

# Coordination in Minimalist Grammars: Excorporation and Across the Board (Head) Movement

**John Torr**

School of Informatics  
University of Edinburgh  
11 Crichton Street, Edinburgh, UK  
john.torr@cantab.net

**Edward P. Stabler**

Nuance Communications  
1198 E Arques Ave  
Sunnyvale, CA 94085  
stabler@ucla.edu

## Abstract

This paper describes how coordination has been integrated into a broad coverage statistical Minimalist Grammar parser currently under development, and presents a unified analysis for a number of coordinate (and related) constructions sometimes considered problematic for transformational syntax; these include across-the-board (ATB) head and phrasal movements, argument cluster coordination, right node raising and parasitic gaps. To accommodate all these structures, a number of novel extensions are introduced into the formalism, including a mechanism for *excorporation* which enables ATB head movement; this supplements a variant of Kobele’s (2008) mechanism for ATB phrasal movement. The weak expressive power of the formalism is shown to be unaffected by these extensions.

## 1 Introduction and Background

This paper documents the core mechanisms that have been implemented within *MGP*Parse, a broad coverage (Extended Directional) Minimalist Grammar (MG) parser currently being developed at the University of Edinburgh<sup>1</sup>. Minimalist Grammars (Stabler, 1997) are formally a kind of highly succinct and lexicalized Multiple Context Free Grammar (Seki et al. 1991), and constitute a mildly context sensitive interpretation of many aspects of Chomsky’s (1995) Minimalist Program. The mechanisms presented below enable *MGP*Parse to generate structures for a range of coordination (and related) phenomena sometimes considered problematic for movement-based approaches to syntax. For example, Gazdar et

<sup>1</sup>*MGP*Parse extends Harkema’s (2001) CKY variant.

al. (1985) state that “transformational grammar has never been able to capture a unitary notion of coordination, for reasons that were endemic to the framework.” Considered particularly troublesome are constructions which involve movement to a single position of two or more constituents which do not stand in a c-command<sup>2</sup> relation with one another, as shown in schematic form in fig 1.

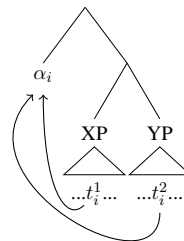


Figure 1: Across-the-board Movement Schema

Examples of constructions arguably involving this configuration are given below<sup>3</sup>:

1. I know who<sub>i</sub> [<sub>TP</sub> Jack likes t<sub>i</sub>] and [<sub>TP</sub> Mary hates t<sub>i</sub>]. (ATB Phrasal Movement)
2. Who<sub>i</sub> does<sub>j</sub> [<sub>TP</sub> Jack t<sub>j</sub> like t<sub>i</sub>] and [<sub>TP</sub> Mary t<sub>j</sub> hate t<sub>i</sub>]? (ATB Head and Phrasal Movement)
3. [<sub>TP</sub> [<sub>TP</sub> Jack likes t<sub>i</sub>] and [<sub>TP</sub> Mary hates t<sub>i</sub>], [Pete’s sister]<sub>i</sub>]. (Right Node Raising)
4. He [<sub>VP</sub> gave<sub>i</sub> [<sub>VP</sub> Pete t<sub>i</sub> a book] and [<sub>VP</sub> Mary t<sub>i</sub> a flower]]. (Argument Cluster Coordination)

One approach to ATB phenomena has been to introduce a mechanism of *sideward movement* (Nunes (1995), (2001), (2004)) into the grammar.

<sup>2</sup>A node c-commands its sister and its sister’s descendants.

<sup>3</sup>We adopt the Verb Phrase Internal Subject Hypothesis (Koopman and Sportiche, 1991), according to which AGENTS are generated in the verb phrase before moving to their surface subject position. We also adopt the Movement Theory of Control (Hornstein (2001)), which avoids the need for additional meaning postulates to derive indices on PRO, which is now simply treated as a trace of A-movement.

This operation moves elements cyclically between trees before those trees are merged together into a single structure. For example, in fig 1,  $\alpha$  could first move sideways from  $t^2$  to  $t^1$  prior to the merger of XP with YP, before undergoing standard (upward) movement to its final surface position.

Two further constructions that have been argued to involve sideward movements, and hence the configuration in fig 1, are adjunct control and parasitic gap structures, as in 5 and 6.

5. [ $TP$   $He_i$  [ $vP$  [ $vP$   $t_i$  filed the paper] [ $PP$  without [ $vP$   $t_i$  reading it]]]]. (Adjunct Control)

6. [Which paper] $_i$  did [ $TP$   $he_j$  [ $vP$  [ $vP$   $t_j$  file  $t_i$ ] [ $PP$  without [ $vP$   $t_j$  reading  $t_i$ ]]]]? (Parasitic Gap)

Under certain assumptions, both 5 and 6 feature movement out of an adjunct in apparent violation of Huang's (1982) *Condition on Extraction Domains* (CED). However, if these movements occur prior to adjunction<sup>4</sup> taking place, i.e. before the adjunct PP actually becomes an adjunct, then arguably CED is never violated. Stabler (2006) shows how sideward movement can be incorporated into MGs to accommodate just such an analysis of adjunct control. Unfortunately, this formulation of sideward movement is severely restricted to moving just a single element as an integral part of adjunction. As a result, it cannot accommodate example 6 which involves *two* elements moving out of the adjunct.

Kobele (2008) introduces an approach to leftward ATB phrasal movement for MGs which can accommodate these cases by 'unifying' any identical movers inside the dependent and main clause structures. However, Kobele does not extend his analysis to examples arguably involving ATB head movement (2 and 4) or Right Node Raising (RNR) (3). Moreover, as things stand, this system also appears to overgenerate 7 below, which features illicit ATB leftward phrasal movement from two different structural case positions.

7. \*I know  $who_i$  [ $TP$  Jack likes  $t_i$ ] and [ $TP$   $t_i$  hates Mary].

To accommodate such examples, we will augment Kobele's system with mechanisms for rightward movement, case valuation and excorpora-

<sup>4</sup>Here and throughout, *adjunction* refers to the merger of an adjunct dependent with a main structure and should not be confused with the operation by that name used in Tree Adjoining Grammar.

tion<sup>5</sup>. Excorporation is argued to exist for example in Roberts (2010) in the context of a discussion of Romance clitics such as the French object pronoun 'l' in *je l'ai vu* (*I have seen him/her*). Clitics are interesting because they behave like heads in being affixal and adjoining to other heads, but they are also capable of moving over much greater distances than typical heads and in this sense behave more like phrases. Here, for instance, the clitic has moved from the object position, past its governing verb, and adjoined to the auxiliary. This, Roberts argues, is achieved via excorporation, which here we will extend to cases of ATB head movement.

The rest of the paper is arranged as follows: section 2 introduces the MG formalism together with some extensions; next, section 3 provides a general framework for coordination; section 4 then presents the analysis of ATB phenomena; finally, section 5 concludes the paper.

## 2 Minimalist Grammars

### 2.1 Introduction to MGs

Minimalist Grammars (Stabler, 1997) are a derivational, lexicalized and feature-driven formalism. Structures are built bottom up using the two operations of binary merge and unary move, which each check and delete features on lexical items and derived constituents, while reordering and concatenating strings. Each partially built structure is represented in the collapsed tree format of Stabler (2001a) and Harkema (2001), in which the actual geometry of derived phrase structure is largely discarded<sup>6</sup> and only the strings/spans and features of the head and any moving elements are retained. For instance, consider the collapsed tree representation for the vP *Jack helped who* in which *who* and *Jack* will later move to check -wh and -case respectively:

[[[cause] *help* : +case v, *who* : -wh, *Jack* : -case]

Each collapsed tree, or *expression*, is composed of between 1 and  $k + 1$  *chains*, where  $k$  is the size

<sup>5</sup>Two further important differences between Kobele's framework and ours are: 1. We do not adopt a GPSG-style slash-feature mechanism; and 2. We do not handle control into complements via ATB movement (this is reserved for control into adjuncts); instead, we allow selectee x features (see section 2.2) to persist as licensees after initial selection and to check further selector =x/x= features (now also control licensors) via standard movement; for complement control, the base position of the controller therefore always commands the base position of the controllee in our system.

<sup>6</sup>The full derived phrase structure tree, with indices, is deterministically recoverable from the derivation tree.

of the set of licensee features<sup>7</sup>. The first chain of the expression (here, [cause] *help* : +case v) is the head of the expression, while any other chains are movers (their ordering being irrelevant). Each chain is in turn composed of a string and a feature sequence. Moving chains are kept separate only until their syntactic features have all been checked and deleted, at which time their strings are concatenated with the head chain’s string and they cease to exist. Importantly, the collapsed tree representation entails that MGs as defined above are type-driven as opposed to structure-driven: all of the information that is input to the rules is contained within the category labels themselves. As well as affording MGs some important computational advantages (the set of MG derivation trees is a regular set), this fact will be crucial in the account of coordination that follows.

## 2.2 A simple Directional MG

A Directional Minimalist Grammar (DMG)<sup>8</sup> is defined as a quadruple  $(V, Cat, Lex, F)$  s.t.<sup>9</sup>:

1.  $V = P \cup I$  is a finite set of non-syntactic features ( $P =$  phonetic features,  $I =$  semantic features).
2.  $Cat = selectees \cup selectors \cup licensees \cup licensors$  is a finite set of syntactic features, s.t. for each feature  $x \in selectees$  there are features  $(=x, x=) \in selectors$ , and for each feature  $-y \in licensees$  there is a feature  $+y \in licensors$ .
3.  $Lex$  is a finite set of axioms (lexical items) over  $V \cup Cat$ , with the  $Cat$  features on each simplex tree strictly ordered from left to right.
4.  $F$  is a set consisting of the structure building functions *MERGE* and *MOVE* (the deductive rules of inference), defined as the union of their respective sub-functions, given in figures

<sup>7</sup>Licensee features are located on moving elements. In LCFRS terminology,  $k + 1$  is the *fan-out* of the grammar.

<sup>8</sup>Directional MGs are MGs in which the directionality of selection is determined by a feature on the selecting head, rather than by the complement vs. adjunct/specifier status of the dependent. Following Ernst (2002), and contra Kayne’s (1994) LCA, we therefore allow both leftward and rightward complements and adjuncts, as well as rightward movement to adjoined positions. There is good evidence for the non-existence of rightward specifiers, however (perhaps for processing reasons), such as the lack of reverse V2 languages and the extreme marginality of OS languages; these are therefore currently disallowed by MGParse. Note that Directional MGs are similar to Categorical Grammars up to movement.

<sup>9</sup>In order to unify the notation for merge and move, we adopt the convention that all diacritics appear on the side of the Part of Speech (PoS) symbol on which selection occurs; hence  $x=$  indicates rightward selection,  $=x$  leftward selection.

2 and 3, where expressions are contained within square brackets, chains are separated by commas,  $\alpha_1, \dots, \alpha_k$  is a (possibly empty) set of moving chains,  $\delta$  and  $\gamma$  are feature sequence suffix variables, with  $|\delta| \geq 1$  and  $|\gamma| \geq 0$ ,  $s$  and  $t$  are string variables, and string/feature separators indicate whether a chain represents an unmerged lexical head ( $::$ ) or a derived element ( $:$ ), or can be either ( $:$ ).

$$\frac{[s::x=\gamma] \quad [t;x, \alpha_1, \dots, \alpha_k]}{[st:\gamma, \alpha_1, \dots, \alpha_k]}(merge1(comp))$$

$$\frac{[t;x, \alpha_1, \dots, \alpha_k] \quad [s::=x\gamma]}{[ts:\gamma, \alpha_1, \dots, \alpha_k]}(merge2(comp))$$

$$\frac{[t;x] \quad [s:=x\gamma, \alpha_1, \dots, \alpha_k]}{[ts:\gamma, \alpha_1, \dots, \alpha_k]}(merge3(spec))$$

$$\frac{[s::x=\gamma] \quad [t;x\delta, \alpha_1, \dots, \alpha_k]}{[s:\gamma, t:\delta, \alpha_1, \dots, \alpha_k]}(merge4(comp))$$

$$\frac{[t;x\delta, \alpha_1, \dots, \alpha_k] \quad [s::=x\gamma]}{[s:\gamma, t:\delta, \alpha_1, \dots, \alpha_k]}(merge5(comp))$$

$$\frac{[t;x\delta] \quad [s:=x\gamma, \alpha_1, \dots, \alpha_k]}{[s:\gamma, t:\delta, \alpha_1, \dots, \alpha_k]}(merge6(spec))$$

Figure 2: Sub-functions of MERGE

$$\frac{[s:+f\gamma, \alpha_1, \dots, \alpha_{i-1}, t:-f, \alpha_{i+1}, \dots, \alpha_k]}{[s:\gamma, \alpha_1, \dots, \alpha_{i-1}, \alpha_{i+1}, \dots, \alpha_k]}(move1)$$

$$\frac{[s:+f\gamma, \alpha_1, \dots, \alpha_{i-1}, t:-f\delta, \alpha_{i+1}, \dots, \alpha_k]}{[s:\gamma, \alpha_1, \dots, \alpha_{i-1}, t:\delta, \alpha_{i+1}, \dots, \alpha_k]}(move2)$$

Figure 3: Sub-functions of MOVE

For a given Minimalist Grammar  $G = Lex$ , the language  $L(G)$  is the closure of  $Lex$  under the structure building functions  $\{MERGE, MOVE\}$  in accordance with the Shortest Move Constraint:

**The Shortest Move Constraint (SMC):** no expression may contain two chains with precisely the same initial feature.

Notice that only where a selector is a head word/morpheme (and so consequently contains no movers) can its selectee contain movers. This encodes the specifier part of CED, according to which only complements ( $\approx$ objects) allow extraction of their contents, not specifiers ( $\approx$ subjects) (or adjuncts<sup>10</sup>). A complement is formally defined as the first argument dependent to be merged

<sup>10</sup>Adjuncts are discussed in section 2.3.1.

$[who :: d -wh]$	$[[pres] :: v = +case t]$
$[Jack :: d -case]$	$[[int] :: t = +wh c]$
$[likes :: d = =d v]$	

Table 1: A DMG lexicon

with a head; all subsequently merged arguments are specifiers. With the simple lexicon in table 1, this grammar will generate the derivation tree in fig 4 for the embedded clause *who Jack likes*<sup>11</sup>.

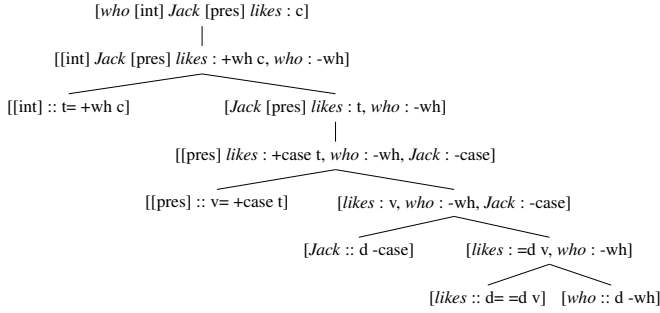


Figure 4: DMG Derivation tree for *who Jack likes*

## 2.3 Extending a Directional MG

MGParse incorporates a number of extensions to the simple DMG presented above which are discussed briefly below. We will refer to a DMG which includes these and the mechanisms to be introduced in sections 3 and 4 as an Extended Directional Minimalist Grammar (EDMG).

### 2.3.1 Adjunction

Linguists have proposed that in addition to complements and specifiers, a third type of *adjunct* dependent can be distinguished. Adjuncts are usually (though not exclusively) semantically adverbial, and include adverbs themselves as well as modificational PPs and (at least some) relative clauses. They also display the following properties: they are usually optional, iterative, type-preserving and opaque to extraction (CED). MGParse incorporates the approach to leftward adjunction of Frey and Gärtner (F&G) (2002) and extends it with rightward adjunction. F&G introduce a new category into the lexicon, which we will refer to here as an *adjunctizer*. Adjunctizers select other categories which are then effectively transformed into adjuncts; they can be null or overt. Below are the categories for two adjunct-

<sup>11</sup>Head strings enclosed in square brackets indicate silent morphemes; t = *tense*; c = *complementizer*.

izers which map clauses (CPs) and PPs respectively into rightward adjuncts that adjoin to VPs.

$$[because :: c = \approx v] \quad [[adjunctizer] :: p = \approx v]$$

We must also add the ADJOIN rules in fig 5 to the grammar to accommodate the new  $\approx x/x \approx$  adjunction selector features. Notice that these rules involve *asymmetric* checking in contrast to the earlier MERGE rules: only the selector feature, not the selectee feature, is deleted; this captures the optional, iterative and type-preserving properties of adjunction. Note also that this time the head features of the mother derive from the selectee not the selector, making the selectee the head. Consequently, only the selectee may contain movers, thereby observing CED.

$$\frac{[t ; x \approx] \quad [s ; x \gamma, \alpha_1, \dots, \alpha_k]}{[ts : x \gamma, \alpha_1, \dots, \alpha_k]} (adjoin1)$$

$$\frac{[s ; x \gamma, \alpha_1, \dots, \alpha_k] \quad [t ; \approx x]}{[st : x \gamma, \alpha_1, \dots, \alpha_k]} (adjoin2)$$

$$\frac{[t ; x \approx \delta] \quad [s ; x \gamma, \alpha_1, \dots, \alpha_k]}{[s : x \gamma, t : \delta, \alpha_1, \dots, \alpha_k]} (adjoin3)$$

$$\frac{[s ; x \gamma, \alpha_1, \dots, \alpha_k] \quad [t ; \approx x \delta]}{[s : x \gamma, t : \delta, \alpha_1, \dots, \alpha_k]} (adjoin4)$$

Figure 5: Sub-functions of ADJOIN

### 2.3.2 Rightward Movement

Rightward movement was at one time the standard tool in analyses of constructions such as heavy NP shift and extraposition, but the rise in popularity of Kayne's (1994) Linear Correspondence Axiom resulted in its almost total abandonment in Minimalism<sup>12</sup>. However, abandoning rightward movement often necessitates using additional silent heads and elaborate sequences of multiple (remnant) leftward movements for what is intuitively a single weight-theoretic requirement: that a heavy constituent appear sentence-finally. We therefore retain rightward movement (of phonetic features only) to adjoined positions. To do this we adapt F&G's (2002) approach to German leftward scrambling, which makes use of a  $\sim x$  scrambling licensee, and introduce the rightward movement licensee  $x \sim$ . Selectee x features will now serve a second purpose as licensors for rightward movement. The R\_MOVE rule is given in fig 6.

<sup>12</sup>Ernst 2002 is a notable exception.

$$\frac{[s:x \gamma, \alpha_1, \dots, \alpha_{i-1}, t:x\sim, \alpha_{i+1}, \dots, \alpha_k]}{[st:x \gamma, \alpha_1, \dots, \alpha_{i-1}, \alpha_{i+1}, \dots, \alpha_k]}$$

Figure 6: R\_MOVE

$x\sim$  features enter the derivation on null *extraposer* heads which map items into rightward-moving versions of themselves. For example, the category for an extraposer causing a DP to move rightward and adjoin to the closest dominating TP is:  $[[extraposer] :: d= +case d -case t\sim]$ <sup>13</sup>.

Note that we can obtain a *Right Roof Constraint*<sup>14</sup> (RRC) (Ross, 1967) preventing rightward movement from crossing (non-small, non-defective<sup>15</sup>) clause boundaries by assuming that  $x\sim$  features cannot persist, i.e. they always delete immediately upon being checked.

### 2.3.3 Head Movement

MGParse incorporates Stabler’s (2001b) head movement rules. Head movement is a highly local operation that causes the head of a selector’s complement to adjoin to the head of the selector as an integral part of MERGE, the vanilla case being subject-auxiliary inversion in English main clause questions. Stabler’s key insight is that the lexical head string of an expression must be kept separate from its left and right dependent strings until that expression has itself been merged/adjoined as a dependent, in case the head has to move. We therefore introduce new feature diacritics  $>$  and  $<$  to indicate head movement with adjunction onto either the left or the right of a governing head. Fig 7 gives the MERGE rules for rightward selection with leftward adjoining head movement.

$$\frac{[(e, s_h, e) :: >x= \gamma] \quad [(t_l, t_h, t_r); x, \alpha_1, \dots, \alpha_k]}{[(e, t_h s_h, t_l t_r) : \gamma, \alpha_1, \dots, \alpha_k]}(merge\_hm1)$$

$$\frac{[(e, s_h, e) :: >x= \gamma] \quad [(t_l, t_h, t_r); x \delta, \alpha_1, \dots, \alpha_k]}{[(e, t_h s_h, e) : \gamma, (t_l t_r) : \delta, \alpha_1, \dots, \alpha_k]}(merge\_hm2)$$

Figure 7: MERGE\_HM functions for rightward selection with leftward adjoining head movement

<sup>13</sup>Note the simulation of feature percolation without actual percolation here. This is desirable since Kobele (2005) shows that genuine feature percolation results in Type 0 MGs. This mechanism is also used by MGParse to achieve pied-piping, e.g., of *to* by *whom* in *to whom did you complain?*

<sup>14</sup>Whether the RRC actually exists has been contested (see e.g. Gazdar (1981)). However, for the purposes of practical parsing, this constraint significantly improves efficiency and hence is currently adopted by MGParse.

<sup>15</sup>Defective clauses are bare TPs lacking a CP layer (standardly assumed to exist in ECM and raising structures).

### 2.3.4 Covert Movement

MGParse includes covert movement rules to accommodate a range of phenomena including apparent long-distance agreement<sup>16</sup>, case checking of prepositional objects and Quantifier Raising in a strictly monotonic/type-driven system. We follow Stabler (1997) in treating covert movement as moving just syntactic (and semantic) features, not phonetic features (cf. Chomsky’s (1995) Move-F). The applicability of overt vs. covert movement is determined by the licenser feature:  $+f$  licenses covert movement and  $+F$  overt movement. Merge rules are added to the grammar splitting an expression into its syntactic and phonetic parts. For example, *merge4* and *merge5* have the corresponding *phonetic merge* rules in fig 8 which fuse the selectee’s string to the selector’s head chain but keep the selectee’s syntactic features separate.

$$\frac{[s::x= \gamma] \quad [t; x \delta, \alpha_1, \dots, \alpha_k]}{[st : \gamma, e : \delta, \alpha_1, \dots, \alpha_k]}(p\_merge1(comp))$$

$$\frac{[t; x \delta, \alpha_1, \dots, \alpha_k] \quad [s::=x \gamma]}{[ts : \gamma, e : \delta, \alpha_1, \dots, \alpha_k]}(p\_merge2(comp))$$

Figure 8: Two sub-functions of P\_MERGE

The moving chains now contain no phonetic material, and so their movements will not be visible in the string, but may have an impact on the semantics. Note that since it is licensers, not licensees, which determine whether movement is overt or covert, both MERGE and P\_MERGE options initially have to be pursued by the system.

## 3 Coordination

MGParse’s EDMG adopts the binary Xbar theoretic view of coordinate structures<sup>17</sup> proposed most recently by Zhang (2010), in which the coordinator (Coord) is the head, its complement is the rightmost conjunct and all leftward conjuncts are in (multiple) specifier positions. Zhang also assumes that Coord heads inherit the PoS category<sup>18</sup> of their (leftmost) conjuncts, which here we will simply precompile into the lexicon.

We take coordination to be a ‘recursive transitive closure over same types’ (Partee and Rooth,

<sup>16</sup>Note that Chomsky’s (2000) long-distance Agree operation is non-monotonic and structure-driven rather than type-driven, hence incompatible here.

<sup>17</sup>See Appendix C for a discussion of the problem posed by lexical  $X^0$  head coordination for Xbar theoretic accounts.

<sup>18</sup>Selectee  $x$  features indicate PoS category in MGs.

1983), where ‘type’ here refers to the cluster of syntactic features on the entire expression (though see Appendix B), not just the PoS/selectee feature of its head chain. The abstract feature sequence schema for all coordinators is:  $x = \bar{x} x$ <sup>19</sup>. This is similar to the Combinatory Categorical Grammar (CCG) (Steedman, 2000) approach to coordination, except that here it is not formally treated as involving adjunction, and full type uniformity is not enforced by the selector features alone, but instead falls out from the interaction of two constraints on rules: CED, and the Coordinate Structure Constraint (CSC) (Ross, 1967) (see Appendix B on the like-types constraint)<sup>20</sup>.

One problem for the analysis so far if we assume that all D elements carry -case is that this needs to be checked for all DP conjuncts. We cannot achieve this by adding covert +case features to the Coord head because this would require the use of potentially infinite sequences of the form  $(=d +case)^+$  to ensure that all specifier conjuncts can check case without triggering SMC. To solve this problem, we exploit a null prepositional dative head, independently used by MGParse to avoid SMC violations in *promise*-type subject control structures<sup>21</sup>. This head has the category  $[[dat] :: d = +case p]$ , which covertly<sup>22</sup> checks the -case feature of its DP complement; the resulting PP is then selected by a P-selecting Coord with PoS category D (i.e. with a d selectee feature).

Notice that this implies that coordinators are able to inherit the PoS category of their complement’s complement (D), rather than that of the

<sup>19</sup>The overline is a diacritic enabling  $=x$  selector features to optionally persist after checking to generate structures such as *Jack, Pete and Mary*. Note that examples with multiple conjuncts such as *Jack and Pete and Mary* are also generated: CoordP with D conjuncts is in reality just DP, hence it can be selected as the complement of a higher D-selecting Coord head.

<sup>20</sup>Unlike in CCG, selector features in MGs are never complex, i.e. we cannot define an abstract coordinator category equivalent to  $(X \setminus X) / X$  and then reify it as, e.g.,  $((S / NP) \setminus (S / NP)) / (S / NP)$  where both conjuncts are specified as being clauses containing object holes/traces; all we can specify is that both conjuncts are clauses (i.e. that they have a c selectee as their first feature), using the sequence:  $c = \bar{c} c$ . We can, however, ensure that a pair of MG expressions entering into a binary merge rule have identical sets of moving chains (see section 4.1 on ATB); this, together with CED and CSC as constraints on the form of MERGE rules, derives the like-types constraint on conjuncts (see Appendix B).

<sup>21</sup>This implements an analysis in Boeckx et al. (2010).

<sup>22</sup>All prepositions are assumed here to trigger *covert* movement of their objects to spec-P to check case, since overt movement would yield postpositions (this only really matters for overt prepositions of course).

complement (P) itself. Again, we simply pre-compile this into the lexicon, permitting coordinate schemas of the form  $[x^T = \bar{x}^T y]$  where the PoS category of the coordinator differs from that of its conjuncts. Since we do not formally treat coordination as adjunction, sacrificing this aspect of type-preservation becomes possible<sup>23</sup>. Interestingly, this move may not be entirely ad hoc: arguably, *Jack and me went home* is more natural than the prescriptively ‘correct’ *Jack and I went home*, as evidenced by the fact that *I and Jack went home* seems awkward, whereas *me and Jack went home* is informal but perfectly fine. This is explained if nominal conjuncts in English are in fact PPs with null dative case-checking P heads.

A further problem for the analysis of coordination so far is that there exist structures which do not appear to adhere to the like-types restriction on conjuncts, such as 8 below.

8. Jack is  $[_{VP}$  working] and  $[_{PP}$  in the garden].

9. \*Jack  $[_{VP}$  works] and  $[_{PP}$  in the garden].

As 9 indicates, coordination of a VP with a PP is generally not permitted, and yet in 8 it is allowed. It is in fact a general feature of the verb *be* that its complement can be a coordinate phrase with apparently unlike conjuncts. However, somewhat tellingly, only predicative categories can be coordinated following *be*. For instance, while *Jack is happy and in the garden* is fine, \**Jack is happily and in the garden* is ungrammatical because *happily* is adverbial rather than predicative. In fact, only VPs, PPs, AdjPs and DPs can be coordinated in this way. One approach pursued in the literature (e.g. Jacobson (1987)) is therefore to assume that the expressions entering into such coordinate structures are in fact of the same super Prd category. We implement this here by adding the null *predicativizers* in table 2 to our lexicon<sup>24</sup>.

$[[prd] :: d = +case =d prd]$	$[[prd] :: >v = =d prd]$
$[[prd] :: p = =d prd]$	$[[prd] :: adj = =d prd]$

Table 2: Predicativizers

These are essentially unary functions which map expressions of a given PoS category into expressions with the Prd PoS category<sup>25</sup>. We can

<sup>23</sup>We must, however, impose heavy restrictions in the lexicon to rule out many unwanted cases here.

<sup>24</sup>Observe that each  $[prd]$  element base generates the DP subject as its specifier.

<sup>25</sup>A similar approach incorporating null adverbializing

then simply coordinate the resulting PrdPs in the usual manner<sup>26</sup>.

## 4 ATB Head and Phrasal Movement

### 4.1 ATB Phrasal Movement

Consider deriving just the embedded clause from example 1, given as 10 below.

10.  $who_i$  [ $TP$  Jack likes  $t_i$ ] and [ $TP$  Mary hates  $t_i$ ].

In terms of the schema in fig 1, *who* corresponds to  $\alpha_i$  and the two TP conjuncts to XP and YP. Recall that our problem here is to derive the fact that the two traces have only one overt antecedent and yet neither c-commands the other. Adapting an approach in Kobele (2008), we can accomplish this as follows: first we construct each TP conjunct (cf. fig 4 up to the first unary branching node). This yields the following two expressions:

[*Jack*, [pres], *likes* : t, *who* : -wh]  
 [*Mary*, [pres], *hates* : t, *who* : -wh]

Next, we merge the right conjunct *Mary hates who* as the complement of the conjunction [*and* :: t=  $\bar{t}$  t], which after feature deletion yields:

[*and*, *Mary* [pres] *hates* :  $\bar{t}$  t, *who* : -wh]

This is where things become interesting. Notice that when the conjunction head merged with its complement, the mover inside the complement was transferred into the resulting expression. If this were to also happen when we merged the specifier, the result would be an SMC violation as we would now have two elements in the same tree whose first feature was -wh. Moreover, transferring a mover out of a specifier is in any case impossible with the rules as currently formulated in accordance with CED. To solve this, we will bleed both SMC and CED by allowing the system to simply drop<sup>27</sup> any mover inside any dependent if that mover's features exactly match those of a mover already inside the governing structure. Dropping the occurrence of *who* from the left conjunct and merging the latter into the main structure

[adv] heads, e.g. with category [p= adv], accommodates coordination of unlike modifiers in MGParse: *Jack works happily and with great speed*.

<sup>26</sup>A reviewer notes that an alternative analysis treats unlike coordination as ATB head movement, e.g. of *be* out of multiple remnant coordinated verb phrases which it heads. This is straightforwardly implementable using the mechanism for ATB head movement introduced in section 4.2.

<sup>27</sup>Alternatively, we can see this as the unification of two sets of feature sequences, with sets only admitting single instances of their members.

will then yield the following TP coordinate phrase, containing only one occurrence of *who*<sup>28</sup>:

[*Jack* [pres] *likes*, and, *Mary* [pres] *hates* : t, *who* : -wh]

We can now merge this expression with a null interrogative [[int] :: t= +WH c] head and move *who* to spec-CP in the usual manner. The updated MERGE rules for specifiers are shown in fig 9<sup>29</sup> (along with the updated version of *adjoin2*, *adj\_atb1*, which derives 5 and 6 - the derivation for 6 is given in Appendix A) where the string ( $\alpha^s$ ) and syntactic ( $\alpha^f$ ) parts of the  $\alpha$  chains have been separated and identity is enforced only on syntactic features. This is because the same language is generated whether or not we stipulate string identity<sup>30</sup>, but not doing so results in a standard MCFG rule and therefore the proof of MCFG-equivalence<sup>31</sup>. Note that by combining *mrg\_atb1* with the rightward movement mechanism introduced in section 2.3.2, we are also able to generate the RNR in 3 as rightward ATB movement (though see section 4.3 for an alternative analysis).

$$\frac{[t : x, \alpha_1^f, \dots, \alpha_l^f(\alpha_1^s, \dots, \alpha_l^s)] \quad [s : =x \gamma, \alpha_1^f, \dots, \alpha_k^f(\alpha_1^s, \dots, \alpha_k^s)]}{[s : \gamma, \alpha_1^f, \dots, \alpha_k^f(\alpha_1^s, \dots, \alpha_k^s)]} \quad (mrg\_atb1)$$

$$\frac{[t : x \delta, \alpha_1^f, \dots, \alpha_l^f(\alpha_1^s, \dots, \alpha_l^s)] \quad [s : =x \gamma, \alpha_1^f, \dots, \alpha_k^f(\alpha_1^s, \dots, \alpha_k^s)]}{[s : \gamma, t : \delta, \alpha_1^f, \dots, \alpha_k^f(\alpha_1^s, \dots, \alpha_k^s)]} \quad (mrg\_atb2)$$

$$\frac{[s : x \gamma, \alpha_1^f, \dots, \alpha_k^f(\alpha_1^s, \dots, \alpha_k^s)] \quad [t : \approx x, \alpha_1^f, \dots, \alpha_l^f(\alpha_1^s, \dots, \alpha_l^s)]}{[s : x \gamma, \alpha_1^f, \dots, \alpha_k^f(\alpha_1^s, \dots, \alpha_k^s)]} \quad (adj\_atb1)$$

$$\frac{[s : x \gamma, \alpha_1^f, \dots, \alpha_k^f(\alpha_1^s, \dots, \alpha_k^s)] \quad [t : \approx x \delta, \alpha_1^f, \dots, \alpha_l^f(\alpha_1^s, \dots, \alpha_l^s)]}{[s : x \gamma, t : \delta, \alpha_1^f, \dots, \alpha_k^f(\alpha_1^s, \dots, \alpha_k^s)]} \quad (adj\_atb2)$$

Figure 9: Left merge and right adjoin ATB rules

Note that for coordination, we enforce  $l = k$  whereas for all other cases  $l \leq k$ . This ensures

<sup>28</sup>Note that although we drop an occurrence of *who* in the syntax, the fact that in the semantics its trace must be co-indexed with the other trace (and the antecedent) is deterministically recoverable from the derivation tree.

<sup>29</sup>A variant of *mrg\_atb1* allows  $\bar{x}$  to persist and generate, e.g., *who does Jack like, Mary hate and Pete despise?*

<sup>30</sup>For practical purposes, however, we allow MGParse to also enforce string identity, since otherwise many partial parses are generated in which a moving substring in the dependent is dropped which does not phonetically match some moving substring in the main structure, and such a strategy can clearly never result in the recognition of a sentence.

<sup>31</sup>If we view the syntactic part of the head chain plus the  $\alpha^f$ s as a single atomic category symbol, then all we are saying in effect here is that combining a category of type A with a category of type B results in a category of type C, which is no different from any other MCFG rule. Seki et al.'s (1991) lemma 2.2 shows that banning variables that become erased during a derivation has no effect on expressive power.

that the like-types constraint (see Appendix B) applies only to coordination and not, for instance, to parasitic gaps such as *which celebrity did pictures of disgrace?* or 6 above, where we only require the  $\alpha^f$ s in the specifier or adjunct to be a (possibly empty) subset of those in the main structure. That parasitic gaps are not subject to precisely the same constraints as coordination structures is evident from the fact that it is possible to fill a parasitic gap, leaving just the trace in the main clause, as in *which paper did Jack file without reading its title*, whereas we cannot extract from one conjunct but not the others (*\*who does Jack like and Mary hate Pete*) (part 2 of CSC)<sup>32</sup>; we assume that both parasitic gaps and ATB-coordinate structures involve the ATB-dropping mechanism, but differ in that only coordination is subject to a like-types constraint owing to its semantics.

Finally, recall that example 7 featured illicit coordination of two conjuncts containing traces in different structural case positions. To disallow these structures, we adopt Kobele’s (2008) Earley-style dotted feature mechanism so that features remain visible after they are ‘deleted’. Additionally, we assume that -case features are valued as acc., nom., gen. etc., when checked. This will then distinguish the two movers in 7 and prevent ATB-drop from applying<sup>33</sup>. To implement this, licensee and licensor features will be split into attribute/value pairs: e.g., -f represents an unvalued licensee feature, while  $-f^v$  is the valued equiv-

<sup>32</sup>We must also ensure that for conjunct (but not other) specifiers, only *mr<sub>g</sub>\_atb1* can apply, i.e.  $|\delta| = 0$ , and similarly for the comp merge rules in fig 2 if the complement is a conjunct. These restrictions capture the fact that while ATB extraction of the identical contents of conjuncts is possible, extraction of conjuncts themselves is not (*\*who<sub>i</sub> does mary like t<sub>i</sub> and t<sub>i</sub>*, *\*who<sub>i</sub> does Mary like Pete and t<sub>i</sub>*) (part 1 of CSC). We achieve these restrictions via the use of a diacritic on coordinator projections. For example, instead of :: and : coordinator projections are marked by ::̄ and :̄ separators. We can then formulate coordination-specific MERGE rules to ban moving complement and specifier conjuncts.

<sup>33</sup>Gazdar (1981) notes that if either gap is in an embedded clause, it no longer matters if the traces have different structural cases (*I know a man who Mary likes and hopes will win*). MGParse also correctly generates these constructions by assuming that the valuations of case are only temporary, and that as soon as an item moves to a higher landing site it reverts to (checked but unvalued) -case. This, combined with the standard assumption of successive cyclic wh-movement via intermediate spec-CPs (implemented with ‘suicidal’ licensors (+F<sub>?</sub>) on intermediate C heads which attract but do not check licensees and which self-delete if they themselves are unchecked), yields the correct result, assuming that -f matches  $-f^v$  for ATB-drop. Evidence that case valuation is indeed temporary is provided by the following contrast: *\*the man whom likes Mary* vs. *the man whom Pete said likes Mary*.

alent (e.g. -case<sup>nom</sup>). We then reformulate our rules using this new notation. Fig 10 gives the reformulated version of move-2, showing both the valuation and dotted feature mechanisms, where  $\beta$  and  $\zeta$  are feature sequence prefix variables. To avoid clutter, for the rest of the discussion we will omit this valuation notation from any rules.

$$\frac{[s: \beta \cdot +f^v \gamma, \alpha_1, \dots, \alpha_{i-1}, t: \zeta \cdot -f \delta, \alpha_{i+1}, \dots, \alpha_k]}{[s: \beta +f^v \cdot \gamma, \alpha_1, \dots, \alpha_{i-1}, t: \zeta -f^v \cdot \delta, \alpha_{i+1}, \dots, \alpha_k]} (\text{move2'})$$

Figure 10: Move-2 with valuation and dotted feature mechanisms

## 4.2 ATB Head Movement

We still need to derive 2 and 4, both of which by hypothesis involve ATB head movement: T-to-C in 2 and V-to-v in 4. Note that the head movement rules presented in section 2.3.3 are insufficient here because there the moving head fused immediately with the head it adjoined to, making head movement a highly local operation. In general, this appears descriptively correct since heads cannot usually skip other heads (*\*Have you would helped?*) (cf. Travis’ (1984) *Head Movement Constraint*). There are, however, certain arguable exceptions to this constraint, such as clitic climbing in Romance and the ATB head movement in 2 and 4. Consider again example 2, repeated below with the CoordP (in reality a TP) now shown.

2. Who<sub>j</sub> does<sub>i</sub> [<sub>CoordP</sub> [<sub>TP</sub> Jack t<sub>i</sub> like t<sub>j</sub>] [<sub>Coord</sub> and] [<sub>TP</sub> Mary t<sub>i</sub> hate t<sub>j</sub>]]?

The derivation for this sentence initially proceeds precisely as in section 4.1 (except that T is now overt). However, when the conjunction head merges with its right TP conjunct, the T head (*does*) of that conjunct will become fused either with its dependents as before or with the Coord head, rendering it inaccessible to C. The situation for the left conjunct is even worse as our rules do not allow for head movement out of specifiers (which would violate CED). Our solution is to extend the grammar with a mechanism for excorporation which allows the head of a complement to move successive cyclically through the governing head rather than incorporating with it (see fig 11).

To implement excorporation, we add a new diacritic  $\hat{\ }$  to the selector which once again causes the complement’s head to move (we also add conjunctions with the feature sequence:  $\hat{x} = \hat{=} \bar{x} x$ ). This time, however, the raising head will become



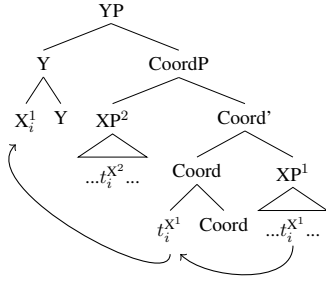


Figure 11: Excorporating ATB Head Movement

the new head of the selecting phrase, with the old head being fused onto the remnant complement string. This sets the stage for the new head to subsequently raise further, leading to successive cyclic head movement. The heads of any specifier conjuncts will simply be dropped<sup>34</sup>, just as their  $\alpha$  chains are dropped for ATB phrasal movement. The two rules are given in fig 13<sup>35</sup>. The first involves the complement case, hence the selector cannot yet contain any  $\alpha$  movers. The second rule shows the specifier case, and is rather like the specifier rule for ATB phrasal movement in that it involves dropping any  $\alpha$  chains in the selectee under feature identity with those in the selector. This time, however, the excorporation diacritic on the selector causes the head string of the selectee also to be dropped. Again only feature identity is required, hence the rule is MCFG-equivalent. The derivation for example 2 is given in fig 12 (only the leftmost conjunct's derivation is given in full)<sup>36</sup>.

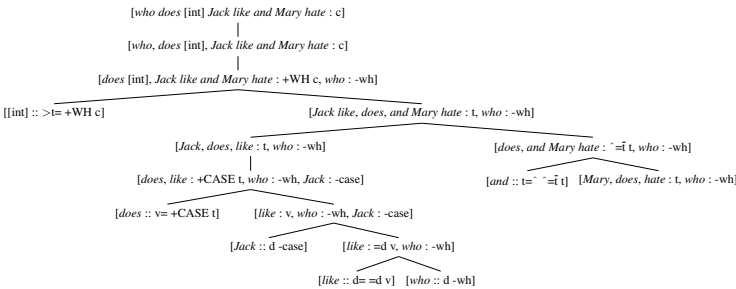


Figure 12: *who does Jack like and Mary hate*

### 4.3 Right Node Raising

In section 4.1, we stated that the ATB mechanism combined with the rightward movement rules pre-

<sup>34</sup> Again, co-indices on all head traces are deterministically recoverable from the derivation tree.

<sup>35</sup> Again, an additional rule is needed to allow  $\bar{x}$  to persist.

<sup>36</sup> Note that the final step in this derivation is a unary rule fusing together the three string parts of a head chain iff it is the only chain in the expression and it has just one feature and that feature is a c (equivalent to reaching the S node).

$$\frac{[(e, s_h, e) \bar{\bar{t}} : \hat{x} = \hat{x} \gamma]}{[(e, t_h, s_h t_i t_r) \bar{\bar{t}} : \hat{x} \gamma, \alpha_1, \dots, \alpha_k]} (mrg\_excorp)$$

$$\frac{[(t_l, t_h, t_r) : x, \alpha_1^f, \dots, \alpha_k^f (\alpha_1^s, \dots, \alpha_k^s)] [(s_l, s_h, s_r) \bar{\bar{t}} : \hat{x} \gamma, \alpha_1^f, \dots, \alpha_k^f (\alpha_1^s, \dots, \alpha_k^s)]}{[(t_l t_r s_l, s_h, s_r) \bar{\bar{t}} : \gamma, \alpha_1^f, \dots, \alpha_k^f (\alpha_1^s, \dots, \alpha_k^s)]} (mrg\_hm\_atb)$$

Figure 13: Excorp/ATB Head Movement Rules

sented in section 2.3.2 could generate RNR structures. However, rightward movement analyses of RNR are not without theoretical problems (see, e.g., Abels (2004) and Gazdar (1981)). For example, it is well known that RNR is always order-preserving, that it does not exhibit many island effects, and that the shared material (which has a focused interpretation) cannot survive VP ellipsis in the rightmost conjunct with which it appears to associate despite also scoping over all traces.

Fortunately, an alternative ATB strategy is available, under which the mover inside the rightmost conjunct undergoes *covert* leftward focus movement (to spec-CP where it c-commands and scopes over all the traces), while the movers inside the other conjuncts begin to undergo *overt* focus movement. Then, when ATB-drop of these overt movers occurs, once again only the string of the mover belonging to the rightmost conjunct remains, though this time in its base position (see Appendix A for derivations and full discussion).

This analysis is closer to the *external remerge* or ellipsis proposals in the linguistic Minimalist literature (e.g. de Vries (2009), Abels (2004)) and can better account for all of the aforementioned properties of RNR<sup>37</sup>. All that is needed is to relax the parser's constraint on string identity (fn.30) slightly, so that the empty string of a covert mover does not trigger a mismatch with the strings of overt movers for the purposes of ATB-drop.

## 5 Conclusion

We have presented the core mechanisms of MGParse and shown how coordination can be incorporated into an EDMG that uses relatively few (currently around 45) MCFG-equivalent rules to assign expressive structural descriptions to a wide range of construction types. While many open questions remain (gapping was not addressed, for instance), this is an important step towards our goal of constructing of a practical MG parser with both broad and deep coverage.

<sup>37</sup> Covert movement has been observed to escape certain island effects (see, e.g., Richards (2000)).

## Acknowledgments

A very big thank you to Mark Steedman for many hours of inspiring and insightful discussions on the topics contained in this paper. We would also like to thank the three anonymous reviewers for their comments and suggestions, several of which have been incorporated into the final paper. This work was funded by the Engineering and Physical Sciences Research Council (EPSRC).

## References

- Anne Abeille. 2006. In defense of lexical coordination. *Empirical Issues in Syntax and Semantics* 6, pages 7–36.
- Klaus Abels. 2004. Right node raising: Ellipsis or across the board movement. *Proceedings of NELS*, 34.
- Cedric Boeckx, Norbert Hornstein, and Jairo Nunes. 2010. *Control as Movement*. Cambridge University Press, Cambridge, UK.
- Robert Borsley. 2005. Against conjp. *Lingua*, 115:461–482.
- Noam Chomsky. 1995. *The Minimalist Program*. MIT Press, Cambridge, Massachusetts.
- Noam Chomsky. 2000. Minimalist inquiries: The framework. In Roger Martin, David Michaels, and Juan Uriagereka, editors, *Step by Step: Essays in Minimalist Syntax in Honor of Howard Lasnik*, pages 89–155. MIT Press, Cambridge, MA.
- Mark de Vries. 2009. On Multidominance and Linearization. *Biolinguistics*, 3.4:344–403.
- Lisa deMena Travis. 1984. *Parameters and Effects of Word Order Variation*. Ph.D. thesis, MIT, Cambridge, Massachusetts.
- Thomas Ernst. 2002. *The Syntax of Adjuncts*. Cambridge University Press, Cambridge, UK.
- Werner Frey and Hans-Martin Gärtner. 2002. On the treatment of scrambling and adjunction in minimalist grammars. In Gerhard Jäger, Paola Monachesi, Gerald Penn, and Shuly Wintner, editors, *Proceedings of Formal Grammar*, pages 41–52.
- Gerald Gazdar, Ewan H. Klein, Geoffrey K. Pullum, and Ivan A. Sag. 1985. *Generalized Phrase Structure Grammar*. Harvard University Press, Cambridge, Massachusetts.
- Gerald Gazdar. 1981. Unbounded dependencies and coordinate structure. *Linguistic Inquiry*, 12:155–184.
- Hendrik Harkema. 2001. *Parsing Minimalist Languages*. Ph.D. thesis, UCLA, Los Angeles, California.
- Norbert Hornstein. 2001. *Move! A Minimalist Theory of Construal*. Blackwell Publishing.
- C.-T. James Huang. 1982. *Logical relations in Chinese and the theory of grammar*. Ph.D. thesis, MIT, Cambridge, Massachusetts.
- Pauline Jacobson. 1987. Review of gerald gazdar, ewan klein, geoffrey k. pullum, and ivan a. sag. generalized phrase structure grammar. *Linguistics and Philosophy*, 10:389–426.
- Richard S. Kayne. 1994. *The Antisymmetry of Syntax, Linguistic Inquiry Monograph Twenty-Five*. MIT Press, Cambridge, Massachusetts.
- Gregory M. Kobele. 2005. Features moving madly: A formal perspective on feature percolation in the minimalist program. *Research on Language and Computation*, 3(4):391–410.
- Gregory M. Kobele. 2008. Across-the-board extraction in minimalist grammars. In *Proceedings of the Ninth International Workshop on Tree Adjoining Grammar and Related Formalisms (TAG+9)*, pages 113–128.
- Hilda Koopman and Dominique Sportiche. 1991. The position of subjects. *Lingua*, 85(2-3):211–258.
- Jairo Nunes. 1995. *The Copy Theory of Movement and Linearisation of Chains in the Minimalist Program*. Ph.D. thesis, University of Maryland, College Park.
- Jairo Nunes. 2001. Sideward movement. *Linguistic Inquiry*, 32:303–344.
- Jairo Nunes. 2004. *Linearization of Chains and Sideward Movement*. MIT Press, Cambridge, Massachusetts.
- Barbara Partee and Mats Rooth. 1983. Generalized conjunction and type ambiguity. In *Meaning, Use, and Interpretation of Language*. Berlin: de Gruyter.
- Norvin Richards. 2000. An island effect in japanese. *Journal of East Asian Linguistics*, 9:187285.
- Ian Roberts. 2010. *Agreement and Head Movement: Clitics, Incorporation and Defective Goals, Linguistic Inquiry Monograph 59*. MIT Press, Cambridge, Massachusetts.
- John Robert Ross. 1967. *Constraints on variables in syntax*. Ph.D. thesis, MIT, Cambridge, Massachusetts.
- Hiroyuki Seki, Takashi Matsumura, Mamoru Fujii, and Tadao Kasami. 1991. On multiple context free grammars. *Theoretical Computer Science*, 88:191–229.

Edward Stabler. 1997. Derivational minimalism. In Christian Retoré, editor, *Logical Aspects of Computational Linguistics (LACL'96)*, volume 1328 of *Lecture Notes in Computer Science*, pages 68–95, New York. Springer.

Edward P. Stabler. 2001a. Minimalist grammars and recognition. In Christian Rohrer, Antje Ross-deutscher, and Hans Kamp, editors, *Linguistic Form and its Computation*. CSLI Publications, Stanford, California.

Edward P. Stabler. 2001b. Recognizing head movement. In P. De Groot, G. Morrill, and C. Retoré, editors, *Logical Aspects of Computational Linguistics: 4th International Conference, LACL 2001, Le Croisic, France, June 27-29, 2001, Proceedings*, volume 4, pages 245–260.

Edward P. Stabler. 2006. Sideways without copying. In *Proceedings of the 11th Conference on Formal Grammar*, pages 133–146.

Mark Steedman. 2000. *The Syntactic Process*. MIT Press, Cambridge, Massachusetts.

Niina Ning Zhang. 2010. *Coordination in Syntax, Cambridge Studies in Linguistics Series 123*. Cambridge University Press, Cambridge, UK.

## Appendix

### A Right Node Raising and Parasitic Gap Derivations

Figures 14–16 give the derivations for the RNR example 3 and parasitic gap example 6 from the text. Here, as in the text, we have made certain simplifications, for example by removing -case features from objects (checked by big V in MGParse) and ignoring the little v causative head. Example 3 has two alternative derivations, corresponding to the rightward movement and leftward focus movement approaches discussed in sections 4.1 and 4.3. The rightward movement analysis works in precisely the same way as the leftward phrasal ATB mechanism discussed in section 4.1, except that now all the movers involved are undergoing rightward instead of leftward movement.

In the alternative focus movement analysis, the mover from the rightmost conjunct undergoes covert leftward focus movement to spec-CP, leaving behind its phonetic material inside the rightmost conjunct. Simultaneously, the movers inside the leftmost conjuncts begin to undergo overt leftward focus movement, but (both their syntactic and phonetic components) are dropped under

identity with the covert mover when those conjuncts are merged with the main structure - recall that for ATB, MGParse enforces string identity, in addition to syntactic identity, as doing so improves efficiency (see fn.30) without affecting weak generative capacity; however, we must relax the string identity requirement here slightly so that the empty string in the main structure (which originated in the complement conjunct) will not cause a mismatch for ATB-drop when compared with the overt strings in the specifier conjuncts.

Thus in the final structure, the silent syntactic/semantic component of the mover from the rightmost conjunct c-commands and therefore scopes over all the traces, while its phonetic component (and semantic trace) remains inside the rightmost conjunct and would therefore be deleted were VP ellipsis to occur. Instead of the rightward movement-based bracketing given in 3, then, the focus-based RNR analysis assigns this sentence the following structure (outline font indicates a silent, covertly moved constituent):

[*CP* [*Pete's sister*]<sub>*i*</sub>] [*TP* [*TP* Jack likes *t<sub>i</sub>*] and [*TP* Mary hates [*Pete's sister*]<sub>*i*</sub>]].

Notice that if we reversed the situation here so that it was the movers inside the *specifier* conjuncts which underwent covert movement while the mover from the complement underwent overt movement, then simply dropping the covert moving chains via ATB would overgenerate sentences like *\*Pete's sister, Jack likes Pete's sister and Mary hates*. We must therefore constrain the application of ATB-drop so that it cannot apply to covertly moving elements.

One way to do this is to allow the ATB rules to make reference to the empty vs. non-empty status of the string itself, and indeed this is precisely how MGParse operates. Grammatical rules do not usually make reference to string information, however, and in order to prove the MCFG-equivalence of our EDMG it is useful to show that we could encode this information in the category system itself. This can straightforwardly be done by marking the chain type of the covert mover in the output of the phonetic merge rules with a diacritic indicating its covert status. The ATB rules could then be reformulated so as to only apply to movers without this diacritic. This would clearly lead to a doubling of certain rules, but since the increase is finite, expressive power would be unaffected.

At present, MGParse uses the rightward movement approach to RNR because allowing covert focus movement into the system introduces lots of ambiguity which severely impacts on the parser’s efficiency. We are, however, currently investigating ways to restrict the application of covert focus movement, perhaps using statistics. Note that both of the approaches to RNR described here are capable of deriving sentences which feature both RNR *and* overt leftward ATB, such as *The policeman to whom I offered, and may give, a flower* (Steedman, 2000).

Finally, note that the adjunct control example 5 from the text is derived in a very similar way to the parasitic gap construction in fig 17, except that now it is only the subject which moves and hence undergoes ATB. The argument cluster coordination in example 4 from the text is derived along the lines of example 2 and is left as an exercise (hint: little *v* is required); precisely the same mechanism can accommodate cases of argument-adjunct cluster coordination, as in *I saw Harry yesterday and Peter today*.

## B Enforcing the like-types constraint on conjuncts

In section 3 (paragraph 2 and fn.20) we stated that full type uniformity between conjuncts could not be enforced by the selector features in an MG, unlike in a CCG where such features can be complex and specify, for instance, that all conjuncts must be clauses/verb phrases with object holes. We also noted that in our EDMG, the like-types constraint falls out instead from the interaction of the two main constraints on rule formation that we have adopted (CED and CSC), together with the ATB mechanisms presented in section 4. In this section, we would like to elaborate on these remarks.

First, we will define more precisely what we mean by ‘type’ here. Recall that MG expressions are composed of a head chain and up to *k* moving chains. Each chain is in turn composed of a string element and a feature sequence. Consider again the example from section 2.1, repeated below:

[[cause] *help* : +CASE *v*, *who* : -wh, *Jack* : -case]

There are two senses of ‘type’ to be defined here. First, we will define an ‘expression type’ as all and only the syntactic components of the entire expression, with all non-initial features replaced by feature suffix (and, if we were to include the dotted

feature mechanism, prefix) variables. The expression type of our example is thus as follows, where the ordering of the non-head chains is irrelevant:

[+CASE  $\gamma$ , -wh, -case]

In addition to the type of the entire expression, each chain can be viewed as having its own individual ‘chain type’, which is its (fully reified) feature sequence. For instance, the type of the head chain in our example is: +CASE *v*. The question to be addressed is how we can ensure that in our EDMG, coordination applies to two or more conjuncts with the same expression types *and* the same chain types (i.e. the features in the suffix and prefix variables matter).

Leaving aside the question of how to enforce identity between the head chains of the conjuncts for a moment, consider how we ensure that all conjuncts have identical sets of moving chain types. Recall that CED prohibits rules enabling extraction from within specifiers and that in section 4.1 we proposed a way to bleed this constraint by allowing movers inside a dependent structure to be dropped under identity with movers inside the governing structure. Recall further that for coordination, we stated that the rules must be formulated in such a way as to ensure that the number of movers inside the dependent is the same as the number of movers inside the main structure (coordinator projections being identified by  $\bar{\bar{\cdot}}$  and  $\bar{\cdot}$ ). This enforces part 2 of CSC, which states that it is not possible to extract the contents of any conjunct (even the complement conjunct), with one exception to this being cases of ATB movement.

Now, when the complement conjunct is merged with the Coord head, any movers which that conjunct contains will be transferred into the resulting expression. Subsequently, as additional (specifier) conjuncts are merged into the main structure, their sets of moving chains will be compared with the set of moving chains which originated inside the complement conjunct and either be dropped if they exactly match this set or lead to a doomed derivation if they do not (because no rule will ever allow them to escape the specifier conjunct). This ensures that all conjuncts must have identical sets of moving chain types.

The situation with the head chains is different: because the head chain of the complement conjunct cannot contain any licensee features (no conjuncts may move - part 1 of CSC), as soon as it is merged with the coordinator it will cease to exist

(though its string will be fused with the coordinator's string). Therefore, it will be unavailable for comparison with the head chain of any incoming specifier conjuncts. How can we overcome this problem?

First, observe that the syntactic feature sequences in MGs have a tripartite structure composed of requirements (=selectors+licensors), the selectee (PoS category) feature, and the licensee features. These three parts are strictly ordered from left to right; that is, an element must have all its argument positions fully saturated before it can be selected as the argument of a higher head in accordance with Xbar theory (unlike in CCG, where unsaturated elements can be selected), and only after it is selected can it undergo movement.

Now consider how we enforce the like-types constraint between the head chains of conjuncts. The selectee/PoS category feature is straightforwardly matched by the constraint on the lexicon that all coordinators must follow the general schema:  $x = \bar{x} x$  (or in certain special cases perhaps  $x = \bar{x} y$ ), where both rightward and leftward selector features must have the same PoS category. As noted in Steedman (2000), this (and the like-types constraint more generally) derives from the semantics of coordination, which is a 'recursive transitive closure over same types' (Partee and Rooth, 1983). The licensee features, meanwhile, are trivially matched by the fact that they are disallowed on all conjunct heads owing to part 1 of CSC: no conjuncts may move.

What of the requirement features? Interestingly, there is no way in our current rule system to ensure that the selector and licensor features of each of the conjuncts' head chains match. Thus we predict that the like-types constraint on conjuncts is not absolute: conjuncts may have different types, but only with respect to the selector and licensor features on their head chains.

There is some evidence that this may be correct. For instance, it is possible to coordinate a yes-no interrogative with an interrogative featuring wh-movement, as in *Pete asked [who had been at the party] and [whether Jack had seen Mary]*. Assuming *whether* to be an interrogative complementizer, only the (silent) complementizer head in the first embedded clause will contain a +WH feature triggering movement. Thus the requirements of the two C heads would seem to differ here, suggesting that we have a genuine case of coordina-

tion of (partially) unlike-types. Another example would be the coordination of a ditransitive with an intransitive VP (*Jack remained and gave Mary his ticket*).

Of course, there are other ways to derive such sentences, for instance by assuming for our first example that an additional projection layer (perhaps ForceP) exists above the phrase hosting the wh-element (perhaps FocusP) and that it is actually ForcePs which are coordinated here. Nevertheless, the data at the very least does not appear to conflict with the approach to the like-types constraint on conjuncts described here.

## C On the coordination of lexical $X^0$ heads

Xbar theory requires all complements and specifiers to be fully saturated, maximal XP projections. As pointed out in Borsley (2005), this poses a serious challenge to the Xbar theoretic view of coordinate structures, given that the coordination of unsaturated  $X^0$  lexical heads is apparently also possible, as in *Hobbs [criticized and insulted] his boss*. In an attempt to rescue Xbar theory here, Kayne (1994) proposes that lexical coordination is only apparent, arguing that such examples feature ellipsis within the left XP conjunct. However, as Borsley notes, there are other cases which do not appear amenable to this analysis. For example, *Hobbs whistled and hummed similar tunes* clearly does not mean the same thing as *Hobbs whistled similar tunes and hummed similar tunes*.

Another strategy sometimes pursued here is to assume that apparent lexical head coordination is actually an instance of RNR (this is in fact how the Penn Treebank analyses such constructions). However, as discussed in Abeille (2006), RNR and lexical head coordination have rather different prosodic and semantic properties, meaning that this analysis too faces problems. We will therefore take  $X^0$  coordination at face value, and in this section propose a solution within the EDMG formalism that makes crucial use of the dotted feature mechanism introduced in section 4.1 (and adopted from Kobele (2008)).

Within MGs, the Xbar theoretic requirement that all arguments be maximal XP projections is encoded by the fact that all requirement features (=selectors+licensors) must precede all selectee (and licensee) features. That is, a given head must have all its requirements checked and deleted be-

fore itself being selected as a dependent. In other words, taking  $\beta$  to be requirements, and  $\gamma$  to be licensees, the only abstract head chain type which can be selected is the following, where the dot immediately precedes the selectee feature:

$$[\beta \cdot x \gamma]$$

Assuming  $\beta$  to be non-empty, on standard assumptions the above category could only have been derived via the application of MERGE and perhaps also MOVE operations. However, formally, nothing prevents us from allowing this category type to appear directly on  $X^0$  heads, or from defining a unary function which ‘type-saturates’ unsaturated  $X^0$  heads. Such items would not truly be saturated semantically, of course, but this is fine provided they can only be selected for by a coordinator with a matching set of requirement (and licensee) features; the matching requirement features on the coordinator projection can then subsequently satisfy the semantic requirements of all its  $X^0$  conjuncts in one fell swoop.

The rules for type-saturation and coordination of complement and (multiple) specifier  $X^0$  conjuncts are given in fig 14, where the asterisk is equivalent to the dot, except that it uniquely identifies the type-saturated heads so that they are only ever selected for by coordinators. As noted in fn.32, an overline on the feature separators ( $\overline{\cdot}$ ) indicates a coordinator projection.

$$\frac{[s \overline{\cdot} \cdot \beta x \gamma]}{[s \overline{\cdot} \beta * x \gamma]} \text{(type-saturation)}$$

$$\frac{[s \overline{\cdot} \cdot x = \overline{x} \beta x \gamma] [t \overline{\cdot} \beta * x \gamma]}{[st \overline{\cdot} x = \cdot \overline{x} \beta x \gamma]} \text{(h\_coord1(comp))}$$

$$\frac{[t \overline{\cdot} \beta * x \gamma] [s \overline{\cdot} x = \cdot \overline{x} \beta x \gamma]}{[ts \overline{\cdot} x = \overline{x} \cdot \beta x \gamma]} \text{(h\_coord2(spec))}$$

$$\frac{[t \overline{\cdot} \beta * x \gamma] [s \overline{\cdot} x = \cdot \overline{x} \beta x \gamma]}{[ts \overline{\cdot} x = \cdot \overline{x} \beta x \gamma]} \text{(h\_coord3(spec))}$$

Figure 14: Lexical head type-saturation and coordination rules

Clearly this approach is very close to the CCG analysis of lexical head coordination, except that in CCG coordinated heads are formally as well as (at the point of coordination) semantically unsaturated. Notice too that without the dotted feature mechanism, the subcategorization frame of a

head would be lost following the type-saturation operation, as the  $\beta$  requirements would simply be deleted. However, we would then have no way to ensure that we were only coordinating heads of the same valency.

Figure 15: Rightward movement analysis: *Jack likes and Mary hates Pete's sister*

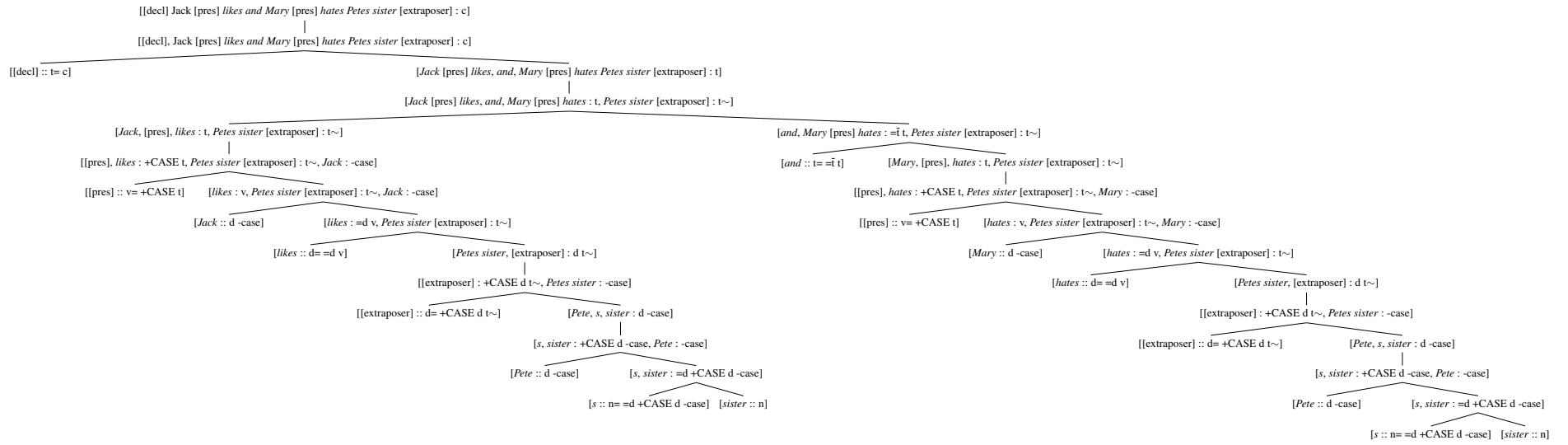


Figure 16: Focus movement analysis: *Jack likes and Mary hates Pete's sister*

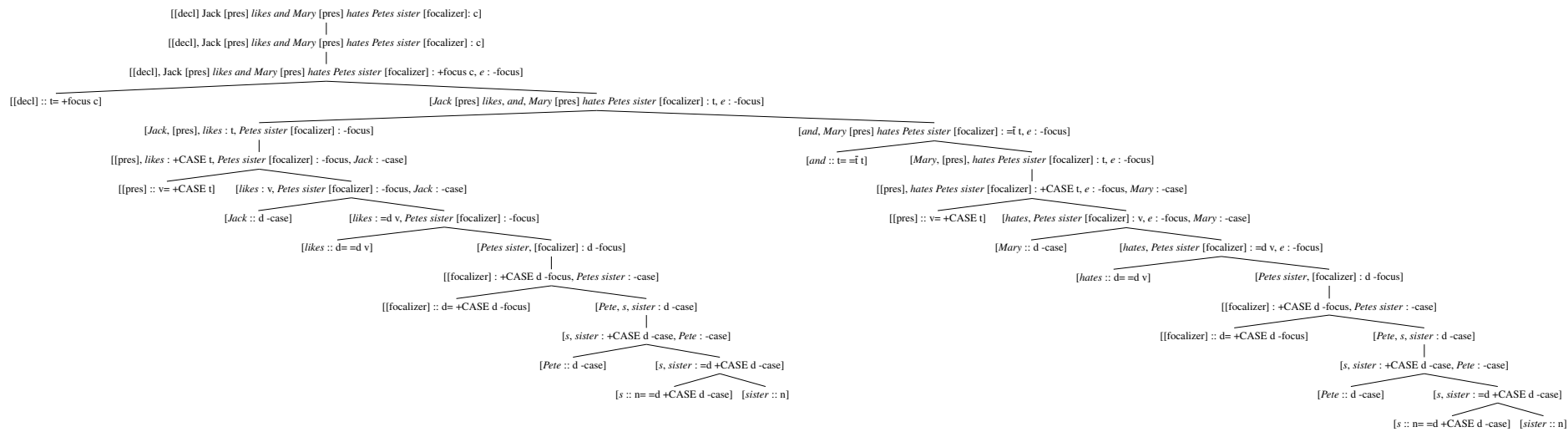




Figure 17: Which paper did he file without reading

