eds. 2004. *Proceedings of the LFG04 Conference*. Stanford, CA: CSLI Publications.

Carpenter, Bob. 1997. *Type-Logical Semantics*. Cambridge, MA: MIT Press.

Chomsky, Noam. 1981. *Lectures on Government and Binding*. Dordrecht: Foris.

—. 1995. *The Minimalist Program*. Cambridge, MA: MIT Press.

Cinque, Guglielmo, and Richard S. Kayne, eds. 2005. *The Oxford Handbook of Comparative Syntax*. Oxford: Oxford University Press.

Crouch, Richard, Mary Dalrymple, Ronald M. Kaplan, Tracy King, John T. Maxwell, III, and Paula Newman. 2011. *XLE Documentation*. Palo Alto Research Center (PARC), Palo Alto, CA.

Dalrymple, Mary, ed. 1999. *Semantics and Syntax in Lexical Functional Grammar: The Resource Logic Approach*. Cambridge, MA: MIT Press.

Dalrymple, Mary. 2001. *Lexical Functional Grammar*. San Diego, CA: Academic Press.

Dalrymple, Mary, Vaneet Gupta, John Lamping, and Vijay Saraswat. 1999a. Relating Resource-Based Semantics to Categorial Semantics. In Dalrymple 1999, 261–280.

Dalrymple, Mary, Ronald M. Kaplan, and Tracy Holloway King. 2004. Linguistic Generalizations over Descriptions. In Butt and King 2004, 199–208.

Dalrymple, Mary, Ronald M. Kaplan, John T. Maxwell III, and Annie Zaenen, eds. 1995. *Formal Issues in Lexical-Functional Grammar*. Stanford, CA: CSLI Publications.

Dalrymple, Mary, John Lamping, Fernando Pereira, and Vijay Saraswat. 1999b. Overview and Introduction. In Dalrymple 1999, 1–38.

—. 1999c. Quantification, Anaphora, and Intensionality. In Dalrymple 1999, 39–89.

Dalrymple, Mary, John Lamping, and Vijay Saraswat. 1993. LFG Semantics via Constraints. In *Proceedings of the Sixth Meeting of the European ACL*, 97–105. European Chapter of the Association for Computational Linguistics, University of Utrecht.

Girard, Jean-Yves. 1987. Linear Logic. *Theoretical Computer Science* 50(1): 1–102.

Gisborne, Nikolas. 1996. English Perception Verbs. Ph.D. thesis, University College London.

—. 2010. *The Event Structure of Perception Verbs*. Oxford: Oxford University Press.

Halvorsen, Per-Kristian, and Ronald M. Kaplan. 1988. Projections and Semantic Description in Lexical-Functional Grammar. In *Proceedings of the International Conference on Fifth Generation Computer Systems*, 1116–1122. Institute for New Generation Systems, Tokyo. Reprinted in Dalrymple et al. (1995: 279–292).

Heine, Bernd, and Heiko Narrog, eds. 2009. *The Oxford Handbook of Linguistic Analysis*. Oxford: Oxford University Press.

Heycock, Caroline. 1994. *Layers of Predication*. New York: Garland.

Horn, Laurence R. 1981. A Pragmatic Approach to Certain Ambiguities. *Linguistics and Philosophy* 4: 321–358.

Jacobson, Pauline. 1999. Towards a Variable-Free Semantics. *Linguistics and Philosophy* 22(2): 117–184.

Jäger, Gerhard. 2005. *Anaphora and Type Logical Grammar*. Dordrecht: Springer.

Kaplan, Ronald M. 1987. Three Seductions of Computational Psycholinguistics. In Peter Whitelock, Mary McGee Wood, Harold L. Somers, Rod Johnson, and Paul Bennett, eds., *Linguistic Theory and Computer Applications*, 149–181. London: Academic Press. Reprinted in Dalrymple et al. (1995: 339–367).

—. 1989. The Formal Architecture of Lexical-Functional Grammar. In Chu-Ren Huang and Keh-Jiann Chen, eds., *Proceedings of ROCLING II*, 3–18. Reprinted in Dalrymple et al. (1995: 7–27).

Kaplan, Ronald M., and Joan Bresnan. 1982. Lexical-Functional Grammar: A Formal System for Grammatical Representation. In Joan Bresnan, ed., *The Mental Representation of Grammatical Relations*, 173–281. Cambridge, MA: MIT Press. Reprinted in Dalrymple et al. (1995: 29–135).

Kokkonidis, Miltiadis. 2008. First-Order Glue. *Journal of Logic, Language and Information* 17(1): 43–68.

Landau, Idan. 2009a. Predication vs. Aboutness in Copy Raising. Ms., Ben Gurion University.

—. 2009b. This Construction Looks Like a Copy is Optional. *Linguistic Inquiry* 40(2): 343–346.

Lev, Iddo. 2007. Packed Computation of Exact Meaning Representations. Ph.D. thesis, Stanford University.

Matushansky, Ora. 2002. Tipping the Scales: The Syntax of Scalarity in the Complement of *Seem*. *Syntax* 5(3): 219–276.

McCloskey, James. 2002. Resumption, Successive Cyclicity, and the Locality of Operations. In Samuel David Epstein and T. Daniel Seeley, eds., *Derivation and Explanation in the Minimalist Program*, 184–226. Oxford: Blackwell.

Michaelis, Laura. 2010. Sign-Based Construction Grammar. In Heine and Narrog 2009, 139–158.

Moortgat, Michael. 1997. Categorial Type Logics. In Johan van Benthem and Alice ter Meulen, eds., *Handbook of Logic and Language*, 93–177. Cambridge, MA: MIT Press. Co-published with Elsevier Science B.V., Amsterdam.

Morrill, Glyn V. 1994. *Type Logical Grammar*. Dordrecht: Kluwer.

Pollard, Carl, and Ivan A. Sag. 1987. *Information-Based Syntax and Semantics*. CSLI Publications.

—. 1994. *Head-driven Phrase Structure Grammar*. Chicago, IL and Stanford, CA: The University of Chicago Press and CSLI Publications.

Postal, Paul M. 1974. *On Raising: One Rule of English Grammar and its Theoretical Implications*. Cambridge, MA: MIT Press.

Potsdam, Eric, and Jeffrey T. Runner. 2001. Richard Returns: Copy Raising and its Implications. In Mary Andronis, Chris Ball, Heidi Elston, and Sylvain Neuvel, eds., *CLS 37: The main session*, vol. 1, 453–468. Chicago, IL: Chicago Linguistic Society.

Rogers, Andy. 1973. Physical Perception Verbs in English: A Study in Lexical Relatedness. Ph.D. thesis, UCLA.

—. 1974. A Transderivational Constraint on Richard? In *Papers from the Tenth Regional Meeting of the Chicago Linguistic Society*, 551–558.

Sag, Ivan A. 2010. English Filler-Gap Constructions. *Language* 86(3): 486–545.

NORMS. 2005–2010. Nordic Center of Excellence in Microcomparative Syntax. http://norms.uit.no/.