A MULTIPLE DOMINANCE ANALYSIS OF SHARING COORDINATION CONSTRUCTIONS USING TREE ADJOINING GRAMMAR

by

David Keith Potter

B.A., Portland State University, 2006

THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ARTS

IN THE DEPARTMENT

OF

LINGUISTICS

SIMON FRASER UNIVERSITY

Spring 2010

APPROVAL

Name: David Keith Potter

Degree: Master of Arts

Title of Thesis: A Multiple Dominance Analysis of Sharing Coordination Constructions Using Tree Adjoining Grammar

Examining Committee: Dr. María Teresa Taboada (Chair)

Dr. Chung-hye Han,
Associate Professor of Linguistics
Simon Fraser University
Senior Supervisor

Dr. Nancy Hedberg,
Associate Professor of Linguistics
Simon Fraser University
Supervisor

Dr. Francis Jeffry Pelletier,
Emeritus Professor of Linguistics and Philosophy
Simon Fraser University
Supervisor

Dr. Fei Xia,
Assistant Professor of Linguistics,
University of Washington
External Examiner

Date Approved: November 26th, 2009

Declaration of Partial Copyright Licence

Abstract

This thesis examines the syntax of sharing coordination constructions, in which a single phonologically overt element at the periphery of one conjunct is interpreted to belong syntactically and semantically to both conjuncts. I argue for a multiple dominance analysis of these constructions, against ellipsis and literal movement approaches, which I formalize in a lexicalized tree adjoining grammar (LTAG) framework. This analysis extends the empirical domain of previous TAG research beyond shared arguments to provide an account of shared modifiers, lexical items and derivationally non-local sharing. Finally, I define a linearization algorithm that linearizes the terminals of multiple dominance structures and produces the novel result of deriving the peripherality conditions on both left and right sharing constructions.

Keywords: Syntax, Tree Adjoining Grammar, Coordination, Sharing Coordination Constructions, Right Node Raising, Across the Board Movement, Linearization

To Angela

Acknowledgments

I would first like to thank my senior supervisor, Dr. Chung-hye Han, to whom I am indebted for being generous with patience, candid with guidance and instructive in her academic rigour. She has contributed tremendously to the quality of my thesis and my education at SFU. Second, I would like to thank my committee. Dr. Nancy Hedberg has provided me the advice, instruction and opportunity for which I consider her an outstanding mentor. This thesis has also benefited from Dr. Jeff Pelletier's astute and encouraging comments. Finally, I would like to thank my external examiner, Dr. Fei Xia, for her insightful and engaging questions and comments.

The department of Linguistics at Simon Fraser University has proven an exceptionally rewarding place to pursue my studies. Dr. Yue Wang has been incredibly generous in her intellectual support and in making the Language and Brain Lab an inclusive and engaging research environment. My fellow graduate students have also been a constant source of stimulating conversation and inspiration; you all are amazing. Finally, I cannot overstate the impact that Angela Cooper has had on my time in Vancouver. She, more than anyone else, has been the catalyst for my learning, and I am profoundly grateful for her presence in my life.

I would also like to gratefully acknowledge the financial support I have received during my time at SFU. I have been funded by SSHRC Standard Research Grant 410-2007-2169 and NSERC Discovery Grant RGPIN/341442-2008 to Dr. Han, SSHRC Grant 410-2007-0345 to Dr. Hedberg and NSERC Discovery Grant 5525-2008 to Dr. Pelletier.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis examines the syntax of certain verbal coordination constructions, in which a single phonologically overt element at the periphery of one conjunct is interpreted to belong syntactically and semantically to both conjuncts. For example, in (1), while the DP *Alexi* appears in the left conjunct, it is shared as the subject of both *cooked* and *ate*. Moreover, the right-shared DP *the bacon* is object to both verbs but is pronounced in the right conjunct.

(1) *Alexi* cooked and ate *the bacon*.

In addition to the shared arguments of (1), many other types of elements can be shared: whole CPs (2a), modals (2b), verb phrases (2c), verbs (2d), adverbs (2e), and nouns (2f).

(2) a. Tim suspects and Spencer is sure *that Portland is one of the greatest cities in the world*.
b. James knows *what would* make Sal happy and please Sue.
c. The kid knows what would and what wouldn't *kill the wolf-man*.
d. Spencer will and Alexi probably won't *visit*.
e. *Alexi usually* wraps his dinner in bacon and drinks beer.
f. Jamie likes the white and I like the red *fish*.

While sharing is possible in several other types of structures which do not include verbal coordinating conjunction, as in (3), taken from Hudson (1976), the focus of this thesis will be on sharing in verbal coordinate constructions. Furthermore, I limit my discussion to English examples in which there are only two conjuncts. Thus, the sentences above are representative of the constructions examined in this thesis.

(3) a. Noam is *a very important* linguist and social critic

b. Of the people questioned, those who liked outnumbered by two to one those who disliked *the way in which the devaluation of the pound had been handled*.

c. I'd have said he was sitting on the edge of rather than in the middle of *the puddle*.

d. It's interesting to compare the people who like with the people who dislike *the power of the big unions*.

1.1 Peripherality Conditions

It is a robust generalization about sharing coordination constructions that the shared element must appear at the periphery of its conjunct (Oirsouw (1983), Sabbagh (2007), Wilder (1994, 1997, 1999)). This constraint is formalized in (4).

(4) Peripherality Conditions (adapted from Sabbagh (2007))

a. In the configuration $[_A$...X...] CONJ $[_B$...Y...]

b. If X is shared, then it must be at the left edge of A, and if Y is shared, then it must be at the right edge of B.

In other words, right shared elements must appear at the right edge of the right conjunct. In (5a), the right-shared DP *beans* is pronounced at the periphery of the right conjunct. Similarly, left-shared elements surface at the left edge of the left conjunct, as in (6a). If the shared element is displaced from the periphery by non-shared material, the sentence is rendered ungrammatical, as exemplified by (5b) and (6b). In the latter case, the shared subject *Spencer* is preceded by the non-shared Wh-word *what* and auxiliary *did*. These Peripherality Conditions will serve as the key diagnostic for identifying cases of sharing coordination constructions.

(5) a. I like and Peter hates *beans*.

b. * I like in the morning and Peter hates *beans* in the evening.

(6) a. *What did Spencer* cook and eat?

b. * What did *Spencer* cook and who did meet?

1.2 Excluding Non-Sharing Constructions

Two other constructions, gapping and verb phrase ellipsis, have often been associated with sharing coordination constructions (Oirsouw (1987), Ha (2008), Wilder (1997)). However, there are several crucial characteristics that differentiate these cases from sharing constructions. In gapping, a verb or verb phrase is elided from the right conjunct and must be followed by a constituent, as in (7a) and (7b). In (7a), the verb *watched* is gapped in the right conjunct, while the direct object *a movie* remains. A similar example, (7c), in which the gapped verb and argument are not followed by a constituent, is ungrammatical. Unlike sharing constructions, which can target a variety of elements, gapping can only act on verbs or verbal constituents. Thus, gapping only the DP *the DVD* in example (7d) results in ungrammaticality.

(7) a. Tina watched a show and Angela ~~watched~~ a movie.

b. Tina watched a show in bed and Angela ~~watched a show~~ on the couch.

c. * Tina watched a show and Angela ~~watched a movie~~.

d. * Tina borrowed a DVD on Monday and Angela returned ~~a DVD~~ on Thursday.

Verb phrase ellipsis targets the whole verb phrase and requires that the auxiliary position be filled, as illustrated by the do-support in example (8a). A single element from a verb phrase cannot be elided to the exclusion of the other elements, exemplified in (8b) and (8c). Similar to gapping constructions, the elided element must follow the antecedent, such that in coordinate structures the verb phrase can only be elided from the right conjunct.

(8) a. Angela likes mashed potatoes and Tina does too.

b. * Angela likes mashed potatoes and Tina ~~likes~~ mashed potatotes too.

c. * Angela likes mashed potatoes and Tina likes ~~mashed potatotes~~ too.

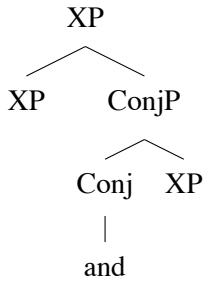
Thus, it appears that gapping and verb phrase ellipsis are subject to different constraints than sharing coordination constructions. Both gapping and verb phrase ellipsis only target verbal constituents, and the gaps must follow their antecedents. On the other hand, sharing constructions can freely target a host of different categories, including nouns, modals and clauses. Furthermore, a shared element can appear in either the left or right conjunct, but the Peripherality Conditions must be respected. This diverges from gapping and verb phrase ellipsis, which are not constrained by

these conditions. For these reasons, I consider gapping and verb phrase ellipsis separate constructions from sharing coordination, and in this thesis, I limit my discussion to the latter.

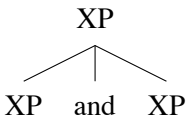
1.3 Asymmetric Coordination

Throughout this thesis, I assume asymmetric coordinate structures, whereby coordinators are the heads of binary branching functional projections, following Munn (1993) and Johannessen (1998). These structures, referred to as ConjP, select the right conjunct and adjoin into the left conjunct, illustrated in (9). Such a structure contrasts with a symmetric analysis, depicted in (10), which posits a headless ternary branching coordination structure.

(9)



(10)



This assumption is motivated by scope and binding evidence, which suggests that the first conjunct c-commands the second (Hartmann (2000)). In examples (11a) and (12a), the quantifiers in the left conjuncts bind those in the following conjuncts; whereas, in (11b) and (12b), it is impossible for a quantifier in the right conjunct to bind into the left conjunct.

(11) a. Peter welcomed every woman$_i$ and her$_i$ companion.

b. * Peter welcomed her$_i$ companion and every woman$_i$.

(12) a. Every woman$_i$ and her$_i$ companion liked the party.

b. * Her$_i$ companion and every woman$_i$ liked the party.

Similarly, in (13a) and (13c), a pronoun in the right conjunct is bound by an R-expression in the left. However, an R-expression in the right conjunct is ungrammatical when co-indexed with a pronoun

in the left conjunct. These facts are unexpected in a symmetric analysis, where each conjunct c-commands the other. Instead, they support the asymmetric approach assumed here.

(13) a. Everybody knows Greg$_i$ and his$_i$ work.

b. * Everybody knows his$_i$ work and Greg$_i$.

c. Max$_i$ and his$_i$ friends are kind and generous.

d. * His$_i$ friends and Max$_i$ are kind and generous.

1.4 Research Question

Sharing coordination constructions have been discussed extensively in previous literature, as early as Chomsky (1957). The debate has centered on how to appropriately derive coordination, and a variety of analyses have been proposed, utilizing base-generated phrase structure rules (e.g. Dougherty (1970, 1971)), transformational derivations (e.g. Gleitman (1965), Oirsouw (1983)), Generalized Phrase Structure Grammar (Gazdar (1981)) and Combinatory Categorial Grammar (CCG, Steedman (1985)). From this debate, three approaches have emerged in recent years, namely literal movement, reduction and multiple dominance. Literal movement analyses coordinate small conjuncts (e.g. TP, vP) and have two derivational sources for shared material, across-the-board movement and base-generation (e.g. Williams (1978), Postal (1998), Sabbagh (2007)). Reduction accounts typically involve clausal coordination and derive the appearance of sharing by reducing one of two underlying copies of the shared material (e.g. Hartmann (2000), Kayne (1994), Ha (2008)). Finally, multiple dominance approaches posit that a single shared element is simultaneously present in both conjuncts (e.g. McCawley (1982), Goodall (1983), Wilder (1999)).

This thesis addresses the question of how best to analyze sharing coordination constructions in conceptual and empirical terms. Specifically, the most appealing analysis will require the fewest construction-specific mechanisms and will provide the broadest empirical coverage for English sharing constructions. Upon reviewing the three aforementioned approaches, I ultimately argue in favour of a multiple dominance account, noting that this mechanism is independently-motivated and correctly captures the results of several structural diagnoses which place the shared element in-situ within the conjuncts. The multiple dominance account proposed in this thesis is formalized within a Lexicalized Tree Adjoining Grammar framework (LTAG) (Joshi and Schabes (1997), Frank (2002)).

1.5 Organization of the thesis

In Chapter 2, specific proposals from the literal movement, reduction, and multiple dominance approaches are discussed. I describe how certain core cases of sharing coordination are derived in each proposal, followed by an evaluation of any conceptual or empirical limitations that arise. This review yields the conclusion that a multiple dominance account is conceptually sound and empirically justified; however, there remain outstanding issues regarding the linearization of multiple dominance structures. Chapter 3 is devoted to the formalization of a multiple dominance approach within an LTAG framework. I first outline several crucial definitions and assumptions necessary for my proposed analysis and review a previous LTAG account of sharing constructions by Sarkar and Joshi (1996). I adapt key aspects of their analysis and illustrate how the resulting account can derive cases of argument sharing. While the account proposed in Chapter 3 can derive shared argument constructions, there are several additional structures, namely derivationally non-local sharing, as well as modifier and lexical item sharing, that remain problematic for the analysis. Thus, in Chapter 4, I present a series of extensions that will enable the account to derive these challenging cases. In Chapter 5, I propose a novel algorithm for linearizing the terminals of multiple dominance structures and illustrate how it can successfully linearize examples that were problematic for previous accounts, as well as derive the appropriate peripherality conditions. In the final chapter, I discuss the major conclusions reached in this thesis and outline directions for future research.

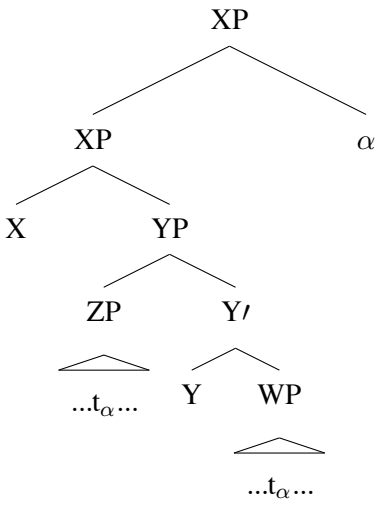
Chapter 2

Approaches to Sharing Coordination

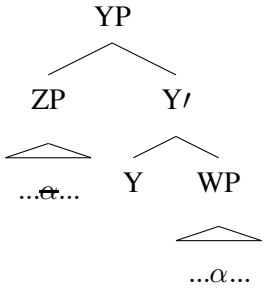
In this chapter, I review three approaches to the analysis and formalization of sharing coordination constructions: literal movement, reduction, and multiple dominance. The discussion focuses on two key aspects of sharing, namely whether the shared item is in-situ or ex-situ and whether the shared item is derived from one or two underlying copies.

In a literal movement approach, the shared material is derived from two underlying copies through a movement operation called across-the-board (ATB) movement (Sabbagh (2007)). The shared item is ex-situ, as the underlying copies are raised out of each conjunct by ATB movement and reduced to a single copy outside of the conjunction. This approach is depicted schematically in example (14a). A reduction approach, similar to literal movement, posits the existence of two underlying copies of the shared element (e.g. Ha (2008)). However, the appearance of a single overt shared item is the result of the in-situ syntactic reduction of one of the two underlying copies. Example (14b) illustrates this type of approach. The third approach, multiple dominance, treats the shared item as being literally shared between each conjunct. There exists only one underlying copy of the shared item in the derivation, and this item is in-situ in both conjuncts, as exemplified in (14c) (e.g. Vries (2009), Wilder (1999)).

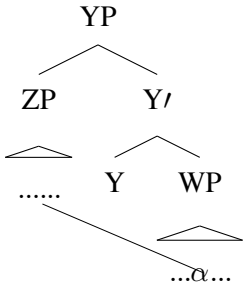
(14) a.



b.



c.



2.1 Literal Movement

2.1.1 Deriving Sharing Coordination Constructions

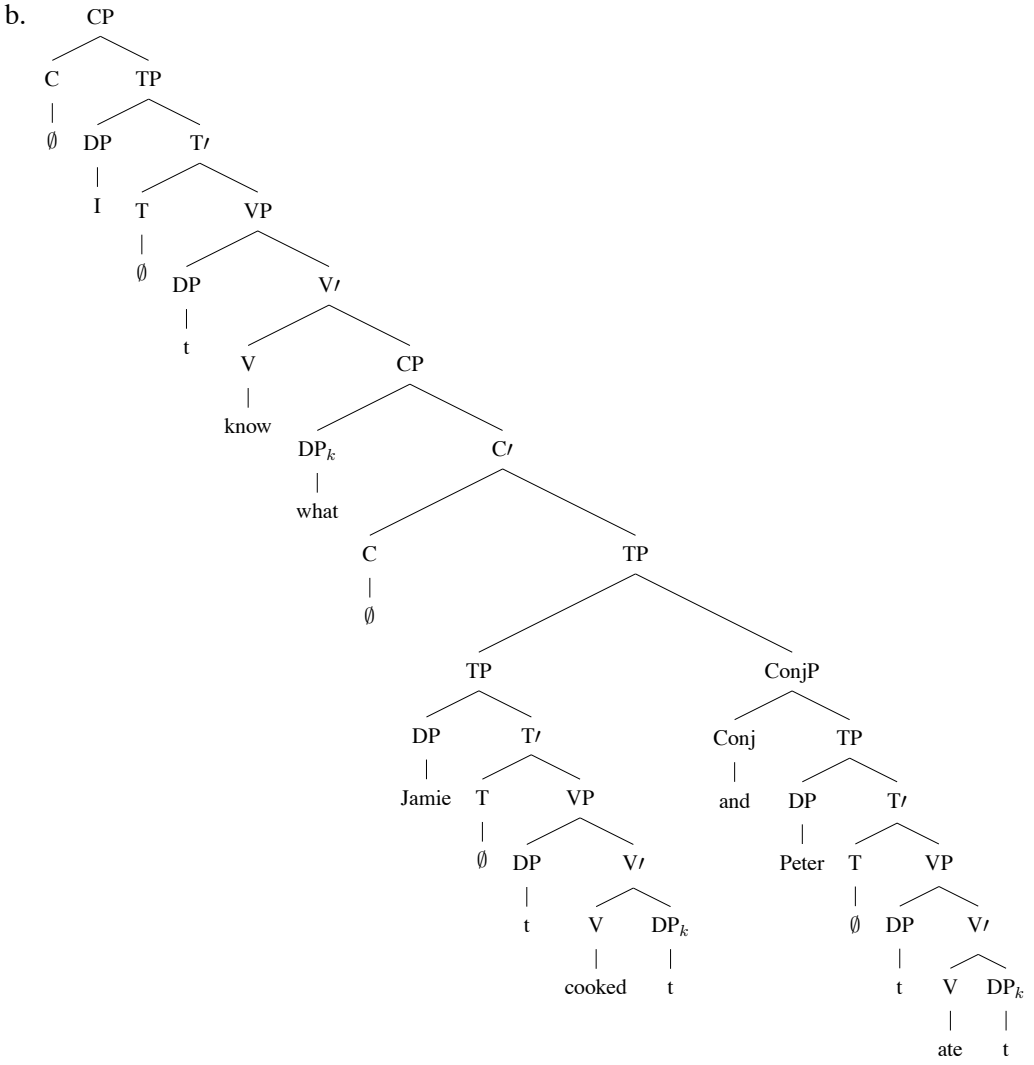
Ross (1967) proposed a rule of Conjunction Reduction, which deletes some element that is present at the edge of each conjunct and adjoins a copy of the deleted element to the matrix S. This notion that shared elements originate in each conjunct and end up outside of the conjunction came to be known as ATB movement and was subsequently adopted in various forms by Williams (1978),

Postal (1974), Abbott (1976), and, more recently, Postal (1998) and Sabbagh (2007). I discuss only Sabbagh in detail, in section 2.1.2, because his is the most recent literal movement account that directly addresses one of the strongest arguments against literal movement. However, many of my comments on his analysis apply equally to all accounts which assume literal movement.

A literal movement account derives sharing coordination constructions with two distinct mechanisms. First, ATB movement raises multiple underlying copies of the shared item simultaneously from each conjunct to a single landing site outside of the conjunction, typically adjoining at CP. Second, the multiple copies of the shared item are reduced to a single copy during the course of movement, necessitating that the moved items be identical. Sharing coordination constructions have straightforward derivations in a literal movement account. For instance, in the case of embedded object Wh-questions, such as example (15a), a copy of the object Wh-word has been merged in each conjunct, as exemplified in (15b). These copies then undergo ATB movement to SpecCP. Wh-questions, such as that in (16a), are derived similarly, with the addition that the shared head of T may require ATB movement to C, shown in (16b).

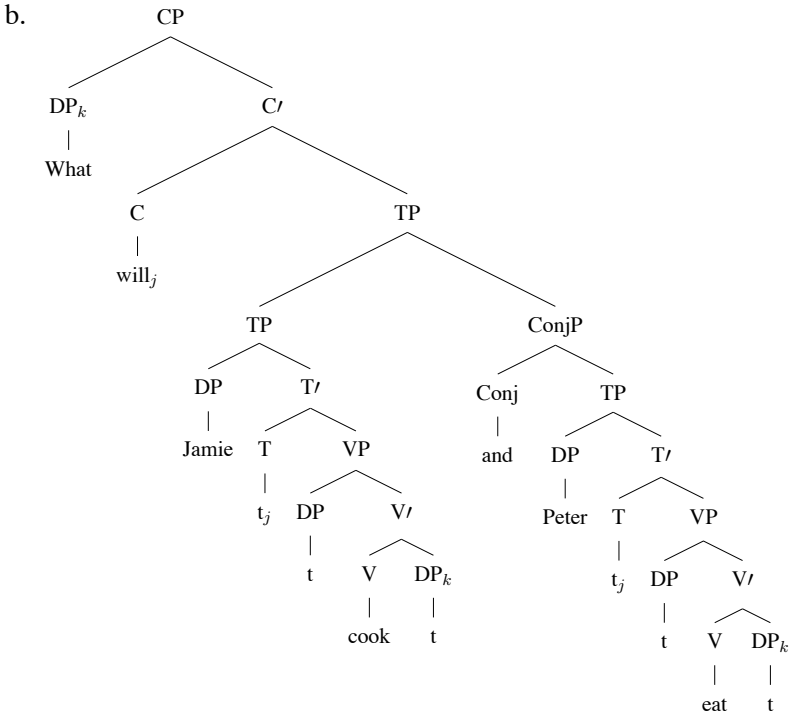
(15) a. I know what Jamie cooked and Peter ate.

b.



(16) a. What will Jamie cook and Peter eat?

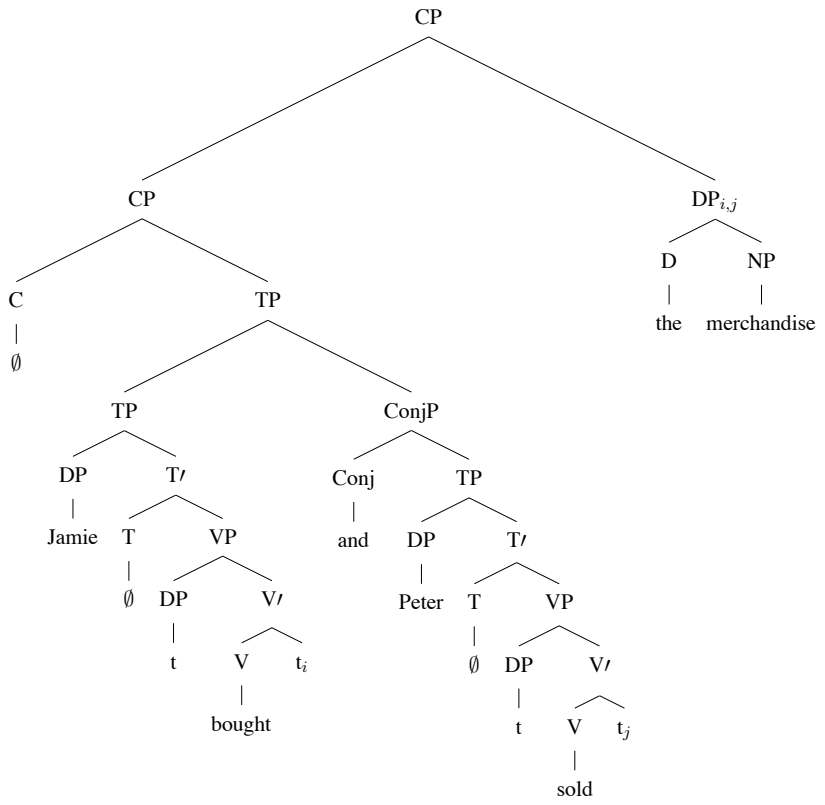
b.



Right sharing constructions can be derived in a similar manner. In (17a), the object *the merchandise* is shared between both *bought* and *sold*. A copy of the object is ATB-moved from each conjunct to a position outside of the conjunction. Sabbagh assumes that the landing site of rightward ATB movement is a position adjoined to the matrix CP, as in (17b). Shared subjects can likewise be derived by literal movement. Assuming the VP-internal subject hypothesis for shared subjects (Burton and Grimshaw (1992); McNally (1992)), the subject *Jamie* in example (18a) has undergone ATB movement to a shared SpecTP position from the specifier of VP in each conjunct, shown in (17b). It should be noted that examples such as this require VP coordination, instead of TP coordination.

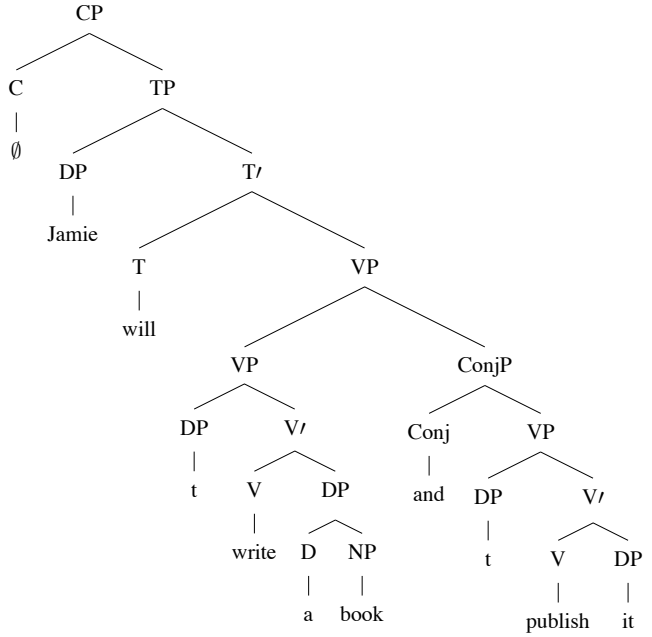
(17) a. Jamie bought and Peter sold the merchandise.

b.



(18) a. Jamie will write a book and publish it.

b.



One of the strongest arguments against literal movement has been that rightward ATB movement exhibits two particular behaviors markedly different from that of typical movement and leftward ATB. Rightward ATB movement can freely violate island constraints (Wexler and Culicover (1981), Levine (1985)), and the Right Roof Constraint (RRC) (Hartmann (2000) and Sabbagh (2007)), while other types of rightward movement cannot. Typically, both rightward and leftward movement are subject to strict movement constraints. Rightward movement, such as extraposition and heavy noun phrase shift (HNPS), conform to the right roof constraint (RRC), outlined in (19) (Ross (1967), Akmajian (1975)).

(19) a. Rightward movement may move and right-adjoin an element X to the cyclic node in which X is merged, but no further.

b. vP, CP and PP are cyclic nodes.

Extraposition can involve the rightward movement of prepositional and relative clause adjuncts as well as prepositional and sentential complements, shown in (20) (Keller (1995)). As exemplified in (21), HNPS moves phonologically heavy noun phrases to the right edge of the phrase.

(20) a. A man t came into the room [with blond hair].

b. Nobody t must live here [who is earning more than twenty pounds a week].

c. Extensive and intensive enquiries t have been made [into whether this fear of this penalty in fact deters people from murdering].

d. There is very great public concern t in Great Britain today (...) [whether the punishments which the courts are empowered to impose are adequate].

(21) a. Josh will eat t raw [almost anything you give him].

b. Josh returned to the library t for Jamie [each of the books she checked out last week.]

The RRC stipulates that the upper bound on such rightward movement is the edge of the cyclic node where the moved element was merged. In (22a), the prepositional adjunct *about verb-movement* has been raised out of the PP *of an article t*. The adjunct relative clause *which interested him* is raised out of a PP cyclic node in (22b). In (22c), the complement PP *from the oven* has been raised out of vP, and in (22d), the sentential complement *that ghosts were real* is raised out the the CP. HNPS is similarly restricted by the RRC, shown in (23).

(22) a. * Josh edited [$_{DP}$ a review [$_{PP}$ of an article t$_i$]] for Jamie [$_{PP_i}$ about verb-movement].

b. * He made informative comments to the author [$_{PP}$ of every paper t$_i$] yesterday [$_{CP_i}$ which interested him].

c. * I [$_{vP}$ removed the prime rib t$_i$ for him] a few minutes ago [$_{PP_i}$ from the oven].

d. * I wasn't sure [$_{CP}$ if it was Peter who believed t$_i$] before he told me so [$_{CP_i}$ that ghosts were real].

(23) a. * Max said that he was going to [$_{vP}$ return to the library t$_i$] yesterday, [$_{DP_i}$ each of the books that he checked out last week].

b. * Max described t$_i$ for Bill drunk, [$_{DP_i}$ a popular Broadway musical].

In contrast, rightward ATB constructions freely violate the RRC, as demonstrated by the examples in (24a) and (24b). This contrast is unexpected under a literal movement analysis, if both rightward movement and rightward ATB movement are derived from the same mechanism. A similar argument can be made for the free violation of islands by rightward ATB.

(24) a. [$_{CP}$ Joss [$_{vP}$ walked suddenly into t$_i$] and Maria [$_{vP}$ stormed quickly out of t$_i$] [$_{DP_i}$the dean's office]].

b. [$_{CP}$ Josh promised that he would [$_{vP}$ give t$_i$ to Jamie] and Joss claimed that he would [$_{vP}$ give t$_i$ to Sue] [$_{DP_i}$ all of the answers to the final exam]].

In general, leftward movement is constrained by island domains, in that Wh-words cannot front out of Wh-islands, as in (25a), or complex noun phrase islands, as in (25b). Leftward ATB conforms to this pattern as well; an island violation produces ungrammaticality, exemplified in (26). This is predicted in a literal movement account, as sharing coordination constructions are derived by ATB movement. In these cases, movement out of islands, whether simple or ATB, produces ungrammaticality. However, by this logic, the insensitivity of rightward ATB to these island constraints, as in (27), is unexpected, a fact first noted by Wexler and Culicover (1981).

(25) a. *[$_{CP}$ What$_i$ does Peter wonder [$_{CP}$ who bought t$_i$]]?

b. *[$_{CP}$ What$_i$ did he meet the professor [$_{CP}$ who taught him t$_i$]]?

(26) a. *[$_{CP}$ What$_i$ does Peter wonder [$_{CP}$ who bought t$_i$] and [$_{CP}$ who sold t$_i$]]?

b. *[$_{CP}$ What$_i$ did he meet the salesman [$_{CP}$ who bought t$_i$] and [$_{CP}$ who sold t$_i$]]?

(27) a. [$_{CP}$ [$_{CP}$ Who does Peter think bought t_i] and [$_{CP}$ who is Mary sure sold t_i] [$_{DP_i}$ the car]]?

b. [$_{CP}$ [$_{CP}$ Did he meet the customer who bought t_i] and [$_{CP}$ did Mary meet the salesman who sold t_i] [$_{DP_i}$ the car]]?

In sum, these data present two problems for a literal movement analysis of sharing coordination constructions. First, the difference between simple rightward movement and rightward ATB movement is not explained. Second, a literal movement account does not predict the contrast between rightward and leftward ATB movement island insensitivity.

In his recent ATB analysis, Sabbagh (2007) responds to these criticisms directly. Sabbagh argues that a literal movement analysis couched within the Cyclic Linearization model of Fox and Pesetsky (2005) is capable of deriving the distinction between ATB movement and typical leftward and rightward movement. Sabbagh posits that all rightward movement is in principle unbounded, but that it is constrained by the Cyclic Linearization principle of order preservation. It so happens that rightward ATB movement is largely vacuous, which allows for the principle of order preservation to be respected in right roof and island constraint violating movement. In general, other types of rightward and leftward movement is not vacuous, resulting in the constrained movement patterns that respect islands and the RRC. In addition to this counter-argument, Sabbagh presents several interesting arguments in favor of his ATB analysis. Parts of the Peripherality Constraint on sharing coordination constructions follow from the principle of order preservation. Additionally certain scope properties of right-shared quantified elements follow if they have been ATB-moved to a position where they c-command both conjuncts.

In the following section, I review Sabbagh's generalizations and analysis as well as the implications that arise from this analysis. I present evidence suggesting that his generalizations on the scope properties of right-shared material are incorrect and demonstrate how his analysis is incapable of accounting for this new data. I then present evidence which disconfirms the account's predictions for the scope properties of extraposition and HNPS. Finally, several other empirical and conceptual arguments against the literal movement approach are discussed.

2.1.2 Cyclic Linearization and Literal Movement

The basic intuition driving Sabbagh's analysis is that rightward movement is generally unbounded, provided that no phonologically overt material is crossed during this movement. If any phonologi-

cally overt material is crossed by a rightward moving element, then the movement is limited to the cyclic node in which it originated, such as the vP, PP or CP. Applying this intuition to examples (22) and (23), it is clear that in all cases of RRC violation, the extrapositioned and heavy noun phrase shifted items have crossed some phonologically overt material outside of their originating cyclic nodes. The rightward sharing examples (24a) and (24b) demonstrate RRC violating rightward movement; however, these cases are still grammatical because no material outside of the originating cyclic node has been crossed.

Sabbagh formalizes this analysis in a Cyclic Spell-out syntactic model based on the work of Fox and Pesetsky (2005). This model includes a forked derivation process, whereby syntactic structures are constructed from an enumeration of syntactic elements via the mechanisms Move and Merge. At the spell-out stage of the derivation, the syntactic structures are delivered to two separate components: 1) Phonological Form (PF), which is the syntactic component that prepares the syntactic structure for phonological processes, and 2) Logical Form (LF), which is the syntactic component that prepares the syntactic structure for semantic processes.

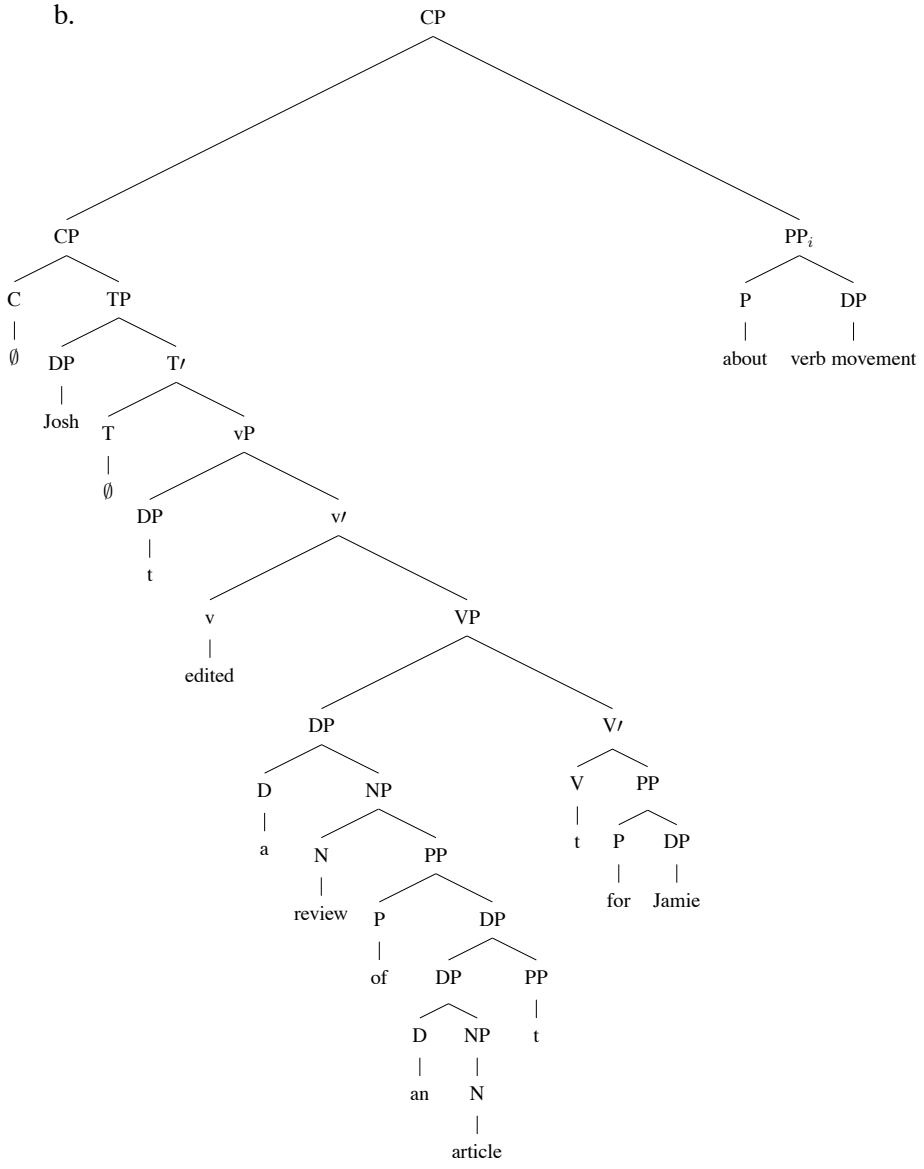
In the Cyclic Spell-out model, instead of sending a complete syntactic structure to spell-out, these structures are sent in cyclic cycles at specific stages of construction. These cyclic cycles are referred to as spell-out domains or phases, which Sabbagh assumes to be CP, vP and PP. When a spell-out domain is complete, the complement of the head of the domain is spelled out. For example, consider a complete CP containing a specifier and a complement. In Sabbagh's model, after the specifier has been merged, the CP is complete, and the complement of the CP, in this case TP, is sent to spell-out. The relevant consequence of a syntactic structure being sent to spell-out is that the linear order of the terminals in that structure is fixed. The spelled-out terminals may participate in further syntactic operations, under the constraint that they must always maintain the relative precedence relations fixed at spell-out.

From this perspective, the distinctions between rightward ATB and simple rightward movement are expected. The RRC on simple rightward movement results from the inability for the linear ordering of spelled-out material to change. In example (28a), the PP *about verb-movement* has raised out of the middle of a spelled-out phase, vP, which requires the reordering of spelled-out material, illustrated in (28b). However, this reordering violates the principle of order preservation because when the VP *a review of an article about verb-movement* is spelled out, its linear order is fixed, as illustrated by (28c). The final linear order of the entire structure is fixed as in (28d), and the derivation crashes due to a linearization contradiction. At a previous spell-out, *about verb-*

movement preceded *for Jamie*, but in the final spell-out, that order is reversed, violating the principle of order preservation.

(28) a. *[$_{CP}$ Josh edited [$_{DP}$ a review [$_{PP}$ of an article t$_i$] for Jamie [$_{PP_i}$ about verb movement]].

b.



c. a>review>of>an>article>about>verb-movement>for>Jamie

d. Josh>edited>a>review>of>an>article>for>Jamie>about>verb-movement

Rightward ATB movement can violate the RRC because no spelled-out material needs to be

reordered; the precedence relations between the spelled-out terminals are maintained. In example (29), an element is ATB moved from each conjunct, adjoining to CP. Unlike example (28a), this movement does not require a change in the order of spelled-out material. In each conjunct, the PPs *into the dean's office* and *out of the dean's office* have moved to the edge of the vP, completing this spell-out domain. The complement of the vP, excluding the PPs, is spelled-out. The DP *the dean's office* at the edge of the PPs then ATB moves and adjoins to CP. The final construction is spelled-out with the order in (30), and the DP *the dean's office* is successfully spelled out at the right edge of the structure without linearization contradictions.

(29) [$_{CP}$ [$_{TP}$Joss walked suddenly into t$_i$] and [$_{TP}$ Maria stormed quickly out of t$_i$] [$_{DP_i}$ the dean's office]].

(30) Joss>>walked>>suddenly>>into>>and>>
Maria>>stormed>>quickly>>out-of>>the>>dean's>>office

Thus, the difference between simple rightward movement and rightward ATB movement is epiphenomenal, in that both constructions must obey the principle of order preservation. The differences between rightward and leftward ATB movement follow in this account as well. While movement islands result from the requirements of order preservation, these requirements are inapplicable to cases of rightward ATB movement.

The Wh-island violation in example (31) is due to the object Wh-word of the embedded clause moving out from the middle of a spelled-out phase. The Wh-word *what* is merged in the complement of VP and moves to an additional specifier position of vP, thus avoiding spell-out within vP. The Wh-word is blocked from further movement; the specifier position of CP is filled, and English is widely assumed to disallow multiple specifiers of CP (Richards (2001)). Thus, the Wh-word *what* must remain to the right of *who* within the embedded TP, where it is subsequently spelled out. A derivation of (31) would require that *what* move past *who* and arrive at the specifier of the matrix CP. The final linearization of this structure would fail, as *what* would both precede the entire structure, at SpecCP of the matrix clause, as well as follow it, in the complement of the embedded VP.[1]

(31) *[$_{CP}$ What$_i$ does Peter wonder [$_{CP}$ who bought t$_i$]]?

[1]It is an interesting question why *what* cannot be spelled out in the additional specifier position of the embedded vP, as it is linearizable in this position. This problem is relevant for all phase theory accounts of island effects, not just Sabbagh's account.

Island effects obtain in leftward ATB movement for the same reasons as in the previous example. In example (32), each copy of the object Wh-word *what* cannot raise into SpecCP of the embedded clauses because these positions are filled. Therefore, they are spelled-out in the complement of VP. ATB movement of these Wh-words out of the conjunction to the specifier of the matrix CP violates the linear order established by prior spell-outs. This renders the example ungrammatical.

(32) *[$_{CP}$ What$_i$ does Peter wonder [$_{CP}$ who bought t$_i$] and [$_{CP}$ who sold t$_i$]]?

On the other hand, rightward ATB movement has no difficulty escaping Wh-islands; the right-shared element can adjoin directly to the matrix CP, without any reordering. In (33), the object DP *the car* is rightmost in each conjunct, and at each spell-out, it is ordered to the right of all other elements. The DP then ATB moves and adjoins to the highest CP, at which point it is spelled-out to the right of both conjuncts. Thus, no ordering conflict ever develops. Example (34) is derived in the same manner.

(33) [$_{CP}$ [$_{CP}$ Who does Peter think bought t$_i$] and [$_{CP}$ who is Mary sure sold t$_i$]] [$_{DP_i}$ the car]?

(34) [$_{CP}$ [$_{CP}$ Did he meet the customer who bought t$_i$] and [$_{CP}$ did Mary meet the salesman who sold t$_i$] [$_{DP_i}$ the car]]?

One of the more interesting conclusions that Sabbagh draws from his analysis is that it allows the Peripherality Constraint to be derived as a consequence of the same mechanisms which produce sharing constructions. In general, movement from one spell-out domain to another is grammatical only when it originates from the edge of the lower spell-out domain, in order to not violate the constraint on order preservation. Sabbagh notes that it follows from this analysis that any element to be shared must be at the edge of its conjunct in order to ATB move outside of the conjunction.

However, it is not necessarily the case that an element must be at the very edge of the spell-out domain to be available for subsequent movement. Sabbagh assumes that when a phase is spelled-out, it is the complement of the head of the spell-out domain that is linearized. However, as the specifiers and adjuncts of the phase are not included in the complement to the head, these elements are not yet linearized. If there are multiple specifiers or adjuncts, then it is possible for those which are non-peripheral to be ATB moved to a position outside of the conjunction.

For example, in (35), the heavy noun phrase *a movie about the zombies that couldn't die* moves to the right edge of the vPs in each conjunct by HNPS. Then, the adjuncts *with his wife* and *with*

her brother adjoin at their respective vPs, above the heavy noun phrase, resulting in the heavy noun phrase no longer being at the periphery of the conjuncts. Because these adjuncts are not complement to the head of the phase, they are not spelled-out. The derivation continues, and the copies of the heavy noun phrase are ATB moved out of the conjuncts, despite not being at the periphery. This case changes the generalization of the Peripherality Constraint that is necessary in Sabbagh's analysis: the gaps for the shared item must either be conjunct peripheral, or they must be dominated by one or more adjuncts that are rightmost in the conjunct.[2]

(35) [$_{CP}$ [$_{TP}$ Peter [$_{vP}$ watched at the theater t$_i$] with his wife] and [$_{TP}$ Mary [$_{vP}$ watched at the zoo t$_i$] with her brother] [$_{DP_i}$ a movie about the zombies that couldn't die]].

Additionally, Sabbagh notes that right-shared quantified noun phrases can scope over other elements in the sentence. The universal quantifier *every* in example (36) scopes over the non-shared existentially quantified DPs *a flu shot* and *a blood test*, providing the reading where the treatments co-vary with the patients. This reading is unavailable when the universal quantifier is not shared, as in example (37), as the only reading available is one in which every patient receives uniform treatment. This contrast follows in the literal movement account because the shared quantified noun phrase in (36) has been raised outside of the conjunction and adjoined to CP, where it c-commands and scopes over the rest of the clause.

(36) [$_{CP}$ [$_{CP}$ Some nurse will either [$_{vP}$ give a flu shot to t$_i$], or [$_{vP}$ administer a blood test for t$_i$]], [$_{DP_i}$ every patient who was admitted last night]]. (Sabbagh example 30a)

(37) [$_{CP}$ Some nurse will either [$_{vP}$ give a flu shot to every patient], or [$_{vP}$ administer a blood test for every patient]]. (Sabbagh example 30b)

Sabbagh also argues against in-situ approaches, claiming that a shared quantifier can scope out of an island. Example (38) demonstrates that the relative clause is an island for the universal quantified noun phrase *every Germanic language*, in that the universal quantifier cannot take wide scope over the existential quantifier *someone*. In contrast, when the universal quantifier is shared,

[2] In fact, the formulation of the Peripherality Constraint also applies to multiple specifiers. Thus, in languages that permit multiple multiple Wh-fronting, such as Bulgarian (Richards (1997)), it is predicted that a lower Wh-word may ATB move out of a coordinate structure, while still satisfying the principle of order preservation. If this is indeed possible, such movement would prove an interesting exception to the constraint on superiority on multiple Wh-fronting in this language. If impossible, it is not clear how such derivations could be blocked.

as in (39), it can take widest scope. This reading where a shared quantifier takes wide scope outside of an island is problematic for in-situ analyses. Both shared and non-shared elements cannot move out of islands in these in-situ analyses, and therefore these wide scope readings are unpredicted.

(38) [$_{TP}$ Josh knows someone [$_{CP}$ who speaks every Germanic language]]. (Sabbagh example 33)

(39) [$_{CP}$ [$_{TP}$ John knows someone [$_{CP}$ who speaks t$_i$]]] and [$_{TP}$ Bill knows someone [$_{CP}$ who wants to learn t$_i$]] [$_{DP_i}$ every Germanic language]. (Sabbagh example 34a)

However, these readings are not uncontroversial; Abels (2004) notes that informants who accept the wide scope reading in (39) also tend to accept it in (38). Without independently verifiable evidence on the scope possibilities of these constructions, it seems premature to use such disputed evidence to supporting a literal movement approach.

Furthermore, Sabbagh's analysis makes strong predictions on the possibilities of scope interactions in all rightward movement constructions, and some of these constructions provide stronger test cases. Sabbagh assumes that there are only two possible landing sites for rightward movement: the matrix CP and the right edge of the phase in which the shared element was initially merged. Once the rightward moving material is at the right edge of its spell-out domain, it continues to move, unbounded, to the matrix CP. It follows then that all material which moves rightward and adjoins at the CP can have scope interactions with other quantifiers in the sentence. However, this prediction does not seem to be borne out. In (40), the quantified noun phrase *each novel* is right shared. In Sabbagh's analysis, it would then be adjoined to the matrix CP, where it is predicted to be able to take wide scope over the matrix subject *someone*. This reading, where the believers co-vary with the novels, is unavailable, and the only available reading is where *someone* takes widest scope.

(40) [$_{CP}$ [$_{CP}$ Someone believes [$_{CP}$ that Mary will sell t$_i$] and [$_{CP}$ that Peter will buy t$_i$]] [$_{DP_i}$ each novel]]. (each novel $\not>$ someone)

Sabbagh's analysis predicts the possibility of similar scope interactions in non-sharing rightward movement. All rightward movement, not just rightward ATB movement, is bounded only by the principle of order preservation. This means that extraposed or heavy noun phrase shifted elements that have been vacuously raised out of islands and adjoined to CP should be able to scope over the rest of the sentence. However, the examples in (41) clearly show that these scope interactions are

impossible[3].

(41) a. Every football player came into the room with some kind of beer. (some $\not>$ every)

b. A good chef will eat t raw [almost anything you give him]. (anything $\not>$ a good chef)

c. Someone returned to the library t for Jamie [each of the books that were checked out last week]. (each of the books $\not>$ someone)

In sum, there is no clear way to constrain unbounded rightward movement such that the appropriate scope properties obtain. Sabbagh's analysis over-predicts the scope interactions in both right sharing and rightward movement constructions. In examples (40) and (41), the wide scope readings of the rightward-moved element cannot be excluded.

Additionally, cases of VP ellipsis pose an empirical challenge to this account. According to a literal movement analysis, a rightward ATB moved object raises out of the VP and adjoins to CP. If a shared object has raised out of the VP, then VP ellipsis should be able to target the verb to the exclusion of the shared element (McCawley (1982)). As the examples in (42) illustrate (Abels (2004)), this does not seem to be the case. Example (42a) demonstrates a typical right sharing construction, and example (42b) a typical VP ellipsis construction. Example (42c) illustrates that verb phrase ellipsis cannot strand a right-shared element. Thus, a literal movement account overgenerates, incorrectly predicting that examples such as (42c) are grammatical.

(42) a. [$_{CP}$ Jane [$_{vP}$ talked about t$_i$] and Frank didn't [$_{vP}$ talk about t$_i$] [$_{DP_i}$ the achievements of the syntax students]].

b. [$_{CP}$ Jane [$_{vP}$ talked about the achievements of the syntax students] and Frank didn't].

c. *[$_{CP}$ Jane [$_{vP}$ talked about t$_i$] and Frank didn't [$_{vP}$ ~~talk about t$_i$~~] [$_{DP_i}$ the achievements of the syntax students]].

Finally, it is conceptually problematic that the ATB mechanism is specific to sharing coordination constructions. Syntactic mechanisms should be broadly applicable to a host of syntactic phenomena, and the explanation of individual constructions should follow from the interaction of general syntactic mechanisms (e.g. deriving subject movement to SpecTP, specifying binding domains in terms of c-command). Furthermore, previous literature has not delineated any formal

[3]See also Culicover and Rochemont (1990) for arguments that extraposed elements are located no higher than vP.

mechanisms to motivate ATB movement. For instance, the impetus for subject raising and Wh-fronting has traditionally been explained in terms of feature checking; however, there has been no attempt to motivate why rightward ATB movement should occur, or why two elements are permitted to move simultaneously. Additionally, the process of deriving one overt element from two underlying copies in the course of movement is never made formally explicit.

2.2 Reduction Approaches

A second type of approach to sharing coordination constructions is the reduction approach. In these accounts, an underlying copy of the shared material is present in each conjunct. One of these copies is realized overtly, and the other is phonologically reduced or deleted. Such analyses have been proposed by An (2008), Mukai (2002), Levine (1985), Wexler and Culicover (1981), Kayne (1994), Hartmann (2000), Féry and Hartmann (2005) and Ha (2006, 2008).[4] There is no "sharing" in these approaches, as the overt copy of the shared element is located in-situ and only has syntactic and semantic dependencies within that conjunct. This is exemplified in (43), where two copies of *the beans* are present. I will refer to the reduced copy as the *target*, indicated by the ~~strikeout~~, and the phonologically realized copy as the *antecedent*, marked by [brackets]. The fundamental issue for the reduction approach is the definition and justification of the licensing conditions for reduction. In this section, I limit my discussion to Ha (2008), as it is the most recent reduction account that comprehensively addresses both left and right sharing coordination constructions.

(43) Peter cooked ~~the beans~~ and ate [the beans].

Ha (2008) argues that sharing constructions are the result of a special type of ellipsis. Adapting the syntactic ellipsis feature proposed by Merchant (2001), Ha claims that a Right Node Raising ellipsis feature, E_{RNR}, is responsible for deriving sharing coordination constructions. This feature is present on contrastively focused elements immediately preceding the shared item. In order to be activated, it must be in a specific syntactic configuration and fulfill a mutual entailment condition. Following Merchant (2001), Ha posits that the mutual entailment condition is fulfilled when the contrastively focused elements are replaced with existentially bound variables and when the

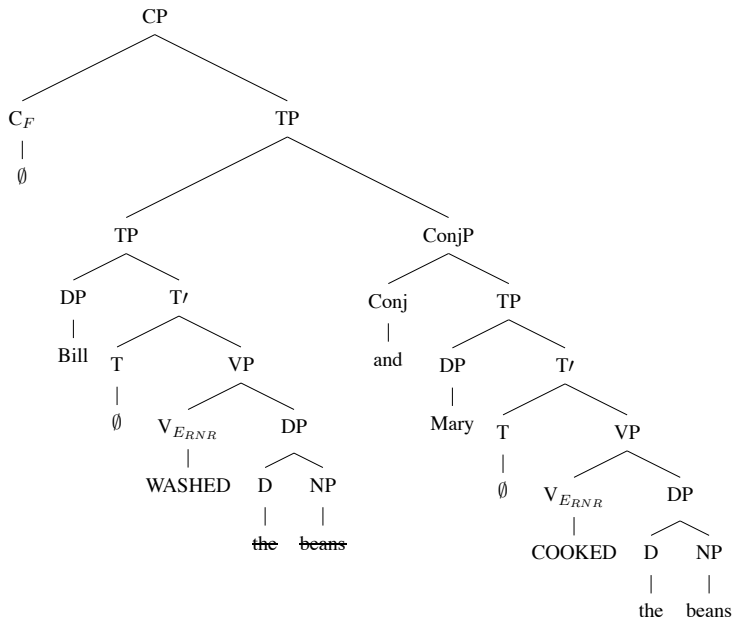
[4] Bošković and Franks (2000) and Munn (1992, 1999) advance an approach in which the target is in-situ while the corresponding non-target is not a phonologically reduced copy of the target, but a null or otherwise reduced element. For reasons of space, I cannot comment on these approaches here.

smallest constituents containing these variables entail each other. These constituents are referred to as F-closures. Ha assumes that all predicate coordination is at most TP coordination and in some cases VP coordination. The head of CP contains a probe which searches for and activates the nearest E_{RNR} feature. Ha stipulates that a probe may only activate one feature. Consequently, the sister of an activated E_{RNR}-featured node is phonologically deleted. The principle advantage for this approach is that the notion of an ellipsis feature and a mutual entailment licensing condition are independently motivated by Merchant (2001) for other ellipsis constructions. The only new mechanism that needs to be added to the grammar is the feature activation function.

In Ha's approach, a typical case of right sharing, such as in example (44a), is derived as follows. The elements *WASHED* and *COOKED* are contrastively focused and therefore bear the E_{RNR} feature. As depicted in (44b), the probe at the head of CP activates the nearest E_{RNR} feature, which is present on *WASHED*. Because the probe can only activate one E_{RNR} feature, the one on *COOKED* does not become activated. At spell-out, the syntactic structure is sent to PF and LF. In LF, the F-closures of both conjuncts are verified for mutual entailment. Because they entail each other, PF is instructed to delete the phonological representation of the sister of *WASHED*.

(44) a. Bill WASHED ~~the beans~~ and Mary COOKED [the beans].

b.



In cases where the F-closure containing the target entails the F-closure containing the antecedent, but not vice versa, ellipsis is not licensed. Instead, this configuration licenses the deac-

centing of the antecedent. In (45), the F-closure containing the target *went to the market* is entailed by the one containing the antecedent, *drove his new Corolla to the market*. As a result, *went to the market* is deaccented.

(45) JOHN *went to the market*, and SUE drove her new Corolla to the market.

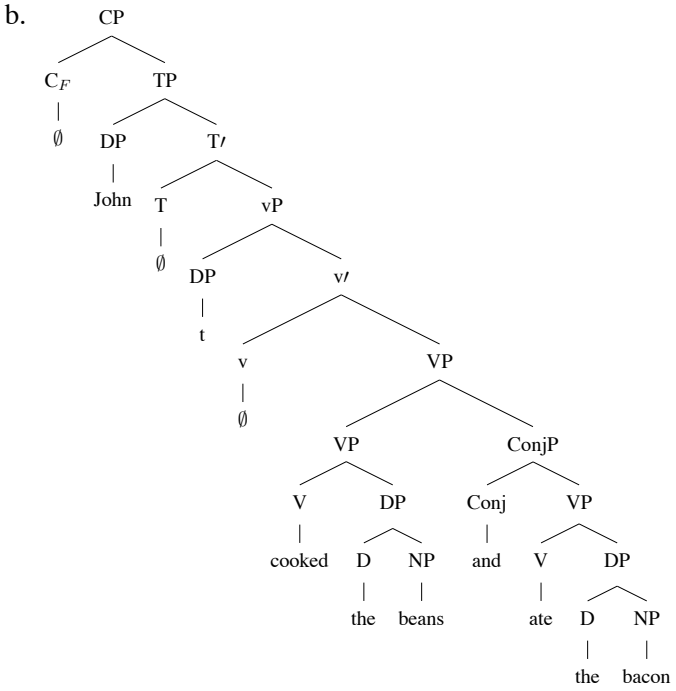
Ha argues that this ellipsis account can be generalized to all left sharing constructions except subject sharing. For instance, in example (46), the object Wh-word is shared. This construction is derived in two steps. First, similar to right sharing constructions, ellipsis occurs in the left conjunct. Ha assumes that *WATCH* and *EMULATE*, being contrastively focused, bear E_{RNR} features. Because the probe on the head of CP activates the feature on *WATCH*, and the F-closure of the conjuncts entail each other, the Wh-word in the left conjunct is elided. This is followed by the movement of the object Wh-word from the right conjunct, to a position outside of the conjunction. Following the coordinate structure constraint of Fox (2000), Ha argues that the Wh-object can move cyclically outside of the conjunction to SpecCP, driven by typical movement mechanisms (e.g. uninterpretable Wh and +Q features).

(46) [$_{CP}$ [Who$_i$] does [$_{TP}$ Peter WATCH ~~who~~] and [$_{TP}$ James EMULATE t$_i$]]?

Furthermore, examples such as (47a) are not analyzable as subject sharing constructions under Ha's analysis. There is no element preceding the position in which the subjects are merged, assumed to be SpecvP, which could receive the E_{RNR} feature. Thus, Ha assumes that these constructions are the result of VP coordination. A single vP then projects above the coordination, and a single subject is merged. This leads to the conceptually unappealing conclusion that ConjP behaves unlike other functional projections in that its position within the extended projection is not fixed (Grimshaw (2000)). In structures such as (47a), the ConjP is dominated by TP, while in examples like (44a), ConjP dominates TP. This diverges from the behavior of other functional projections such as CP and TP, which always appear in the same dominance relations.

(47) a. John cooked the beans and ate the bacon.

b.



2.2.1 Conceptual Challenges

While it seems that Ha's extension of Merchant's 2001 ellipsis feature explains sharing coordination constructions, several empirical and conceptual challenges remain. Ha's central claim that sharing is due to an independently motivated ellipsis mechanism is based on a flawed premise: ellipsis and sharing coordination are quite different constructions. While sharing coordination constructions can target DPs, CPs and TPs, VP ellipsis only elides VPs. Critically, VP ellipsis cannot target DPs, which are some of the most natural sounding targets for sharing coordination constructions. The fact that VP ellipsis only elides VPs, while the E_{RNR} mechanism elides a host of different categories (e.g. DPs, CPs, TPs), suggests that these constructions are derived differently. This undermines the notion that sharing constructions are derived using an independently motivated mechanism. If sharing coordination constructions are derived from ellipsis, then this sharing-ellipsis mechanism appears to be construction-specific, in that it is applicable only to the derivation of sharing coordination constructions.

2.2.2 Empirical Challenges

In addition to the conceptual complications, Ha's analysis faces a number of empirical limitations. Ha assumes that every contrastively focused item has an E_{RNR} feature. To avoid generating examples where every ellipsis feature is activated, causing inappropriate deletion of antecedents and targets, as in (48b) and (48c), Ha stipulates that the probe may only activate the single nearest E_{RNR} feature. While this stipulation correctly derives (48a) and rules out (48b) and (48c), it also predicts that (49a) is ungrammatical. Furthermore, Ha's analysis only allows a single E_{RNR} feature to be activated, the ungrammatical sentence (49b) is predicted to be grammatical. In order to derive the grammatical structure in (49a), two E_{RNR} features would have to be activated.[5]

(48) a. I COOKED ~~the beans~~ and she ATE [the beans].

b. * I cooked ~~the beans~~ and she ate ~~the beans~~.

c. * I cooked [the beans] and she ate ~~the beans~~.

(49) a. Peter washed ~~the beans~~, Jamie cooked ~~the beans~~, and Justin ate [the beans].

b. * Peter washed ~~the beans~~, Jamie cooked [the beans] and Justin ate [the beans].

A similar problem arises in multiple left sharing constructions. In example (50a), an object Wh-word and modal are shared. In order for both of these elements to be elided in the left conjunct, both the subjects and the verbs must have an E_{RNR} feature. However, Ha stipulates that the probe may only activate the single closest E_{RNR} feature. Thus, only the feature on *Peter* would be activated, thereby licensing the ellipsis of *would*. This leaves an overt copy of *who* in each conjunct. Finally, the modal and the nearest Wh-word would raise out of the conjunction, yielding the ungrammatical sentence given in (50b).

(50) a. [$_{CP}$ [Who$_j$] [would$_i$] [$_{TP}$ Peter ~~would~~ WATCH ~~who~~] and [$_{TP}$ James t$_i$ EMULATE t$_j$]]?

b. *[$_{CP}$ [Who$_j$] [would$_i$] [$_{TP}$ Peter ~~would~~ WATCH t$_j$] and [$_{TP}$ James t$_i$ EMULATE who$_j$]]?

[5]Note the difference between examples like (49b) and *Peter WASHED and Jamie COOKED the beans, and Justin ate the beans*. The latter is the case of multiple coordination; where a coordinate structure, in which sharing of *the beans* has taken place, is coordinated with another sentence. The lack of an overt coordinator in examples like (49b) seem to force the simple, three-way coordinate reading in which any sharing must occur between all three conjuncts. See Wagner (2008) for some additional discussion on the differences between *x and y and z* and *x, y and z* constructions.

An additional empirical problem is that sharing in CP coordination constructions is predicted to be impossible. Activation of the E_{RNR} feature is dependent on a probe outside of the conjunction, located in the head of CP. However, in CP coordinate structures, there is simply no head outside of the coordination at which the probe could reside, yet all of the constructions in (51) are grammatical. Ha's account is therefore unable to derive any of these examples.

(51) a. Who rented ~~the movies~~ and who returned [the movies]?

b. What did Mark loan to ~~their neighbors~~ and what did Mary borrow from [their neighbors]?

c. Did Peter eat ~~the beans~~ or did she eat [the beans]?

Furthermore, because Ha derives left sharing from right sharing constructions, it is predicted that the originating position of the left shared element is at the right periphery of each conjunct. Ha claims that the contrast between examples (52) and (53) demonstrates the validity of this prediction. However, example (54) is perfectly acceptable, even though the origin of the left shared item is not at the right periphery. Thus, the Peripherality Constraint does not seem to affect those right-sharing constructions from which left-sharing is derived.

(52) *[$_{CP}$ [Who$_i$] did [$_{TP}$ John give ~~who~~ an apple] and [$_{TP}$ Bill give t$_i$ a pizza]]?

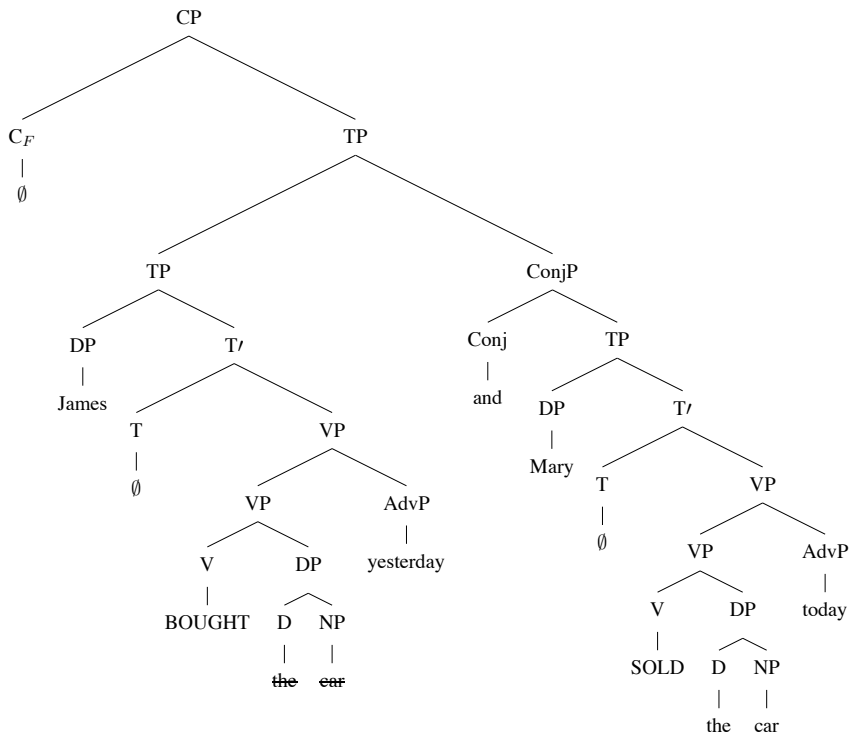
(53) [$_{CP}$ [How much money$_i$] did [$_{TP}$ John give Mary ~~how much money~~] and [$_{TP}$ Mary give CHRIS t$_i$]]?

(54) [$_{CP}$ [Who$_i$] did you [$_{VP}$ CALL ~~who~~ yesterday] and [$_{VP}$ SEE t$_i$ today]]?

Finally, it is worth noting that the Peripherality Constraint cannot be derived from this ellipsis analysis. Consider the example in (55a), where the shared element is not peripheral in the conjuncts. However, Ha's analysis is capable of deriving this construction, as depicted in (55b). The elements *BOUGHT* and *SOLD* are contrastively focused, and their sisters entail each other. The E_{RNR} feature in the left conjunct is activated by the probe F on the head of CP, licensing the phonological deletion of *the car* in the left conjunct. Nothing in this account ensures that the elided element be peripheral in the conjunct. Ha instead assumes that the Peripherality Constraint is independently derived.

(55) a. * James BOUGHT ~~the car~~ yesterday and Mary SOLD [the car] today.

b.



In sum, Ha's analysis does not provide a convincing argument for the conceptual and empirical validity of this reduction approach. The structural constraint provided by the E_{RNR} feature is unable to account for sharing in CP coordination, in constructions with three or more conjuncts or in cases of multiple left shared items. It also predicts that a left shared element must have a source at the right edge of the right conjunct.

2.3 Multiple Dominance

A third approach to sharing coordination constructions is multiple dominance, which posits that there is a single in-situ element with a mother in each conjunct. This contrasts with the literal movement approach in which the shared element is ex-situ, having been raised outside of the conjunction by ATB movement (Sabbagh (2007)). Furthermore, the multiple dominance approach is distinct from reduction accounts which posit two underlying copies of the shared element (Ha (2008)). The fact that a multiple dominance approach proposes a single underlying copy that remains in-situ proves to be advantageous for the analysis of sharing coordination constructions.

The formulation of an in-situ shared element is supported by a number of arguments. First,

no construction-specific mechanisms, such as ATB movement, are required to derive sharing coordination constructions. As will be discussed in section 2.3.2, multiple dominance has been independently motivated for the analysis of movement. Second, an ex-situ approach to right-sharing constructions such as (56a) is necessarily more complex, in that it must explain why rightward ATB movement can violate the island constraints. The grammaticality of (56a) follows in an in-situ account because the right-shared element does not move out of the island. Furthermore, as discussed in section 2.1, the interaction of VP ellipsis and right-sharing coordination is problematic for ex-situ analyses, which incorrectly predict that a VP may be elided to the exclusion of a right-shared element. For an in-situ approach, however, examples such as (56b) are expected to be ungrammatical. The right-shared item does not leave the VP, thus it must be elided as part of the VP constituent. Finally, in an example such as (56c), an ex-situ account posits that the right-shared complement clause raises and c-commands the conjunction. It therefore incorrectly predicts that the R-expression *Mary*, which is contained in the complement clause, can bind the pronoun *She* within the conjunction (Levine (1985)). On the other hand, in an in-situ approach, the R-expression does not raise and is c-commanded by the pronoun, rendering the sentence ungrammatical.

(56) a. Did he$_j$ meet [the customer]$_i$ who bought and did Mary meet [the salesman]$_k$ who sold the car?

b. * Jane talked about and Frank didn't the achievements of the syntax students.

c. * She$_i$ said and I happen to agree that Mary$_i$ needs a new car

The above discussion highlights the limitations of ex-situ analyses such as literal movement; however, this leaves two in-situ approaches (reduction and multiple dominance) to be considered. While reduction approaches utilize two underlying copies of the shared element, there are two advantages to proposing only a single underlying copy. First, it foregoes the need to propose licensing requirements on the reduction mechanism, thereby avoiding the conceptual and empirical limitations reviewed in section 2.2. In a multiple dominance account, the existence of a single, phonologically overt shared item directly follows from the presence of a single underlying element. Second, in a reduction approach, an example such as (57a) would be derived from (57b), predicting that these sentences share the same interpretation. However, this prediction is not borne out, as the distributive reading, which is available for (57a), is unobtainable for (57b). This interpretational contrast is expected in a multiple dominance account, in that (57a) and (57b) are not derived from each other.

(57) a. Peter likes and Mary hates similar paintings.

b. * Peter likes similar paintings and Mary hates similar paintings. (On distributive reading)

Early multiple dominance analyses, such as McCawley (1982) and Goodall (1983), recognized the advantage of utilizing this approach, but failed to address the principal problem for multiple dominance: the linearization of such structures can no longer be assumed to follow from a simple linear ordering between terminals. The shared element is in both conjuncts at once; thus, it is not clear at which conjunct it should be pronounced. In this section, I review two approaches to the linearization of multiple dominance structures and demonstrate that neither of them can fully account for the data.

2.3.1 Wilder and Complete Dominance

Wilder (1999) was one of the first analyses to address the linearization problem. Wilder adapted the Linear Correspondence Axiom (LCA) of Kayne (1994) to accommodate and linearize multiple dominance structures, claiming that the Peripherality Constraint follows from the algorithm. Additionally, Wilder adopted the view that linearizability is a constraint on grammaticality, in that constructions that cannot be linearized are considered ungrammatical.

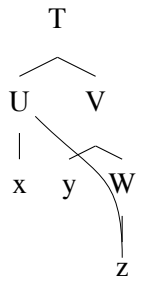
The core of Wilder's linearization algorithm is a modified version of the Kayne's (1994) LCA, which states that asymmetric c-command determines the linear order of terminals, such that for any pair of nodes X and Y, if X asymmetrically c-commands Y, then all the terminals dominated by X precede those dominated by Y. In a typical phrase containing a specifier, head and complement, the specifier will asymmetrically c-command the head, and the head will asymmetrically c-command the complement. Consequently, the terminals of the specifier precede the head, which in turn precedes the terminals of the complement.

Multiple dominance structures cannot be linearized by this algorithm because the multiply dominated element is linearized to precede itself. In structure (58), the node z is multiply dominated by U and W. U asymmetrically c-commands W, thus the terminals dominated by U, $\{\mathrm{x}, \mathrm{z}\}$, precede those dominated by W, $\{\mathrm{z}\}$. This linearization produces a reflexivity violation, as z precedes itself.

Asymmetric c-command Relations	Terminal Precedence
U > y, W, z	x > y, z
y > z	y > z
Final Linearization:	x > y > z

Table 2.1: Linearization for example (58)

(58)



Wilder adapts the LCA for use with multiple dominance structures by redefining the algorithm to refer to the notion of full dominance instead of dominance. For any pair of nodes X and Y, if X asymmetrically c-commands Y, then the terminals *fully dominated* by X precede those *fully dominated* by Y. Wilder defines full dominance in terms of node sharing, such that a node α is shared by X and Y iff neither of X and Y dominates the other, and both X and Y dominate α. Accordingly, X fully dominates α iff X dominates α, and X does not share α. Wilder also modifies the definition of c-command such that a node X may now c-command a node Y even if X dominates Y, provided that X does not fully dominate Y[6]. For instance, in example (58), *U* c-commands *z*, despite dominating it.

Wilder's algorithm returns a linear ordering of all the nodes in (58), as illustrated in Figure 2.1. The left column delineates the tree's asymmetric c-command relations, and the right column includes the corresponding terminal precedence relations. The final linearization, provided in the bottom row of the table, is the union of these individual precedence relations; the final ordering for this example is x > y > z.

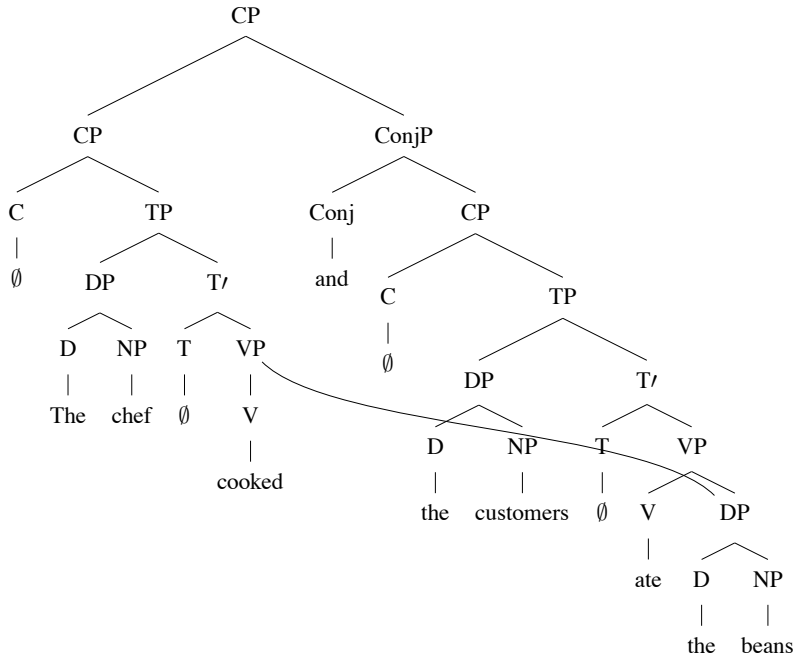
Wilder uses examples such as (59a) to illustrate how his algorithm can successfully derive right sharing constructions. This example contains a shared object *the beans*, and the Table 2.2 illustrates the linearization process. The asymmetric c-command relations in the left conjunct, illustrated in (59b), place *the beans* after *the chef* and *cooked*, and the relations in the right conjunct also place *the*

[6]Wilder also assumes, following Kayne (1994), that bar-level projections do not participate in c-command relations.

beans after the other elements in the conjunct. The fact that the algorithm linearizes fully-dominated terminals is critical when considering the c-command relations between the left CP and the right conjunct. The left CP dominates the shared object, which is also dominated in the right conjunct. The proviso that it is only the fully dominated terminals that are linearized prevents the object from being linearized before itself. The union of the resulting precedence relations places the shared object in the right conjunct.

(59) a. The chef cooked and the customers ate the beans.

b.



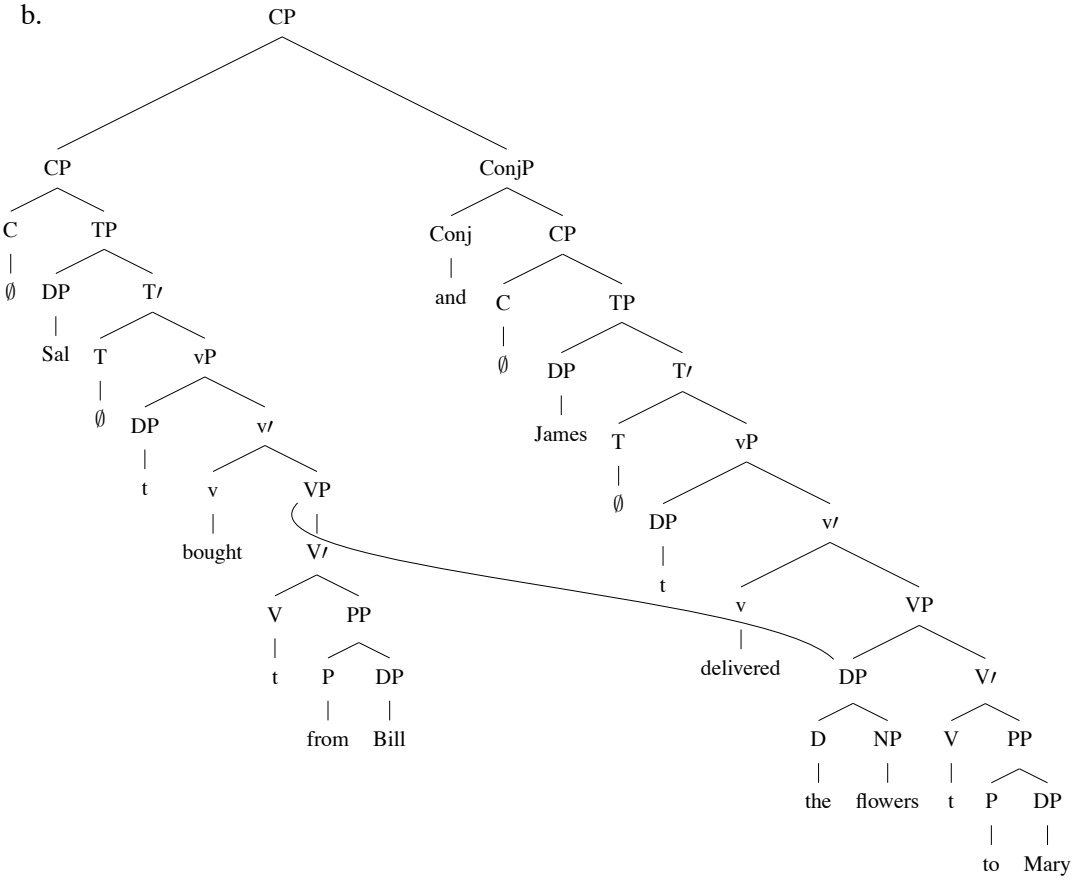
On the basis of examples such as (60a), Wilder also argues that the right Peripherality Constraint follows from the formulation of the linearization algorithm. The direct object *the flowers* is shared and is not right peripheral in either conjunct. This configuration, in (60b) results in the object being linearized in the middle of each conjunct, indicated in Table 2.3. The object follows both verbs, being asymmetrically c-commanded by each v, and it precedes each indirect object, which it asymmetrically c-commands. This constitutes a reflexivity violation.

(60) a. * Sal bought from Bill and James delivered the flowers to Mary.

Asymmetric c-command Relation	Terminal Precedence
CP > ConjP, Conj, CP, C, TP, DP, D, NP, T′, T, VP, V, DP, D, NP	the, chef, cooked > and, the, customers, ate, the, beans
C > TP, DP, D, NP, T′, T, VP, V, DP, D, NP,	∅ > the, chef, cooked, the, beans
DP > T′, T, VP, V, DP, D, NP,	The, chef > cooked, the, beans
D > NP	the > chef
T > VP, V, DP	∅ > cooked, the, beans
V > DP, D, NP,	cooked > the, beans
Conj > CP, C, TP, DP, D, NP, T′, T, VP, V, DP, D, NP	and > the, customers, ate, the, beans
C > TP, DP, D, NP, T′, T, VP, V, DP, D, NP,	∅ > the, customers, ate, the, beans
DP > T′, T, VP, V, DP, D, NP,	the, customers > ate, the, beans
D > NP	the > customers
T > VP, V, DP, D, NP,	∅ > ate, the, beans
V > DP, D, NP,	ate > the, beans
D > NP	the > beans
Final Linearization:	The chef cooked and the customers ate the beans

Table 2.2: Linearization for example (59)

b.



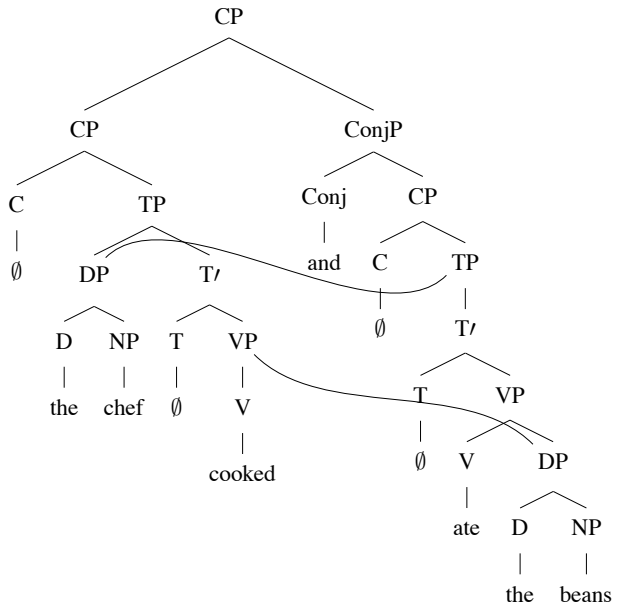
However, there are two problems with Wilder's linearization algorithm: it cannot linearize certain well-formed right-sharing or any left-sharing constructions. Wilder assumes that left-sharing constructions, such as that in (61a), with the structure in (61b), are due to multiple dominance, yet they cannot be linearized by this algorithm. As illustrated by the asymmetric c-command relations given in Table 2.4, the terminals fully dominated by the Conj node precede those fully dominated by all of the nodes asymmetrically c-commanded by Conj, including D and NP. Thus, *and* must precede *the* and *chef*. Contradictorily, *the* and *chef* must also precede *and* by virtue of the asymmetric c-command relations in the left conjunct. The DP asymmetrically c-commands V, which means that *the* and *chef* precede *cooked*, while the left CP asymmetrically c-commands Conj, which means that *cooked* precedes *and*. By transitivity, *the* and *chef* must precede *cooked* and *and*, as well as follow them. In following Kayne (1994), Wilder assumes that structures which produce reflexivity violations are ungrammatical, yet (61a) is considered fully grammatical.

(61) a. The chef cooked and ate the beans.

Asymmetric c-command Relation	Terminal Precedence
CP > Conj, CP, C, TP, DP, T′, T, vP, v, VP, DP, D, NP, V′, V, PP, P, DP	Sal, bought, from Bill > and, Bill, delivered, to Mary
C > TP, DP, T′, T, vP, v, VP, DP, D, NP, V′, V, PP, P, DP	∅ > Sal, bought, from, Bill
DP > T′, T, vP, v, VP, DP, D, NP, V′, V, PP, P, DP	Sal > bought, from, Bill
T > vP, v, VP, DP, D, NP, V′, V, PP, P, DP	∅ > bought, from, Bill
v > VP, DP, D, NP, V′, V, PP, P, DP	bought > from, Bill
DP > V′, V, PP, P, DP	the, flowers > from, Bill
D > NP	the > flowers
V > PP	∅ > from, Bill
P > DP	from > Bill
Conj > CP	and > James, delivered, to Mary
C > TP, DP, T′, T, vP, v, VP, DP, D, NP, V′, V, PP, P, DP	∅ > James, delivered, to, Mary
DP > T′, T, vP, v, VP, DP, D, NP, V′, V, PP, P, DP	James > delivered, to, Mary
T > vP, v, VP, DP, D, NP, V′, V, PP, P, DP	∅ > delivered, to, Mary
v > VP, DP, D, NP, V′, V, PP, P, DP	delivered > to, Mary
DP > V′, V, PP, P, DP	the, flowers > to, Mary
V > PP	∅ > to, Mary
P > DP	to > Mary
Final Linearization:	Sal bought (the flowers) from Bill and James delivered (the flowers) to Mary

Table 2.3: Linearization for example (60a)

b.

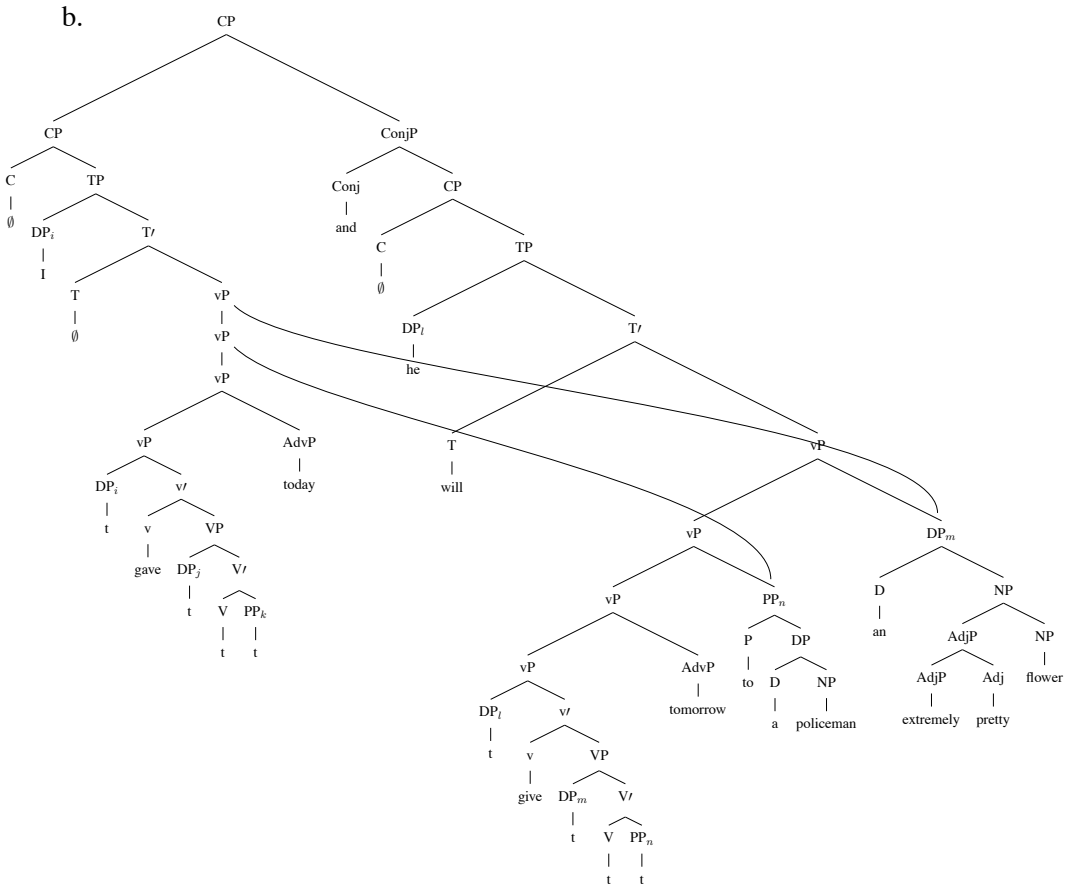


Additionally, right-sharing constructions such as that in (62a) are grammatical, yet they cannot be linearized by Wilder's algorithm. In this example, the shared elements *to a policeman* and *an extremely pretty flower* are right-adjoined to vP in each conjunct, shown in (62b). I assume that the vPs asymmetrically c-command these right-adjoined elements. Thus, the terminals fully dominated by the vP containing *to a policeman* precede those fully dominated by the object DP *an extremely pretty flower*. However, the terminals *to a policeman* are not fully dominated by this vP and are thus not linearized with respect to the terminals *an extremely pretty flower*. Consequently, the final linear ordering is incomplete, and the structure is unlinearizable by Wilder's algorithm.

(62) a. I gave today and he will give tomorrow to a policeman an extremely pretty flower. (Steedman (1996))

Asymmetric c-command Relation	Terminal Precedence
CP > ConjP, Conj, CP, C, TP, DP, D, NP, T′, T, VP, V, DP, D, NP	the, chef, cooked > and, the, chef, ate, the, beans
C > TP, DP, D, NP, T′, T, VP, V, DP, D, NP,	∅ > the, chef, cooked, the, beans
DP > T′, T, VP, V, DP, D, NP,	The, chef > cooked, the, beans
D > NP	the > chef
T > VP, V, DP	∅ > cooked, the, beans
V > DP, D, NP,	cooked > the, beans
Conj > CP, C, TP, DP, D, NP, T′, T, VP, V, DP, D, NP	and > the, chef, ate, the, beans
C > TP, DP, D, NP, T′, T, VP, V, DP, D, NP,	∅ > the, chef, ate, the, beans
DP > T′, T, VP, V, DP, D, NP,	the, chef > ate, the, beans
T > VP, V, DP, D, NP,	∅ > ate, the, beans
V > DP, D, NP,	ate > the, beans
D > NP	the > beans
Final Linearization:	(The chef) cooked and (the chef) ate the beans

Table 2.4: Linearization for example (61a)



Though Wilder's account is able to linearize the majority of right-sharing constructions, the inability to linearize every grammatical right-sharing construction is problematic. Furthermore, the algorithm cannot be extended to any cases of left-sharing, rendering it unsuitable for use in a multiple dominance account of both right and left-sharing coordination constructions. In the next subsection, I examine a different approach to linearization, in which the linearization of shared elements is determined by two types of multiple dominance.

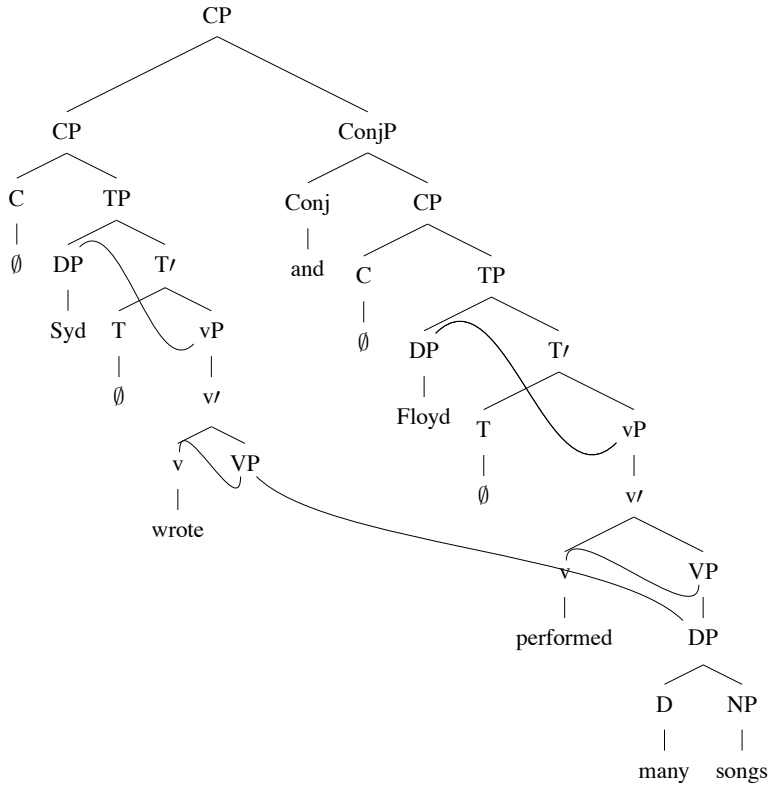
2.3.2 Vries and Internal and External Remerge

One downside of Wilder's approach is that the use of the multiple dominance mechanism is construction-specific, as it is introduced into the grammar for the sole purpose of analyzing sharing coordination constructions. However, Vries (2009) and Kluck (2007) address this issue by extending the domain of multiple dominance to all cases of movement. When an element "moves", there is actually no movement occurring. The "moved" item is multiply dominated, in that it obtains an additional

mother. Vries defines two mechanisms that produce multiple dominance structures: Internal and External Remerge. Remerge operates on a merged node by creating an additional immediate dominance relation, thereby "remerging" the node. Consequently, a remerged node has two mothers. If both mothers of the remerged element are in the same syntactic structure, it is considered to be a case of Internal Remerge. This type of remerge encodes "movement" relations, as exemplified by the subjects in (63a). In this example, the DPs *Syd* and *Floyd* are first merged at the SpecvP and are Internally Remerged at SpecTP. On the other hand, External Remerge occurs when the mothers are in separate syntactic structures at the time of remerge. In example (63a), the object DP *many songs* is merged as complement to VP in the left conjunct and externally remerged as complement to VP in the right conjunct, as in (63b). At the time of this operation, the conjuncts are separate syntactic structures, which are subsequently connected by the ConjP.

(63) a. Syd wrote and Floyd performed many songs.

b.



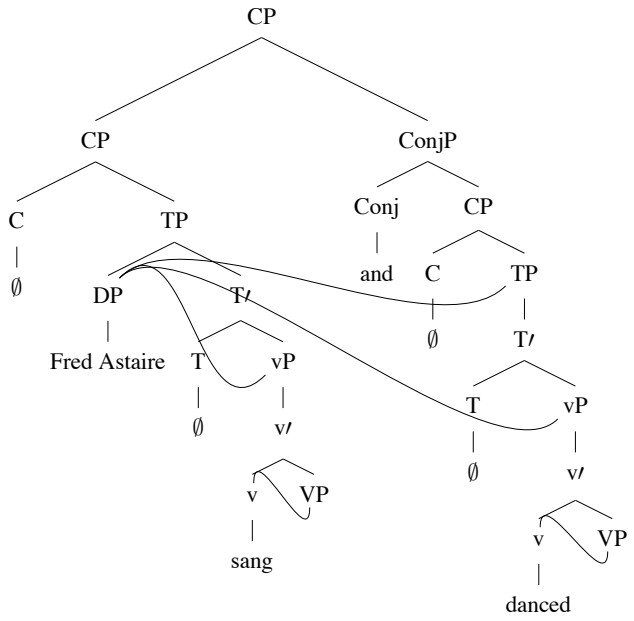
Vries (2009) and Kluck (2007) observe that the linearization requirements of External Remerge are distinct from those of Internal Remerge. If a node has been externally remerged, it is linearized in the rightmost position. For example, the object DP *many songs* in (63a) must be linearized to

the right of both *wrote* and *performed*. Nodes that have been either internally remerged, such as the subjects of (63a), or both internally and externally remerged are linearized at their highest merged position.

Both Internal and External Remerge are necessary for the derivation of left-sharing constructions. While right shared elements can often be derived exclusively with External Remerge, most left-shared elements also require Internal Remerge. In example (64a), *Fred Astaire* has been first externally remerged at both SpecvPs and is subsequently internally remerged at SpecTP, depicted in (64b). In contrast with example (63a), the left shared subject DP *Fred Astaire* has undergone both External and Internal Remerge. Consequently, Vries' algorithm linearizes the shared item at the highest merged position, SpecTP of the left conjunct.

(64) a. Fred Astaire sang and danced.

b.



Unlike Wilder (1999), Vries' and Kluck's approaches linearize every possible multiple dominance structure. Their algorithm searches for each remerged element and linearizes it appropriately. Thus, there are no constraints on the types of permissible structures. In Wilder's approach, not every structure can be linearized; linearizability is a condition for grammaticality. Wilder claims that the types of multiple dominance structures ruled out by this constraint correspond to violations of the Peripherality Constraint. In abandoning linearizability as a condition for grammaticality, Vries is forced to adopt an alternative explanation for the Peripherality Constraint. Vries and Kluck pro-

pose that the Peripherality Constraint follows from prosodic and semantic constraints on sharing coordination constructions.

Additionally, the algorithm makes incorrect predictions about the position at which externally remerged elements are pronounced. The algorithm linearizes these elements at their rightmost position; however, the sentences in (65) all contain externally remerged elements which are pronounced in the left position. The modals in (65a) and (65b) and the do-support in (65c) are externally remerged in the head of TP, while the negation in (65c) is externally remerged in the head of NegP[7]. The determiner in (65d) is merged at the head of DP, and the adjectival phrase in (65e) is adjoined to NP. None of these elements are commonly assumed to involve Internal Remerge; thus following Vries' algorithm, they are predicted to be pronounced at their rightmost position; however, as illustrated in (66), this results in ungrammaticality.

(65) a. I **could** call and visit.

b. She **might** prepare and present the paper.

c. The driver **did not** stop at any stop-sign or use his turn signal.

d. She had **a** boy and girl.

e. **Big** cars and trucks used to be quite popular.

(66) a. * I call and **could** visit.

b. * She prepare and **might** present the paper.

c. * The driver stop at any stop-sign or **did not** use his turn signal yet.

d. * She had boy and **a** girl.

e. # Cars and **big** trucks used to be quite popular. (Under reading where both cars and trucks are big.)

It seems that a simple mapping from remerge type to linear order fails to predict certain sharing constructions, such as those listed in (65). Furthermore, because the algorithm is based exclusively on remerge type, ignoring all structural information, there is no principled way to extend the algorithm to account for these types of cases.

[7]These examples could alternatively be derived by vP coordination and the merge of a single shared element; but this analysis is not available to Vries and Kluck, who assume CP coordination.

In this chapter, I examined three approaches to the sharing coordination construction: literal movement, reduction, and multiple dominance. I argued that the literal movement's ex-situ analysis of shared elements requires a construction-specific mechanism and makes incorrect predictions about scope interactions and VP ellipsis in these constructions. I reviewed the reduction approach and argued that equating sharing constructions to ellipsis phenomena is empirically and conceptually inadequate. Finally, I demonstrated that multiple dominance is well motivated; however, I also pointed out the outstanding problem that linearization poses. The linearization algorithms employed by two recent multiple dominance accounts either lack empirical coverage or fail to provide an explanatory basis for the linearization.

Chapter 3

Syntactic analysis using LTAG

This chapter is devoted to the development of a multiple dominance analysis of sharing coordination constructions in the framework of a Lexicalized Tree Adjoining Grammar (LTAG). LTAGs are mathematically explicit formalisms with well-known formal properties. In their application to natural languages, LTAGs have proven to be interesting frameworks for linguistic inquiry, having been applied to the analysis of syntactic problems such as scrambling, reflexives, clefts, and Wh-movement, as well as semantic problems such as scope and the denotation of questions (Becker et al. (1991); Frank (2006, 2008); Han and Hedberg (2008); Joshi et al. (2004); Ryant and Scheffler (2006); Storoshenko et al. (2008)).

The central intuition behind LTAG is that lexical items are associated with a finite syntactic structure, collectively known as elementary structures. Syntactic relations within the elementary tree are considered to be local, and compared to phrase structure grammar, the elementary tree provides LTAG with an extended domain of locality for syntactic dependencies. The operations which combine elementary trees allow all long-distance dependencies to be reduced to elementary tree-local dependencies. The principal advantages of using LTAG for linguistic analysis are that the domain of local syntactic dependencies can be stated explicitly, and that recursion and non-local dependencies are reduced to local dependencies.

Although LTAG-based linguistic theories have been successfully applied to provide insightful analyses for a variety of syntactic constructions, sharing coordination constructions pose a particular challenge to an LTAG analysis. While the LTAG operations that compose elementary trees can only involve the composition of two elementary trees, a multiple dominance analysis of sharing coordination constructions seems to require that three trees compose at a time. For example, a

multiple dominance analysis of a subject shared between two verbs would require that the single shared subject simultaneously compose with two verbal elementary trees. This sort of composition operation is not defined for standard LTAG. The incompatibility of standard LTAG and sharing constructions was first noted by Sarkar and Joshi (1996), who modified LTAG to allow for the derivation of multiple dominance structures. Their approach is able to derive several core cases of sharing coordination but is limited in empirical scope.

The analysis that I present in this chapter can be viewed as an evolution of the approach presented in Sarkar and Joshi (1996). I adapt their account to a modified version of the LTAG-based linguistic framework proposed by Frank (2002). This framework allows the mechanism which derives sharing coordination constructions to be restated in a simplified manner, and it facilitates the incorporation of recent work on the syntactic structure of coordination.

In section 3.1, I present the LTAG formalism, followed by a description of the linguistic framework of Frank (2002) in section 3.2. In 3.3, two revisions to this linguistic framework are proposed. First, the terminals of elementary trees are treated as bare heads to which an explicit lexical insertion operation adds lexical items. Second, elementary tree internal movement is recast as multiple dominance. With this framework established, I discuss the mechanism introduced by Sarkar and Joshi (1996) to derive sharing coordination constructions in 3.4. In 3.5, I incorporate aspects of their analysis into my proposed framework.

3.1 Lexicalized Tree Adjoining Grammar

LTAG is a lexicalized tree-rewriting system that can model syntactic competence (Joshi and Kulick (1997), Frank (2002)). The atomic units of syntactic composition are tree structures. Linguistic theories based on LTAG generally specify that one frontier node be a lexical item, which is designated the lexical anchor. The elementary tree consists of syntactic structure associated with this lexical anchor, which, at minimum, specifies the argument structure of the lexical anchor. Assuming such a structure to be a fundamental syntactic element allows the elementary trees to be encoded with the syntactic relations of their lexical anchors, such as number and person agreement and argument subcategorization. Elementary trees are therefore lexicalized and exhibit an extended domain of locality for the specification of syntactic relations. The central hypothesis of LTAG is that this extended domain of locality can capture all syntactic relations.

In contrast, Minimalist-style frameworks do not derive the domain of locality for syntactic re-

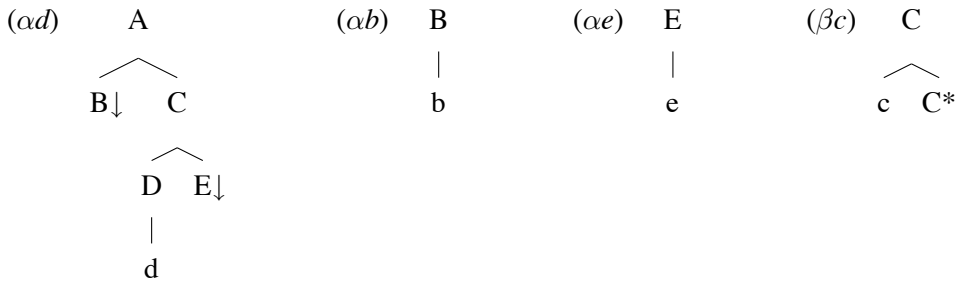


Figure 3.1: Elementary Trees

lations (e.g. movement, agreement, case assignment) from the formalism that these frameworks employ. Instead, these linguistic facts must either be stipulated or argued to follow from other aspects of the theory. The advantage for LTAG is that, if its central hypothesis is accurate, it will have captured a generalization about the relationship between the formalization of natural languages and the domain of syntactic locality, which cannot easily be captured in Minimalist-style frameworks.

3.1.1 Composition Operations: Substitution and Adjoining

The structures in Figure 3.1 are schematics of elementary trees. Elementary trees consist of a root node, one or more frontier nodes and labeled tree-internal nodes. The root node is the node that is not dominated by any other, and the frontier nodes are those along the bottom edge of the tree. Frontier nodes can also be called "leaves" and are classified as either terminal or non-terminal. Terminal frontier nodes are generally words, traces or empty heads; in this example, they are denoted with lower-case letters. There exist two types of non-terminal frontier nodes: substitution sites and foot nodes. Substitution sites are indicated by a downward arrow (↓), and foot nodes are indicated by an asterisk (*). The label of a foot node always matches the label of the root node. These non-terminal nodes are the sites at which elementary trees are composed, which will be discussed shortly.

Elementary trees are classified into one of two categories based on the presence or absence of a foot node. Initial trees are elementary trees which lack a foot node, while auxiliary trees possess them. Both initial and auxiliary trees may contain substitution sites. Elementary trees are named on the basis of type of tree and anchor. α indicates an initial-type tree, such as trees (αd), (αb) and (αe), and β indicates an auxiliary-type tree, such as (βc). The non-terminal frontier nodes $B\downarrow$ and $E\downarrow$ in (αd) are substitution sites. The non-terminal frontier node C^* in (βc) is a foot node.

Elementary trees are composed with one of two operations, substitution or adjoining. In the

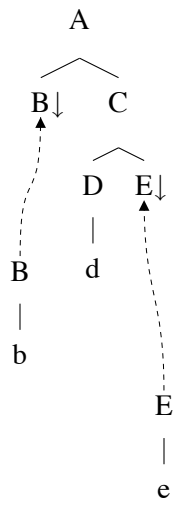


Figure 3.2: Substitution

substitution operation, the root node of an initial tree is identified with the substitution site of another tree. Identification in this context means that the two nodes are unified to become a single node as a result of the substitution. Thus, in Figure 3.2, the root nodes B of initial tree (αb) and E of initial tree (αe) are identified with the substitution sites B$\downarrow$ and E$\downarrow$ in (αd), respectively. The other operation, adjoining, splits an elementary tree at a non-terminal node into a top and bottom portion, leaving two copies of the split node. The root node of an auxiliary tree is identified with the top half of this split node, and the foot node of the auxiliary tree is identified with the bottom half of the split node. Thus when (βc) adjoins into (αd), the C node of (αd) is split into a top and bottom half, as depicted in Figure 3.3. The root node C of (βc) is identified with this top half, while its foot node is identified with the bottom half. The split elementary tree is then reconstructed, with an auxiliary tree inserted into its interior, as in Figure 3.4.

LTAG compositions produce two trees: a derived tree and a derivation tree. The derived tree is a phrase-structure tree resulting from the composition of elementary trees by adjoining and substitution. The derivation tree is a history of which elementary trees were composed and the manner of their composition. The derived tree in Figure 3.4 is the result of the composition of the elementary trees of Figure 3.1 as recorded in the derivation tree of Figure 3.5. In a derivation tree, for a given mother-daughter node pair, the daughter has adjoined or substituted into the mother[1]. The linear

[1]Typically, a label on the line between the pair indicates at which node the daughter has combined with the mother. These labels have been omitted for simplicity.

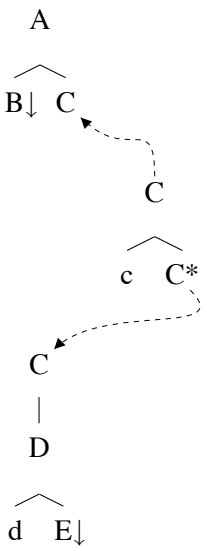


Figure 3.3: Adjoining

order of the daughters in the derivation tree is not significant.

3.2 The LTAG Linguistic Framework

The LTAG-based linguistic framework proposed by Frank (2002) defines a set of conditions on the well-formedness of elementary trees. Frank posits that elementary trees are derived from an enumeration, using the Minimalist-style operations Move and Merge. The elementary trees are limited in size to the maximal projection of the lexical anchor and are thus typically bounded to the CP,

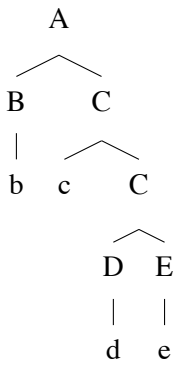


Figure 3.4: Derived Tree "*b c d e*"

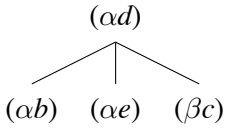


Figure 3.5: Derivation Tree "*b c d e*"

DP or PP projections. The incorporation of a Move and Merge-based derivation of elementary trees provides a clear rationale for the importation of many assumptions about syntactic structure from Minimalism. For example, elementary trees are assumed to involve binary branching structures. Additionally, theta roles are assigned within the vP, and movement is motivated by the feature checking requirements.

In the remainder of this section, I will discuss Frank's well-formedness constraints on elementary trees, referring to the trees in Figure 3.6. These trees will be used to derive the example given in (67), thereby demonstrating that the apparently long-distance dependency between the fronted Wh-word *who* and the embedded verb *like* is reduced to a local dependency within the (α*like*) elementary tree.

(67) Who did Peter say you like?

Every elementary tree contains a lexical anchor, which is a lexical class word, such as a noun, adjective, verb or adverb. The other terminal nodes include determiners, auxiliaries, modals and complementizers, which head their associated functional projections. For example, the auxiliary tree (β*did_say*) contains the do-support auxiliary *did*. The tree consists of the extended projection of this lexical anchor, which includes its set of hierarchical lexical and functional projections (Grimshaw (2000)). For instance, the extended projection of nouns includes the DP and PP, and that of verbs includes the TP and CP. Thus, the elementary tree (α*like*) in Figure 3.6 projects to the CP projection, and the (α*Peter*) tree projects to DP. It is not necessary for an elementary tree to extend to the highest possible projection; the auxiliary tree (β*did_say*) only projects to C′ for example. The maximal projection thus constitutes an upper-bound on the size of an elementary tree.

In addition to the terminal frontier nodes, an elementary tree may contain a number of non-terminal frontier nodes, which are either substitution sites or foot nodes. Substitution sites are placeholders for arguments; if an elementary tree has a DP substitution site, the lexical anchor selects for a DP argument at that node. For instance, the tree (α*like*) selects for a DP argument in the specifier of CP and another DP argument in the specifier of TP. The substitution site is filled

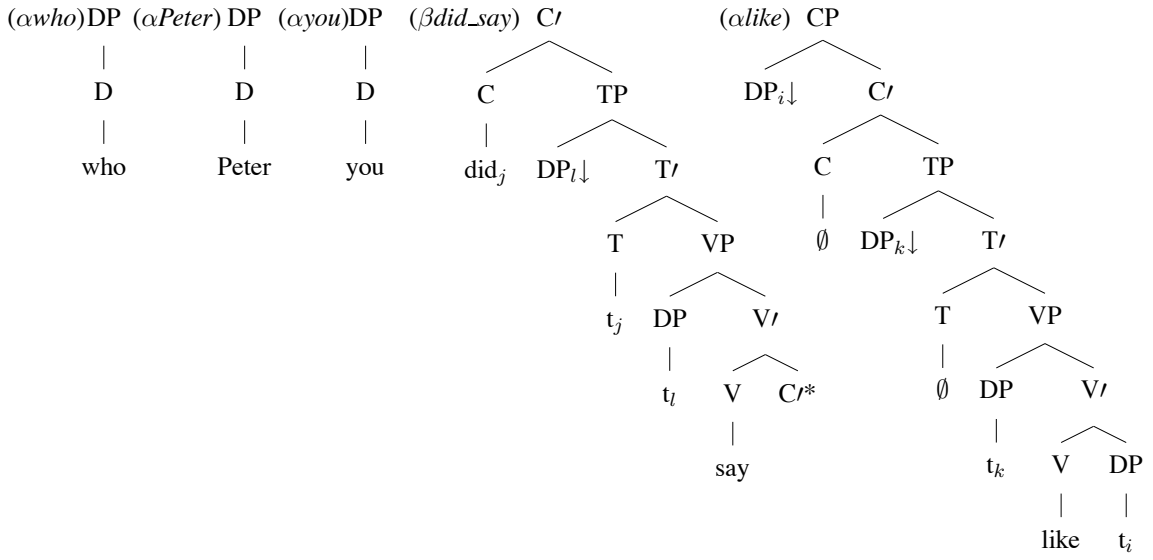


Figure 3.6: Elementary trees for *"Who did Peter say you like?"*

when an appropriate initial tree is substituted at this node. All substitution sites in an elementary tree must be assigned theta-roles by the lexical anchor. Similarly, the foot node of an auxiliary tree must conform to the subcategorizational requirements of the lexical anchor. For example, (*βdid_say*) selects a C′ node as a complement.

Frank assumes that the lexical and functional projections of an elementary tree are generated from an enumerated set of terminal and non-terminal nodes, referred to as the enumeration, through the application of the feature-driven Minimalist-style operations Move and Merge (Chomsky (1995)). The nodes that compose the elementary tree, including substitution sites, root and foot nodes, are built into the tree to satisfy feature requirements.

Operations on single elementary trees, such as Move and Merge, are called lexical operations, which may involve the addition, re-ordering or removal of material from a single elementary tree. A lexical operation may not function on two or more elementary trees simultaneously. In contrast, syntactic operations, such as substitution and adjoining, must operate simultaneously on two elementary trees.

This definition localizes syntactic dependencies to a single elementary tree. Examples such as (67) exemplify long-distance Wh-movement and can be given an analysis where the Wh-movement is localized to an elementary tree (Frank (2006)). In Minimalist approaches (e.g. Richards (1997)), "*who*" is merged as the complement of "*like*" and is subsequently cyclically moved through the

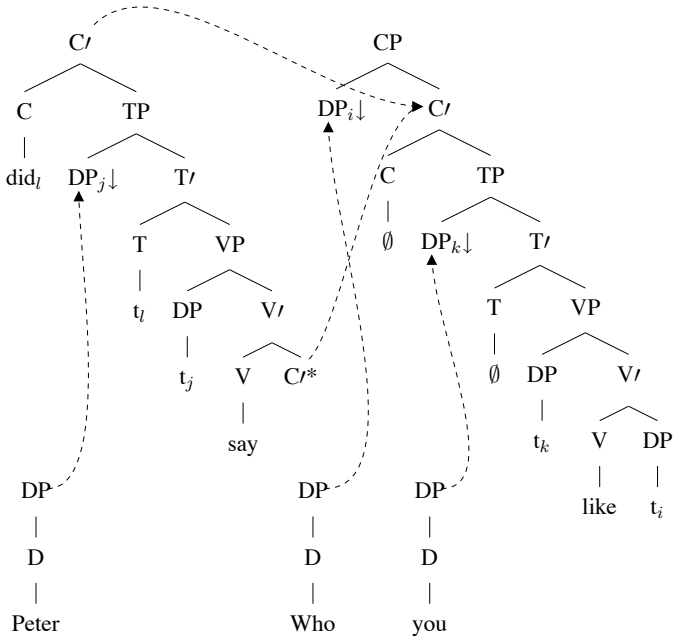


Figure 3.7: Deriving "*Who did Peter say you like?*" from the Elementary Trees in Figure 3.6

specifier of the embedded CP to the specifier of the matrix CP. In LTAG, cyclic movement and the accompanying constraints are unnecessary to derive such examples.

The elementary trees in Figure 3.6, combined using substitution and adjoining, are sufficient to generate (67). (α*like*) includes the substitution sites for the embedded subject and the Wh-moved object. The object has moved from the complement of V to the specifier of CP, and this movement is clearly local to the elementary tree. The appearance of long-distance movement and the corresponding word order changes are the result of adjoining. (β*did_say*) is an auxiliary tree that is recursive on C′; it adjoins to (α*like*) at the C′ node, stretching the distance between the Wh-moved object and its trace. The individual operations that derive this example are illustrated in Figure 3.7, and the derived and derivation trees are given in Figure 3.8.

Frank (2002) considers LTAG's fundamental hypothesis to be the localization of all syntactic dependencies to the elementary tree. The conditions on the well-formedness of elementary trees, combined with the operations substitution and adjoining, predict that all syntactic dependencies can be reduced to tree-local relations. In the next subsection, I propose two revisions to Frank's conditions on elementary tree well-formedness, in order to expand the empirical coverage of sharing coordination constructions and of multiple dominance in general.

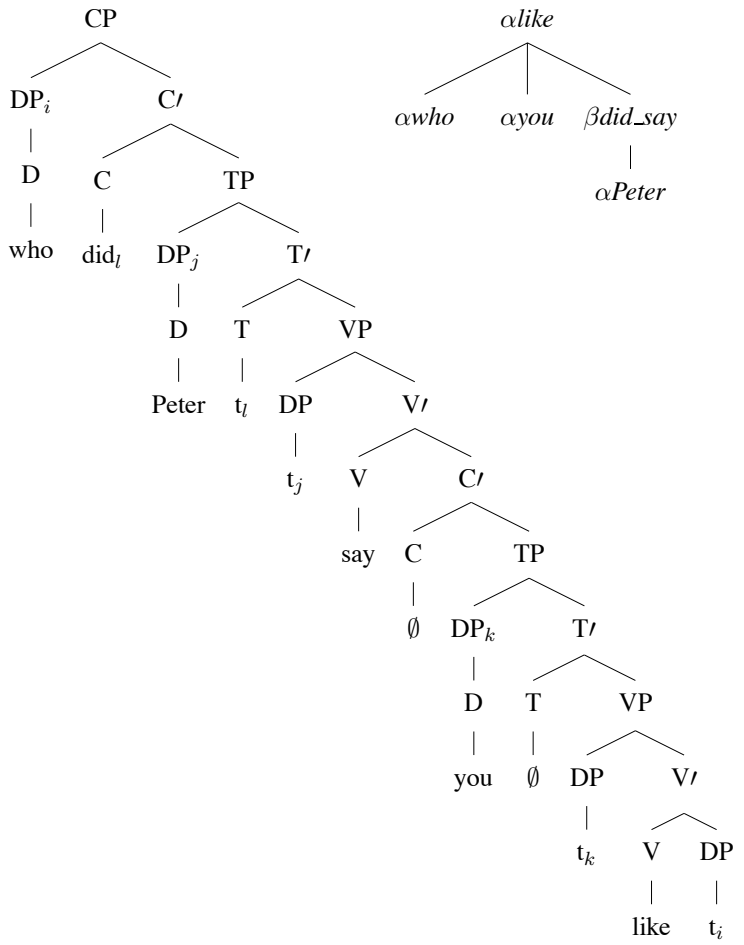


Figure 3.8: Derived and Derivation Trees for "*Who did Peter say you like?*"

3.3 Revised Conditions on Elementary Tree Well-formedness

The current account follows the framework of Frank (2002) with two exceptions. The first change involves the adoption of a multiple dominance theory of movement, as first discussed in section 2.3.2, in which the appearance of movement results from the derivation of elementary tree-internal multiple dominance structures. Elements that appear to have been moved do not leave copies or traces in previously occupied positions because the moved element has not actually been dislocated. The element remains in-situ, and an additional immediate dominance relation is added to it; this process is termed Remerge (Vries (2009)). As a result of this operation, remerged elements are multiply dominated. Incorporating this notion of Remerge extends the empirical domain of multiple dominance beyond the relatively uncommon sharing coordination constructions to the pervasive mechanism of movement. The ability to describe the properties of both common and uncommon structures strengthens the case for the existence of multiple dominance as a structural configuration.

The second change involves a shift in the conception of the lexicalization of elementary trees. Instead of being filled with simple lexical items, the terminal frontier nodes of elementary trees are actually heads devoid of lexical items. An operation, lexical insertion, inserts lexical material into these heads, similar to the Distributed Morphology model of grammar proposed by Halle and Marantz (1993)[2]. The adoption of lexical insertion in this thesis is motivated by the fact that it will permit the derivation of constructions in which a lexical item is shared between conjuncts, as in example (68). In section 3.3.2, I define the operation of lexical insertion, and in section 4.3, I describe in further detail the derivation of examples like (68).

(68) Peter likes the big and Mary likes the small dog.

3.3.1 Movement as Multiple Dominance

There have been several conceptions of movement utilized for syntactic theories, including trace theory (e.g. Chomsky (1981); Frank (2002)), copy theory (e.g. Chomsky (1995)) and multiple

[2]The Distributed Morphology research program has produced a number of interesting results, as discussed in Harley and Noyer (1999). The adoption of a lexical insertion operation has a number of potential benefits. One would be the possibility of porting over the results of the Distributed Morphology program to an LTAG framework. Another is the possibility of radically reducing the number of elementary trees necessary to describe the grammar of a language, by allowing various lexical items to insert into standard elementary tree templates. These avenues are interesting for both linguistic and computational research using LTAG but are well beyond the scope of this thesis.

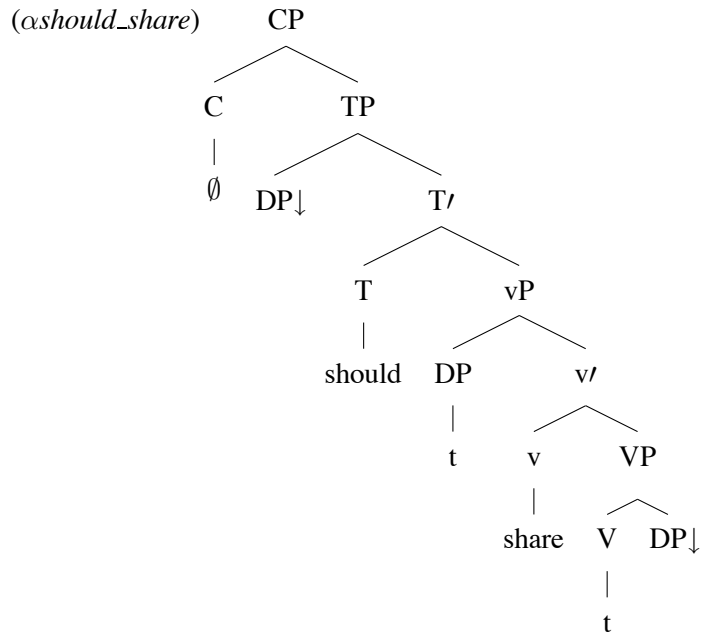


Figure 3.9: Trace Theory of Movement: Verbal Elementary Tree (α*should_share*)

dominance (e.g. Kluck (2007), Vries (2009)).

Frank (2002) adopts the trace theory of movement, in which moved elements are depicted at the position at which they last arrived, and previously occupied positions are filled with a trace. For instance, in Figure 3.9, the subject DP substitution site was merged at the specifier of vP, where it received a theta-role, and was subsequently moved to the specifier of TP to satisfy an EPP feature on T. Similarly, the verb *share* was merged at V but has raised to occupy v, leaving a trace in V. This analysis follows the theory of movement developed in Chomsky (1981). A second, widely-adopted theory is the copy theory of movement, detailed in Chomsky (1995). In this theory, moved elements leave copies of themselves in-situ, instead of traces.

A third option for the formal depiction of movement is multiple dominance. This approach, adopted by Gärtner (1997), Kluck (2007), Vries (2009) and Starke (2001), treats a moved element as being multiply dominated. As formalized by Vries (2009) and as discussed in section 2.3, movement is recast as *Remerge*, an operation which selects nodes and adds an additional immediate dominance relation on the selected element. A remerged element is dominated by its original mother, as well as by a second mother, acquired as a consequence of Remerge.

Figure 3.10 illustrates the elementary tree (α*should_share*) reformulated with movement de-

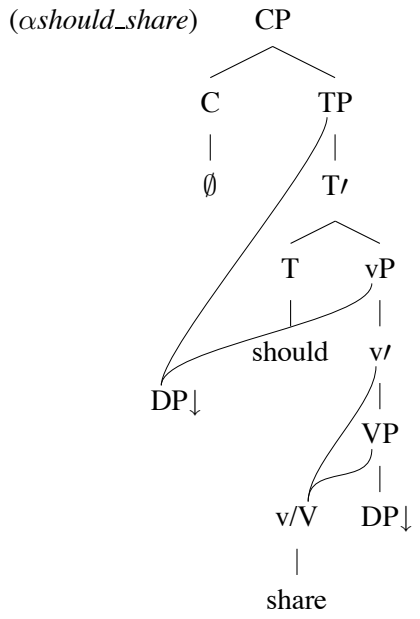


Figure 3.10: Multiple Dominance Theory of Movement: Verbal Elementary Structure (α*should_share*)

picted as multiple dominance. As a result of adopting multiple dominance, these structures are not trees, but directed graphs. For simplicity, I will refer to both trees and graphs as *structures*. The subject DP substitution site is multiply dominated by vP and TP. I assume that the DP substitution site receives a theta-role at specifier of vP and satisfies the EPP features at the specifier of TP. The verb *share*, as head of VP, assigns a theta-role to the object DP substitution site and, as head of vP, assigns a theta-role to the subject DP substitution site. I assume that the category of multiply-dominated heads is simply the combination of positions which the head occupies. In Figure 3.10, the verb is head of vP and VP, resulting in the category label v/V.

The adoption of a multiple dominance theory of movement is conceptually attractive. If multiple dominance is, as I have argued, necessary for an analysis of sharing coordination constructions, then a multiple dominance theory of movement extends the empirical domain of the multiple dominance analysis. This provides independent support for the existence of this type of derivational mechanism.

Additionally, a multiple dominance theory of movement can be effectively integrated into Frank's LTAG-based linguistic framework without disrupting its central hypothesis, whereby syntactic dependencies are localized to individual elementary structures. In section 3.2, I demonstrated that in sentences like that in (69), the relationship between the Wh-word *who* and the verb *like* is reduced

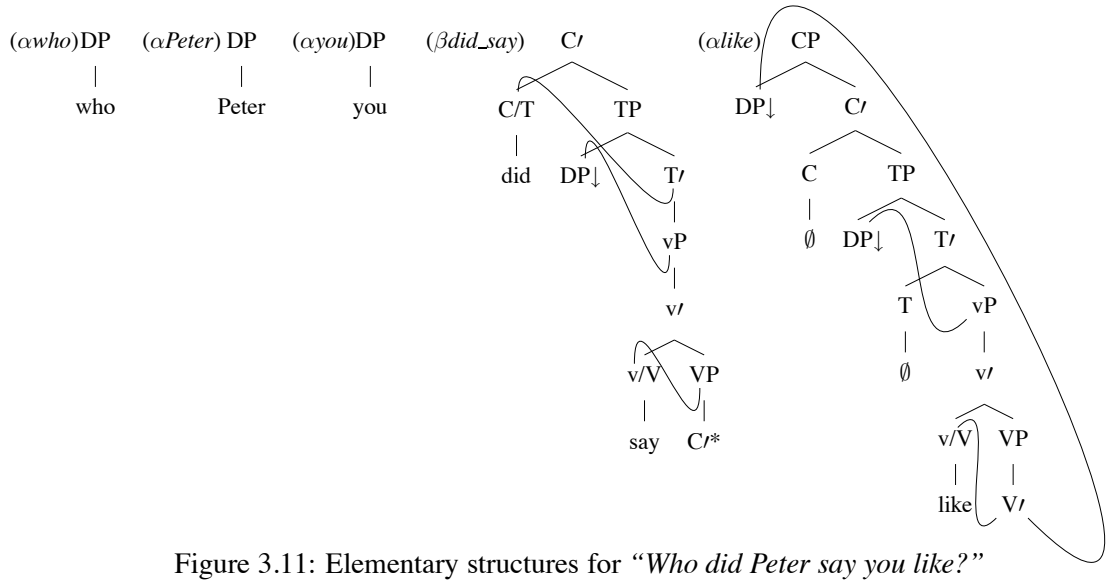


Figure 3.11: Elementary structures for *"Who did Peter say you like?"*

to an elementary structure-local relationship. As demonstrated in the following paragraphs, the adoption of a multiple dominance theory of movement maintains this structure-local relationship.

The elementary structures in Figure 3.11, where movement has been represented as multiple dominance, are the structures necessary to derive example (69). The lexical item *did* is multiply dominated as the head of T and C in the elementary structure (β*did_say*). The lexical anchors *say* and *like* are multiply dominated as the heads of vP and VP in their respective elementary structures. Similarly, the subject DP substitution site in (β*did_say*) is dominated by both vP and TP, and the DP subject and object substitution sites in (α*like*) are also multiply dominated.

The derivation and derived structures are given in Figure 3.12. The auxiliary elementary structure (β*did_say*) has adjoined to the initial verbal structure (α*like*) at the C′ node, and the Wh-object DP initial structure (α*who*) has substituted into (α*like*) at the object substitution site. While the derivation structure reflects the multiple dominance of the Wh-object DP, this is not indicated in the derivation structure because the Wh-object DP has simply substituted into a single substitution site in the (α*like*) structure. Multiple dominance within an elementary structure is derived during its construction.

(69) Who did Peter say you like?

The adoption of multiple dominance to replace movement operations also accentuates the need for an accurate and robust linearization mechanism. As discussed in section 2.3, to the best of my

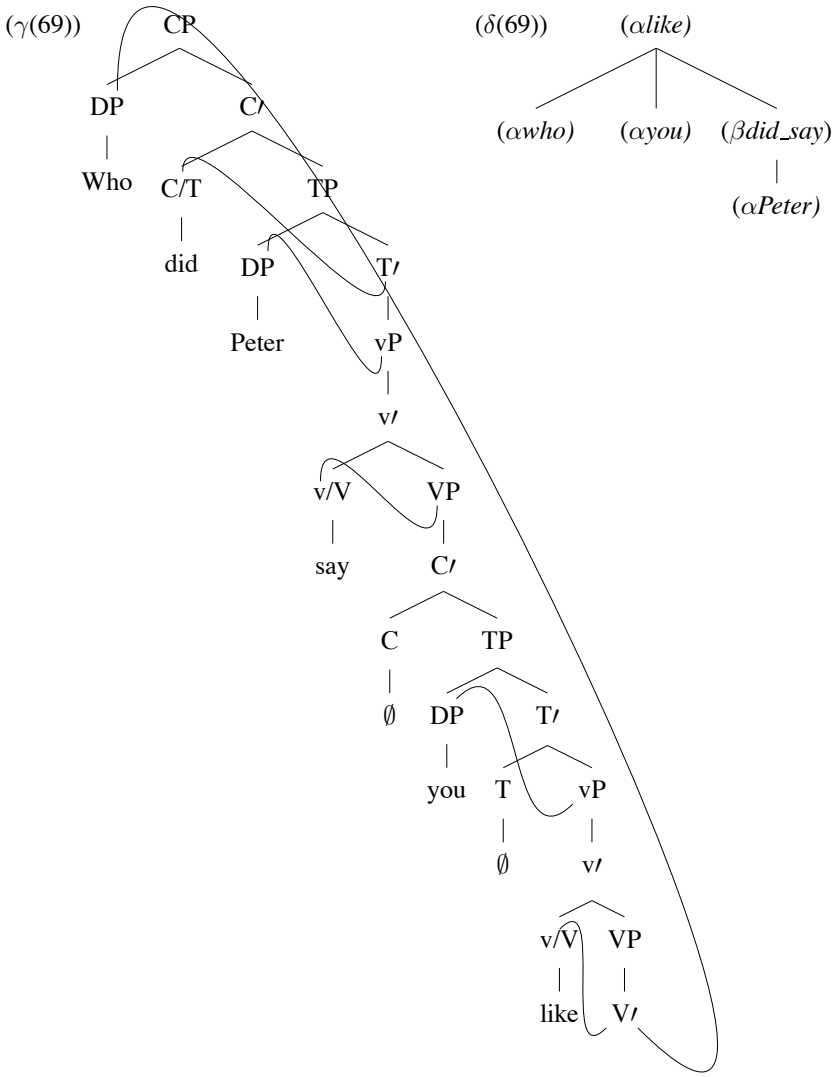


Figure 3.12: Derived and Derivation structures for "*Who did Peter say you like?*"

knowledge, no currently proposed linearization algorithm can correctly linearize all cases of multiple dominance structures. Thus, if one assumes movement as multiple dominance, linearization poses a problem not just to the relatively uncommon sharing coordination constructions, but also to any construction that would otherwise contain movement. In Chapter 5, I address this issue by presenting a linearization algorithm that will linearize the complete set of grammatical multiple dominance structures.

3.3.2 Lexical Insertion

The second departure from the framework of Frank (2002) concerns lexical insertion. Specifically, I postulate that there is a lexical insertion operation that introduces lexical items into elementary structures at lexical insertion sites after the LTAG syntactic operations.

This notion of lexical insertion is not new to LTAG or to Generative Grammar in general. Halle and Marantz (1993) propose a model of grammar in which syntactic operations function on bundles of features to which phonological material is added at a morphological stage of the derivation. Sarkar and Joshi (1996) also have a notion of lexical insertion in which elementary structures are treated as templates lacking lexical anchors. Sarkar and Joshi applied this notion to an analysis of gapping, where a verbal lexical insertion site was targeted by the conjoin operation, allowing a single lexical anchor to be shared between two elementary structures. Though I do not treat gapping as an instance of sharing[3], the approach to lexical insertion that I pursue here can be seen as a reformulation and extension of the analysis proposed by Sarkar and Joshi.

The principal change to LTAG that is necessary to accommodate a distinct lexical insertion operation involves the composition of the elementary structures. Previously, an elementary structure contained a single lexical anchor (e.g. a noun, verb or adjective) and potentially several functional heads within the maximal projection of the lexical anchor. In the present approach, each head is a lexical insertion site, at which a lexical item will be introduced. The head does not dominate any material; it is a frontier node. I also assume that heads are terminal nodes, as the lexical insertion operation is not a syntactic operation; that is, it does not compose two LTAG elementary structures.

Figure 3.13 contains examples of LTAG elementary structures as defined in this analysis; these structures will be used to derive the sentence in example (70). The type of elementary structure is still specified in its name: α denotes an initial structure, and β indicates an auxiliary structure.

[3]See Chapter 1 for arguments to this effect.

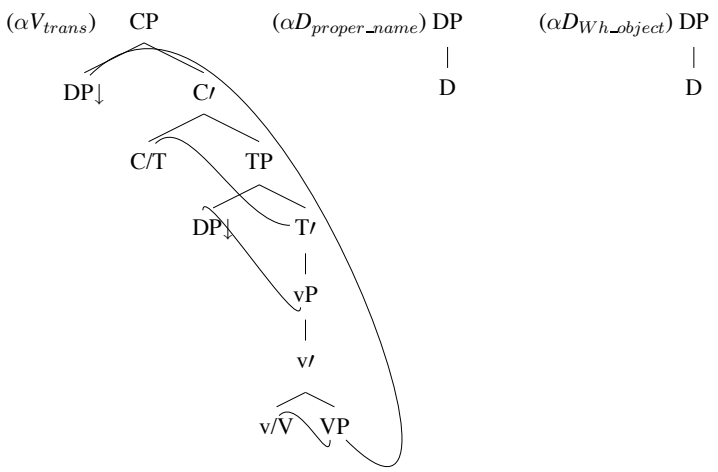


Figure 3.13: LTAG elementary structures necessary to derive example (70): "What did Peter buy?"

However, the names no longer include the lexical anchor, as the elementary structures do not contain lexical anchors. Instead, the names include the category of the anchor lexical insertion site, subscripted with some information about the structure's syntactic role or type. For example, (αV_{trans}) is an initial structure anchored by a transitive verb, and (αD_{proper_name}) is an initial structure anchored by a proper name.[4] The derived and derivation syntactic structures produced from these elementary structures are included in Figure 3.14. Removing the lexical items from elementary structures has not changed the syntactic derivation, it simply adds an additional step.

(70) What did Peter buy?

Elementary structures are constrained by a modified version of Frank's (2002) conditions on elementary structure minimality. Each elementary structure must contain at least one lexical insertion site. Exactly one of these insertion sites must be a site for a lexical anchor (e.g. a noun, adjective or verb), and this lexical insertion site may be dominated by functional projections up to the appropriate extended projection, as defined by Grimshaw (2000). If these functional projections are present, their heads are also lexical insertion sites, where a functional lexical item must be inserted.

Lexical items are associated with structures that contain exactly two nodes. The terminal frontier node is a lexical item, and the root node is the corresponding category label. These types of structures are called lexical item structures, and their names are prefixed with an η. Lexical item

[4] I assume that proper names and Wh-words will be inserted at D, and in these cases, that the D is the anchor of the elementary structure.

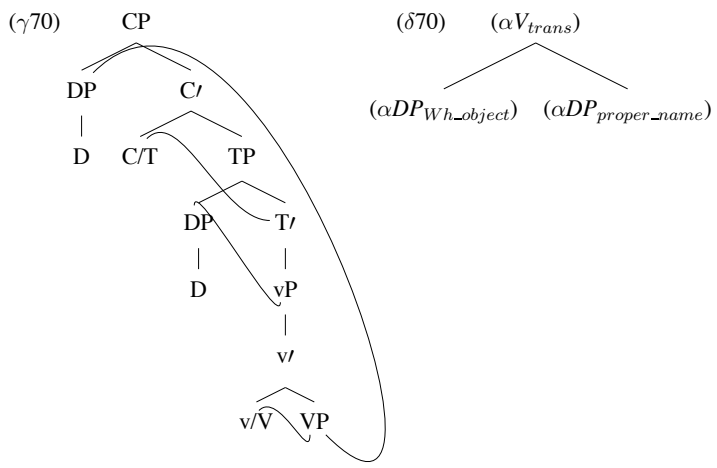


Figure 3.14: Derived and Derivation structures of (70): “What did Peter buy?”

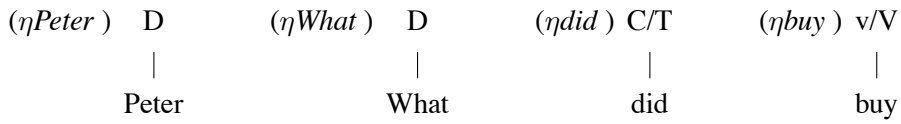


Figure 3.15: Lexical Item structures for example (70): “What did Peter buy?”

structures are composed at lexical insertion sites in a composition operation called lexical insertion. Lexical insertion is a node rewriting operation analogous to substitution, whereby a lexical insertion site is identified with the root node of a lexical item structure.

The lexical item structures necessary to complete the derivation of example (70) are given in Figure 3.15, and the lexically complete derivation and derived structures are given in Figure 3.16. The resulting derived structure is indistinguishable from that which would be produced by the derivational process described in section 3.3.1. The derivation structure, on the other hand, does reflect the process of lexical insertion, recording which lexical items are inserted into which insertion sites.

As mentioned in the introduction to this section, the principal motivation for the adoption of the lexical insertion operation was to derive certain types of sharing coordination constructions, namely those that share a lexical item. However, apart from the shared lexical item constructions that will be discussed in Chapter 4, lexical insertion does not directly impact the other constructions discussed in this thesis. Therefore, I will generally omit the explicit presentation of the lexical insertion operation. I will present elementary structures which contain lexical items, with the understanding that these items are actually inserted by lexical insertion after the LTAG operations. This also has the beneficial effect of reducing the presentational complexity of the following discussions, allowing

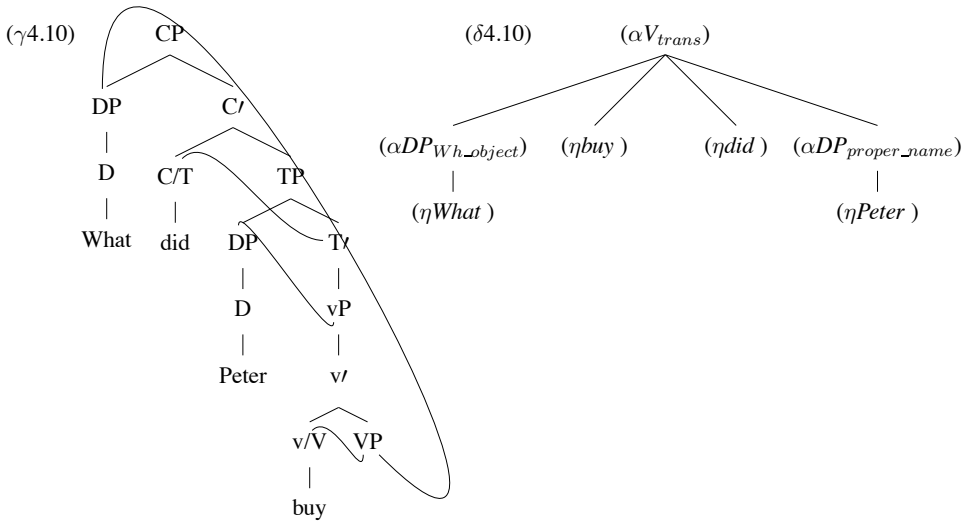


Figure 3.16: Lexically Complete Derived and Derivation structures of (70): "What did Peter buy?"

focus to remain on the most relevant aspects of the derivations.

3.4 Sarkar and Multiple Dominance in TAG

While the LTAG framework presented sections 3.2 and 3.3 does derive elementary structure internal multiple dominance, sharing coordination constructions cannot be derived. The operations substitution and adjoining are defined as the combination of two elementary structures, but sharing coordination constructions seem to require that three elementary structures compose simultaneously. This problem was first noted by Sarkar and Joshi (1996), in which several mechanisms were proposed that allow a multiple dominance analysis of sharing coordination structures in an LTAG framework.

Sarkar and Joshi posit that multiple dominance structures can be derived by the simultaneous composition of more than two elementary structures. In the case of substitution, an initial structure is substituted into two substitution sites at once. The authors propose that this derivation is the result of a novel syntactic operation, conjoin, as well as a set of two lexical operations, Build Contraction and Find Root, which function on individual elementary structures. Build Contraction marks a set of nodes as "contracted", and these nodes will be the site of multiple composition operations in the syntax. These nodes are marked in the elementary structure by a circle and are subscriptionally noted in the name of the elementary structure. Find Root identifies the lowest internal node that dominates all non-contracted terminal nodes. The node returned by Find Root is indicated in italics.

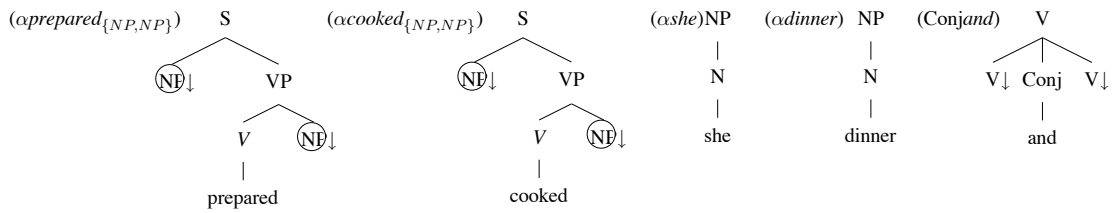


Figure 3.17: Elementary structures to derive (71).

Conjoin is a syntactic operation which combines three elementary structures simultaneously and merges the contracted nodes identified by Build Contraction. In addition to these novel operations, Sarkar and Joshi assume conditions on the well-formedness of elementary structures that differ from those of Frank (2002). Elementary structures only include a single terminal frontier node, the lexical anchor, and these structures only project far enough to include the argument substitution sites of the lexical anchor.

The elementary structures in Figure 3.17 will be used to derive example (71), in which the subject and object are shared. The verbal elementary structures project to an S and include substitution sites for both subject and object NPs. Build Contraction has marked the subject and object NP substitution sites in both verbal elementary structures as contracted nodes; these contracted nodes are specified in the names of the verbal elementary structures: (α*prepared*$_{\{NP,NP\}}$) and (α*cooked*$_{\{NP,NP\}}$). Find Root has identified the V nodes as the highest nodes to dominate non-contracted terminals, and these nodes are italicized.

(71) *She* prepared and cooked *dinner*.

In addition to the verbal and argument elementary structures, Sarkar and Joshi assume a coordinator elementary structure. This coordinator structure contains two special non-terminal frontier nodes which match the root node in category and which are indicated as substitution sites with the same down-arrow (↓). These are the nodes at which the coordinator structure combines with the other two structures. The coordinator structure in Figure 3.17 has V frontier and root nodes; although other instantiations are possible, such as VP or V′ coordination. The appropriate version of the coordinator structure will have root and frontier nodes which match the category returned by the operation of Find Root on the verbal elementary structures.

The coordinator elementary structure combines with the verbal elementary structures with the novel operation conjoin. The non-terminal frontier nodes of the coordinator structure are identified

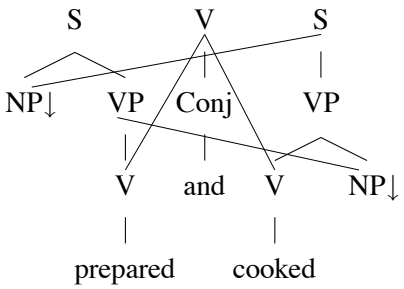


Figure 3.18: (α*prepared*), (α*cooked*) and (CONJ*and*) after Conjoin Operation

with the V nodes in each verbal elementary structure. This aspect of conjoin is similar to substitution, except that the latter only operates on the root nodes of initial structures and only combines two structures at once. Conjoin can operate on internal or root nodes and always simultaneously combines more than two structures.

Conjoin has a second function, which I refer to as the contraction function. The contraction function merges the contracted nodes, as well as the combination operations which occur on these nodes (p.c. Anoop Sarkar). Because (α*prepared*$_{\{NP,NP\}}$), (α*cooked*$_{\{NP,NP\}}$) and (Conj*and*) are combined by conjoin, the subject and object NP substitution sites of the two verbal elementary structures are merged. Substitution at these merged nodes happens at both nodes simultaneously; the merged nodes are unified as a single node.

Figure 3.18 illustrates an intermediary derived structure in which the elementary structures (α*prepared*$_{\{NP,NP\}}$) and (α*cooked*$_{\{NP,NP\}}$) have combined with the coordinator structure (Conj*and*) at the V nodes.[5] As a result, the nodes in the contraction sets of (α*prepared*$_{\{NP,NP\}}$) and (α*cooked*$_{\{NP,NP\}}$) have been merged and a single subject and object NP remains. Also, the V nodes, at which the (Conj*and*) structure conjoined, are multiply dominated. In addition to the VP nodes which dominate the V nodes in the spine of the verbal elementary structures, the V nodes are also dominated by the root V of the coordinator elementary structure.

The substitution of the subject and object NPs into the merged substitution sites in the structure in Figure 3.18 yields the derived structure in Figure 3.19. The derivation structure is given in Figure 3.20. Finally, Sarkar and Joshi do not propose an algorithm for the linearization of multiple dominance structures.

[5]This intermediary derivation structure is given for illustrative purposes only. In LTAG, such intermediary structures have no theoretical reality, the combination operations occur between the elementary structures, not intermediary constructions.

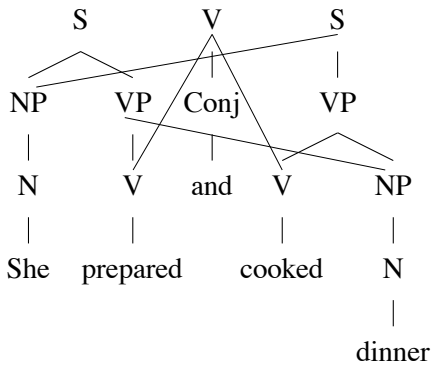


Figure 3.19: Derived Graph for (71):"*She prepared and cooked dinner.*"

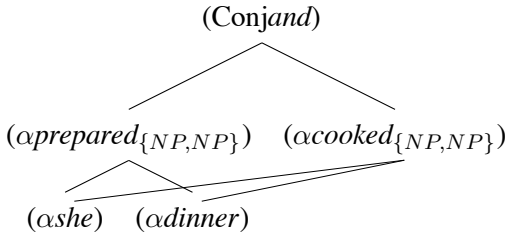


Figure 3.20: Derivation Graph for (71): "*She prepared and cooked dinner.*"

While the conjoin approach derives a significant portion of the data, there are several empirical and theoretical reasons why the approach needs to be reevaluated. From a conceptual point of view, it is problematic for the conjoin operation to be added to the grammar as a third syntactic operation. The grammar would then contain three operations: two of which are very general and a third which only derives sharing coordination constructions. In a system which otherwise makes use of only two syntactic operations, this seems to be an unwieldy solution.

Sarkar and Joshi address this issue, proposing that an alternative conception of their analysis could split the function of conjoin into two components. One component, a lexical operation, would add a coordinator to a verbal elementary structure and yield an auxiliary structure of the type given in Figure 3.21. The coordinator is added such that it dominates the node returned by Find Root, in this case, the V node. Build Contraction operates as previously defined; in Figure 3.21, the subject NP substitution site has been identified for sharing. This new lexical operation would replace the aspect of conjoin in which the coordinator elementary structure was combined with two verbal elementary structures. In this alternative conception, the coordinator is added to one verbal elementary structure in a lexical operation, and the resulting auxiliary structure is combined with another verbal

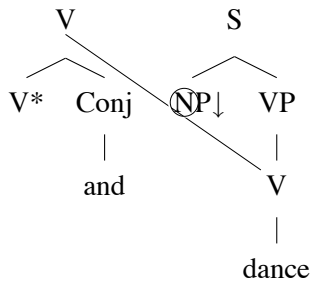


Figure 3.21: Auxiliary Coordination structure

elementary structure by the normal TAG operation adjoining. The contraction function of conjoin is incorporated into the adjoining operation. As the auxiliary coordination structure adjoins into another verbal elementary structure, the nodes identified by Build Contraction are merged.

Another conceptually challenging aspect of this approach is the treatment of the coordinator. It is widely assumed in syntactic approaches which utilize trees or phrase structures as formal devices that coordinators are the heads of asymmetric endocentric projections, as in Munn (1993), Zoerner (1995), Johannessen (1998) (but see Progovac (1998) for a dissenting view). Furthermore, coordinators are functional items, and therefore should be incorporated as functional heads of projections within the extended projection of a lexical anchor. The coordinator in Sarkar and Joshi's approach is treated as the daughter of an exocentric structure. The authors do not provide any support for this view of coordination, which is problematic under the assumption that all phrases must be singularly headed.

The final problem with this approach to multiple dominance in TAG is that examples such as (72) cannot be derived. This approach predicts that only arguments of coordinated verbal elementary structures can be shared.[6] The sharing of arguments between two coordinated verbal elementary structures is possible because the substitution sites in the coordinated structures have been merged by the contraction function. This function requires that the structures containing the sites to be merged are derivationally-local. That is, either both verbal structures must conjoin into the same

[6]The shared elements can be shared as the result of merged substitution sites, as in the cases discussed here. Sarkar and Joshi (1996) also discuss cases in which the foot nodes of auxiliary verbal elementary structures are merged. In these cases the coordinated verbal auxiliary structures adjoin into the same node, therefore sharing the same argument of modification.

coordinator elementary structure, or one structure must merge with the other, as in the alternative derivation. The examples in (72) would require that the merged nodes be contained in elementary structures that are not derivationally local to each other. For example, in (72a), the shared PP argument *of the analysis* would have to substitute into a substitution site shared by the DPs *the discussion* and *the report*. However, these DPs are not derivationally-local; they substitute into their respective verbal structures, which are then composed, either by conjoin or adjoining. Consequently, the contraction function cannot merge the nodes marked for contraction in the DP structures because they are not derivationally-local.

(72) a. Mary heard the discussion and Peter didn't hear the report *of the analysis*.

b. Mary read a report and Peter wrote a book *about coordination*.

c. Peter likes a professor who taught and Mary likes a student who debunked *the theory*.

In Chapter 4, I present a solution to this problem. I argue that the licensing of contracted nodes is necessarily local and that the appearance of non-locality in the examples of (72) is derivable under an extension to LTAG called Delayed-Tree Local Multi-Component TAG, as developed by Chiang and Scheffler (2008).

In the next section, I adapt the approach of Sarkar and Joshi (1996) to the framework proposed in the section 3.2 and 3.3, such that coordinators are analyzed as heads of functional projections that take CPs as complements. I then demonstrate that this analysis can derive the basic sharing coordination constructions.

3.5 Multiple Dominance in LTAG

As discussed in the previous section, Sarkar and Joshi (1996) propose an alternative to their conjoin analysis, where a lexical operation combines a verbal initial structure with a coordinator elementary structure resulting in a coordination auxiliary structure, shown in Figure 3.22. This alternative is attractive in that it eliminates the need for the construction-specific syntactic operation conjoin. The contraction function of conjoin becomes a function of adjoining. The other function of conjoin, which is the composition of the coordinator elementary structure with the verbal elementary structures, has become a lexical operation.

In this section, I propose a multiple dominance analysis of sharing coordination constructions adapted from the mechanisms described by Sarkar and Joshi (1996). In addition to the LTAG frame-

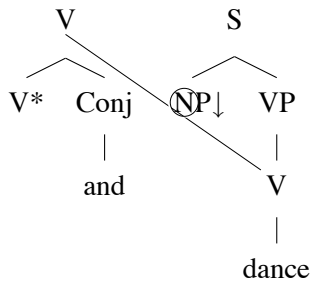


Figure 3.22: Auxiliary Coordination Tree

work of Frank (2002) and the two proposed revisions outlined in sections 3.3.1 and 3.3.2, there are several coordination specific assumptions that need to be discussed. I will first outline these additional assumptions, which will be followed by a series of illustrative sample derivations.

Following Munn (1993), Zoerner (1995) and Johannessen (1998), I assume that the coordinator is the head of a functional projection, ConjP, and that ConjP takes CP as complement, forming part of the maximal projection of the verbal lexical anchor. The coordinator and its projection ConjP may therefore be included in elementary structures anchored by verbs.[7] One consequence of assuming that ConjP takes CP as complement is that the Find Root operation becomes unnecessary. As discussed in section 3.4, Find Root identifies the lowest node which dominates all non-shared frontier nodes in each conjunct, and coordination occurs at the nodes returned by Find Root. In the present proposal, verbal coordination is always CP coordination, which is conceptually simpler and therefore preferred over an analysis in which ConjP must take various maximal and bar level projections.

In addition, ConjP, as a functional projection in the extended projection of verbal lexical anchors, takes CP as complement. I assume that the left conjunct is also always a CP and, following Munn (1993), that ConjP right-adjoins to this CP, becoming the right conjunct. In the LTAG formalism, this means that a verbal elementary structure which projects a conjunction phrase is an auxiliary structure. The structure (β*and_shared*) in Figure 3.23 is a typical example of a verbal coordination

[7]See Han et al. (2008) for an analysis of DP coordination where ConjP is treated as the functional projection of a nominal lexical anchor. In this analysis, nominal coordination is derivationally distinct from verbal coordination; see section 6.2 for a discussion on how the current analysis could be constrained to disallow generation of nominal coordination from underlying verbal coordination.

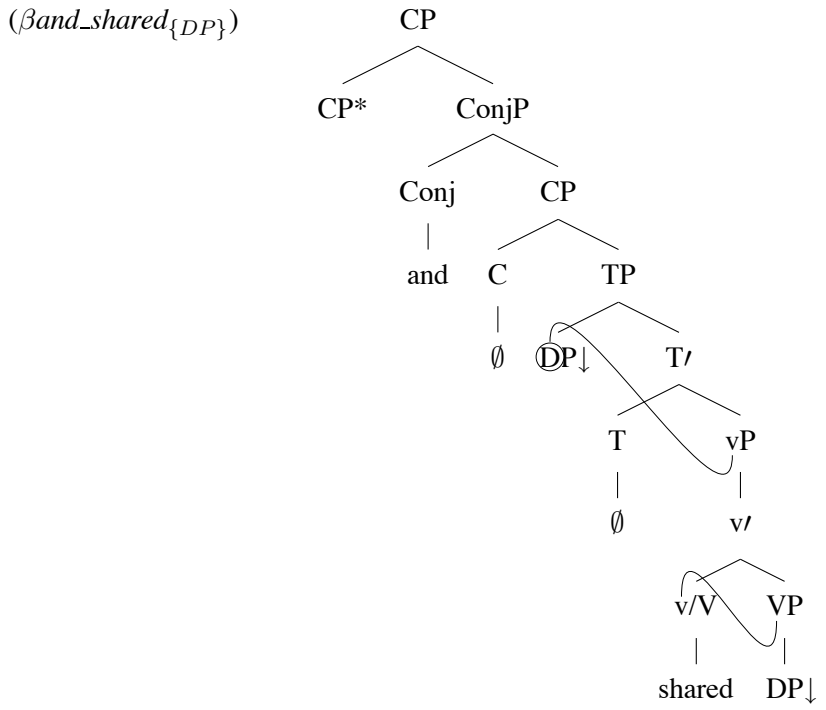


Figure 3.23: Coordination Auxiliary Tree (β*and_shared*$_{\{DP\}}$)

elementary structure in my proposal. The lexical anchor *shared* is dominated by both v′ and VP. *shared* takes two arguments, a subject DP substitution site, which is shared between TP and vP, and an object DP substitution site as the complement of VP. The coordinator *and* heads the functional projection ConjP, which takes the CP as complement. This ConjP then adjoins to another CP, as indicated by the presence of a CP root node and CP* foot node.

In this example, the subject DP substitution site is marked for contraction, denoted by the circle around this node and the subscript DP in the structure name. Sarkar and Joshi (1996) proposed the lexical operation Build Contraction to identify nodes for contraction in elementary structures. While I could assume a similar operation, I instead assume that these nodes identified for contraction are already present in the enumeration from which elementary structures are constructed. In other words, the enumeration consists of frontier and internal nodes, some of which are identified for contraction. Elementary structures are constructed from this enumeration by Merge and Remerge, which operate on all types of nodes, both those identified for contraction and those which are not. Therefore, it is not necessary that Build Contraction specify these nodes, and Merge and Remerge can be maintained as the only lexical operations.

In accordance with Sarkar and Joshi's (1997) alternative analysis, I assume the contraction function to be part of the adjoining operation. In the course of deriving a sharing coordination construction, a coordination verbal auxiliary structure adjoins into a verbal initial structure at the CP node. As a function of this adjoining operation, the nodes marked for contraction are merged, including their features structures and any operations which take place at these nodes. If two substitution sites are merged, then an initial structure which Substitutes into the merged node is simultaneously inserted into both verbal elementary structures.

In summary, my proposal builds on the LTAG framework outlined in Frank (2002). Elementary structures are required to conform to the conditions on elementary structure minimality. This entails that the arguments of a lexical anchor are encoded in the elementary structure, and that the elementary structure may project to the extended projection of the lexical anchor. Additionally, I assume a multiple dominance theory of movement, where sharing within elementary structures is derived during the generation of the elementary structures. My proposal considers coordinators as heads of the functional projection ConjP, which always takes CP as complement. Following Munn (1993), ConjP adjoins into the left conjunct, which I assume to be a CP. Formalized in LTAG, the coordination verbal elementary structure is an auxiliary structure with a CP root node and CP* foot node. In the current approach, the lexical operations Find Root and Build Contraction are both unnecessary because coordination is always between CPs and nodes identified for contraction are merged into the elementary structures as ordinary nodes. In addition, adjoining and substitution are the only syntactic operations, in that these are the only mechanisms in which two elementary structures can compose. Finally, Merge and Remerge are the only lexical operations by which elementary structures are constructed.

In order to demonstrate how the current account derives sharing coordination constructions, I will outline the derivations for a series of left and right-sharing examples. The sentence in example (73) contains a left-shared subject, and Figure 3.24 includes the elementary structures necessary to derive this example. (α*bought*$_{\{DP\}}$) is an initial structure, as indicated by the α portion of the name, and the verb *bought* is the lexical anchor. This structure has a subject DP substitution site marked for contraction, which is circled and noted in the structure name. (β*and_shared*$_{\{DP\}}$) is a verbal coordination auxiliary structure, where the verb *shared* is the lexical anchor, and *and* is the head of ConjP. The root node CP and foot node CP* indicate that this structure will adjoin into a CP node. Similar to the structure (α*bought*$_{\{DP\}}$), (β*and_shared*$_{\{DP\}}$) has a subject DP substitution site that is marked for sharing.

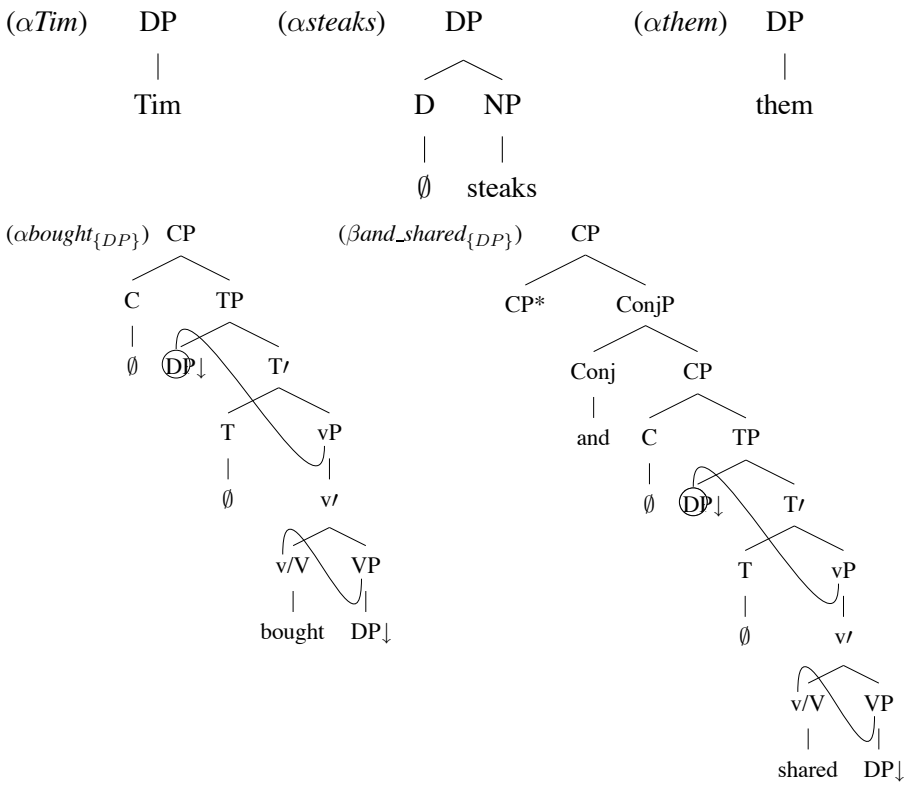


Figure 3.24: Elementary structures to Derive Example (73): "Tim bought steaks and shared them."

(73) *Tim* bought steaks and shared them.

Figure 3.25 includes the derived and derivation structures for example (73). In the derived structure (γ(73)), the subject DP is dominated by the TP of both the left and right conjuncts. The conjuncts are CPs, and the ConjP phrase has been adjoined to the root of the left CP conjunct. The derivation structure (δ(73)) records the information on how the derived structure was composed. The auxiliary verbal coordination structure (β*and_shared*$_{\{DP\}}$) has adjoined into the verbal initial structure (α*bought*$_{\{DP\}}$). The contraction function of adjoining merges the DP substitution sites marked for contraction in each of these structures. Thus, the initial structure (α*Tim*) substitutes into the DP substitution sites of both verbal initial structures simultaneously. The object DP initial structure (α*steaks*) substitutes into (α*bought*$_{\{DP\}}$), and the initial structure (α*them*) substitutes into (β*and_shared*$_{\{DP\}}$) at their respective object DP substitution sites.

(74) Spencer mixed and we drank *the martinis*.

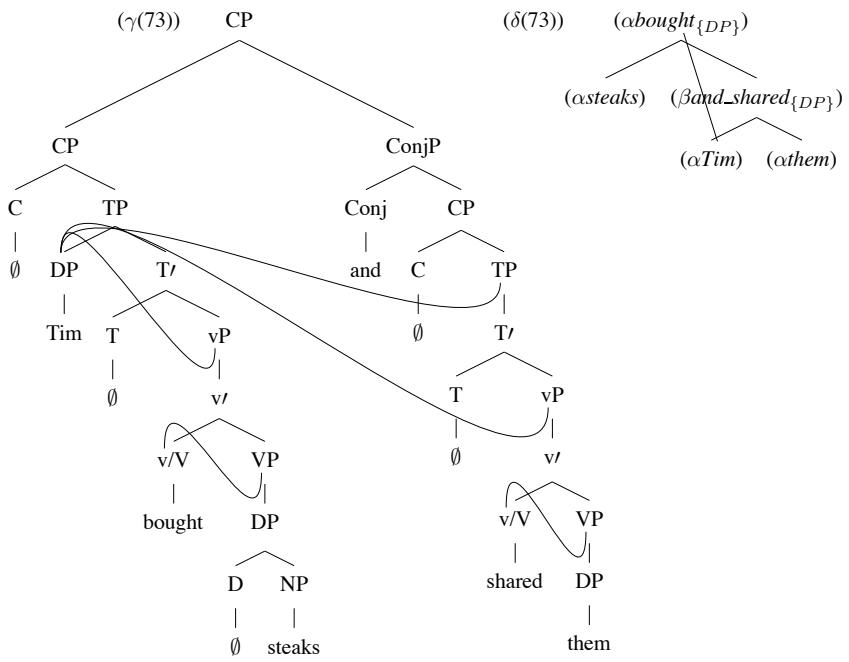


Figure 3.25: Derived and Derivation Structures for Example (73): "Tim bought steaks and shared them."

Similarly, the right-sharing construction in example (74) is derived in the same manner as example (73), using the elementary structures in Figure 3.26. The verb *mixed* anchors the initial elementary structure (α*mixed*$_{\{DP\}}$), where the object DP substitution site is marked for contraction. The coordinator *and* heads the ConjP in the auxiliary structure (β*and_drank*$_{\{DP\}}$), which also contains an object DP substitution site that is marked for contraction. The derivation of example (74) proceeds like a mirror image of the derivation for (73): the verbal coordination auxiliary structure (β*and_drank*$_{\{DP\}}$) adjoins into the verbal initial structure (α*mixed*$_{\{DP\}}$), and the object DP substitution sites are merged. The subject DP initial structures (α*Spencer*) and (α*we*) substitute into the subject substitution sites of the verbal elementary structures, while the object initial structure (α*the_martinis*) substitutes simultaneously into the verbal elementary structures at the merged object substitution site. In the derived structure (γ(74)), the object DP is dominated by the VP of each conjunct. Similar to the derived structure (γ(73)), the conjuncts are both CPs, and ConjP has adjoined to the left CP. The derived and derivation structures for this example are provided in Figure 3.27.

Example (75) has a similar derivation, with the exception that both subject and object are shared.

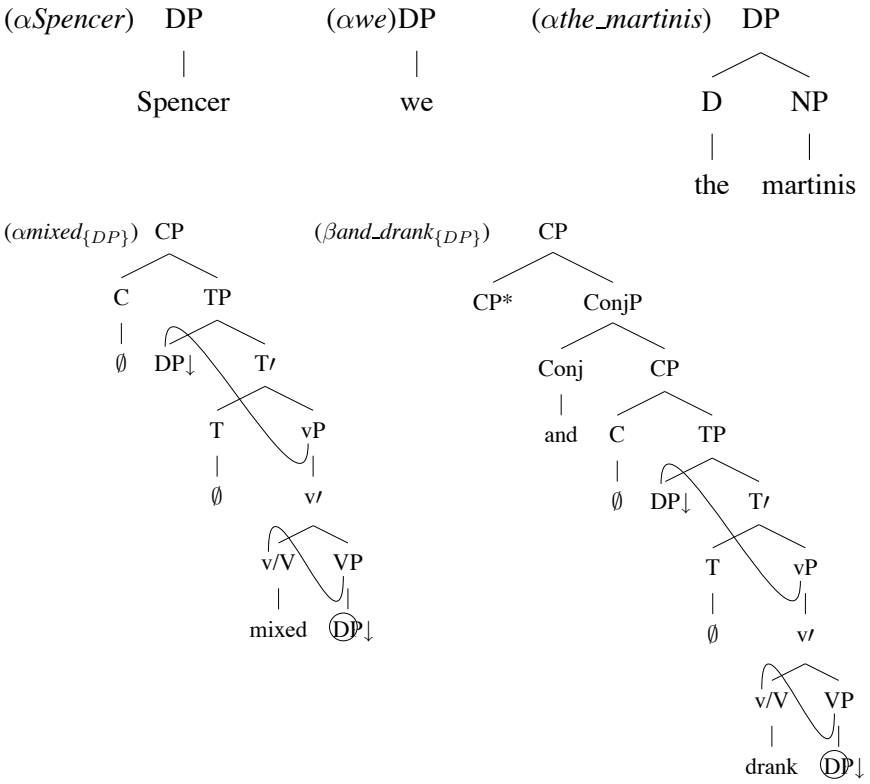


Figure 3.26: Elementary structures to Derive Example (74): "Spencer mixed and we drank the martinis."

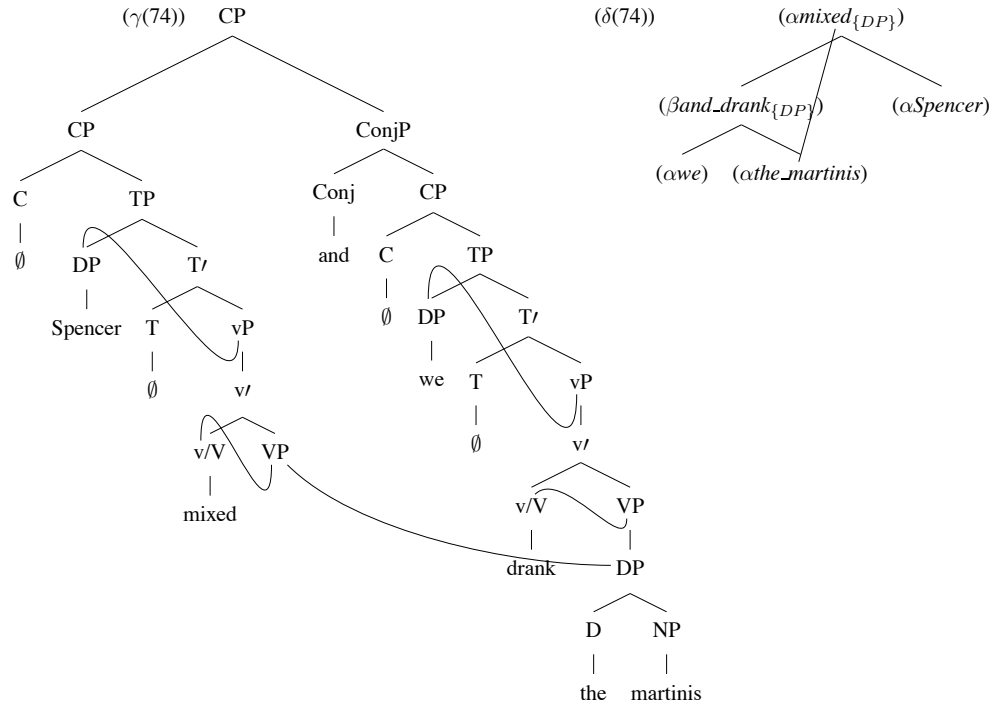


Figure 3.27: Derived and Derivation Structures for Example (74): "Spencer mixed and we drank the martinis."

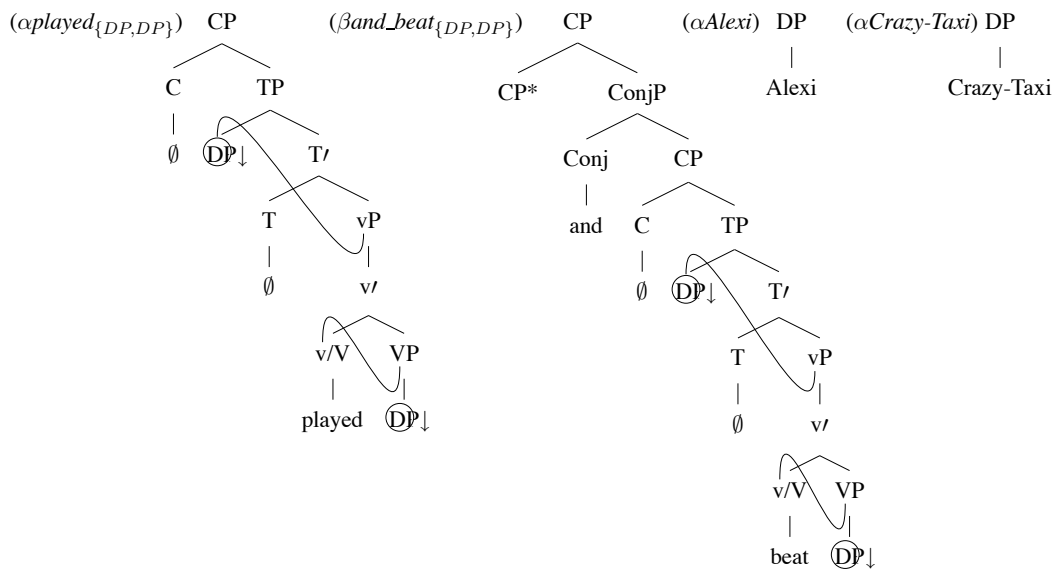


Figure 3.28: Elementary structures to Derive Example (75): "Alexi played and beat Crazy-Taxi."

Thus, both of the verbal elementary structures (α*played*$_{\{DP,DP\}}$) and (β*and_beat*$_{\{DP,DP\}}$) in Figure 3.28 contain subject and object substitution sites that are marked for contraction. In the derived structure given in Figure 3.29, the initial structure (α*Alexi*) has substituted at the merged DP subject substitution site of both verbal elementary structures, while the structure (α*Crazy-Taxi*) has substituted into both verbal structures at the merged DP object site.

(75) *Alexi* played and beat *Crazy-Taxi*.

The present approach can also account for shared elements in non-canonical positions. In example (76), the coordinate structure is an embedded Wh-question complement of the verb *know* that contains both a shared fronted Wh-argument and a shared subject. The elementary structures to derive this example are given in Figure 3.30. In the verbal elementary structures (α*bought*$_{\{DP,DP\}}$) and (β*and_sold*$_{\{DP,DP\}}$), the Wh-object substitution site is dominated by the VP as well as the CP and the subject and object substitution sites are marked for sharing. The elementary structure (α*knows*) contains two substitution sites: a subject DP dominated by TP and vP, and a CP as complement to VP.

(76) Angela knows what Tina bought and sold.

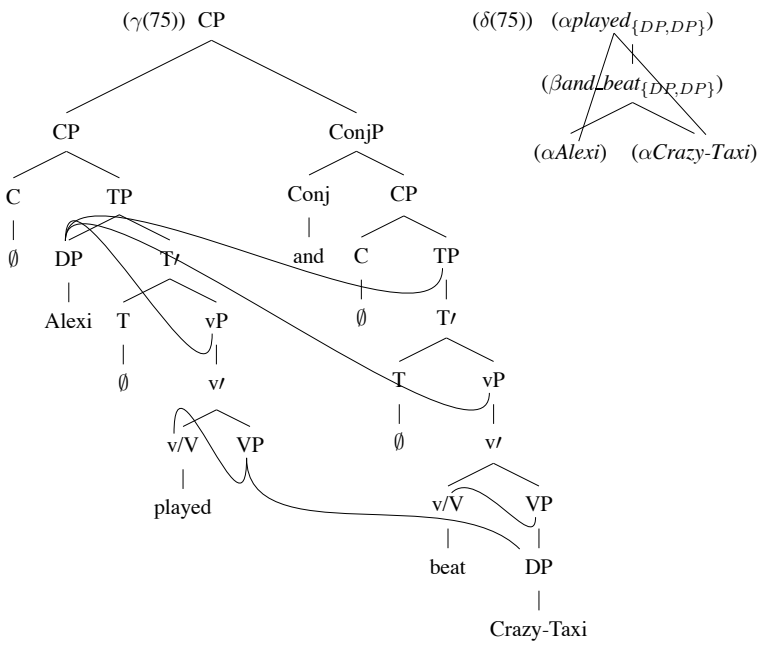


Figure 3.29: Derived and Derivation Structures for Example (75): "Alexi played and beat Crazy-Taxi."

The derived and derivation structures for this example are given in Figure 3.31. The auxiliary structure (β*and_sold*$_{\{DP,DP\}}$) adjoins into the initial structure (α*bought*$_{\{DP,DP\}}$), at which point the nodes marked for contraction are merged. The Wh-object initial structure (α*what*) and the subject initial structure (α*Tina*) substitute into the merged nodes. The initial structure (α*Angela*) substitutes at the subject DP substitution site of (α*knows*), and the initial structure (α*bought*$_{\{DP,DP\}}$) substitutes at the VP complement substitution site. Thus in the derived structure, the conjunction is complement to *knows*, while the Wh-object and DP subject *what* and *Tina* are shared between the conjuncts.

This proposal can derive both left and right argument sharing, as demonstrated by examples (73) through (76). The multiple dominance derived structures account for the central properties of shared arguments in sharing coordination constructions: a single item is an in-situ argument to two conjuncts. However, while this proposal can successfully account for argument sharing, there are additional sharing coordination constructions that fall outside the scope of the account presented thus far. In the following chapter, I detail a series of extensions to the account in order to broaden its empirical coverage.

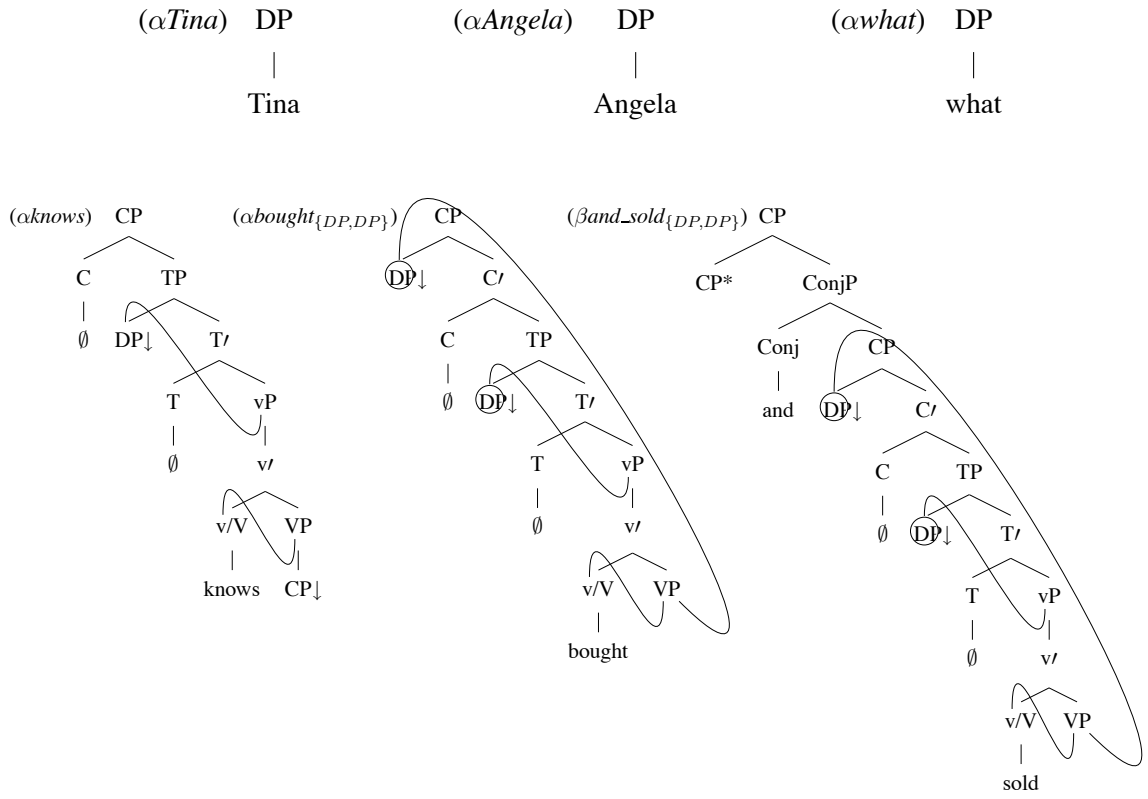


Figure 3.30: Elementary structures to Derive Examples (76): “Angela knows what Tina bought and sold.”

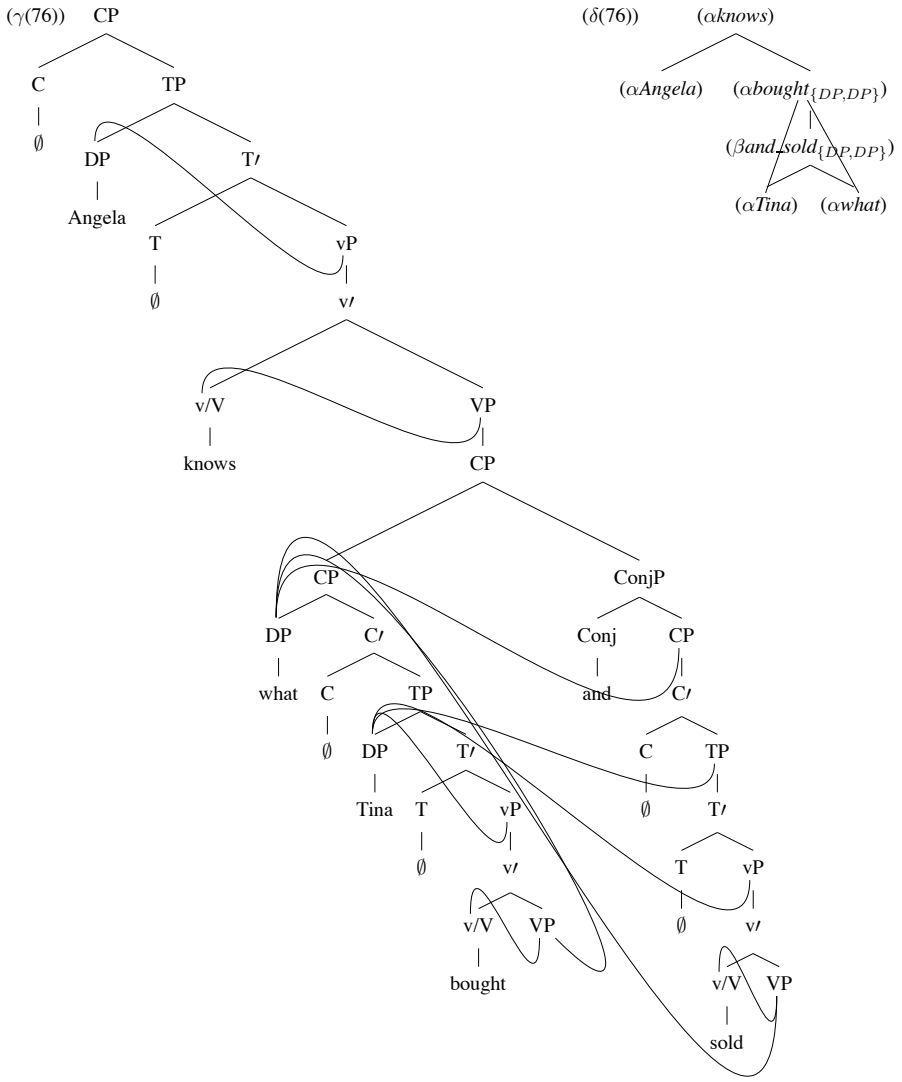


Figure 3.31: Derived and Derivation Structures for Example (76): “Angela knows what Tina bought and sold.”

Chapter 4

Extending the Empirical Coverage

The analysis proposed in Chapter 3 is designed to derive sharing coordination constructions in which an argument is shared between conjoined verbs. However, the empirical range of coordination sharing constructions includes a wider variety of constructions than just argument sharing. This chapter is devoted to extending the analysis beyond shared arguments to account for three additional constructions, including modifier and lexical item sharing as well as derivationally non-local sharing constructions. In section 4.1, I discuss modifier sharing and propose that the derivation of shared modifiers involves the contraction of adjoining sites, which results in an adjoining operation that is functionally distinct from adjoining at non-shared sites.

Section 4.2 addresses derivationally non-local sharing constructions, where the shared element does not directly compose with the conjoined verbal structures. Instead, the shared element combines with elementary structures that do not compose with each other and are therefore derivationally non-local. In example (77), the shared element *of the analysis* is an argument of the DPs *the discussion* and *the report*. Thus, the structures which contain the marked substitution sites do not adjoin into each other, precluding the possibility that the contraction function of adjoining merges the substitution sites marked for contraction. The sharing of the PP *of the analysis* is therefore unexplained by the approach presented in Chapter 3. In order to account for these derivationally non-local constructions, I apply the notion of delayed structure local multi-component TAG (Chiang and Scheffler (2008)), permitting the shared elements to combine locally.

(77) Mary heard the discussion and Peter didn't hear the report *of the analysis*.

The third type of sharing construction, shared lexical anchors, is discussed in section 4.3. To

derive these constructions, I utilize the lexical insertion operation, introduced in section 3.3.2, in which lexical items are inserted into derived and derivation structures after the LTAG syntactic operations. I extend the contraction function to lexical insertion sites, which allows lexical items to be shared in a manner analogous to argument sharing.

4.1 Identification for Adjoining Sites

In addition to shared arguments, as discussed in Chapter 3, modifiers can also be shared, as shown in (78). Each of these sentences is three-way ambiguous, and in two of these readings, the modifier is shared. In the first reading of example (78a), only the nearest conjunct is modified. One event, the grilling of steaks, is asserted to usually happen, while the frequency of the other event, the roasting of chicken, is not mentioned. It seems that the modifier is not shared between the conjuncts in this case. In another reading, both events often take place, though not necessarily at the same time: Tim may regularly grill steaks, but he also often roasts chicken. In the third reading, both of these events occur frequently and at the same time. In these last two readings, the modifier is shared between the conjuncts, which are being modified either together or individually.

(78) a. *Tim often* grills steaks and roasts chicken.

b. *Mary sometimes* visits her grandparents and calls her mom.

c. *John never* eats meat and drinks alcohol.

In the LTAG framework outlined in Chapter 3, modifiers such as “often”, “sometimes” and “never” are analyzed as the anchors of auxiliary elementary structures, as exemplified by (β*often*) in Figure 4.1. The root and foot nodes of this structure indicate that it must adjoin at a T$\prime$ adjoining site. However, as proposed in section 3.5, the derivation of shared elements is defined for substitution sites only. Sharing occurs as the result of two substitution sites being merged by the contraction function of adjoining, and an argument initial structure substituting at the merged node. Therefore, the analysis presented in Chapter 3 predicts that modifiers cannot be shared, which is not borne out by the examples in (78).

A logical first attempt to derive shared modifiers would extend the mechanism developed for sharing substitution sites to the sharing of adjoining sites. In such an approach, adjoining sites would be marked for contraction in the elementary structures, as was proposed for substitution sites in Chapter 3. Then, these adjoining sites would be merged as the verbal coordination auxiliary

structure adjoins into the initial verbal structure, and the modifier would subsequently adjoin into the merged adjoining site. However, this analysis produces an incorrect derived structure, as will be presently demonstrated.

The elementary structures necessary for this type of derivation of (78a) are given in Figure 4.1. In addition to the modifier auxiliary structure (β*often*), there are three argument initial structures, (α*Tim*), (α*steak*) and (α*chicken*), as well as two verbal elementary structures, (α*grills*$_{\{DP,T\prime\}}$) and (β*and_roasts*$_{\{DP,T\prime\}}$). In both verbal elementary structures, the subject substitution sites and the T′ nodes are marked for contraction, as indicated by the subscription of these nodes in their names and their circling in the elementary structures.

The derived and derivation structures in Figure 4.2 represent a critical intermediate stage in the derivation of (78a). As the derivation structure indicates, the verbal coordination auxiliary structure (β*and_roasts*$_{\{DP,T\prime\}}$) has adjoined into (α*grills*$_{\{DP,T\prime\}}$) at the CP node. The contraction function of this adjoining operation has merged the T′ nodes of these verbal elementary structures. However, the problem with the derived structure is apparent: the merge has resulted in a quaternary branching lower T′ node. The Ts and vPs from either conjunct have become sisters in a multiply-headed structure, and the spines of the two verbal elementary structures have become crossed. The final derived and derivation structures are given in Figure 4.3. While the derivation structure is correct, as (α*Tim*) and (β*often*) are both shared between the verbal elementary structures, the derived structure (γ(78a)) is not. The constituency is incorrect, as there is a single T′ dominating four daughters.

The derived structure (γ(78a)) given in Figure 4.4 reflects the correct constituency. The adverbial phrase is multiply dominated, being shared between the T′s of both conjuncts. Assuming that the elementary structures in Figure 4.1 are correct, it seems that the contraction function for adjoining sites cannot be the simple analogue of that for substitution sites.

I will assume then that adjoining sites marked for contraction are not merged by the contraction function. Instead, this function links nodes marked for contraction such that an auxiliary structure which adjoins into one linked adjoining site is simultaneously adjoined into the paired linked adjoining site. Consequently, the contraction function of adjoining serves two purposes: 1) merging substitution sites marked for contraction, and 2) linking adjoining sites marked for contraction. This simultaneous adjoining at distinct but linked adjoining sites requires a modified conception of adjoining.

I propose that adjoining at linked adjoining sites operates differently than adjoining at non-linked sites. Instead of identification between the root and foot nodes of the auxiliary structure with

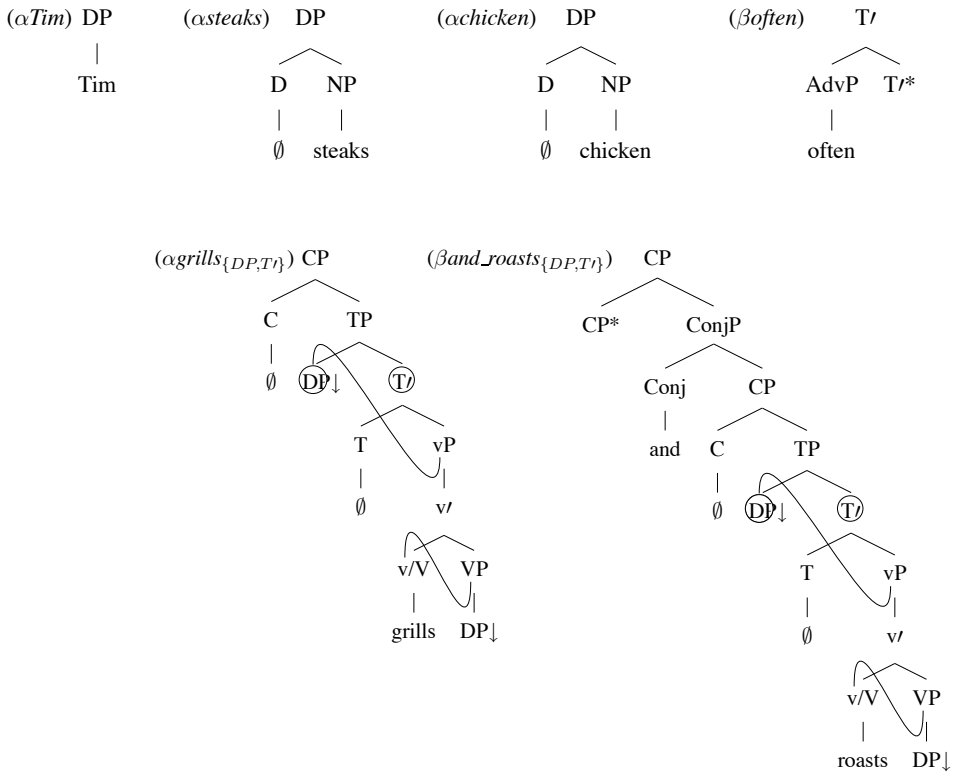


Figure 4.1: Elementary structures to Derive Example (78a): "Tim often grills steaks and roasts chicken."

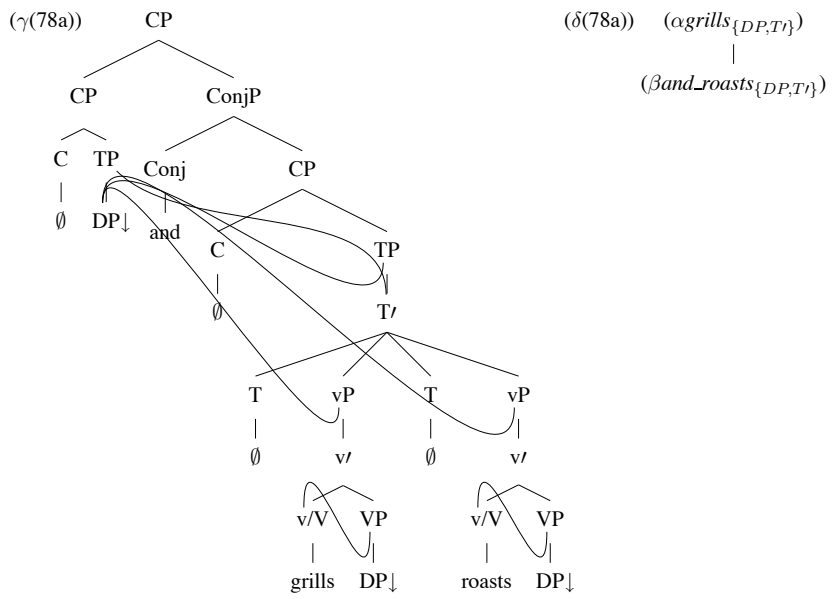
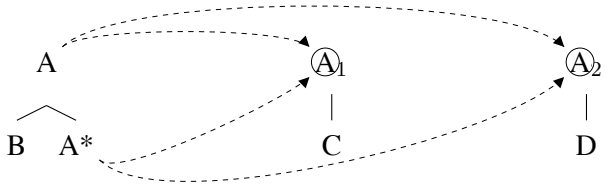


Figure 4.2: Intermediate Derived and Derivation Structures for Example (78a): "Tim often grills steaks and roasts chicken."

the top and bottom halves of the split adjoining site, a replacement operation occurs between these nodes. The root and foot nodes of the auxiliary structure are replaced with the top and bottom halves of the adjoining site, respectively. (79) includes an example of adjoining at linked nodes. Assume that nodes A_1 and A_2 are linked. The auxiliary structure anchored by B adjoins into both linked nodes simultaneously. The resulting derived structure is given in example (80). The critical aspect of this special adjoining operation is that the linked adjoining sites have not been merged with each other. Instead, the simultaneous adjoining of the B auxiliary structure adds a mother to B; thus, it is now multiply dominated by A_1 and A_2.

(79)



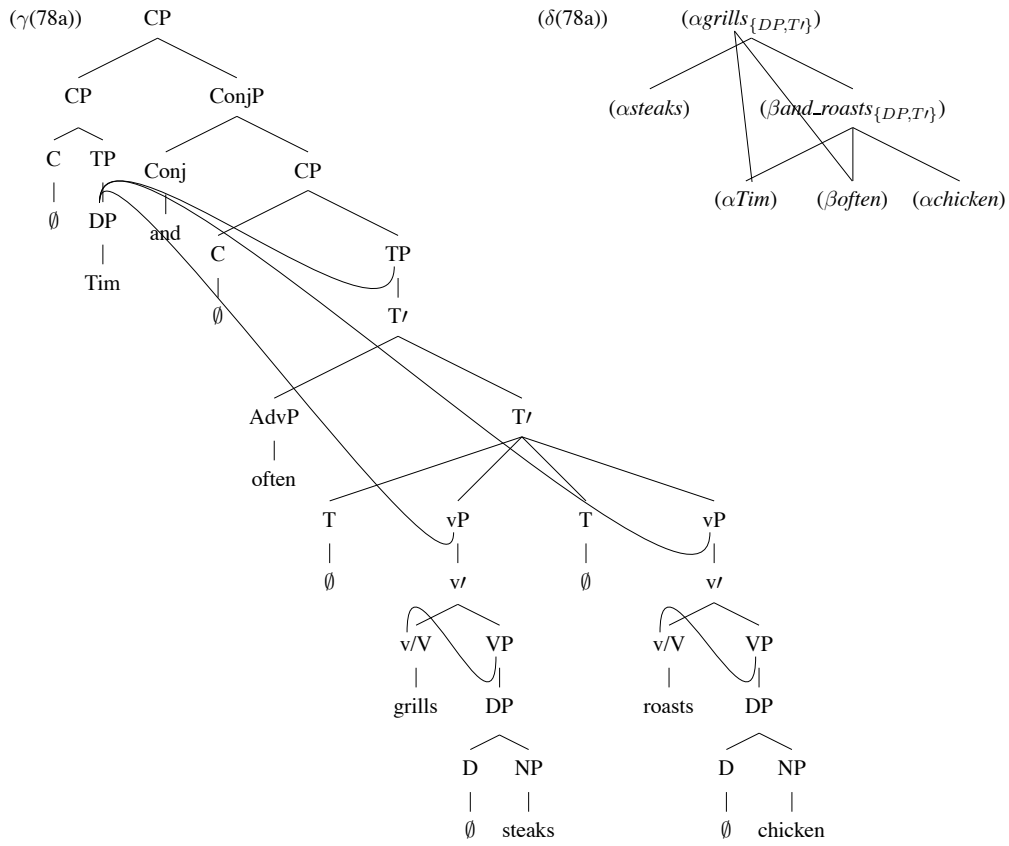
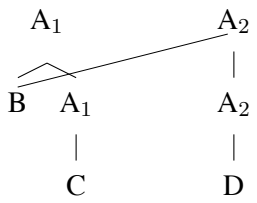


Figure 4.3: Derived and Derivation Structures for Example (78a): "Tim often grills steaks and roasts chicken."

(80)



This definition of adjoining at linked adjoining sites allows the auxiliary structure node that is immediately dominated by the root to be multiply dominated, instead of the root or foot nodes. When used in the derivation of (78a), this operation yields the correct derived structures. The auxiliary structure (*βoften*) simultaneously but separately adjoins into the linked T′ nodes in each conjunct. As it adjoins into the left conjunct, the root and foot nodes of (*βoften*) are replaced with the top and bottom halves of the split linked adjoining site, and similar operations occur in the right conjunct. The effect of these operations is that AdvP node of (*βoften*) becomes multiply dominated by the T′ nodes of each conjunct. The derived structure in Figure 4.4 can capture both readings of (78a) in which the modifier is shared; the disambiguation between these readings will occur as a semantic process[1].

This re-analysis of adjoining at linked sites creates a contrast between the function of substitution at nodes marked for contraction and the function of adjoining at these types of node. The contraction function merges substitution sites but links adjoining sites; however, such a contrast is not surprising considering the differences between substitution and adjoining. Substitution rewrites the substitution sites with an initial structure. Even if an initial structure is substituted into two substitution sites that have been merged, the only material that is being shared is the initial structure. Adjoining, on the other hand, inserts material into the spine of an elementary structure. Thus, if adjoining sites were merged instead of linked, portions of their spines would be shared as well. As we saw above, this manner of derivation is problematic. The re-analysis of adjoining at linked adjoining sites excludes spine sharing, such that the foot and root nodes of the auxiliary structure are not shared. In sum, the extension of the contraction function of adjoining to adjoining sites can accurately account for modifier sharing constructions, which were otherwise underivable in the proposal presented in Chapter 3.

[1]The difference between these readings seems to be the scope of the modifier; in the reading of (78a), in which the two events both occur frequently but not necessarily at the same time, the modifier has scope under the coordinate structure. In the third reading, where the actions denoted by the verbs occur as one event, which occurs often, the modifier has scope over the coordinate structure. See Han et al. (2008) for a syntax-semantics interface in which these readings can be underspecified.

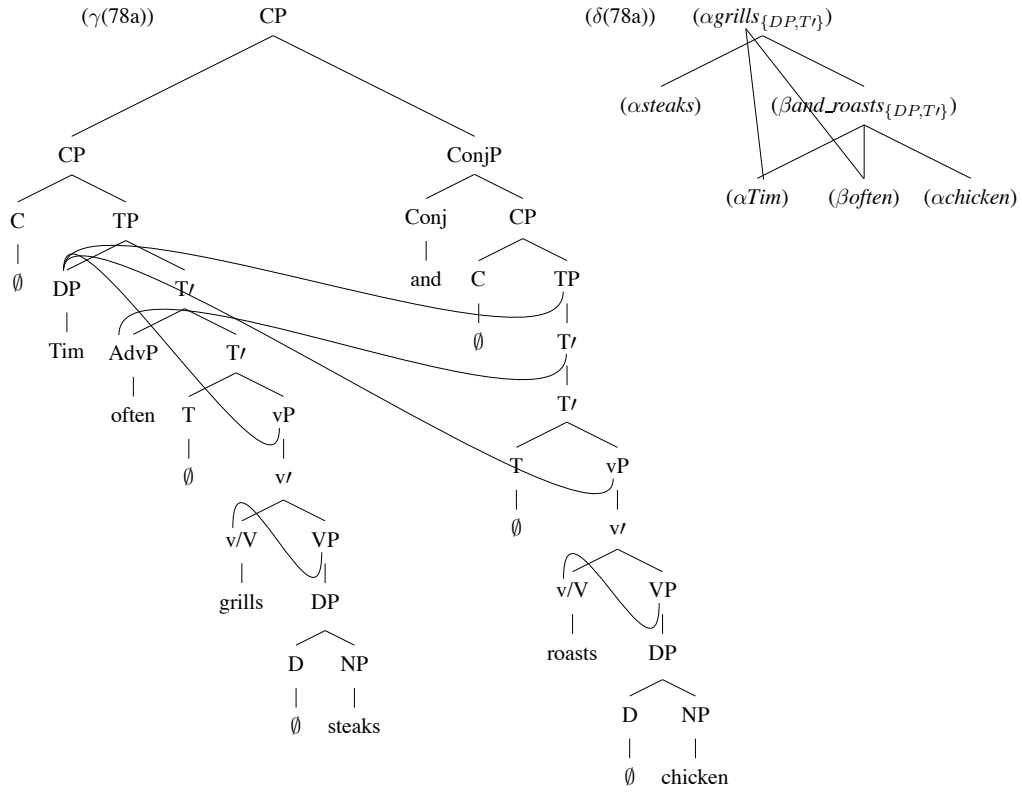


Figure 4.4: Proper Derivation and derived structure for Example (78a): “Tim often grills steaks and roasts chicken.”

4.2 Derivationally Non-local Sharing

The second type of construction that is problematic for the analysis presented in Chapter 3 involves derivationally non-local sharing. In these cases, exemplified in (81), the shared elements are not arguments nor are they modifiers of the verbs that participate in coordination. For example, in (81a), the PP *of the analysis* is shared between the DPs *the discussion* and *the report*.

(81) a. Mary heard the discussion and Peter didn't hear the report *of the analysis*.

b. Mary read a report and Peter wrote a book *about coordination*.

c. Peter likes a professor who taught and Mary likes a student who debunked *the theory*.

I assume for the moment that the elementary structures given in Figure 4.5 are used to derive (81a); however, I will later present revised elementary structures. The argument substitution sites of (α*the_discussion*$_{\{PP\}}$) and (α*the_report*$_{\{PP\}}$) are marked for contraction. The derivation and derived structures given in Figure 4.6 are the correct structures, but they can only be produced from the elementary structures in Figure 4.5 if the contraction function of adjoining is permitted to function in a derivationally non-local manner. That is, the adjoining of the verbal coordination auxiliary structure (β*and_didn't_hear*) into the verbal initial structure (α*heard*) would have to license the merge of the marked substitution sites in (α*the_discussion*$_{\{PP\}}$) and (α*the_report*$_{\{PP\}}$).

This apparent need to allow for non-locality between the structures which contain nodes marked for contraction can be resolved using an extension to LTAG known as multicomponent TAG (MC TAG). MC TAG was first proposed by Joshi (1987) and elaborated in Weir (1988). Since then, MC TAG has been used in the analysis of many structures, including quantifier scope interactions (Frank (2002), Kallmeyer and Joshi (2003)).

In MC TAG, elementary structures are sets of structures which together are treated as atomic units for LTAG composition, as exemplified in Figure 4.7. This MC elementary structure set contains two elementary structures, the initial structures (αD) and (αE). Singleton elementary structure sets, which only contain a single structure, are also possible. Multicomponent elementary structure sets are subject to the same conditions on well-formedness as singleton sets. The individual structures in an MC TAG elementary structure set are represented in the derivation structure as individual nodes. Thus, in the derived structure (δB) in example (82), each of the nodes represent individual elementary structures, though the structures (αD) and (αE) belong to a single MC set.

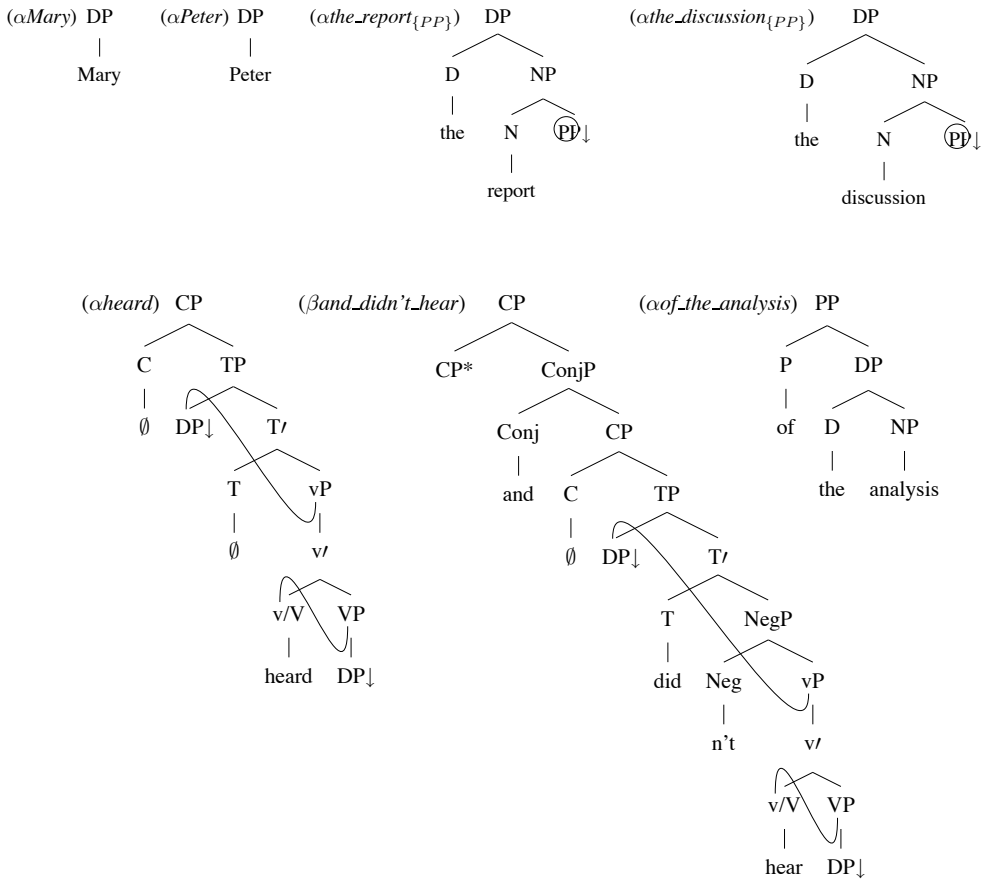


Figure 4.5: Elementary structures to Derive Example (81a): "Mary heard the discussion and Peter didn't hear the report of the analysis."

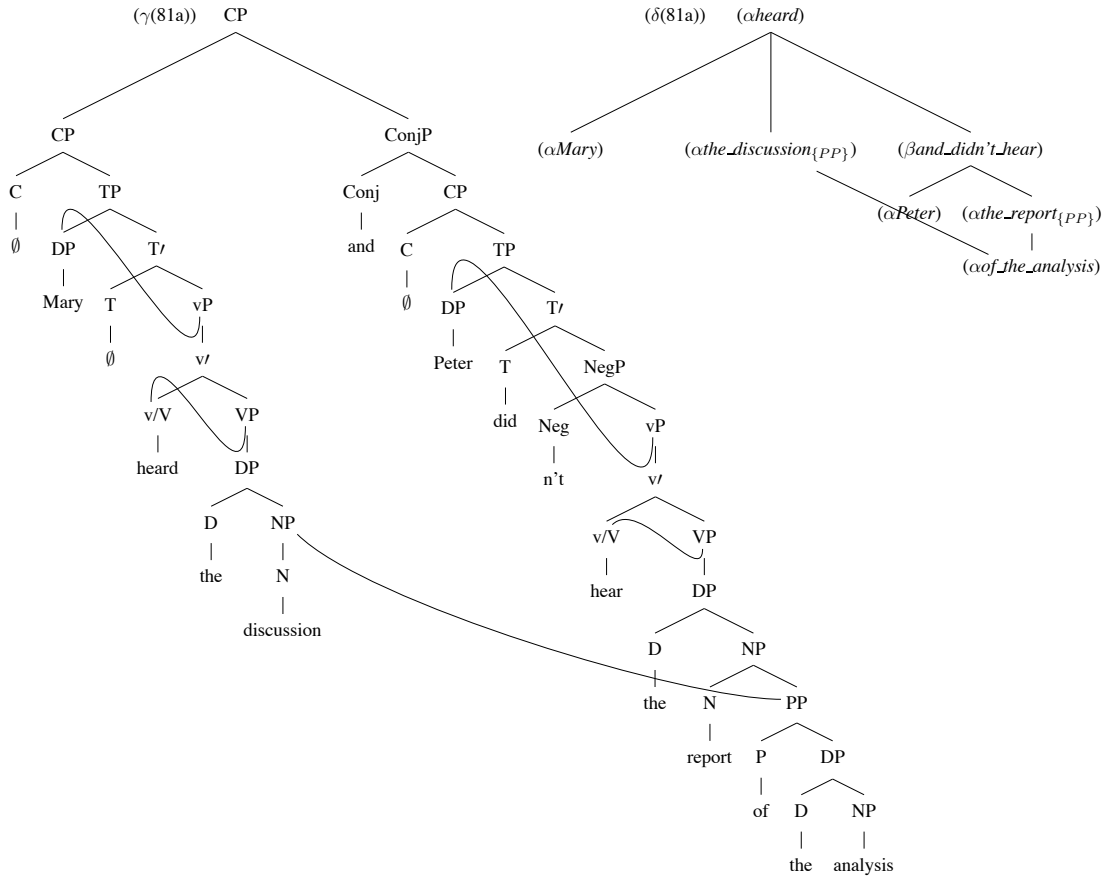
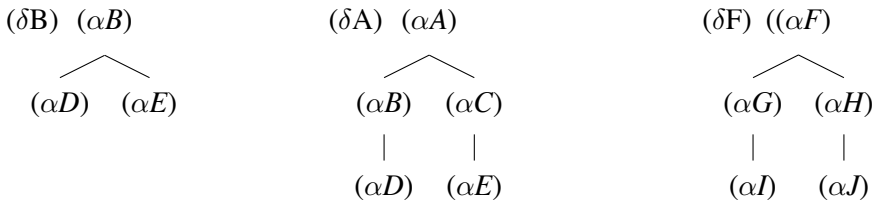


Figure 4.6: Derivation and Derived Structures of (81a) "Mary heard the discussion and Peter didn't hear the report *of the analysis*

$$\left\{ \begin{array}{cc} (\alpha D)\ \mathrm{D} & (\alpha E)\ \mathrm{E} \\ | & | \\ \mathrm{d} & \mathrm{e} \end{array} \right\}$$

Figure 4.7: MC structure set

(82)



The derivation recorded in the derivation structure (δB) is called a structure-local MC TAG derivation because all of the elementary structures of an MC structure set compose with the same elementary structure. In contrast, the derivation structure (δA) in (82) is the result of a non-structure-local derivation, where the MC set $\{(\alpha D), (\alpha E)\}$ has composed with two separate structures, (αB) and (αC) respectively. Weir (1988) demonstrated that structure-local MC TAG derivations are weakly equivalent to LTAG but that non-structure-local MC TAG can be generatively more powerful than standard LTAG.

However, Chiang and Scheffler (2008) show that not all non-structure-local MC TAG derivations result in more generative power. They define a subset of non-structure-local MC TAG called Delayed Structure-Local MC TAG that is weakly equivalent to standard structure-local MC TAG and LTAG[2]. In delayed structure-local MC TAG, non-structure-local derivations are subject to a structural constraint, expressed in terms of a *delay*, which is defined on the derivation structure. For any derivation involving an MC elementary structure set, a delay is the set of nodes on each path from the elements in the MC structure set to the lowest node that dominates all such nodes, excluding this lowest dominating node. For example, in the derivation structure (δA) in (82), the nodes (αD) and (αE) are members of an MC set. (αA) is the lowest node which dominates both (αD) and (αE). Thus, the delay of the MC set $\{(\alpha D), (\alpha E)\}$ is the set of nodes on the paths from this MC set to (αA): $\{(\alpha D), (\alpha E), (\alpha B), (\alpha C)\}$. This is a one delay structure-local derivation because no node is a member of more than one delay. In derivation structure (δF), assume that $\{(\alpha G), (\alpha H)\}$ and $\{(\alpha I), (\alpha J)\}$ are each multicomponent structure sets. The delay of $\{(\alpha I), (\alpha J)\}$ is the set of nodes $\{(\alpha G), (\alpha H), (\alpha I), (\alpha J)\}$, and the delay of $\{(\alpha G), (\alpha H)\}$ is the set $\{(\alpha G), (\alpha H)\}$. Thus, the nodes $\{(\alpha G), (\alpha H)\}$ are contained in two delays, that of the MC set $\{(\alpha G), (\alpha H)\}$ and that of $\{(\alpha I), (\alpha J)\}$, making this is a two delay structure-local derivation. Chiang and Scheffler (2008) demonstrate that derivations involving up to two delays are weakly equivalent to structure-local derivations.

[2]Chiang and Scheffler use the name Delayed Tree-Local MC TAG, as they do not discuss directed graphs. I will continue to use *structure* in place of both *tree* and *graph*.

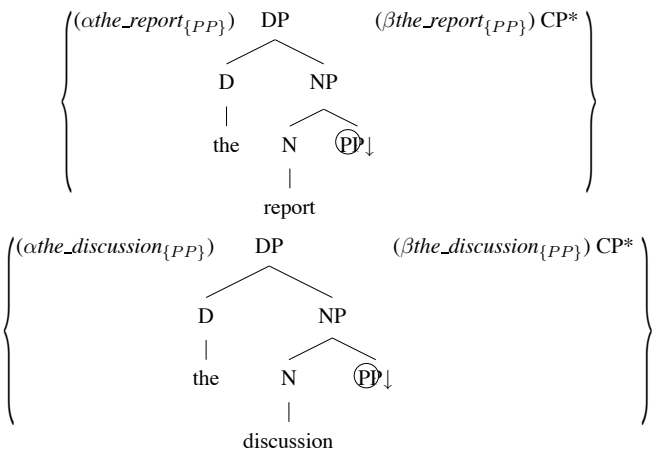


Figure 4.8: Elementary structures for Delayed-structure local Derivation of (81a)

Recall that in example (81a), here repeated as (83), the PP argument *of the analysis* is shared between the DPs *the discussion* and *the report*. However, these DPs are not derivationally local; that is, neither one of these DPs adjoins into the other. Thus, following the definition of adjoining adopted in Chapter 3, their substitution sites cannot be merged by the contraction function of adjoining.

(83) Mary heard the discussion and Peter didn't hear the report *of the analysis*.

However, it is possible for the DPs to be derivationally local, if the DPs are assumed to be members of MC structure sets that include an additional "defective" elementary structure. In essence, this analysis ensures that the DPs will be derivationally local through the a delayed structure-local composition of these defective structures, thus permitting the contraction function to merge the substitution sites of the DPs. The defective element is an auxiliary structure that consists of only a foot node labeled CP*. Thus, the DP elementary structures (α*the_report*$_{\{PP\}}$) and (α*the_discussion*$_{\{PP\}}$) given in Figure 4.5 become the multicomponent structure sets $\{$(α*the_report*$_{\{PP\}}$), (β*the_report*$_{\{PP\}}$)$\}$ and $\{$(α*the_discussion*$_{\{PP\}}$), (β*the_discussion*$_{\{PP\}}$)$\}$, illustrated in Figure 4.8.

The derived and derivation structures for example (81a) are given in Figure 4.9. Note that the derived structure in Figure 4.9 matches the one in 4.6; the defective auxiliary structures do not substantively change the derived structure. As recorded in the derivation structure (δ(81a)), the defective structure (β*the_discussion*$_{\{PP\}}$) adjoins to the root node of the (α*heard*) structure, and the defective structure (β*the_report*$_{\{PP\}}$) adjoins to the defective structure (β*the_discussion*$_{\{PP\}}$). Thus, the

elementary structures with nodes marked for contraction compose through adjoining, allowing the contraction function to merge the marked nodes. Subsequently, (α*of_the_analysis*) substitutes simultaneously into and is consequently shared by both ((α*the_discussion*$_{\{PP\}}$) and ((α*the_report*$_{\{PP\}}$).

The derivation of example (81a) in Figure 4.9 is a one-delay structure local derivation. The delay of the MC set {(α*the_discussion*$_{\{PP\}}$), (β*the_discussion*$_{\{PP\}}$)} is the set of nodes {(α*Mary*), (α*the_discussion*$_{\{PP\}}$), (β*the_discussion*$_{\{PP\}}$)}. The delay of the MC set {(α*the_report*$_{\{PP\}}$), (β*the_report*$_{\{PP\}}$)} is the set of nodes {(β*and_didn't_hear*), (α*the_report*$_{\{PP\}}$), (β*the_report*$_{\{PP\}}$)}. Thus, no node in this derivation belongs to more than one delay.

This analysis is attractive for two principal reasons: 1) the defective elementary structures enable the derivationally non-local constructions to be analyzed as being local, and 2) it allows both local and non-local sharing constructions to be derived using the same mechanism, the contraction function of adjoining, and only the elementary structures of these constructions differ.[3] That is, the difference between these constructions is limited to the composition of the elementary structures containing nodes marked for contraction. However, one consequence of this analysis is that a significant number of additional elementary structure sets are necessary for the grammar.

4.3 Shared Lexical items

A key assumption in LTAG has been that sharing is due to the combination of whole elementary structures. In Chapter 3, I demonstrated how an argument could be shared between two verbal elementary structures by merging the substitution sites at which the argument is substituted. The argument would therefore simultaneously substitute into both elementary structures. In section 4.1, I discussed shared modifiers and illustrated that auxiliary structures, which generally encode modification and recursion, could be shared between elementary structures by linking the adjoining sites at which shared auxiliary structures adjoin. Thus far, sharing in LTAG has been defined to apply to whole elementary structures, either initial or auxiliary. However, it appears that portions of elementary structures can also be shared, as shown in (84). In this section, I demonstrate that the contraction function of adjoining can be extended to lexical insertion sites, introduced in section 3.3.2, in order to derive these sub-elementary structure sharing constructions.

[3]While Delayed-Structure local MC TAG and Flexible composition MC TAG (Chiang and Scheffler (2008)) are weakly equivalent, Delayed-structure local MC TAG can derive these constructions, while Flexible composition MC TAG cannot.

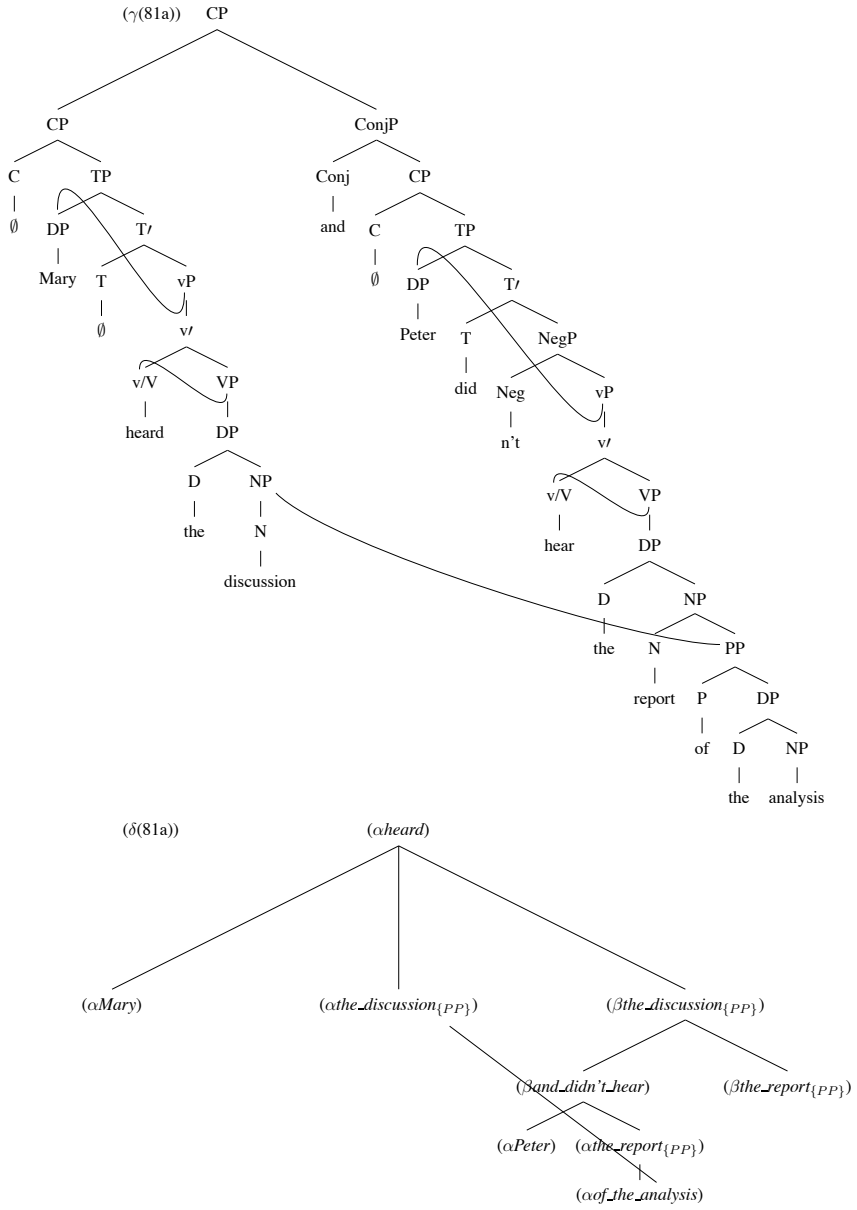


Figure 4.9: Delayed-Structure local Derivation and Derived Structures of (81a) "Mary heard the discussion and Peter didn't hear the report *of the analysis*

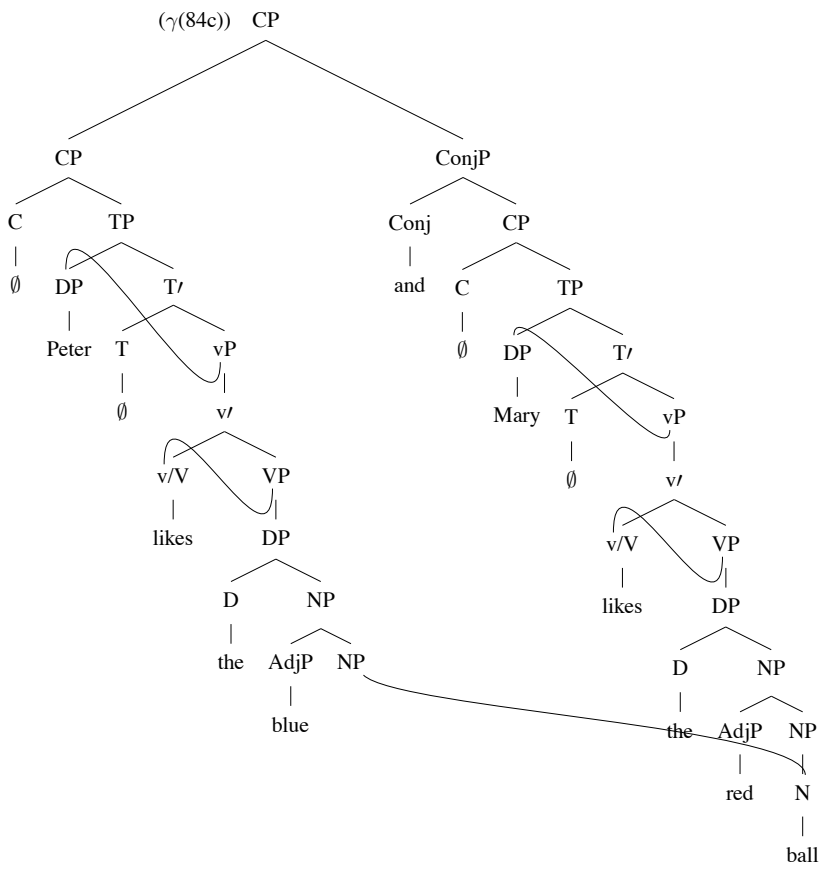


Figure 4.10: Derived structure for Example (84c): "Peter likes the blue and Mary likes the red ball."

(84) a. *Peter can* sing and dance.

b. *What did* Peter buy and Mary sell?

c. Peter likes the blue and Mary likes the red *ball*.

In all of the sentences in example (84), a lexical item is shared between two conjuncts. The modal *can* and the do-support auxiliary *did* are shared between the conjuncts in (84a) and (84b), respectively. In (84c), the lexical anchor *ball* is shared between the object DPs of each conjunct. Assuming the structure for (84c) given in Figure 4.10, *ball* is the lexical anchor for both the object DP modified by *blue* and that modified by *red*.

As a consequence of adopting lexical insertion, the terminal frontier nodes of elementary structures are bare heads (i.e. lacking lexical material) of the functional and lexical projections. These heads are the sites for lexical insertion. Lexical material is formalized as structure structures con-

taining two nodes: the terminal node, which is the lexical material (e.g. the word), and the root node, which is the category label of the lexical item (e.g. N, V). The lexical insertion operation, which is defined analogously to substitution, introduces these Lexical Item structures into the lexical insertion sites after the LTAG operations have occured.

Similar to substitution and adjoining sites marked for contraction, I assume that lexical insertion sites may also be marked for contraction in the enumeration. As elementary structures are constructed, these marked lexical insertion sites are merged into the structure. These nodes are indicated in the elementary structure with a circle, and the elementary structure name is subscripted with the label of the marked node. The identification notation and naming convention for nodes marked for contraction are the same for lexical insertion sites as for substitution or adjoining sites.

Lexical insertion sites marked for contraction can also be merged by the contraction function of adjoining, similar to marked substitution sites. During the LTAG derivation, these lexical insertion nodes are then merged by the contraction function of adjoining, as one elementary structure containing a marked node adjoins into another. This process yields a single multiply dominated merged node. The contraction function was initially defined only for substitution sites and was later expanded to adjoining sites in section 4.1. Here, it is extended again to merge lexical insertion sites.

Once the lexical insertion sites are merged by the contraction function of adjoining, a single lexical item is free to be inserted into the merged site, resulting in the multiple dominance and sharing of that lexical item. Thus, these cases of lexical item sharing will be derived in much the same manner as other sharing coordination constructions.

Consider the elementary structures in Figure 4.11, used to derive example (85). In the initial structure ($\alpha V_{trans\{DP,C/T\}}$) and in the verbal coordination auxiliary structure (β*conj*_$V_{trans\{DP,C/T\}}$), the fronted Wh-object and head of C/T are marked for contraction. As these two structures compose, the marked nodes DP$\downarrow$ and C/T are merged. (αDP_{Wh_object}) has substituted into the merged Wh-object substitution sites, and the two (αD_{proper_name}) structures have substituted into the subject DP substitution sites. The syntactic derived and derivation structures are provided in Figure 4.12.

(85) *What did* Peter buy and Mary sell?

The lexical item structures in Figure 4.13 compose with the syntactic derivation structure of Figure 4.12 to produce the lexically-complete derived and derivation structures shown in Figure 4.14. The key aspect of the derivation is the insertion of the lexical item (η*did*) into the derived structure.

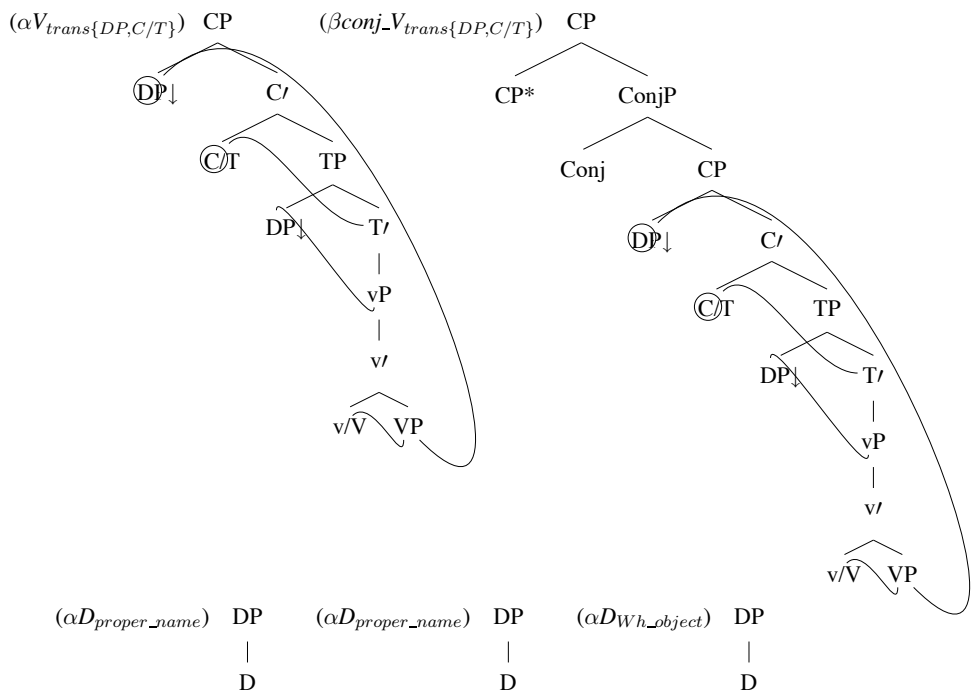


Figure 4.11: Elementary Structures to Derive (85): "What did Peter buy and Mary sell?"

The head nodes of the two verbal elementary structures, which were marked for contraction, were merged during the syntactic derivation by the contraction function of adjoining. The (*ηdid*) structure is subsequently inserted into this single lexical insertion site. In the resulting derived structure, given in Figure 4.14, the lexical item is shared between the two conjuncts and multiply dominated by the C′ and T′ nodes in each conjunct.

In summary, I have demonstrated that shared lexical item constructions can be successfully derived by utilizing a lexical insertion operation and the contraction function of adjoining. As discussed in section 3.3.2, Lexical insertion is defined as a post-syntactic substitution operation (Halle and Marantz (1993)), in which a lexical item structure is introduced into a lexical insertion site. During the syntactic derivation, elementary structures lack lexical items, with the bare heads serving as terminal nodes. The treatment of elementary structures as lexical item-less constructions also opens up the possibility of radically reducing the number of elementary structures necessary for a complete grammar. Each transitive verb in the grammar could potentially share the same basic transitive elementary structure.

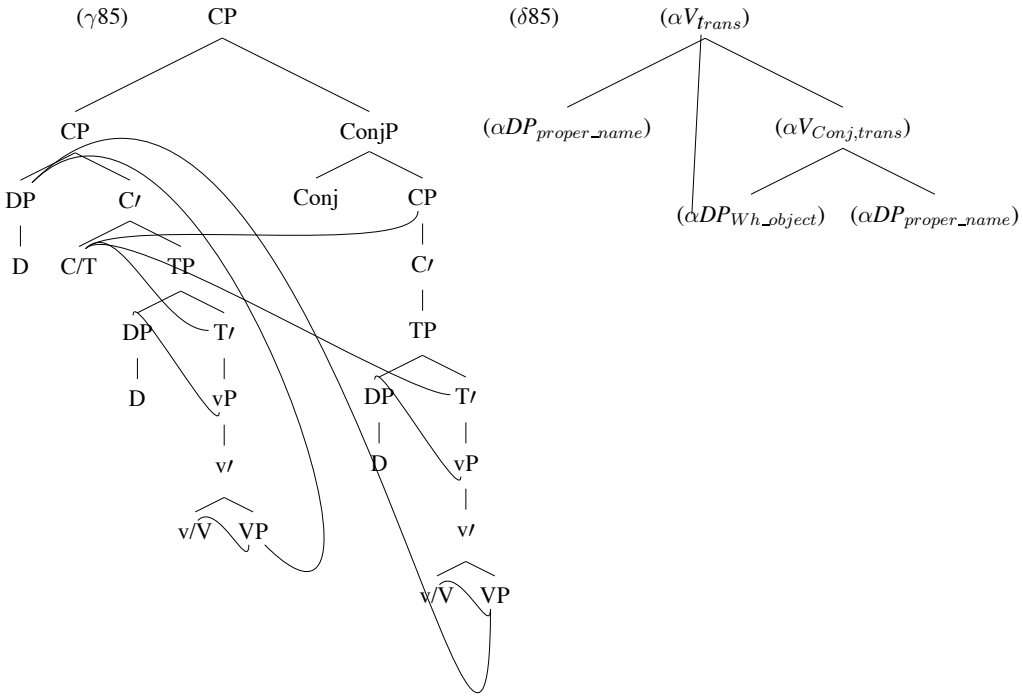


Figure 4.12: Syntactic Derived and Derivation structures of (85): "What did Peter buy and Mary sell?"

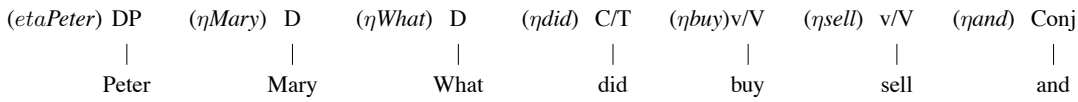


Figure 4.13: Lexical Item structures to Derive (85): "What did Peter buy and Mary sell?"

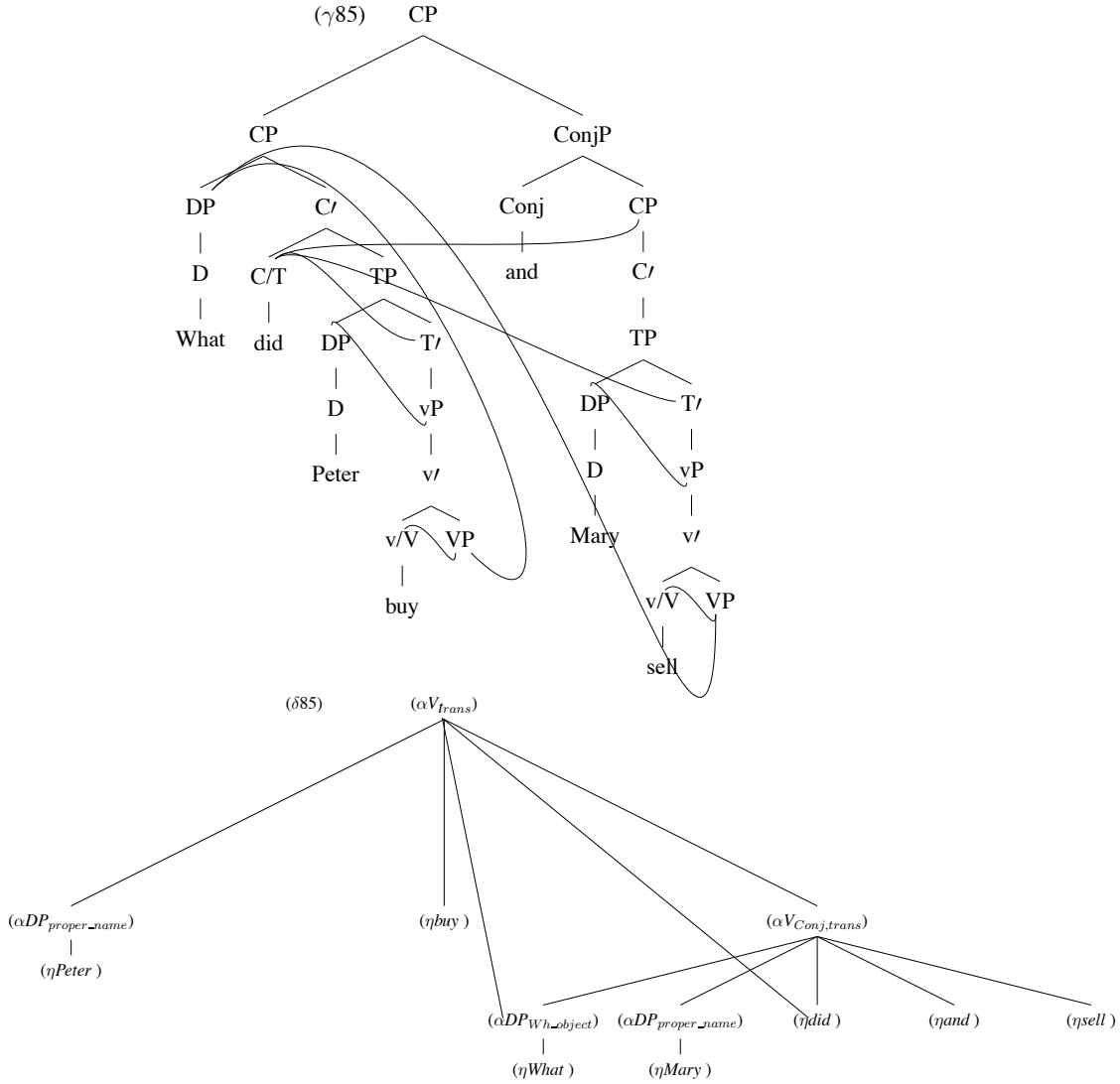


Figure 4.14: Lexically Complete Derived and Derivation structures of (85): "What did Peter buy and Mary sell?"

In Chapter 4, I have expanded the empirical domain of the analysis presented in Chapter 3 to include constructions such as modifier sharing, derivationally non-local constructions and shared lexical items. However, the structures derived by this approach must still be linearized so that the terminals are placed in the correct linear order. The linearization of a shared element must place it in the appropriate position. In the following chapter, I propose a novel linearization algorithm which correctly linearizes all of the constructions discussed in this thesis, including those constructions that were problematic for previous algorithms. Furthermore, I demonstrate that this linearization algorithm derives the Peripherality Conditions on sharing coordination constructions, where previous algorithms could not.

Chapter 5

The Linearization of Multiple Dominance Structures

As discussed in Chapter 2, the linearization of multiple dominance structures is one of the principle problems facing a multiple dominance approach. The classic linearization algorithms of Partee et al. (1990) and Kayne (1994) were designed to operate on single dominance structures and therefore cannot linearize the terminal nodes in multiple dominance structures. Thus, if we are to maintain the assumption that it is the structure of derived trees that determines the precedence relations between their terminal nodes, multiple dominance approaches must provide alternative linearization algorithms.

I discussed two distinct approaches to this linearization problem in Chapter 2 and remarked that not only were there differences between the forms of their algorithms but also in their intended range of empirical coverage. Wilder (1999) modified the Linear Correspondence Axiom to allow for and linearize multiple dominance structures. In doing so, Wilder also followed Kayne (1994) in taking the position that linearizability should be viewed as a constraint on grammaticality. That is, Wilder's algorithm cannot linearize every possible multiple dominance structure, and the structures that cannot be linearized Wilder assumes to be ungrammatical. Wilder also claimed that the Peripherality Conditions of sharing coordination constructions can be derived from the effects of the linearization algorithm.

Unlike Wilder (1999), the approach to linearization taken by Vries (2009) and Kluck (2007) does not directly appeal to the c-command relations of non-frontier nodes and their sets of domi-

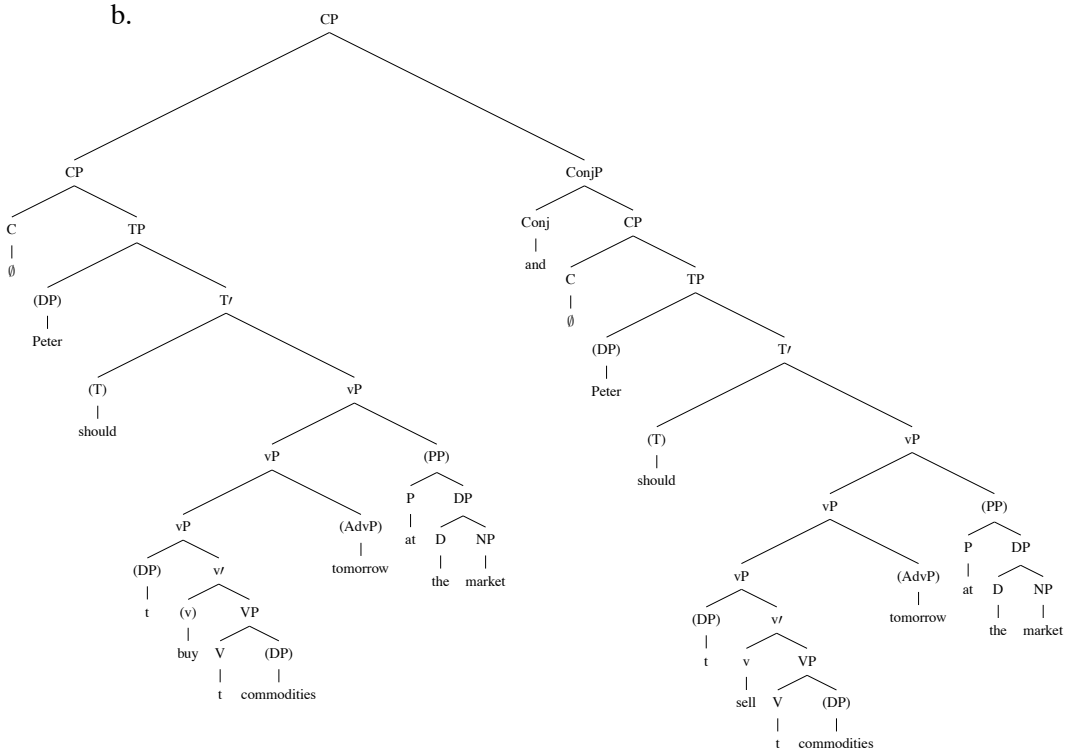
nated terminals. Instead, terminals are linearized as the result of a tree recognition algorithm. As the algorithm parses the tree, the nodes that have been parsed are marked. The position at which multiply dominated nodes are linearized depends on how many time those nodes are marked. Two interesting consequences follow from this approach. The first is that every syntactic structure can be linearized by this algorithm. Consequently, it is not possible to derive the Peripherality Conditions by the algorithm, and another cause for these conditions must be assumed. On the other hand, the algorithm is flexible enough to permit a very useful expansion to the empirical domain of multiple dominance analyses: movement can be analyzed as multiple dominance. As noted in section 3.3, this application of multiple dominance beyond sharing coordination constructions to a ubiquitous phenomenon such as movement provides strong independent motivation for the availability of multiple dominance derivations in the grammar. The algorithm of Wilder (1999), on the other hand, is unable to linearize structures where movement is assumed to involve multiple dominance[1].

However, upon closer examination, neither algorithm is entirely satisfactory. For instance, examples such as (86a) with the structure in (86b) are not correctly derived by either account. Wilder's account cannot linearize left-sharing constructions, thus failing to linearize (86a) due to the left-shared *Peter* and *should*, while that of Vries fails to linearize *should* properly: instead of linearizing it in the left conjunct, it appears in the right. In response to these failings, I will in this chapter propose a linearization algorithm that can accurately linearize all grammatical multiple dominance structures as well as derive the Peripherality Conditions. To do so, I will first define the relevant terms and structural relations and detail the assumptions that underly the algorithm. Then, in section 5.2, I describe the algorithm and provide several illustrative examples. Finally, in section 5.3, I demonstrate how this algorithm successfully derives the Peripherality Conditions on sharing coordination constructions.

(86) a. Peter should buy and sell commodities tomorrow at the market.

[1]The unavailability of movement as multiple dominance in Wilder's account is due to his definition of multiple dominance, on which the algorithm he proposed relies.

b.



5.1 Definitions and Assumptions

Following Kayne (1994), I take a set of terminals to be linearly ordered when the order of the terminals meets three conditions. First, the precedence relations between the terminals must be transitive; if a terminal x precedes terminal y, and if terminal y precedes terminal z, then terminal x must also precede terminal z. In other words, each terminal must be ordered with respect to every other terminal; a lack of total ordering constitutes a transitivity violation. Second, the terminal relations must also be irreflexive; a terminal x may not directly precede itself. Third, the terminal relations must be asymmetric, such that no terminal may precede itself indirectly. An example of an asymmetry violation would be a structure in which a terminal x precedes a terminal y and in which the terminal y also precedes x. The greater-than symbol $>$ is used to denote precedence, in that X$>$Y indicates that X precedes Y. Precedence is defined on terminal nodes.

It is not the case that the strings of every conceivable structure will be given a linear order by the proposed algorithm. The structures for which the terminals are not assigned a linear order I term unlinearizable. Following Kayne (1994), Chomsky (1995) and Wilder (1999), linearizability is considered to be a condition on the grammaticality of syntactic structures. A structure that cannot

be linearized is ungrammatical, even if it is otherwise well-formed.

The proposed linearization algorithm orders terminals nodes on the basis of the sister precedence and dominance relationships of the non-terminals that dominate them. Dominance is a relation between two nodes X and Y: node X dominates node Y if some chain of mother-daughter relations, a *path*, from Y to the root passes through X. The set of terminals dominated by X is denoted by the expression d(X). Dominance is a reflexive relation; a node X dominates itself because each path from X to the root necessarily passes through X. It follows that a node Y dominated by X may have a mother outside of X, as there may be some path from node Y that does not pass through X.

Three additional dominance relations are relevant for the linearization algorithm: full, parent and immediate dominance. Full dominance is defined such that if node X fully dominates node Y, then *every* path from Y to the root passes through X. It follows that if node Y is fully dominated by node X, then Y may have multiple mothers, but all of these mothers must be dominated by X. The set of terminals fully dominated by a node X is denoted by the expression fd(X).

The second dominance relation is immediate dominance. A node X immediately dominates a node Y if X dominates Y, and for at least one path from Y to the root that includes X, there is no node Z such that Z dominates Y and is dominated by X. I will also refer to this relation as motherhood. The set of nodes immediately dominated by node X is denoted by the expression id(X).

This notion of immediate dominance allows us to define sisterhood. Sisterhood is a relation between nodes X and Y, such that nodes X and Y are sisters iff for some node Z, X and Y are members of id(Z) and they form an ordered pair <X, Y>. This definition of sisterhood has some interesting properties. First, sisters are necessarily ordered because sisterhood is defined on ordered pairs. Thus, the sisters <X, Y> and <Y, X> are different pairs of sisters. Even though the two pairs contain the same nodes, they are different because their sister precedence relations are different. It also follows that sisterhood is irreflexive, as I assume that precedence is irreflexive. A node X cannot be its own sister.

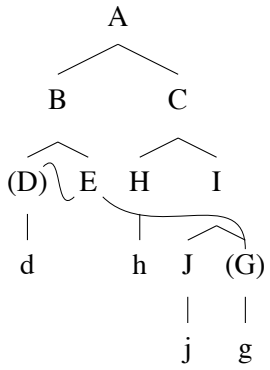
In multiple dominance structures, such as that in example (87), some nodes, such as node (G), may have more than one sister: (D) and J. Note however that (D) and J are not sisters because there is no node that immediately dominates both of them. Sisterhood is therefore not transitive. Furthermore, following the assumption that tree structures must be binary branching (due to the definition of the Merge operation), sisterhood must also be intransitive: there can be no sisters X and Y and Y and Z such that X and Z are also sisters.

In example (87), the subtree rooted by G is dominated by nodes {A, B, E, C, I, G}. The

only nodes that fully dominate G are the root node A and node G itself. Node A fully dominates G because all paths to the root node must pass through A, as A is the root. Similarly, G fully dominates itself because all paths from G to the root must pass through G. Notice also that A and G both dominate and fully dominate G, while the nodes {B, E, C, I} dominate but do not fully dominate G. For each of these nodes that simply dominate G ({B, E, C, I}), there is some path to the root that does not include that node. For example, B dominates G, but the path to the root through C and I does not include B; thus, B does not fully dominate G.

A final, novel dominance relation, parent dominance, is necessary for this linearization algorithm. A node X parent dominates node Y if X dominates Y and no node on any path from Y to X is sister to a node dominating X. The set of terminals parent dominated by node X is denoted by the expression pd(X). Thus in example (87), node E parent dominates nodes (G) and g; however, it does not parent dominate nodes (D) and d because (D) is sister to E, which dominates itself. The set of terminals pd(E) is therefore {g}.

(87)



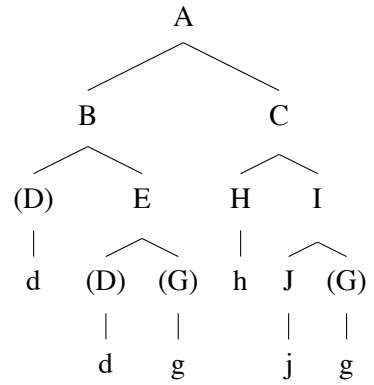
As noted in section 2.3 of Chapter 2, some notions of c-command are incompatible with multiple dominance structures. Accordingly, I provide a definition of c-command in terms of sisterhood and parent dominance, which is compatible with multiple dominance. A node X c-commands node Y if X and Y are sisters, or if Y is parent dominated by the sister of X. In example (87), node (D) c-commands nodes E, (G) and g. Note that (D) does not c-command itself, as (D) is not parent dominated by the sister of (D), E. Similarly, node (G) c-commands nodes (D), d, J and j. A similar sister-containment approach to c-command is used in Chomsky (2000), though the use of parent dominance in this definition allows reflexive c-command to be avoided.

Finally, it is necessary to formally define the terms multiple dominance and single dominance. A node X is multiply dominated if there is more than one node that immediately dominates X, and

a node X is singly dominated if that node is immediately dominated by exactly one node. In (87), the nodes (D) and (G) are multiply dominated, a status which is indicated by the parentheses. It is also interesting to note that while (D) and (G) are multiply dominated, d and g are not. Each of these terminals has exactly one mother. Thus, the children of a multiply dominated parent are not necessarily multiply dominated themselves.

In future examples, the sheer number of multiply dominated nodes in each tree will preclude the manner of depiction shown in (87). Instead, a copy of any subtree rooted by a multiply dominated node will be included as daughter to each of the multiply dominated node's mothers. Thus, (87) will be equivalently depicted as (88).

(88)



5.2 The Linearization Algorithm

The proposed linearization algorithm is provided in (89). The algorithm delineates three types of precedence relations between the terminals or subsets of terminals dominated by pairs of sisters. If the sisters X and Y are both multiply dominated, or if they are both singly dominated, then the fully dominated terminals of X precede those terminals fully dominated by Y. On the other hand, the relevant subsets of the terminals dominated by X and Y are different if either X or Y is multiply dominated while the other is singly dominated. For the multiply dominated sister the relevant subset is that of its fully dominated terminals, and the relevant subset for the singly dominated sister is that of its parent dominated terminals. In either case, it is the relevant subset of the terminals dominated by X that precede those dominated by Y.

(89) For all sisters $<$X, Y$>$:

a. if X and Y are both multiply dominated or are both singly dominated then fd(X) $>$ fd(Y)

Sister Precedence	Terminal Precedence
B > C	d > h, j
(D) > E	d > g
(D) > (G)	d > g
H > I	h > j
J >(G)	j > g
Final Linearization:	d h j g

Table 5.1: Sister and Terminal precedence relations for example (88)

b. if X is multiply dominated and if Y is singly dominated then fd(X) > pd(Y)

c. if X is singly dominated and if Y is multiply dominated then pd(X) > fd(Y)

The results of the algorithm are provided in the form of a table, such as in Table 5.1 for example (88). The left column delineates all of the sister relations in the structure. The right column lists the terminal precedence relations dictated by the algorithm. The sisters <B, C> are both singly dominated, so the fully dominated terminals of B (i.e. d) precede those of C (i.e. h and j). The terminal g is neither fully dominated by B or C. Between the sisters <(D), E>, the sister E is singly dominated, while the sister (D) is multiply dominated; therefore clause (89b) of the algorithm applies. Only the terminal g is parent dominated by E, as the path between the terminal d and E contains (D), which c-commands E. Thus, the terminals fully dominated by (D) precede those parent dominated by E, such that d > g. The final linearization is the union of these terminal precedence relations.

In section 3.5, I presented a multiple dominance analysis of the sentence in example (90a) with its associated derived structure provided in (90b). As noted earlier, the roots of multiply dominated subtrees are marked with parentheses. The subject DP *Tim* is multiply dominated both within each conjunct and between conjuncts, as it is the specifier of the vP and TP in each conjunct. The verbs *bought* and *shared*, as the heads of VP and vP, are multiply dominated within each conjunct.

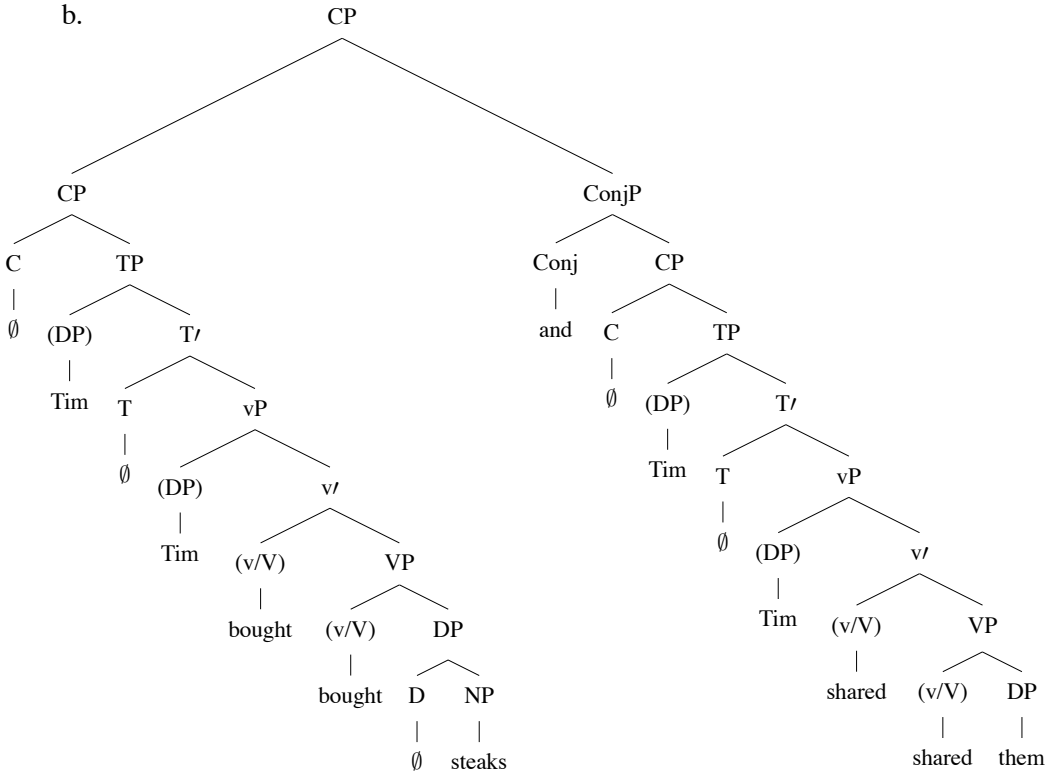
The linearization of (90a) is detailed in Table 5.2 in which, certain sister relations are of particular interest. First, note that the pair <CP, ConjP> are both singly dominated; thus, the terminals fully dominated by CP precede those fully dominated by ConjP. The subject *Tim* is not fully dominated by CP or ConjP and is therefore not linearized with respect to the terminals fully dominated by CP and ConjP. The verbs *bought* and *shared*, while being multiply dominated, are each fully

dominated within their respective conjuncts, resulting in *bought* preceding *shared*. Together, the set of terminals fully dominated by CP is {*bought, steaks*}, and these terminals precede those fully dominated by ConjP, {*and, shared, them*}.

The sisters <(DP), T′> also warrant further discussion. Because (DP) is multiply dominated and T′ is not, the fully dominated terminals of (DP) precede the parent dominated terminals of T′. Note that while T′ dominates the terminal *Tim*, the (DP) node that dominates *Tim* is on the path from *Tim* to T′. Consequently, *Tim* is not parent dominated by T′, ensuring that *Tim* does not precede itself. The terminals that *Tim* does precede are those parent dominated by T′: {bought, steaks}. According to the terms of the linearization algorithm, the structure in (90b) is linearizable; the linearized string is given in the last row of the Table 5.2: "Tim bought steaks and shared them".

(90) a. *Tim* bought steaks and shared them.

b.



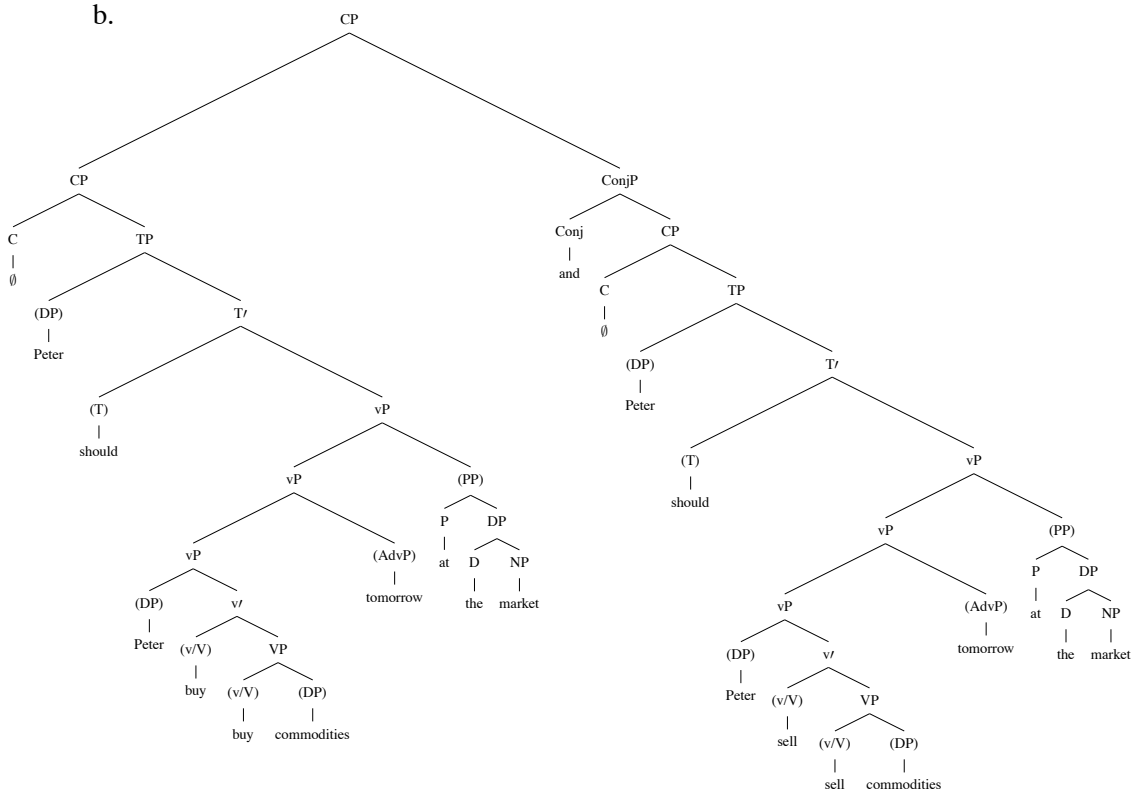
This algorithm can also derive the example given in (91a). As depicted in (91b), the subject DP subtree is multiply dominated, as it is dominated by the vP and TP of both conjuncts. As the head of both vP and VP, the verb is also multiply dominated. The linearization of the structure in (91b) is outlined in Table 5.3, and the algorithm is able to establish a complete and consistent ordering.

Sister Precedence	Terminal Precedence
CP > ConjP	bought, steaks > and, shared, them
C > TP	∅ > bought, steaks
(DP) > T′	Tim > bought, steaks
T > vP	∅ > bought, steaks
(DP) > v′	Tim > bought, steaks
(v/V) > VP	bought > steaks
(v/V) > DP	bought > steaks
D > NP	∅ > steaks
Conj > CP	and > shared, them
C > TP	∅ > shared, them
(DP) > T′	Tim > shared, them
T > vP	∅ > shared, them
(DP) > v′	Tim > shared, them
(v/V) > VP	shared > them
(v/V) > DP	shared > them
Final Linearization:	Tim bought steaks and shared them.

Table 5.2: Sister and Terminal precedence relation for example (90b)

(91) a. Peter should buy and sell commodities tomorrow at the market.

b.



It is notable that this algorithm is capable of linearizing the structure in (91b) where the algorithms of Wilder (1999) and Vries (2009) and Kluck (2007) have failed. The algorithm of Vries (2009) and Kluck (2007) incorrectly linearizes the shared modal subtree *should* in the right conjunct; in the proposed algorithm, the sister pair<(T), vP> in the left conjunct correctly places the modal to the left of the vP. Because (T) is multiply dominated and vP is singly dominated, the fully dominated terminals of (T) precede those parent dominated by vP. This means that *should* precedes the set {*buy, commodities, tomorrow, at, the, market*}. Similarly, in the right conjunct, the sister pair <(T), vP> dictates that *should* precedes the set {*sell, commodities, tomorrow, at, the, market*}. These two sister pairs therefore ensure that the modal is pronounced to the left of both vPs. Meanwhile, the sister pairs <(DP), T′> in the left and right conjuncts ensure that the modal will be pronounced after *Peter*. Finally, the sister pair <Conj, CP> orders only the terminals fully dominated by Conj and CP. Because *should* is not fully dominated by this CP, it is not linearized after *and*, thus avoiding the asymmetry violation where *should* would both precede and follow *and*.

Sister Precedence	Terminal Precedence
CP > ConjP	buy > and, sell
C > TP	∅ > buy
(DP) > T′	Peter > should, buy, commodities, tomorrow, at, the, market
(T) > vP	should > buy, commodities, tomorrow, at, the, market
vP > (PP)	Peter, buy, commodities, tomorrow > at, the, market
vP > (AdvP)	Peter, buy, commodities > tomorrow
(DP) > v′	Peter > buy, commodities
(v/V) > VP	buy > commodities
(v/V) > (DP)	buy > commodities
P > DP	at > the, market
D > NP	the > market
Conj > CP	and > sell
C > TP	∅ > sell
(DP) > T′	Peter > should, sell, commodities, tomorrow, at, the, market
(T) > vP	should > sell, commodities, tomorrow, at, the, market
vP > (PP)	Peter, sell, commodities, tomorrow > at, the, market
vP > (AdvP)	Peter, sell, commodities > tomorrow
(DP) > v′	Peter > sell, commodities
(v/V) > VP	sell > commodities
(v/V) > (DP)	sell > commodities
Final Linearization:	Peter should buy and sell commodities tomorrow at the market.

Table 5.3: Sister and Terminal precedence relation for example (91b)

The algorithm of Wilder (1999) runs into problems with the sets of shared sub-trees at either edge of the conjuncts, in that no linear order is established between the object DP, AdvP and PP subtrees. Additionally, left-sharing constructions produce reflexivity violations and therefore the linearization fails to be consistent and complete. The proposed algorithm, on the other hand, does establish the correct linear ordering. To exemplify this, examine the sister pair <vP, (PP)> in the right conjunct.[2] As vP is singly dominated, while (PP) is multiply dominated, the set of terminals parent dominated by vP, {*sell, commodities, tomorrow*} precede the fully dominated terminals of (PP) {*at, the, market*}. The relationship between the other shared sub-trees is derived in a similar manner by the other sister pairs.

5.3 Peripherality Conditions

As discussed in Chapter 1, sharing coordination constructions are subject to Peripherality Conditions. An externally right-shared element must appear at the right edge of the right conjunct, while a left-shared element is necessarily found at the left edge of the left conjunct. External sharing can be defined as the sharing of an element between two or more conjuncts, and internal sharing refers to within-conjunct sharing. In the grammatical example (92a), *beans* is externally right-shared and appears right-peripheral in the right conjunct. If *beans* appears non-peripherally in the right conjunct, as in (92b), the example becomes ungrammatical. Similarly, an externally left-shared element must be leftmost in the left conjunct, as demonstrated by the contrast between (93a), in which the externally shared material *What did Spencer* is at the left of the left conjunct, and (93b), where the shared subject *Spencer* is not.

Notice that the left most element in example (93b) is an internally shared element. In the multiple dominance approach to movement adopted here, the Wh-word *what* is shared between the CP and VP in the left conjunct. Thus, there is a contrast between internal and external sharing; it appears that only external sharing is constrained by the Peripherality Condition.

(92) a. I like and Peter hates *beans*.

b. * I like in the morning and Peter hates *beans* in the evening.

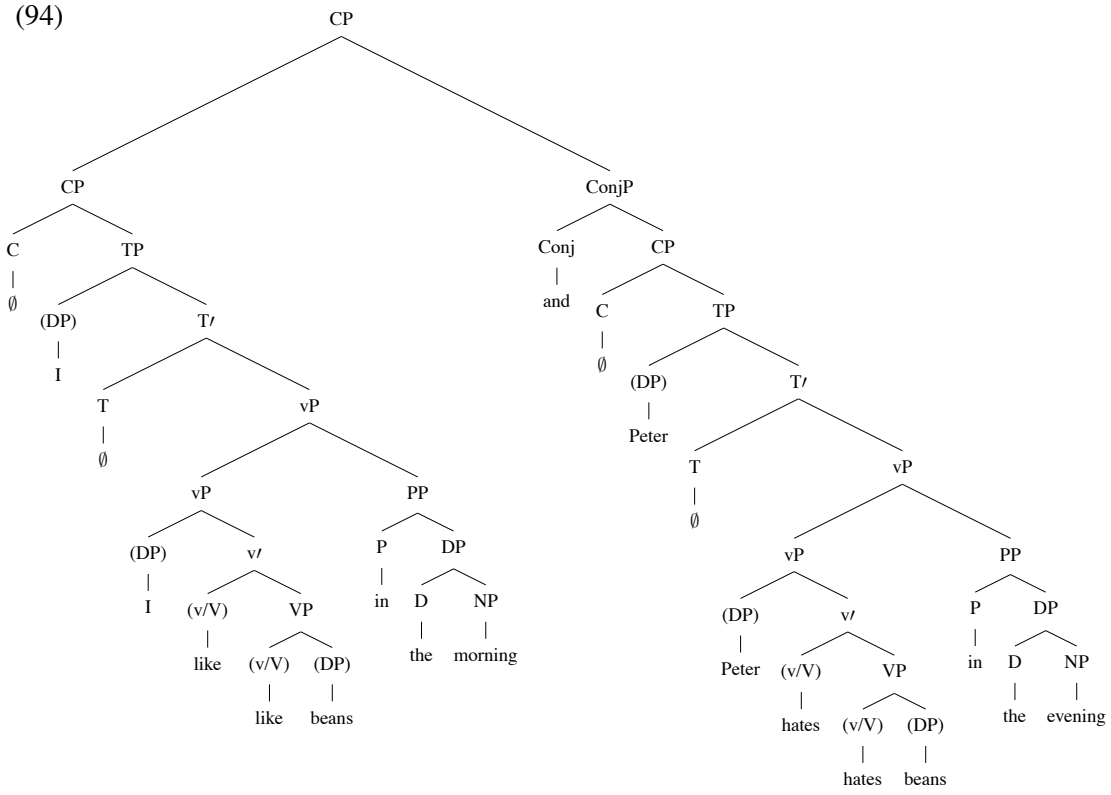
[2] A number of other sister pairs in the tree, e.g. the <vP, (PP)> sister in the left conjunct, or the <vP, (AdvP)> or <(T), vP> sisters in either conjunct, would serve to demonstrate the same point, namely that a precedence relation is established between the shared subtrees in question.

(93) a. *What did Spencer* cook and eat?

b. * What did *Spencer* cook and who did meet?

I assume the structure in example (94) for (92b). The externally shared object *beans* is complement to the VP in each conjunct, and the non-shared PPs are adjoined to each of the vPs. Table 5.4 contains the results of the linearization algorithm applied to (94); however, the ordering of terminals is incomplete. In the right conjunct, the externally shared DP is never linearized with respect to the modifying PP. In the right conjunct, the sister pair <vP, PP> results in {hates} > {in the evening}, while the sisters <(DP), v′>, <(v/V), VP> and <(v/V), (DP)> together yield the terminal relations Peter > hates > beans. Although *beans* is linearized to follow *hates*, no precedence relation is ever established between *beans* and the terminals of the modifying non-shared PP. Therefore, the structure cannot be fully linearized, resulting in ungrammaticality.

(94)



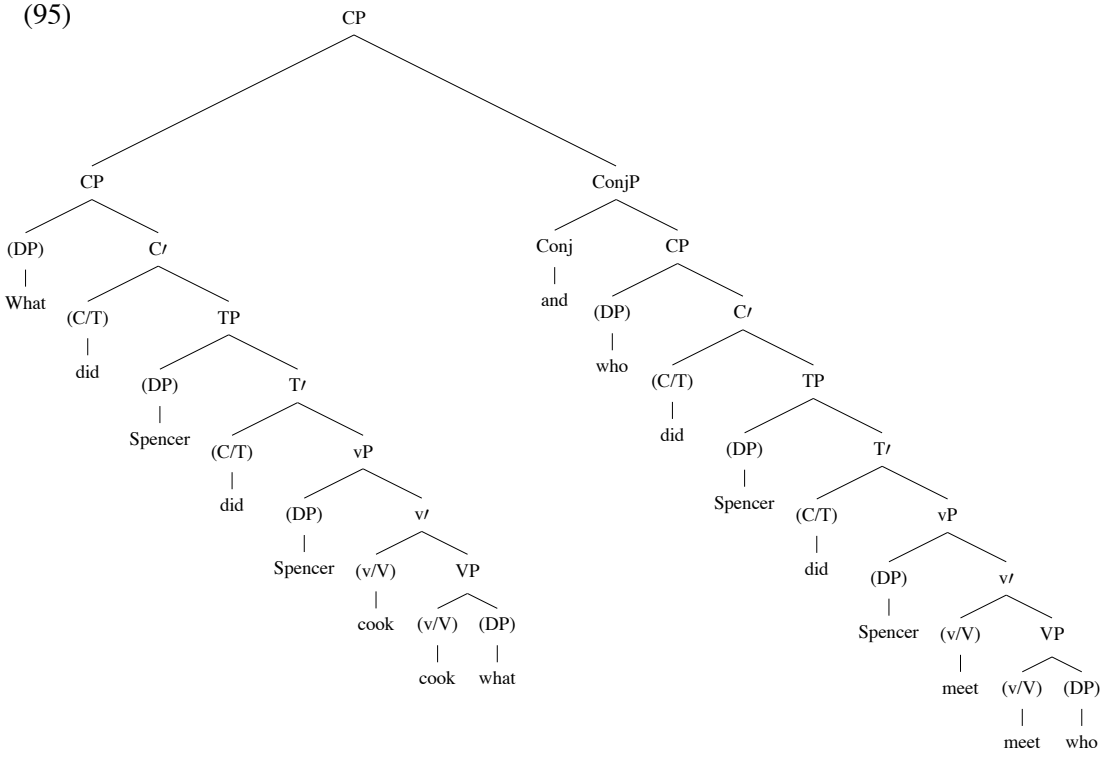
Similarly, the structure for example (93b) cannot be linearized, as demonstrated in Table 5.5. While the structure in (94) was unlinearizable because the terminals were not fully linearized, the unlinearizability of (95) is due to an asymmetry violation. In the left terminal, the sisters <(C/T),

Sister Precedence	Terminal Precedence
CP > ConjP	like > and, hates
C > TP	∅ > like
(DP) > T′	I > ∅, like, beans, in, the, evening
(T) > vP	∅ > like, beans, in, the, evening
vP > (PP)	I, like, beans > at, the, market
(DP) > v′	I > like, beans
(v/V) > VP	like > beans
(v/V) > (DP)	like > beans
P > DP	in > the, morning
D > NP	the > morning
Conj > CP	and > hates
C > TP	∅ > hates
(DP) > T′	Peter > ∅, hates, beans, in, the, evening
(T) > vP	∅ > hates, beans, tomorrow, in, the, evening
vP > (PP)	Peter, hates, beans, tomorrow > in, the, evening
(DP) > v′	Peter > hates, beans
(v/V) > VP	hates > beans
(v/V) > (DP)	hates > beans
P > DP	in > the, evening
D > NP	the > evening
Final Linearization:	?

Table 5.4: Right peripherality violation in example (92b) excluded by linearization

TP> and <(DP), T′ > together order *did* before *Spencer*, and *Spencer* before *cook*. The corresponding sisters in the right conjunct, <(C/T), TP> and <(DP), T′ >, place *Spencer* between *who* and *meet*. This is problematic in that *Spencer* is linearized medially in both conjuncts.

(95)



In sum, the novel algorithm presented in this chapter overcomes a number of challenges faced by previous multiple dominance approaches. The algorithm is capable of successfully linearizing all types of multiple dominance constructions, including left and right sharing coordination constructions as well as movement recast as multiple dominance. Additionally, the peripherality constraints on sharing constructions follow from the formulation of the algorithm, without appealing to external mechanisms.

Sister Precedence	Terminal Precedence
CP > ConjP	what, did, cook > and, who, did, meet
(DP) > C′	what > did, Spencer, cook
(C/T) > TP	did > Spencer, cook
(DP) > T′	Spencer > did cook
(C/T) > vP	did > cook
(DP) > v′	Spencer > cook
(v/V) > VP	cook > ∅
(v/V) > (DP)	cook > ∅
Conj > CP	and > who, did, meet
(DP) > C′	who > did, Spencer, meet
(C/T) > TP	did > Spencer, meet
(DP) > T′	Spencer > meet
(C/T) > vP	did > meet
(DP) > v′	Spencer > meet
(v/V) > VP	meet > ∅
(v/V) > DP	meet > ∅
Final Linearization:	What did (Spencer) cook and who (Spencer) did meet

Table 5.5: Right peripherality violation in example (93b) excluded by linearization

Chapter 6

Conclusion

6.1 Summary

This thesis has been concerned with developing an analysis of sharing coordination constructions, such as those in (96). I argued for a multiple dominance analysis of these constructions and formalized this analysis in a lexicalized tree adjoining grammar (LTAG) framework. This analysis extends the empirical domain of a previous TAG analysis of sharing constructions (Sarkar and Joshi (1996)) beyond shared arguments to provide an account of a broader range of sharing constructions, including shared modifiers, lexical items and derivationally non-local sharing. Finally, I defined a linearization algorithm that produces a linear ordering between the terminals of syntactic structures. Additionally, this algorithm produces the novel result of deriving the peripherality conditions on both left and right sharing constructions.

(96) a. *Alexi* cooked and ate *the bacon*.

b. Tim suspects and Spencer is sure *that Portland is one of the greatest cities in the world*.

c. James knows *what would* make Sal happy and please Sue.

d. The kid knows what would and what wouldn't *kill the wolf-man*.

e. Spencer will and Alexi probably won't *visit*.

f. *Alexi usually* wraps his dinner in bacon and drinks beer.

g. Jamie likes the white and I like the red *fish*.

In chapter 1, I discussed the distinguishing characteristics of sharing coordination constructions and how they differ from other phenomena associated with coordinate structures, such as gapping and verb phrase deletion. The key diagnostic that distinguished sharing coordination constructions are the peripherality conditions, provided in (97). Material in the left conjunct must appear at the left edge of the left conjunct and material in the right conjunct must appear at the right edge of this conjunct. These conditions do not apply to gapping and verb phrase deletion. Additionally, in sharing coordination constructions, but not in gapping or verb phrase deletion constructions, the category and constituency of the shared material is relatively unconstrained. That is, while gapping and verb phrase deletion only target verbal constituents, sharing may target multiple heads or maximal projections of nearly any category. Finally, sharing may occur bidirectionally, while verb phrase deletion and gapping may only delete material which follows the antecedent. Thus, sharing coordination constructions seem to form a natural class, in contrast with gapping and verb phrase deletion constructions.

(97) Peripherality Conditions (adapted from Sabbagh (2007))

a. In the configuration [$_A$...X...] CONJ [$_B$...Y...]

b. If X is shared, then it must be at the left edge of A, and if Y is shared, then it must be at the right edge of B.

These constructions resist analysis in terms of simple clausal or coordinate structures, and it seems that some additional mechanism is necessary. In Chapter 2, I examine three types of approaches to the syntax of sharing coordination constructions. I argue that the literal movement and reduction approaches face serious empirical and conceptual problems, which a multiple dominance approach largely avoids. However, a multiple dominance account does face the problem of linearization, in which the mapping from syntactic structure to the linear ordering of terminals is challenging. I review several approaches to this problem but find them lacking both empirically and conceptually.

In Chapter 3, I present a multiple dominance analysis of sharing coordination constructions formulated in an LTAG framework. This account revises and extends the analysis presented in Sarkar and Joshi (1996), and can successfully account for shared arguments. I adopted LTAG because it provides a mathematically explicit and formally precise framework for the formulation and testing of linguistic hypotheses. Additionally, LTAG is attractive, in that certain linguistic generalizations can be derived from the properties of the framework itself, without further stipulation. The revi-

sions to the analysis of Sarkar and Joshi (1996) were designed to increase the empirical range and conceptual simplicity of the analysis. These revisions include the adoption of a multiple dominance view of movement and of asymmetric coordinate structures, where the coordinator is the head of a functional projection ConjP that takes CP as complement and specifier.

Chapter 4 extends this analysis to include shared modifiers, lexical items, and non-local sharing constructions. The derivation of shared modifiers requires that the adjoining operation that functions on adjoining sites marked for contraction operates slightly differently than typical adjoining. The effect of this shift in functionality is that no portions of the verbal elementary tree spines are shared, resulting in the correct derived structure. Derivationally non-local sharing is accomplished by the adoption of delayed-tree local multi-component TAG, an extension to LTAG in which the non-locality of these cases does not impact the formal generative power of the formalism. Shared lexical items are analyzed by extending the sharing mechanisms described in Chapter 3 to an explicit lexical insertion operation. This extension allows shared lexical items to be analyzed in a manner analogous to shared arguments.

In Chapter 5, I define a linearization algorithm which produces a linear ordering of the terminals in syntactic structures. This algorithm is defined in terms of several relations between nodes: sisterhood, dominance, full dominance and parent dominance. The result is that certain multiple dominance structures can be linearized, including grammatical left and right sharing constructions. However, the algorithm will not linearize all multiple dominance structures, thus acting as a constraint on possible sharing coordination constructions. One important class of sharing coordination constructions ruled out by the linearization algorithm are structures which violate the peripherality conditions. The peripherality conditions are a key characteristic of sharing coordination constructions, and consequently, it is very interesting that both the left and right peripherality conditions can be derived from an aspect of the grammar which is necessary outside of sharing coordination constructions.

6.2 Future Work

Due to space constraints, I have been unable to address certain issues relevant to sharing coordination constructions. One issue concerns examples such as (98), where the shared verb *likes* agrees with the subject of each conjunct separately. Though examples like these are ungrammatical, the account here predicts them to be acceptably derived as right sharing constructions. These unavailable

derivations are not just a problem for multiple dominance analyses of sharing; all of the accounts discussed in this thesis predict that constructions such as these are possible.

(98) * Peter and Jane *likes sandwiches*.

Another interesting aspect of sharing coordination constructions that I have not addressed in this thesis is the possibility that shared elements can be licensed in only one conjunct, as discussed in Ha (2008). That is, in some sharing constructions, such as those given in (99), the shared element may fail to follow typical agreement or licensing requirements. In (99a), the noun *essay* does not agree in number with the numeral of the left conjunct. The negative polarity item *any* in (99b) is licensed by the negation in the right conjunct, while no such licensor exists in the left conjunct. In (99c), the shared R-expression *John* is free in the right conjunct, but C-commanded by a co-indexed R-expression, and thus illicit, in the left conjunct. The corresponding non-sharing constructions of these examples, given in (100), are ungrammatical. Notice that the shared element must be licit for the nearest conjunct; in the ungrammatical (101), the shared elements have been recast as agreeing only with the far conjunct.

(99) a. Bill has to read two, and Mary must write one *essay by tomorrow*.

b. John read, but he hasn't understood *any of my books*.

c. Mary heard that John submitted, but Sue said that Bill actually wrote *the article about John for the magazine*.

(100) a. * Bill has to read two essay, and Mary must write one essay by tomorrow.

b. * John read any of my books, but he hasn't understood any of my books.

c. * Mary heard that John submitted the article about John for the magazine, but Sue said that Bill actually wrote the article about John for the magazine.

(101) a. * Bill has to read two, and Mary must write one essays by tomorrow.

b. * John hasn't understood, but he has read *any of my books*.

c. * Mary heard that John$_i$ submitted, but Sue said that Bill$_j$ actually wrote *the article about himself$_i$ for the magazine*.

It seems to me that, although the examples given in (99) are not completely ungrammatical, in general they are also much less acceptable than sharing constructions in which the shared element

agrees with both conjuncts. This intuition is supported by experimental evidence of VP sharing in Dutch by Kluck and Zwart (2009), who found constructions in which a shared VP agreed in number with only the nearest conjunct were in general less acceptable than sharing constructions in which the shared material agreed with both conjuncts.

I take these differences in acceptability to indicate that these mismatched cases involve some derivational mechanism that is distinct from those of typical matched sharing constructions. The fact that the shared element must at least agree with the nearest conjunct suggests that agreement must be satisfied after linearization; it is only the position in which the shared element is linearized that is relevant for matching effects. One possible explanation for these facts is that the lexical insertion operation, which inserts the appropriate lexical item into syntactic structure, is influenced by the linear order of terminals resulting from the linearization algorithm. Thus, the lexical insertion operation selects a lexical item that is appropriate for the precedence relations of the terminals. While this approach seems plausible, additional data are necessary for these cases, as the judgments are subtle and variable between speakers, languages and constructions.

Thirdly, while I have focused on English in this thesis, sharing constructions exist in a variety of languages (Haspelmath (2000)). Most importantly, it is necessary to test the cross-linguistic validity of the peripherality conditions. The basic approach to sharing coordination constructions taken here assumes that sharing constructions are a natural class identifiable by the adherence to the peripherality conditions. This assumption is supported by the facts of English sharing constructions, which do adhere to these conditions. In Chapter 5, I showed how the conditions can be derived from the algorithm which linearly orders the terminals of syntactic structures. I assume that the linearization algorithm, like the other mechanisms that construct syntactic structures, does not vary cross-linguistically. It follows then that the analysis presented in this thesis predicts that the peripherality conditions should hold cross-linguistically; however additional research is necessary to determine if this prediction is borne out.

The analysis presented here also predicts that sharing constructions are universally available to the syntactic component of the grammar. In a given language if certain sharing constructions are unavailable (e.g. the ungrammaticality of right or left-sharing), it is predicted that some non-syntactic mechanism must be responsible. Such a non-syntactic constraint on sharing constructions can be seen in the degraded acceptability of English right sharing constructions which lack contrastive focus on the word preceding the shared element. For example, Hartmann (2000) attributes the ungrammaticality of (102) to the fact that *I'd* and *he'll* cannot receive contrastive focus.

(102) * I think that I'd and I know that he'll *buy one of those portraits of Elvis*.

Finally, an extension to LTAG, known as Synchronous TAG (STAG), was used to develop a syntax-semantics interface for the argument sharing constructions discussed in Chapter 3 (Han et al. (2008)). That approach should be extensible to the portions of the analysis presented in Chapter 4.

Bibliography

Abbott, Barbara. 1976. Right node raising as a test for constituency. *Linguistic Inquiry* 7:639–642.

Abels, Klaus. 2004. Right Node Raising: Ellipsis or across the board movement? In *Proceedings of the North East Linguistic Society 34*, ed. Keir Moulton and Matthew Wolf, volume 1, 45–60. University of Massachusetts at Amherst: Graduate Linguistic Student Association.

Akmajian, Adrian. 1975. More evidence for an NP cycle. *Linguistic Inquiry* 6:115–130.

An, Duk-Ho. 2008. Lower Copy Pronunciation and Multiple Wh-Fronting in ATB Contexts. *Nanzan Linguistics: Special Issue 3* 2:1–15.

Becker, T., A.K. Joshi, and O. Rambow. 1991. Long-Distance Scrambling and Tree Adjoining Grammars. In *Proceedings of the fifth conference on European chapter of the Association for Computational Linguistics*, 21–26. Association for Computational Linguistics Morristown, NJ, USA.

Bošković, Željko, and Steven Franks. 2000. Across-the-board movement and LF. *Syntax* 3:107–128.

Burton, Strang, and Jane Grimshaw. 1992. Coordination and VP-internal subjects. *Linguistic Inquiry* 23:305–312.

Chiang, D., and T. Scheffler. 2008. Flexible Composition and Delayed Tree-Locality. In *The Ninth International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+ 9)*, 17–24.

Chomsky, Noam. 1957. *Syntactic structures*. Mouton.

Chomsky, Noam. 1981. *Lectures on government and binding*. Dordrecht, The Netherlands: Foris Publications.

Chomsky, Noam. 1995. *The minimalist program*. Cambridge, Massachusetts: MIT Press.

Chomsky, Noam. 2000. Minimalist inquiries: The framework. In *Step by step: Essays on minimalist syntax in honor of Howard Lasnik*, ed. Roger Martin, David Michaels, and Juan Uriagereka, 89–156. MIT Press.

Culicover, Peter W., and Michael S. Rochemont. 1990. Extraposition and the Complement Principle. *Linguistic Inquiry* 21:23–47.

Dougherty, Ray C. 1970. A Grammar of Coördinate Conjoined Structures, I. *Language* 46:850–898.

Dougherty, Ray C. 1971. A Grammar of Coördinate Conjoined Structures, II. *Language* 47:298–339.

Féry, Caroline, and Katherina Hartmann. 2005. The Focus and Prosodic Structure of German Right Node Raising and Gapping. *The Linguistic Review* 22:69–116.

Fox, D., and D. Pesetsky. 2005. Cyclic linearization of syntactic structure. *Theoretical Linguistics* 31:1–45.

Fox, Danny. 2000. *Economy and semantic interpretation*. Cambridge, Massachusetts: MIT Press.

Frank, Robert. 2002. *Phrase Structure Composition and Syntactic Dependencies*. MIT Press.

Frank, Robert. 2006. Phase Theory and Tree Adjoining Grammar. *Lingua* 116:145–202.

Frank, Robert. 2008. Reflexives and TAG Semantics. In *The Ninth International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+ 9)*, 97–104.

Gärtner, Hans-Martin. 1997. Generalized Transformations and Beyond. Doctoral Dissertation, University of Franfurt/Main.

Gazdar, Gerald. 1981. Unbounded Dependencies and Coordinate Structure. *Linguistic Inquiry* 12:155–184.

Gleitman, Lila. 1965. Coordinating Conjunctions in English. *Language* 41:260–293.

Goodall, Grant. 1983. A Three-dimensional Analysis of Coordination. In *Proceedings of the Chicago Linguistic Society 19*.

Grimshaw, Jane. 2000. Extended Projection and Locality. In *Lexical specification and insertion*, ed. P. Coopmans, M. Everaert, and J. Grimshaw. Benjamins.

Ha, Seungwan. 2006. Multiple Dominance CAN'T, but Ellipsis CAN account for Right Node Raising. In *Proceedings of the 42nd annual Chicago Linguistic Society*.

Ha, Seungwan. 2008. Ellipsis, Right Node Raising and Across-the-board Constructions. Doctoral Dissertation, Boston University.

Halle, Morris, and Alec Marantz. 1993. Distributed Morphology and the Pieces of Inflection. In *The view from building 20*, ed. Kenneth Hale and Samuel Jay Keyser, 111–176. Cambridge, Massachusetts: MIT Press.

Han, Chung-hye, and Nancy Hedberg. 2008. Syntax and Semantics of *It*-Clefts: A Tree Adjoining Grammar Analysis. *Journal of Semantics* 25:345–380.

Han, Chung-hye, David Potter, and Dennis Storoshenko. 2008. Compositional Semantics of Coordination using Synchronous Tree Adjoining Grammar. In *The Ninth International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+ 9)*, 33–40.

Harley, Heidi, and Rolf Noyer. 1999. Distributed Morphology. *Glot International* 4:3–9.

Hartmann, Katherina. 2000. *Right Node Raising and Gapping*. John Benjamins Publishing Company.

Haspelmath, Martin. 2000. Coordination. In *Language typology and linguistic description*, ed. Timothy Shopen, 1–51. Cambridge, 2nd edition.

Hudson, Richard A. 1976. Conjunction reduction, gapping and right-node raising. *Language* 52:535–562.

Johannessen, Janne Bondi. 1998. *Coordination*. Oxford University Press.

Joshi, A.K. 1987. An Introduction to Tree Adjoining Grammars. *Mathematics of language* 1:87–115.

Joshi, Aravind K., Laura Kallmeyer, and Maribel Romero. 2004. Flexible Composition in LTAG: Quantifier Scope and Inverse Linking. In *Computing meaning*, ed. Harry Bunt and Reinhard Mustens, volume 3 of *Studies in Linguistics and Philosophy*. Dordrecht: Kluwer.

Joshi, Aravind K., and Seth Kulick. 1997. Partial Proof Trees as Building Blocks for a Categorial Grammar. *Linguistics and Philosophy* 20:637–667.

Joshi, Aravind K., and Yves Schabes. 1997. Tree-Adjoining Grammars. In *Handbook of formal languages, vol. 3: beyond words*, 69–123. New York, NY, USA: Springer-Verlag New York, Inc.

Kallmeyer, Laura, and Aravind K. Joshi. 2003. Factoring Predicate Argument and Scope Semantics: Underspecified Semantics with LTAG. *Research on Language and Computation* 1:3–58.

Kayne, Richard S. 1994. *The Antisymmetry of Syntax*. Cambridge, Massachusetts: MIT Press.

Keller, F. 1995. Towards an Account of Extraposition in HPSG. In *Proceedings of the seventh conference on European chapter of the Association for Computational Linguistics*, 301–306. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.

Kluck, M., and J.W. Zwart. 2009. Good Neighbors or Far Friends: Matching and Proximity Effects in Dutch Right Node Raising. *Groninger Arbeiten zur germanistischen Linguistik* 48.

Kluck, Marlies. 2007. The Perspective of external Remerge on Right Node Raising. In *CamLing 2007*, 130–137.

Levine, Robert D. 1985. Right Node (non)-Raising. *Linguistic Inquiry* 16:492–497.

McCawley, James D. 1982. Parentheticals and Discontinuous Constituent Structure. *Linguistic Inquiry* 13:91–106.

McNally, Louise. 1992. VP Coordination and the VP-Internal Subject Hypothesis. *Linguistic Inquiry* 23.

Merchant, Jason. 2001. *The Syntax of Silence: Sluicing, Islands, and the Theory of Ellipsis*. Oxford: Oxford University Press.

Mukai, Emi. 2002. On Verbless Conjunction in Japanese. In *NELS 33*.

Munn, Alan. 1992. A Null Operator Analysis of ATB Gaps. *The Linguistic Review* 9:1–26.

Munn, Alan. 1993. Topics in the Syntax and Semantics of Coordinate Structures. Doctoral Dissertation, The University of Maryland.

Munn, Alan. 1999. On the identity requirement of ATB extraction. *Natural Language Semantics* 7:421–425.

Oirsouw, Robert R. van. 1983. Coordinate Deletion and "n-Ary" Branching Nodes. *Journal of Linguistics* 305–319.

Oirsouw, Robert R. van. 1987. *The Syntax of Coordination*. New York, New York: Croom Helm.

Partee, Barbara Hall, Alice G. B. ter Meulen, and Robert Eugene Wall. 1990. *Mathematical Methods in Linguistics*. Dordrecht ; Boston: Kluwer Academic.

Postal, Paul. 1974. *On Raising*. Cambridge, Massachusetts: MIT Press.

Postal, Paul M. 1998. *Three Investigations of Extraction*. Cambridge: The MIT Press.

Progovac, Ljilyana. 1998. Structure for Coordination. *GLOT* 3.

Richards, Norvin. 1997. Competition and Disjoint Reference. *Linguistic Inquiry* 28:178–187.

Richards, Norvin. 2001. *Movement in Language: Interactions and Architectures*. Oxford: Oxford University Press.

Ross, John. 1967. Constraints on Variables in Syntax. Doctoral Dissertation, Massachusetts Institute of Technology.

Ryant, Neville, and Tatjana Scheffler. 2006. Binding of Anaphors in LTAG. In *Proceedings of the 8th International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+ 8)*, 65–72. COLING-ACL workshop, Sydney, Australia.

Sabbagh, Joseph. 2007. Ordering and Linearizing Rightward Movement. *Natural Language and Linguistic Theory* 25:349–401.

Sarkar, Anoop, and Aravind Joshi. 1996. Coordination in Tree Adjoining Grammars: formalization and implementation. In *Proceedings of COLING'96*, 610–615. Copenhagen.

Starke, Michal. 2001. Move Dissolves into Merge: A Theory of Locality. Doctoral Dissertation, University of Geneva.

Steedman, Mark. 1985. Dependency and Coordination in the Grammar of Dutch and English. *Language* 61:523–568.

Steedman, Mark. 1996. *Surface Structure and Interpretation*. Cambridge, Massachusetts: MIT Press.

Storoshenko, D.R., C. Han, and D. Potter. 2008. Reflexivity in English: An STAG Analysis. In *The Ninth International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+ 9)*, 149–156.

Vries, Mark de. 2009. On Multidominance and Linearization. *BIOLINGUISTICS* 3:344.

Wagner, Michael. 2008. And, Or and Ø. *Snippets* 17:11–12.

Weir, D.J. 1988. Characterizing Mildly Context-Sensitive Grammar Formalisms. *Univ. of Pennsylvania-Electronic Dissertations* .

Wexler, Kenneth, and Peter Culicover. 1981. *Formal Principles of Language Acquisition*. Cambridge, Massachusetts: MIT Press.

Wilder, Chris. 1994. Coordination, ATB, and Ellipsis. *Groninger Arbeiten zur Germanistischen Linguistik* 37:291–331.

Wilder, Chris. 1997. Some Properties of Ellipsis in Coordination. In *Studies on universal grammar and typological variation*, ed. Artemis Alexiadou and T. Hall, 59–107. Amsterdam: Benjamins.

Wilder, Chris. 1999. Right Node Raising and the LCA. In *WCCFL 18: Proceedings of the 18th West Coast Conference on Formal Linguistics*, ed. Sonya Bird, Andrew Carnie, Jason D. Haugen, and Peter Norquest, 586–598.

Williams, Edwin. 1978. Across-the-Board Rule Application. *Linguistic Inquiry* 9:31–43.

Zoerner, Cyril Edward, III. 1995. Coordination: The Syntax of &P. Doctoral Dissertation, University of California at Irvine.