

ECONOMY OF COMMAND

by

David Peter Medeiros

Copyright © David Peter Medeiros 2012

A Dissertation Submitted to the Faculty of the

DEPARTMENT OF LINGUISTICS

In Partial Fulfillment of the Requirements
For the Degree of

DOCTOR OF PHILOSOPHY

In the Graduate College

THE UNIVERSITY OF ARIZONA

2012

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Dissertation Committee, we certify that we have read the dissertation
prepared by David Peter Medeiros

entitled Economy of Command

and recommend that it be accepted as fulfilling the dissertation requirement for the
Degree of Doctor of Philosophy

Andrew Carnie

Date: 11/17/2011

Andrew Barss

Date: 11/17/2011

Thomas Bever

Date: 11/17/2011

Heidi Harley

Date: 11/17/2011

Massimo Piatelli-Palmarini

Date: 11/17/2011

Final approval and acceptance of this dissertation is contingent upon the candidate's
submission of the final copies of the dissertation to the Graduate College.

I hereby certify that I have read this dissertation prepared under my direction and
recommend that it be accepted as fulfilling the dissertation requirement.

Dissertation Director: Andrew Carnie

Date: 11/17/2011

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at the University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the copyright holder.

SIGNED: David Peter Medeiros

Acknowledgements

This work has benefited from the comments and criticisms of a number of scholars, whose contributions cannot adequately be recorded here. But I would like to single out the following for their help, among many others.

This project would not have been possible without the loving support of my family, who have shown remarkable patience and faith in me (especially my wife, who gave up some of her dreams so that I could pursue this one).

Thanks are due to Cedric Boeckx, for encouraging me to pursue this line of inquiry and supporting it at every turn; Noam Chomsky, for generously taking the time to comment on these ideas; Kleanthes Grohmann, for guidance and (extraordinary) patience; Simin Karimi, for her unflagging support and insightful questions; Terry Langendoen, who showed me the utility and joy of applying mathematics to understanding language; Terje Lohndal, for close reading of drafts of this material and very helpful comments; Jaime Parchment, for early help in computational exploration of phrase structure (and relentless skepticism about the project); Yosuke Sato, for warm friendship and a demonstration of what it means to be a serious syntactician; and Juan Uriagereka, whose ideas directly inspired everything here. I would also like to thank Jennifer Culbertson, Sandiway Fong, Elly van Gelderen, Norbert Hornstein, Scott Jackson, Antxon Olarrea, and the participants in the Arizona Syntax Salon.

A special thanks is due to my committee: Andy Barss, for his penetrating brilliance; Tom Bever, for his awesome knowledge and (to me) surprising enthusiasm for my work; Heidi Harley, for understanding this work better than I do and suggesting many applications; Massimo Piatelli-Palmarini, for believing in me and sharing his far more well-informed vision of biolinguistics; and above all to Andrew Carnie, who inspired me to pursue linguistics in the first place, and has been the finest mentor and friend one could find. To you all, it has been an honor to stand among such giants; I am humbled by your support.

TABLE OF CONTENTS

LIST OF FIGURES.....	9
LIST OF TABLES.....	10
ABSTRACT.....	11
CHAPTER 1: INTRODUCTION.....	12
1.0 Bilingualism and the Minimalist Program.....	12
1.1 An old idea	14
1.1.1 Transformations as structural simplification.....	15
1.1.2 Minimal depth as a desideratum.....	16
1.2 Tree-balancing as a desideratum for language design.....	20
1.3 Predictions for the design of natural language.....	20
1.4 Brief note on perspective and goals.....	22
1.5 Structure of this dissertation.....	23
CHAPTER 2: BALANCED TREES.....	28
2.0 Introduction.....	28
2.1 C-command and minimal search.....	29
2.1.1 Minimality and minimizing links.....	31
2.1.2 An alternative: c-command as a grammatical primitive.....	32
2.2 Divergence of extremes.....	33
2.3 On counting c-command relations.....	39
2.4 Agreement asymmetries and ‘minimizing links’.....	41
2.5 The bushiness of natural language expressions.....	45
2.6 On the spinality of the base.....	48
2.7 Summary.....	53
CHAPTER 3: UNIFORM STRUCTURE AND CYCLIC INTERPRETATION.....	55
3.0 Syntactic assumptions.....	55
3.1 Articulated syntax.....	56
3.2 Uniformity of structure.....	58
3.2.1 Variation I: Extent of projection.....	59
3.2.2 Variation II: Agreement and negation.....	60
3.2.3 Considerations of variation in this work.....	62
3.2.4 Interim summary: Assumptions about structure adopted in this work.....	63
3.3 Cyclic Interpretation.....	63
3.3.1 On copies.....	65
3.3.2 Linearization.....	67
3.3.3 Agreement.....	69
3.3.4 Binding.....	70
3.3.5 Nuclear Stress Rule.....	73
3.4 Conclusions.....	76

TABLE OF CONTENTS – *Continued*

CHAPTER 4: MOVEMENT AS TREE-BALANCING.....	78
4.0 Introduction.....	78
4.1 Previous treatment of syntactic movement.....	82
4.2 What does movement do?.....	86
4.3 The Fundamental Movement Condition.....	88
4.4 Antilocality.....	93
4.5 Size threshold.....	96
4.6 Islands of symmetry.....	104
4.7 Roll-up movement and Malagasy.....	109
4.7.1 The basic pattern.....	109
4.7.2 Malagasy facts.....	111
4.7.3 Dynamics of roll-up movement.....	113
4.7.4 Rightward Object Shift in Malagasy: Against head directionality.....	115
4.7.5 Reconciling Antilocality and strict reversal.....	116
4.8 Iterating patterns of movement.....	118
4.9 Conclusions.....	123
 CHAPTER 5: TREE-BALANCING IN THE DP AND UNIVERSAL 20.....	 126
5.0 Introduction.....	126
5.0.1 DP orders: A brief sketch.....	127
5.0.2 Tree-balancing is a sufficient explanation.....	128
5.0.3 This is a surprising result.....	130
5.0.4 Structure of this chapter.....	132
5.1 DP orders: Facts and analysis.....	133
5.1.1 Cinque's Generalization as Harmonic Bounding of left alignment.....	134
5.1.2 Cinque's Generalization in Artificial Language Learning.....	135
5.1.3 Cinque's Generalization beyond the DP.....	136
5.1.4 On head-complement order and restrictiveness of analysis.....	138
5.2 Assumptions and methodology.....	143
5.2.1 Analytical assumptions.....	144
5.2.2 Philosophical background.....	147
5.2.3 Why look at DP orders?.....	147
5.2.4 Inferring movements from surface orders.....	148
5.2.5 The DP Condition.....	151
5.2.6 Direct demonstration for smallest solution tree.....	153
5.3 Overview of numerical results.....	155
5.3.1 Antilocality.....	155
5.3.2 <i>N</i> is big.....	156
5.3.3 <i>D</i> can be big too, and bushy.....	156
5.3.4 Spinal cartography.....	157

TABLE OF CONTENTS – *Continued*

5.4 Discussion	160
5.4.1 <i>On order (p) NDAM</i>	162
5.4.2 <i>On remnant movement</i>	163
5.4.3 <i>A simpler account, with messier predictions</i>	164
5.5 Conclusions and direction for future research	166
 CHAPTER 6: ECONOMY OF COMMAND IN PHRASE STRUCTURE	167
6.0 Optimal phrasal shape	167
6.0.1 <i>Generalized X-bar phrases</i>	168
6.1 Local comparison: A first pass	172
6.2 Generalized phrases	176
6.2.1 <i>A domain for terminals</i>	177
6.2.2 <i>Possible growth modes</i>	179
6.2.3 <i>Notational conventions</i>	180
6.2.4 <i>Generating all possibilities</i>	181
6.3 Comparing growth modes	186
6.3.1 <i>Comparison sets based on local complexity</i>	186
6.3.2 <i>Direct comparison redux</i>	187
6.3.3 <i>Indirect comparison</i>	188
6.3.4 <i>The ‘bottom of the tree’ problem</i>	189
6.3.5 <i>Top-down generation</i>	191
6.3.6 <i>Some results from indirect comparison</i>	193
6.4 Deriving projection	195
6.5 On projection	197
6.5.1 <i>What is projection, and what does it do?</i>	199
6.5.2 <i>Is projection Minimalist?</i>	200
6.6 Proof of the optimality of generalized X-bar forms	201
 CHAPTER 7: THE GOLDEN PHRASE	206
7.0 Golden syntax	206
7.0.1 <i>The X-bar schema as recursive template</i>	208
7.0.2 <i>Structure of this chapter</i>	210
7.1 A brief introduction to “golden” mathematics	212
7.1.1 <i>The golden mean</i>	212
7.1.2 <i>The Fibonacci numbers</i>	213
7.1.3 <i>The Golden String</i>	216
7.1.4 <i>Golden recurrence</i>	219
7.2 Golden recurrence in the X-bar schema	219
7.2.1 <i>Fibonacci numbers of syntactic categories</i>	220
7.3 The X-bar schema as minimax solution	221

TABLE OF CONTENTS – *Continued*

7.3.1 <i>Local computation, but not too local</i>	223
7.3.2 <i>Optimal trees, but not too optimal</i>	224
7.3.3 <i>The last bushy spine, the last spiny bush</i>	226
7.3.4 <i>The largest endocentric form which is still locally best</i>	227
7.4 <i>The X-bar schema is a golden fractal</i>	232
7.4.1 <i>Phrasal patterns as line division algorithms</i>	233
7.4.2 <i>The Cantor Set</i>	233
7.4.3 <i>The image of X-bar structure is an Asymmetric Cantor Set</i>	235
7.4.4 <i>The golden dimension of the X-bar form</i>	238
7.5 <i>Golden growth in X-bar phrase structure</i>	239
7.5.1 <i>Defining a notion of ‘growth factor’</i>	239
7.5.2 <i>Expressing phrase structure patterns as matrices</i>	241
7.5.3 <i>The growth factor is the characteristic root</i>	242
7.5.4 <i>Growth factors by complexity class</i>	243
7.5.4.1 <i>One non-terminal: pair, spine, or bush</i>	244
7.5.4.2 <i>Two non-terminals: The X-bar class</i>	245
7.5.4.3 <i>Orientation families</i>	247
7.5.4.4 <i>Three non-terminals</i>	250
7.5.4.5 <i>Factorization and composition in degenerate systems</i>	254
7.6 <i>Conclusions</i>	255
CHAPTER 8: CONCLUSION	259
8.0 <i>Overview</i>	259
8.1 <i>Minimizing c-command relations</i>	259
8.2 <i>Movement as tree-balancing</i>	261
8.3 <i>On Cinque’s Generalization</i>	263
8.4 <i>The nature of cross-linguistic syntactic variation</i>	265
8.5 <i>Phrase Structure</i>	267
8.6 <i>Optimizing movement by phase: A-movement and A-bar movement</i>	268
8.7 <i>Final Remarks</i>	271
APPENDIX A: DERIVATION OF THE DP CONDITION	272
A.0 <i>Overview</i>	272
A.1 <i>Leapfrogging</i>	273
A.2 <i>Excluding successive-cyclic movement</i>	277
A.3 <i>Attested Orders</i>	280
A.4 <i>Unattested Orders</i>	284
A.5 <i>Putting it all together</i>	290
A.6 <i>Table of results and selected solutions</i>	292
A.7 <i>Java program</i>	295
REFERENCES	299

LIST OF FIGURES

Figure 1: Derivations for minimal tree with free head-complement order.....	142
Figure 2: Derivations of attested orders considered in this work.....	150
Figure 3: Derivations for minimal tree.....	154
Figure 4: Iterative construction of the Cantor Set.....	234
Figure 5: First stage of mapping X-bar form to line segment.....	236
Figure 6: Second stage of mapping X-bar form to line segment.....	236
Figure 7: Iterative construction of two-scale Cantor Set.....	237
Figure 8: The squeezing lemma illustrated.....	275
Figure 9: Derivations of attested orders considered in this work.....	281
Figure 10: Derivational possibilities with N, A, M.....	286
Figure 11: Excluded routes to unattested orders 1.....	289
Figure 12: Excluded routes to unattested orders 2.....	290

LIST OF TABLES

Table 1: C-command relations by depth.....	37
Table 2: C-command relations in Spine and Bush.....	37
Table 3: Attested and unattested DP orders.....	128
Table 4: Options for comparison set with two non-terminals.....	183
Table 5: C-command relations as a function of terminals in best trees.....	194
Table 6: Characteristic polynomials and growth factors for three non-terminals.....	254
Table 7: Movements involved in deriving each attested order.....	282
Table 8: Results of Program.....	293

ABSTRACT

This dissertation proposes a principle of “economy of command”, arguing that it provides a simple and natural explanation for some well-known properties of human language syntax. The focus is on the abstract combinatorial system that constructs the hierarchical structure of linguistic expressions, with long-distance dependencies determined by the structural relation of c-command. Adopting the assumption of much recent work that properties of syntax reflect very general organizational principles, I propose that syntactic forms with fewer and shorter c-command relations are preferred. Within the boundaries of strict binary branching assumed here, this results in a preference for hierarchical tree structures to be shallow and bushy, rather than deep and narrow. I pursue two broad applications of this principle, to syntactic movement and phrase structure.

I argue that movement, the displacement of material to thematically unrelated positions, is a mechanism to reduce the number and length of c-command relations in the affected structures. I detail the properties we expect if movement is driven by this principle, including antilocality, a size threshold effect, a class of island effects, and feedback effects on iterated patterns of movement. I argue that these predictions align well with recent empirical descriptions of syntactic movement. I develop an account in these terms of the cross-linguistic ordering of elements within nominal phrases. Utilizing a computer program, I show that a single underlying structure common to all languages can give rise to all and only the attested word order possibilities via c-command-reducing movements, and describe the required shape of this underlying structure.

The principle of economy of command also makes predictions about the format of phrase structure. Among the possible ways to build self-similar syntactic structure, the phrasal forms that build trees with the fewest c-command relations are “endocentric”, in the geometric sense that each phrase contains a unique local terminal, and every daughter of the phrase that does not contain its associated terminal is another phrase. This provides a structural basis for the mysterious headedness of phrases. These successes support the validity of the principle, and reinforce the broader project of seeking naturalistic explanation of linguistic properties.

CHAPTER 1: INTRODUCTION

1.0 Biolinguistics and the Minimalist program

This dissertation aims to contribute to the emerging field of biolinguistics, approaching linguistics as a natural science. I argue that some fundamental properties of syntax (the rules governing how words are assembled into phrases and sentences) can be explained by natural laws of form, principles of efficient computation, and the like, matters of ‘Galilean perfection’ that are most obvious in the inorganic world (e.g., in the six-pointed stars of snowflakes). The concerns pursued below can be viewed as a species of ‘Save Wire’ constraint¹, applied to abstract long-distance (c-command and dominance) relations in syntactic structures. The results calls to mind the stunning findings of Cherniak *et al* (2004), who reports human cerebral cortex to be optimized for a far more literal version of ‘Save Wire’ (a one in many billions best solution). It is worth asking whether similar, very general, principles of network optimization are at work both in the neural architecture, and in the mathematical laws of linguistic cognition.

It is increasingly clear that natural law in this sense plays a significant role in explaining organic forms, as argued by D’Arcy Wentworth Thompson (1917) and Turing (1952), an idea lately revived in the ‘evo-devo’ revolution in biology (see e.g. Carroll 2005). This work goes one step further, and argues that natural law is evident in cognition as well (as manifested in human language). The picture that emerges casts the syntactic system (part of the mind) as something like a crystal (Chomsky 2005), robustly

¹ That is, an optimization constraint that minimizes the number and cost of connections in a complex system.

and spontaneously patterned by neither elaborate genetic specification nor environmental shaping, but rather by organizational principles of nature itself (Chomsky's "third factor"). See Boeckx and Piatelli-Palmarini (2005), Freidin & Vergnaud (2001), Uriagereka (1998), among many others.

The third factor consists of "principles not specific to the faculty of language" (Chomsky 2005: 6); these are contrasted with the first factor (genetic) and the second (environment and learning). The third factor instead comprises "principles of structural architecture and developmental constraints that enter into canalization, organic form, and action over a wide range, including principles of efficient computation, which would be expected to be of particular significance for computational systems such as language" (Chomsky 2005:6).

The more we are able to show that deep and very general principles determine the details of the phenomenon, the more we have explained in a principled way. In the case of language, there are particularly strong reasons to suspect that laws of form in some sense (Cherniak's "non-genomic nativism") play a large role in comparison to other factors, especially the first factor (genetic endowment). This suspicion is due to the apparent recent emergence and near-uniformity of the faculty of language across the species, even between groups who diverged tens of thousands of years ago.² Moreover, if we prematurely retreat to explanations in terms of the other two factors, we may end up dismissing as noise or brute accident what may in fact be signs of deep order.

² We must, it seems, now admit some genuine variation, though of a quite peripheral sort, in view of the recent discovery of a genetic component to the distribution of tone languages (Dediu & Ladd 2007). Note that, of the dozens of linguistic features the authors examined, only tone had a significant correlation with the distribution of the (brain growth and development related) alleles they implicated (ASPM-D and MCPH-D).

The present work can be seen as part of the project that Chomsky (2007) describes as “approaching UG [universal grammar] from below,” the idea being to see how close we can get to correctly describing the faculty of language while attributing as spare, and as conceptually natural, a structure to that faculty as conceivable. The effort may succeed or fail, but surely it is worth attempting. This thesis argues that an understanding of how long-distance dependencies accumulate in syntactic representations, and the natural supposition that computation should seek to minimize the burden of long-distance computations, leads to rich and surprisingly correct predictions about the facts of natural language. In this case, I argue that X-bar-like phrase-structural patterns provide the optimal “growth mode” for minimizing the overall number of long-distance relations, while the movements found in natural language uniformly “improve” (reduce the number of long-distance relations in) the structures they transform.

1.1 An old idea

In many ways, the ideas of this thesis represent a return to some older ideas, explored in the earliest days of Generative Grammar. While much current research focuses on the attempt to understand syntactic phenomena in terms of the specific properties (features) of the items involved, the present work follows an old tradition of looking to the structural forms themselves, independent of their contents. The suspicion is that recent work may be missing the trees for the leaves, so to speak: by pursuing the idea that the features of individual syntactic objects determine their syntactic behavior, one misses the possibility that what is driving syntactic phenomena are properties of the trees

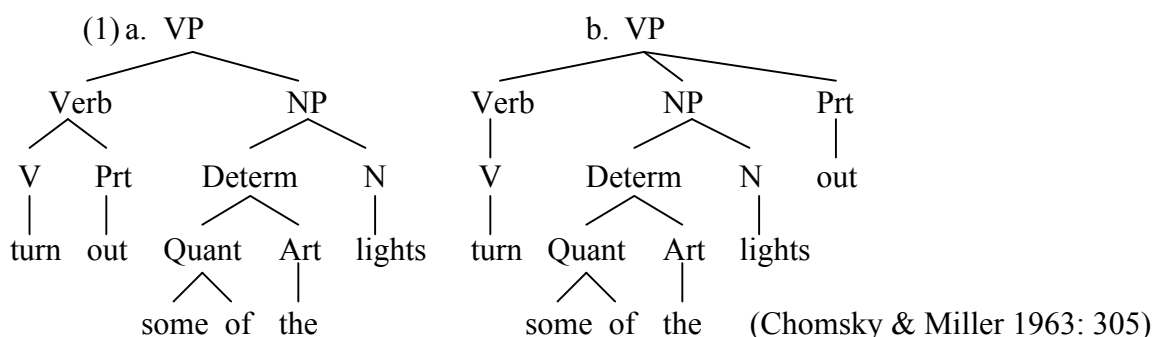
themselves. In this work, I argue that concerns of bare tree form (namely, minimization of long-distance dependencies) suffice to explain many of the core properties of human syntax. Below, I discuss two previous works that have pursued similar ideas.³

1.1.1 Transformations as structural simplification

“A major concern of the minimalist program is the reduction of the computational load in carrying out a derivation. A natural extension of that concern is the reduction of the complexity of the generated objects themselves, such as their degree of embedding, without sacrificing expressive power. Syntactic transformations, as they were first formulated in generative grammar, to some extent had the property of reducing the structural complexity of the generated objects.” (Langendoen 2003: 307).

As Chomsky & Miller (1963: 304) observe, “Singular transformations are often just permutations of terms in the proper analysis. [...] Figure 6 illustrates a characteristic effect of permutations, namely, that they tend to reduce the amount of structure to which they apply.” Their example is *turn out some of the lights* / *turn some of the lights out*, where the transformation is taken to exchange the second and third terms of (V, Prt, NP).

I reproduce their structures below:



³ Certainly Kayne (1994) deserves mention in this regard, as a modern work which points to explanations for syntactic facts in terms of properties of tree structure (and, moreover, in terms of the c-command relations defined over those trees). In this section, I keep to discussion of older work; Kayne’s proposals will be touched on throughout this thesis, in various ways.

This “permutation” transforms a strictly binary-branching structure with the *Prt* and *V* grouped together, into one with a unary-branching constituent dominating the verb and a ternary-branching *VP* at the highest level – this difference constituting the reduction in structure they have in mind.⁴

In what follows, I pursue somewhat similar ideas. The idea that transformations reduce the degree of embedding, in particular (see the quote from Langendoen above), is at the heart of Chapters 4 and 5 below. Nevertheless, the implementation of that intuition here is rather different; for instance, I keep to strict binary-branching throughout (see Kayne 1984). The relevant notion of structural minimization is constructed in terms of long-distance dependencies; put another way, it is about minimizing the depth of syntactic structures. This notion, too, finds a precedent in older proposals.

1.1.2 *Minimal depth as a desideratum*

The present work finds an important predecessor in Yngve (1960).⁵ To a certain extent, the concerns to be pursued in this work echo his, in particular the following points:

- (i) An attempt to explain “non-functional complexity” (ibid., 452) in terms of easing a computational burden associated with language.
- (ii) A preference for one extreme of branching structure over the opposite extreme.

⁴ It is not clear what this is taken to be a reduction of: perhaps branching structure in some sense, but note the total number of branches and nodes is unchanged. What is in fact reduced is the total number of certain long-distance dependencies.

⁵ Thanks to Norbert Hornstein for discussion.

- (iii) A concern to minimize depth (for Yngve, depth is the amount of temporary memory storage needed by his language production device): “depth considerations are among the most important factors in the grammar of any language.” (ibid., 452)
- (iv) The expectation that a limit on depth will be reflected in grammatical conditions, (“We would expect that constructions of less depth would be preferred over equivalent constructions of greater depth.” (ibid., 453)), with readily observable results; “The grammars of all languages will include methods for restricting regressive⁶ structures so that most sentences will not exceed this depth,”(ibid., 452)

Yngve’s work, while solidly grounded in the concerns of computationally implementing a grammar, has not had much impact on current syntactic theory. This relates to the mechanism he proposes to limit the penalized (regressive, or left-branching) structures involves counting in a ‘restricted relabeling’ system:

“A regressive branch can be allowed to grow for a certain length and then stopped if some method is used for automatically counting or keeping track of the number of regressive steps so that the *n*th step can be prevented.” (Yngve 1960: 453)

It is this move that Miller and Chomsky (1963) famously object to, concluding that, to put it as simply as possible, grammars can’t count.⁷ Rather, Miller and Chomsky argue

⁶ Yngve distinguishes regressive from progressive branches: the latter are the rightmost branch of any phrasal expansion, while the former are non-rightmost branches. In his system, regressive structures require increasing memory resources, while progressive structures do not. The production system he details can “forget” information associated with the embedding context (only) when expanding a right branch.

⁷ This conclusion might be rethought in light of the unification of the arithmetic ability and the faculty of language suggested by Hauser, Chomsky, and Fitch (2002). They suggest that a single fundamental

that memory resources constrain only performance, and are not directly reflected in grammatical conditions, as Yngve had envisioned. On their view, there are fully grammatical expressions that happen to be unusable for non-grammatical reasons (for example, multiply center-embedded sentences).

While Yngve examines a range of interesting data that supports his basic “save the hardest for last” prediction (a preference for branching complexity to be limited, as much as possible, to the rightmost branch of a growing production tree), the kinds of “discontinuous constituents” he motivates do not form a grammatically central body of phenomena. That is, the relevant movements are, we would now say, rightward movements. If the phenomena Yngve investigates are indeed movement patterns, they have a ‘peripheral’ feel, quite opposite to the large core of leftward syntactic movement phenomena (see Kayne 1994). As a particularly striking, and I think significant, example of the different predictions of Yngve (1960) and the present account, consider what has become known as the “copy theory” of movement (see Chomsky 1995b, Nunes 1995, Bošković 2001, among others). Leaving details for later, the idea is that syntactic movement creates a chain of identical objects, only one of which, typically the highest, furthest left copy in the chain, is pronounced. Insofar as overt copies are more costly than covert copies (for pronunciation and linearization, almost trivially so), then Yngve’s proposal would seem to predict the opposite preference, as part of the general theme of “save the hardest for last”. Instead, in natural language, the hardest (the pronounced copy in the chain) comes first, not last.

innovation in combinatoric cognition, called Merge, was the evolutionary event that gave our species both abilities. If so, there is a sense in which grammar and counting are, in effect, the same thing.

The central thesis of the present work, that language design should reflect what I call “economy of command” (in effect, tree-balancing/sum-over-depths minimization) is tantalizingly close in spirit to Yngve’s concern for minimizing depth of temporary memory in a sentence-production device. Yet the detailed predictions work out quite differently. In the present work, the central concern is also, as for Yngve (1960), minimization of depth, though measured in a different sense. I will also claim that a number of aspects of the structure of natural language expressions can be fruitfully analyzed as methods to reduce or minimize depth (in my sense). However, rather than temporary memory, the relevant notion of depth is constructed in terms of the relation of c-command.

(2) C-Command: Node A c-commands node B if neither A nor B dominates the other and the first branching node which dominates A dominates B. (Reinhart 1976: 32)

C-command (Reinhart 1976), the syntactic relation taken to play a role in many syntactic phenomena (including binding, agreement, movement, scope, linearization, probe-goal, etc.) has survived a number of radical overhauls of syntactic theory. As Epstein et al (1998: 24) point out, “[C-command] is persistent: despite substantive changes in the theory of syntax, Reinhart’s definition, proposed almost two decades ago, remains linguistically significant.”⁸

⁸ While ‘eliminating’ c-command is a much-practiced sport of late, I think such attempts fare no better than attempts to ‘eliminate’ specifiers (e.g., Starke 2004). While such attempts may achieve a recasting or reanalysis of the relevant relation or structure, there is a core of truth about the traditional formulation that survives intact. Consider Epstein et al (1998)’s derivational account of c-command. Rather than being an arbitrarily defined representational constraint on the domain of syntactic relations, as in Government & Binding Theory (Chomsky 1981), c-command is seen to be a natural “dynamic horizon”; relations can only be established between subtrees linked by Merge, the structure-building operation. That gives us a good answer to *why* the particular structural relation described by c-command should be the correct description.

1.2 *Tree-balancing as a desideratum for language design*

This work claims that certain facts of natural language can be explained naturalistically in terms of “economy of command”, in effect an extension of basic minimal search concerns to minimizing the aggregate load of iterated c-command-based computations. It amounts to a preference for balanced trees, because shallow, bushy structures have fewer total c-command relations than deep, spindly ones. I take this difference to index a lesser load for the computation of long-distance dependencies (empirically irreducible and numerous). There are clear reasons to expect those shapes with fewer vertical relations to be favored: they put a tighter cap on the ‘explosion’ of long-distance relations.

1.3 *Predictions for the structure of natural language*

I argue that *tree-balancing* explains a number of fundamental—and otherwise deeply mysterious—properties of human language. Chief among these are (i) projection/labeling and (ii) syntactic movement. I argue that (i) projection is an epiphenomenon of optimal branching in (the dynamic growth of) truly bare phrase structures, and (ii) syntactic movement can be explained, in a deep sense, in the same terms.

In a sense, it eliminates c-command as a primitive syntactic notion; instead, the relevant notion is derived from a derivational understanding of syntax. But what has not “dissolved” is the concept of a search process accompanying Merge, establishing linguistically significant intra-arboreal relations. On this view, minimization of c-command should matter for computational reasons, as it reduces the length of the search implicit in the derivational definition. That is, relations are established not just between two objects linked by Merge, but between one object and the contents of the other object. Seen this way, the concern of this thesis, economy of command, comes into focus as a condition of minimal search. In structures with fewer c-command relations, the relevant intra-arboreal searches are fewer and shorter.

As for projection (i), I prove that unlabeled geometric forms isomorphic to generalized X-bar phrases (one terminal “head”, one “complement” phrase, some fixed number of “specifier” phrases) are optimal, building more balanced trees than any alternative phrasal shapes of matched complexity. In effect, endocentricity ‘emerges’ as an accident of optimal tree growth; then projection, a stipulation in most current theories, dissolves into more basic machinery, a welcome result for the Minimalist Program (Chomsky 1995b).

Not only does tree-balancing provide a compelling motivation for movement (ii), to balance unbalanced trees, but indeed the detailed phenomenology of movement might fall out from these concerns as well. Supposing that movement is ‘for’ tree-balancing, we predict the following: Anti-locality; phrasal but not head movement; the possibility of roll-up (runaway minimally-antilocal) movement; and, incorporating a notion of syntactic cycle, we predict movement of large XPs to the edge, or just below the edge, of the cyclic domain—as I argue, this provides a structural basis for A-bar and A-movement, respectively.

Moving beyond broad-stroke predictions, I demonstrate at length that the cross-linguistic array of attested and unattested relative orders of demonstrative, numeral, adjective, and noun (cf. Greenberg’s 1963 Universal 20) can be made to follow from constraining movement to improve tree-balance, if the large cartographic hierarchy underlying the coarse four-element hierarchy meets geometric constraints I derive. The predicted tree-shape actually aligns closely with current cartographic proposals for the

nominal domain (Cinque 2005, Svenonius 2008), with further strong predictions, e.g. a large size for the subtree underlying the noun.

1.4 Brief note on perspective and goals

The approach taken throughout departs from the large empirically-motivated body of work that seeks to derive conditions on syntactic form from semantic or lexical characteristics of the expression. Instead, I limit my focus here to considerations of bare branching form, as a matter of idealization in the Minimalist mode. Simply put, my goal is to try to figure out how a minimal language device designed for economy of command would be expected to behave. This should not be read as a serious proposal that this is the *only* factor at work in determining the properties of natural language expressions. But the only way to find out how much can be explained in this way, is to start by pretending as if it were the only factor, to understand what design features would affirm the presence of the hypothesized bias.

Can we ‘grow’ the essential outlines of the properties of human syntax from some austere (computational) ‘physics’ of language? How much can we explain about the syntax of particular expressions, in particular languages, in terms that make no reference whatsoever to the lexical contents thereof, but rather is cast wholly in terms of laws of (tree) form? Surely not everything; of course the relevant phenomenon is nuanced, influenced by many ‘messy’ factors, and enormously complicated. But I submit that the answer is not ‘nothing’, either; the notion of economy of command, applied as I sketch in the rest of the dissertation, seems to buy us something like what we find in natural

language. This dissertation is an attempt to take this perspective seriously, and see how far it might go.

To take a concrete example, in what follows I ignore completely the rich and quite productive body of work exploring the role of syntactic features in determining the details of syntactic movement. Instead, I pursue explanations for movement in a blindly structural sense, where the individual nodes of the tree are taken to be featureless and indistinguishable. This should not be taken to indicate that I think we can ultimately discard features from syntactic explanations.

Nevertheless, there is something unsatisfying about ascribing each movement to an appropriately placed feature, especially where independent semantic or morphological evidence for that feature is thin on the ground (a familiar complaint). The goal of the present work is to go beyond taxonomy, to a place where, even if we must admit features as part of the mechanism of movement (the proximate cause), we can say something insightful about why features are such as to drive the movements they do. This echoes the conclusions of Boeckx:

“[...] the neo-constructionist, realizational, post-syntactic PF-models that are becoming more and more influential should be used to view lexical features not as leaders, but as followers, as stabilizing, rather than dictating the construction of structural options.” (Boeckx 2010: 22)

1.5 Structure of this dissertation.

The remaining chapters are devoted to developing the notion of “economy of command”, working out the predictions that follow it, and comparing them against what is known about human language. I take the results to be broadly encouraging; it appears as though

a great deal of the basic machinery of the language faculty can be rationalized in the present terms, perhaps (eventually) explained in a deep sense. Nevertheless, this is nothing more than an initial exploration of the terrain, and a great deal more work would be required to see how far this sort of thing can really go.

Chapter 2 lays out the basics of the idea. I enumerate some reasons for thinking that the number of c-command (equivalently, dominance) relations in a tree represents a kind of computational cost. The idea is that a large number of linguistic phenomena track along long-distance dependency paths described by these relations, and that the ‘search’ process involved in computing long-distance dependencies is costly, such that minimal search is preferred. I provide a simple exploration of some of the relevant mathematics, examining how the number of c-command (or, if we like, dominance) relations scales with tree size and changes with tree shape, whether it is reasonable to count all c-command relations equally (and some gesturing at what predictions we would get if we pursued some other reasonable alternatives), and further details.

Chapter 3 spells out the theoretical assumptions that underpin this work. There are two broad topics in this chapter. The first concerns the approach to comparative syntax lately called “cartography,” which supposes that syntactic structures are much more finely articulated than traditionally assumed, and that the articulated structure is largely or completely identical for all expressions, and all languages. A particularly important issue here is whether there is any degree of variation at all, and if so how constrained it might be; these concerns return to the forefront in the following chapter. Also with an eye to setting the scene for the rest of this investigation, this chapter takes up the notion

of what I call cyclic interpretation. The basic thrust is that much of the “action” with respect to the c-command and dominance relations happens in the partial representations presented by the syntactic engine to the interfaces, phase by phase. To put it simply, this means that what matters is economy of command as it exists in interface forms, which reflect the results of movement, and where the chain of copies produced by movement collapses onto a single position (for most, but not all processes that rely on hierarchical relations, only the highest copy in the chain “counts”). This perspective combines two core ideas of Chomsky’s 1995b Minimalist Program: a concern for economy conditions, and attention to requirements imposed on syntax by the interfaces (so-called bare output conditions).

Chapter 4 takes up the task of treating syntactic movement as a mechanism for tree-balancing. I discuss the motivations for this view of things, and propose a precise condition governing movement, the Fundamental Movement Condition (FMC). I then move to exploring further aspects of my predictions, with an eye to well-known, often mysterious properties of movement phenomena that bear on these predictions. I motivate a notion of Antilocality (a hard lower limit on how short movement can be), and what I call a size threshold effect, which I relate to several empirical phenomena. I point out that this view of movement also gives us a natural kind of island condition, from which we may derive a form of Ross’s (1967) Coordinate Structure Constraint. I consider the patterns of movement expected under this account, including an analogue of extremely-local roll-up movement.

Chapter 5 provides an existence proof that this account of movement can get us somewhere interesting. In particular, the goal we reach is a sufficient account of the typology of possible and impossible DP orders of the four core elements (demonstrative, numeral, adjective, and noun). As I show, there are shapes of the base tree such that all and only the attested orders have a “positive” derivation, while no unattested orders do (“positive derivation”: one in which the only movements that occur each satisfy the FMC, i.e. achieve better tree-balance). Thus, if the base tree in fact matches my structural predictions, the account looks eminently plausible. I discuss some recent proposals in the cartographic literature that seem to support my predictions, which I take to be most intriguing and encouraging, not least because the predictions are quite specific and restrictive – i.e., they are unlikely to be satisfied “by accident”; we wouldn’t expect the base tree to happen to meet these incredibly detailed predictions if the present account were completely off-base.

In chapter 6, (a reworked version of portions of Medeiros 2008), I apply economy of command as a metric to select among implicitly unconstrained phrase structure patterns, supposing that those phrasal ‘recipes’ that produce the most balanced trees are favored. As I show, requiring nothing more than that phrase structure must accomplish the linkage of terminal atoms with larger, recursively defined structures, economy of command picks out as ‘best’ (relative to patterns of matched complexity) those patterns corresponding to a generalized X-bar schema. In effect, this yields endocentricity as a by-product of optimal packing, a happy result.

Chapter 7 expands considerably on the abstract conception of phrase structure developed in the previous chapter. The emphasis of this chapter is on the relationship between the one-specifier X-bar schema and the mathematics of the golden mean and Fibonacci numbers. I propose there to call the X-bar schema the “golden phrase”, as it manifests a syntactic recurrence relation exactly parallel to the recurrence relations underlying other pieces of “golden” mathematics. As part of highlighting the properties of this special structural scheme, I consider other structural possibilities in some detail. I develop mathematical techniques revolving around a matrix formulation of syntactic recurrence, and provide a number of results of general interest (including a way to define a growth factor for each kind of syntactic recurrence pattern, and a way to compute its fractal dimension when interpreted as a line division scheme in a natural way).

Chapter 8 concludes the dissertation. I review the major results derived throughout, and discuss the implications, returning to the very large-bore concerns about biological explanation already touched on above. I discuss the prospects for future research examining and testing the ideas sketched here, noting the areas that seem most promising, and those most problematic.

CHAPTER 2: BALANCED TREES

2.0 Introduction

The purpose of this chapter is to propose the basic thesis of economy of command. In brief, I will argue for a minimization constraint on the number and depth of long-distance relations in syntactic trees. In spirit at least, this is hardly novel; the basic intuition of a locality condition on long-distance relations underlies Minimality (Rizzi 1990), Shortest Move, Attract Closest, the Minimal Link Condition (the last three from Chomsky 1995b), the Minimal Distance Principle (Rosenbaum 1967), Subjacency (Chomsky 1973), and similar formulations. However, those conditions are understood as restrictions on which of several available c-command “paths” is chosen (namely, the shortest), in some structure determined by unrelated principles. In the present account, it is argued that a preference for path-shortening informs structure-building itself (in both Internal and External Merge, i.e. in movement and phrase structure).

In this chapter, I argue for a particular articulation of this intuition. Section 2.1 lays out the conceptual foundations for the claim, reviewing the notion of c-command, and pointing out that a well-known minimization condition on such relations can be reinterpreted as a constraint on structure-building. In section 2.2, I distinguish between two extremes of tree structure: One, which I call “the Spine”, expresses the maximum number of c-command/containment relations. The other, called “the Bush”, expresses the minimum. Section 2.3 attempts to justify the simple expedient of counting all c-command relations in a syntactic structure as a way to measure their economy of command, noting

reasons for caution. In section 2.4 and 2.5, I sketch how some familiar empirical effects can be understood in terms of economy of command. The first discusses agreement asymmetries, arguing that they can be understood in terms of path-shortening. The second points out a selection of cases (the clause structure of Niuean, the structure of Spanish verbs, and the geometry of phi features), where syntactic forms seem to reflect the hypothesized ideal of the Bush. In section 2.6, I address the apparent problem that the underlying syntactic hierarchy is a Spine. Section 2.7 concludes the chapter.

2.1 C-command and minimal search

In what follows, I will be comparing syntactic structures on the basis of the number of c-command (and dominance) relations they encode. I adopt the familiar definition of c-command, as follows:

1) C-Command (Reinhart 1976: 32)

Node A c-commands node B if neither A nor B dominates the other and the first branching node which dominates A dominates B.

However, in this work I assume strict binary branching, in which case any non-terminal node is necessarily a branching node. I rule out as well relativization of command relations to only some non-terminal nodes (as in the M-command of Aoun and Sportiche (1983), computed relative to maximal projections, or Lasnik's (1976)

Kommand, referring to NP and S).¹ This allows a simpler definition of c-command, which I use throughout:

- 2) C-command (Simplified): Node A c-commands node B if A and B are distinct, A does not dominate B, and all nodes that dominate A dominate B.

Here, as throughout, I am referring only to irreflexive dominance (i.e. nodes do not dominate themselves). It is for this reason that we need not stipulate that B does not dominate A. On the other hand, we must now spell out the distinctness of A and B.² See Carnie (2010) for further discussion of these issues.

The notion of c-command is central to numerous linguistic relations. Langacker (1966) first spoke of the “chain of command” among certain kinds of noun phrases, discussing what we would now call Condition C effects. Wasow (1972) and Reinhart (1976) further developed the concept of c-command in describing the distribution of anaphora. While still central to binding theory, the role of the c-command relation in linguistic theory has been extended to linearization (Kayne 1994), the determination of relative scope (May 1985), and agreement (as in the probe-goal system of Chomsky 2001), to mention some core cases.

Epstein et al. (1998) rationalize the importance of the c-command relation in terms of a derivational view of syntax. As they point out, c-command amounts to the condition that syntactic objects can enter into linguistic relations with the contents of the sub-tree

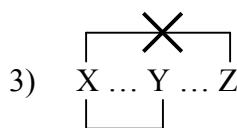
¹ See also Barker and Pullum (1990), who formally define a variety of c-command relations along these lines.

² The distinctness of A and B is handled by the stipulation that A and B do not dominate each other, if we take dominance to include the reflexive case (as in Reinhart’s original definition).

they are merged with. This suggests a view of c-command as following from a search operation accompanying Merge.³

2.1.1 Minimality and minimizing links

The property of Minimality, as encoded by principles such as the Minimal Link Condition, Shortest Move, Attract Closest, and Relativized Minimality (the relevant literature is vast; see Chomsky 1995b, Rizzi 1990, among many others), has also been interpreted in terms of a search algorithm (Chomsky 2000, 2001). It is robustly observed that in configurations like (3), where X could enter into a dependency with either Y or Z, but Y is closer to X than Z is, a dependency may hold between X and Y but not between X and Z.⁴



This closeness is measured by c-command relations: If Y asymmetrically c-commands Z, Y is closer to X than Z is (and, of course, X must c-command Y and Z to potentially enter into relations with them). To a first order of approximation, we might

³ This derivational understanding is not the only way of picking out the c-command relation as special or significant. As Richardson & Chametzky (1985) and Chametzky (1996) discuss, c-command provides a “factorization” of the phrase marker with respect to a node. See those works for details.

⁴ Note that the government relation (Chomsky 1981, 1982, and much subsequent work) was subject to a very similar locality constraint. However, the locality constraint on government was described as favoring a relation between Y and Z over one between X and Z, taking the perspective of looking up the tree, so to speak (from Z to a potential governor in either X or Y). In the derivational perspective of Chomsky (2000 et seq), the relevant locality condition is formulated looking down the tree, regulating long-distance relations between a “probe” X and two potential “goals” Y and Z deeper in the tree.

reasonably say that syntax seems to minimize links. Such a property fits well with the idea that c-command relations are subject to minimization principles, as suggested here.

The present hypothesis, that language is designed for economy of command, can be seen as one species of principle aimed to “reduce ‘search space’ for computation: ‘Shortest Movement/Attract’, successive-cyclic movement (Relativized Minimality, Subjacency), restriction of search to c-command or minimal domains, and so on.” (Chomsky 2000: 99) To make the connection explicit, on a derivational understanding the c-command relation indicates intra-arboreal search accompanying structure-building. Forms with fewer c-command and dominance relations form what we might describe as minimal search surfaces, minimizing the space searched during an extended derivation.

2.1.2 An alternative: c-command as a grammatical primitive

Frank & Kuminiak (2000) and Frank & Vijay-Shanker (2001) explore the idea that c-command is a primitive grammatical relation, rather than a derived relation based on a primitive dominance relation, as usually assumed (for an explicit formulation of the standard view of trees, see Partee, ter Meulen, and Wall 1993). This has the advantage of answering quite directly why the particular relationship of c-command, among all possible derived notions based on dominance, should figure so centrally in so many linguistic phenomena, while other relations derivable from dominance (and precedence) do not seem to play any role in linguistic computation.

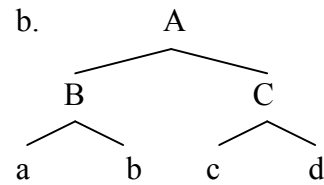
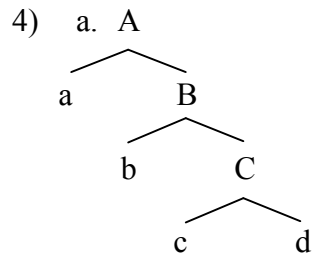
On that view, the concerns of the present thesis could be reformulated in an appealing way. Rather than supposing that the (dominance-based) structure is such as to

minimize the number of resulting, derived c-command relations, “economy of command” could be cast as a preference for syntactic structures with minimal specifications. If c-command relations are a primitive of grammar, and syntactic structures are specified by listing their c-command relations, then structures with fewer such relations are literally “smaller” than structures with an equal number of nodes, but more c-command relations. Such a conception makes a minimization principle on c-command relations especially natural. Nevertheless, in the following chapters I will keep to the more traditional view for the sake of simplifying the exposition.

2.2 *Divergence of Extremes*

Let us start with the observation that different binary-branching arrangements of the same number of nodes may encode different total numbers of c-command or dominance relations. Consider the two trees in (4a & b). In a very real sense, there is more to know about the first tree than the second; it encodes more hierarchical relations, even though they are built from the same number of constituent pieces. For example, there are 12 ‘contains’ relations⁵ in (4a), but only 10 in (4b). We find the same numbers if we count c-command relations, the syntactic relation at the heart of long-distance phenomena (entering into binding, scope, linearization, movement, and other fundamental processes).

⁵ I use “containment” and “irreflexive dominance” interchangeably.



Containment relations:

A contains: a, B, b, C, c, d

B contains: b, C, c, d

C contains: c, d

$$\Sigma = 12$$

A contains: B, C, a, b, c, d

B contains: a, b

C contains: c, d

$$\Sigma = 10$$

C-command relations:

a c-commands: B, b, C, c, d

B c-commands: a

b c-commands: C, c, d

C c-commands: b

c c-commands: d

d c-commands: c

$$\Sigma = 12$$

B c-commands: C, c, d

C c-commands: B, a, b

a c-commands: b

b c-commands: a

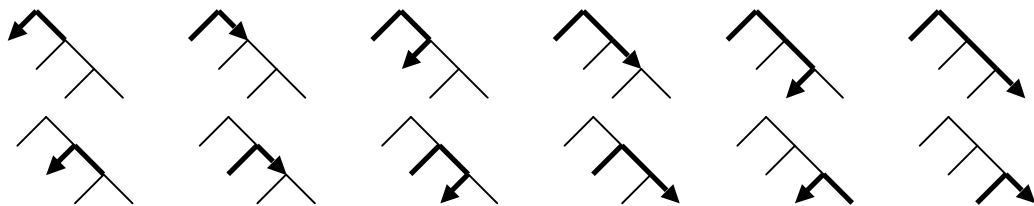
c c-commands: d

d c-commands: c

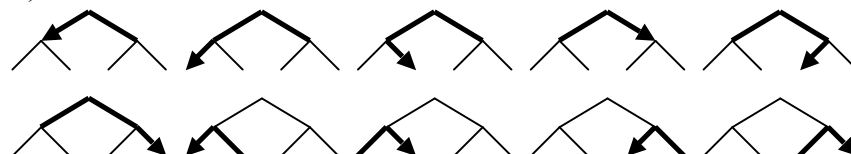
$$\Sigma = 10$$

This difference can be brought out more perspicuously in graphical form. Below, I show each of the c-command relations as a directed arrow; the tail node of the arrow c-commands the tip node. Note that there are more relations in the right-branching “spine” than in the maximally balanced “bush”.

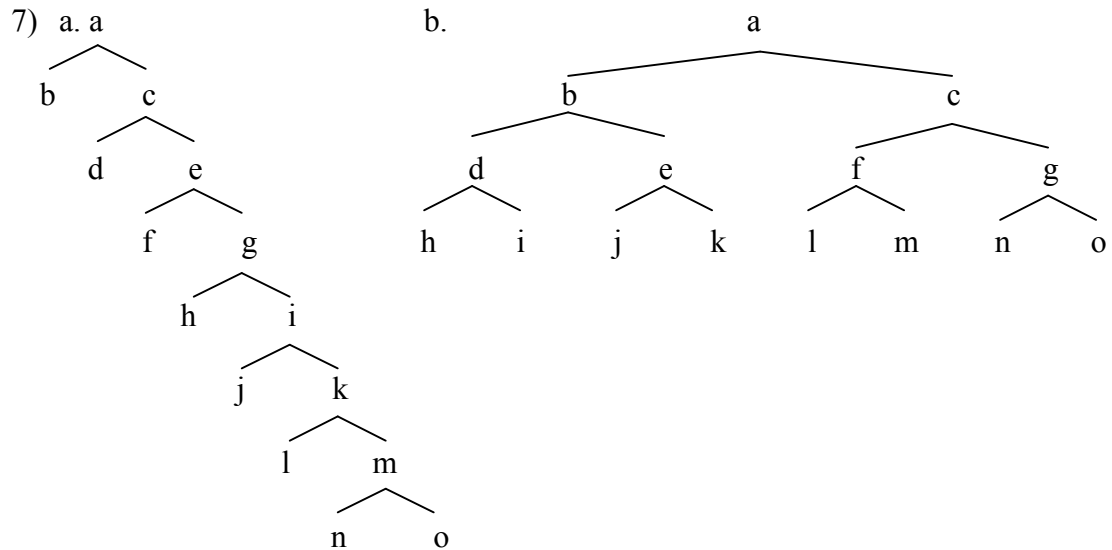
5) Spine: 12 c-command relations.



6) Bush: 10 c-command relations.



The difference between the number of c-command relations in the trees above is small, but for larger trees the difference between the extremes becomes very large. With 8 terminals, we find 56 c-command relations in the strictly right-branching tree, but only 34 in the maximally bushy tree.



C-command relations:

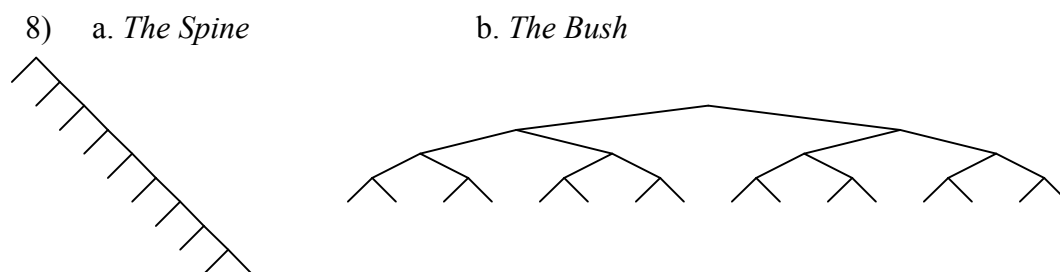
a: -
b: c,d,e,f,g,h,j,i,k,l,m,n,o
c: b
d: e,f,g,h,j,i,k,l,m,n,o
e: d
f: g,h,j,i,k,l,m,n,o
g: f
h: j,i,k,l,m,n,o
i: h
j: k,l,m,n,o
k: j
l: m,n,o
m: l
n: o
o: n

$\Sigma = 56$

a: -
b: c,f,g,l,m,n,o
c: b,d,e,h,i,j,k
d: e,j,k
e: d,h,i
f: g,n,o
g: f,l,m
h: i
i: h
j: k
k: j
l: m
m: l
n: o
o: n

$\Sigma = 34$

Restricting attention to binary-branching (Kayne 1984), we find two extremes of structure that provide the upper and lower boundaries on the total number of c-command relations present: *The Spine* (8a), a maximally deep tree, maximizes the number of c-command relations (or containment relations; their totals are identical in binary-branching structures) for a given number of terminal nodes and *the Bush* (8b), a maximally shallow tree, minimizes the number of c-command (= containment) relations for a given number of terminals. The trees below illustrate these two extremes.



The divergence in c-command/dominance totals is considerable for even moderately large trees; if the cartographic project (see discussion and references in Chapter 3) is correct in identifying dozens or even hundreds of heads in the tree, the difference between best and worst case may be very large indeed, more than an order of magnitude apart. Table 1 below illustrates this divergence, graphing the total number of c-command (or containment) relations defined in the tree as a function of the number of terminal nodes in the tree. The upper boundary curve represents the number of c-command relations in the Spine; the lower boundary is the number of c-command relations in the Bush. Other tree-shapes fall in the shaded area between them.

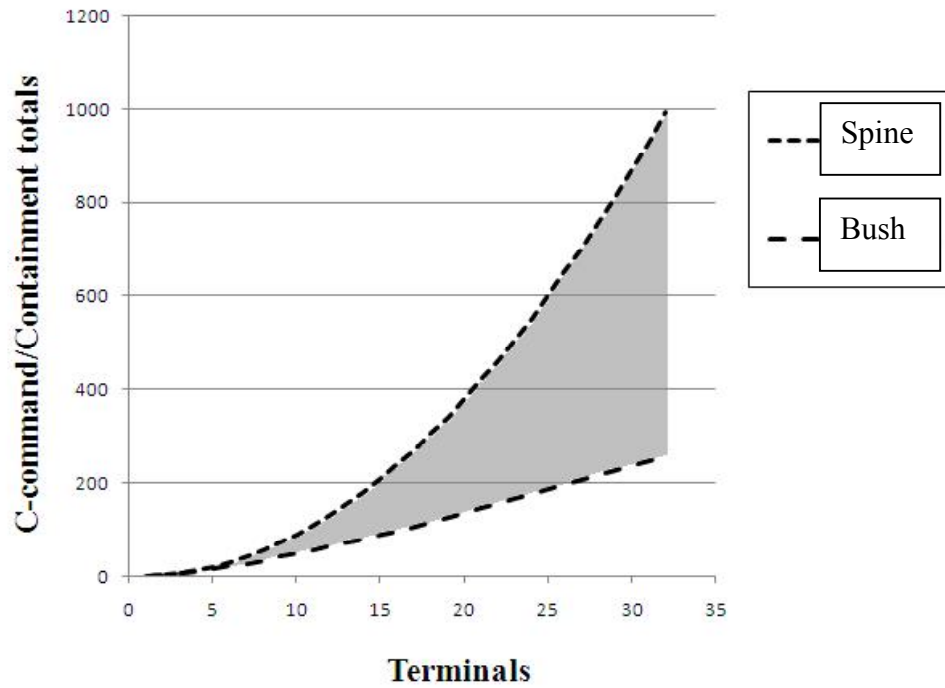


Table 1: C-command relations by depth.

And so on: the larger the tree, the greater the divergence between the extremes. In Table 2 below, I give the total number of c-command relations in Spine and Bush structures of various sizes and the ratio between them, including general formulae.

Number of terminals:	C-command relations in Spine:	C-command relations in Bush:	Ratio of Spine/Bush:
4	12	10	1.2
8	56	34	1.65
16	240	98	2.45
32	992	258	3.84
64	4,032	642	6.28
128	16,256	1,538	10.57
$n=2^k$	$n(n-1)$	$2(n(k-1)+1)$	$\sim 2^{k-1}/(k-1)$ $(n/2)/\log_2(n/2)$

Table 2: C-command relations in Spine and Bush.

Explicitly, the upper curve in Table 1 can be expressed as a function Σ of the number of terminals n , as in (9).

$$9) \quad \Sigma(n) = n(n-1)$$

The formula for the lower curve (the Bush) is rather more complicated (10).

$$10) \quad \begin{array}{l} \text{for } 2^k \leq n < 2^{k+1}, \\ \Sigma(n) = 2(n(k-1)+1) + (k+1)(n-2^k) \\ \text{Where } n = 2^k, \Sigma(n) = 2(n(k-1)+1) \end{array}$$

What values of n are appropriate: how big are real syntactic structures? Consider what Cinque and Rizzi (2010) have to say on the topic.

“A guiding idea of much current cartographic work is that the inventory of functional elements (heads or specifiers of functional projections) is much larger than is generally thought. [...] To judge from Heine and Kuteva’s (2002) four hundred, or so, independent grammaticalization targets, the number of functional elements must at least be of that order of magnitude.” (Cinque & Rizzi 2010)

Suppose then that n (the number of terminals in the tree) = 400; what does the difference between maximum and minimum totals of c-command relations look like at that scale of tree size?⁶

$$\begin{array}{ll} 11) \text{ For } n = 400: & \\ \text{Spine: } \Sigma &= 159,600 \\ \text{Bush: } \Sigma &= 3,090 \\ \text{Spine/Bush: } &\sim 51.65 \end{array}$$

In other words, the number of c-command relations defined over a Spine with 400 terminals is more than fifty times greater than the number of such relations defined over a Bush with the same number of terminals, a very considerable difference.

⁶ Of course, with phases (Chomsky 2000 *et seq*), or some other version of the syntactic cycle, even if the inventory of functional heads is this large, the effective size of trees may not be. The point of this example is just to get a sense for how extreme the difference between best and worst case is, at a point representing a reasonable upper limit on tree size.

This section has explored the mathematical basics of how the number of c-command relations scales with tree size, pointing to two “poles” of structure that bound the possibilities. In the next sections, I discuss some evidence that supports the idea that c-command minimization plays a role in grammatical conditions.

2.3 *On counting c-command relations*

As an empirical matter, it seems likely that for some c-command-based relations, some pathways matter more than others. This is clearest for computations such as agreement, or binding, or scope, and is moreover explicit in the “relativized” portion of Rizzi’s (1990) Relativized Minimality. There seem to be a number of distinct systems that establish relations over the c-command pathways in a tree, with respect to which only certain parts of the tree are relevant. For example, the Binding theory applies to nominals, not other categories. Likewise, the establishment of agreement relations involves a relation between only certain kinds of categories (those bearing phi-features).

However, certain other relations, for which c-command and/or dominance defines the relevant structural pathways, are not relativized in this way. Linearization, in Kayne’s (1994) influential work, is computed from c-command relations of the most general sort, not relativized by category type.⁷ In the computation of phrasal stress, we again see a long-distance/iterated vertical relation that is sensitive to the bare syntactic structure; this computation is not relativized to certain category types.

⁷ This is not quite true; Kayne’s formulation relies on a distinction between terminal and non-terminal nodes, and, to allow specifiers/phrasal adjunction, a segment/category distinction. The point is that the distinctions are no finer-grained than that; in particular, it is not the case that verbs, say, are linearized by principles distinct from those that apply to nouns.

Other authors have suggested that at any point in the derivation, only one direction of c-command is exploited. This has been proposed in conjunction with theories of feature checking, and the idea that all instances of Merge satisfy features, with an incomplete item asymmetrically probing a “saturated” element it Merges with. Another proposal leading to the same conclusion is Uriagereka’s (1999) theory of Multiple Spell Out; in that framework, one or the other of two complex branching objects that Merge must be “flattened”, its derivational cascade terminated and embedded within another, containing the object that “projects”, in traditional terms. This idea finds support in recent proposals like Narita’s (2010) insistence that Merge always takes the form {Head, complement}.

In light of these complications, one approach would be to explore a much “uglier” (but more realistic and empirically refined) view of which c-command and dominance pathways “count”. For example, we can imagine that the c-command relationship between a verb and a subject it agrees with ought to be minimized, because that pathway will bear a costly computation in determining the form of actual expressions. On the other hand, vertical relations that do not carry any obvious “traffic” might be thought not to matter as much, or at all. Or again, why not simply incorporate the asymmetry of c-command discussed above, if that seems to match the empirical picture?

The reason I have not chosen to pursue these complications and refinements is a matter of my goals. The point, again, is to discover what the most austere assumptions about language lead to. In this case, I assume only binary branching, and that long-distance dependencies will be established over the scaffolding of c-command (or

dominance) relations provided. I do not know of any principled way to decide, in advance and at the level of generality pursued here, which such pathways will bear costly computations (e.g., overt agreement), and which will participate only in “background” computations (like linearization and phrasal stress). It seems to me that the only reasonable way to proceed with this project is to simply count all the relations in the structure equally. In particular, it seems misguided, given these goals, to build in from the start all the details of the system. The hope, rather, is that we may gain some insight into how the phenomenon ends up working out, by growing essential details from a minimal, principled basis. I will therefore keep to the simplest assumptions, pointing out again that this work is but a preliminary investigation of these effects. I leave to future research an exploration of what predictions might follow from a more nuanced scheme for counting the cost of alternative configurations with respect to the various “flavors” of long-distance dependencies.

2.4 Agreement asymmetries and ‘minimizing links’

In this section, I review some familiar facts regarding the connection between movement and the richness of agreement. I suggest an interpretation of these patterns as reflexes of structural c-command minimization principles: the principle of economy of command predicts that richer agreement should more strongly trigger movement.

Miyagawa (2010) argues, on the basis of pre-/post-verbal subject agreement asymmetry facts in some Northern Italian dialects and Arabic, that (sufficiently rich) agreement triggers movement. We can describe this as a preference to shorten the c-

command paths actually used to compute agreement. It suggests a penalty for too much “traffic” on long c-command pathways, lending some plausibility to the notion that minimization of c-command relations plays a role in the syntax.

Miyagawa notes that in the Italian dialects Fiorentino (12a) and Trentino (12b), verbs resist agreement with post-verbal subjects. In this configuration, agreement is the default 3rd person masculine singular.

- 12) a. Gli è venuto delle ragazze. (Fiorentino)
 b. E' vegnú qualche putela. (Trentino)
 is come some girls
 ‘Some girls have come.’
 (Miyagawa 2010: 6, citing Brandi and Cordin 1989:121–122)

There is no plural verbal agreement for plural post-verbal subjects:

- 13) a. *Le son venute delle ragazze. (Fiorentino)
 b. *L' è vegnuda qualche putela. (Trentino)
 they are come some girls
 ‘Some girls have come.’ (ibid.)

However, a preverbal subject triggers agreement (in this case, for feminine gender):

- 14) a. La Maria la parla. (Fiorentino)
 b. La Maria la parla. (Trentino)
 the Mary she speaks
 ‘Mary speaks.’ (ibid.)

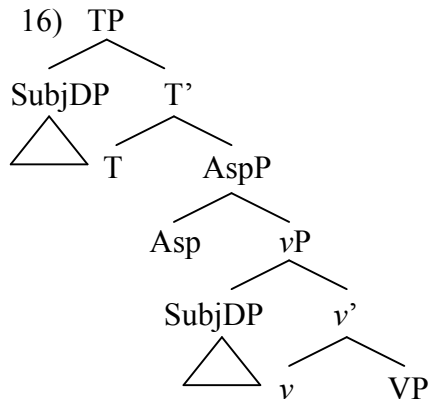
Miyagawa notes a similar agreement symmetry in Arabic. Unlike the all-or-nothing asymmetry observed in the Italian dialects discussed above, in Arabic there is a split between full and partial agreement:

“[P]ostverbal subjects trigger the partial agreement of person and gender (the verb also has the default singular agreement form) whereas preverbal subject triggers the full agreement of person, gender, and number (e.g., Bahloul and Harbert 1993, Benmamoun 1992, Fassi Fehri 1993).” (ibid, 8)

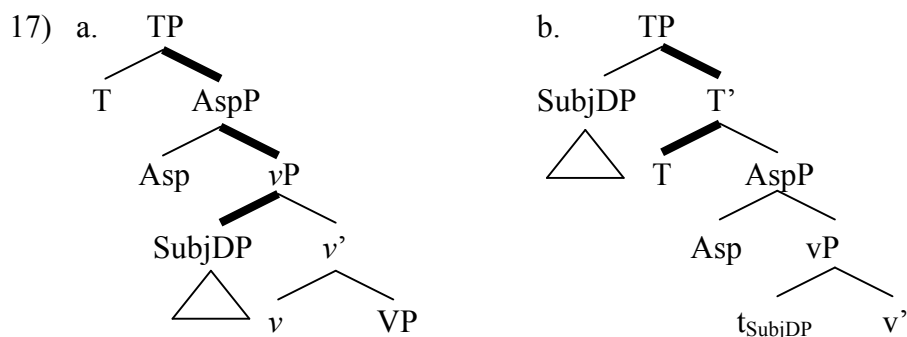
The following examples illustrate:

- 15) a. Qadim-a (/ *qadim-uu) al-ʔawlaadu.
 came -3MS came -MP the-boys-3MP
 ‘The boys came.’
- b. Al-ʔawlaadu qadim-uu (/ *qadim-a) [t]
 the-boys-3MP came -3MP came -3MS
 ‘The boys came.’ (Bahloul and Harbert 1993:15, cited in Miyagawa 2010:8)

At least descriptively, we may say that the c-command path bearing subject-verb agreement is shortened when it bears more traffic (richer agreement). To make this clear, let us sketch some rather conservative assumptions about the structures involved. Standard analyses have the moved, preverbal subject in the specifier of the agreement-bearing head (often taken to be T, elsewhere AgrS). The unmoved subject is taken to be in the specifier of vP, again a standard analysis. There is ample reason to posit additional structure between the T and v layers. For example, Aspect is often taken to be Merged above v but below T. Supposing so, we have a tree schematically like (8) (leaving open the possibility that this may be shorthand for richer structure).



Compare the longer c-command path, from T to the post-verbal subject DP, which bears partial or default agreement (17a), with the shorter path from preverbal subject DP to T, which bears full agreement (17b).



T...Subj: Default/partial agreement

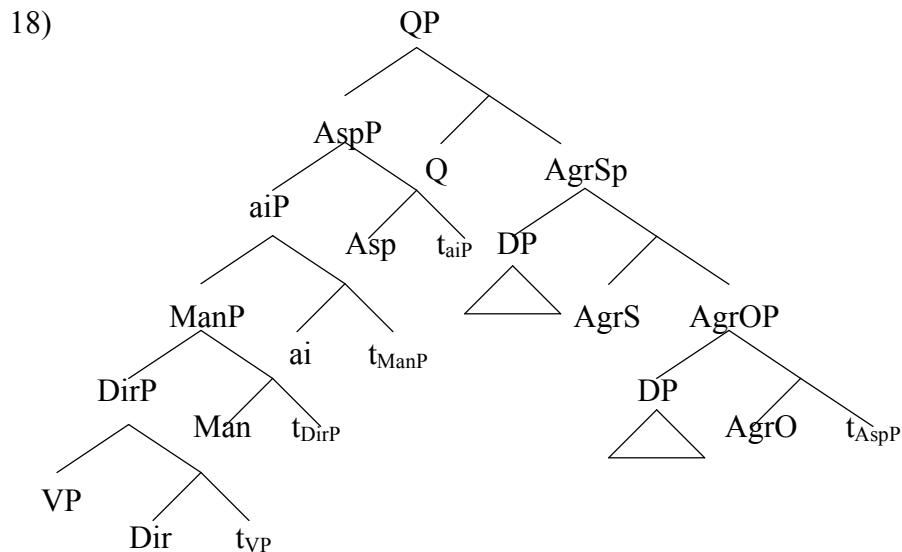
Subj T: Full agreement

This fits well with the perspective of Chapters 4 and 5, where I suggest that movement is cheap and ubiquitous, while c-command relations are costly. In this case, movement brings the agreed-with subject as close as movement can bring it to the agreeing head, by hypothesis from a more remote position (perhaps much) deeper in the tree. This example illustrates one possible reflex of a concern for minimization of the depth of c-command relations in grammatical conditions.

Notice that although movement involves adding a new, linguistically relevant c-command relation (the one bearing the long-distance relation between the moved object and its deleted copy or trace), it nevertheless serves to reduce the overall “wirelength” (the sum of the lengths of all of the relevant c-command relations). In particular, movement creates a configuration in which the c-command paths along which agreement is established are shortened. Thus, despite the “extra step” involved in doing movement, the resulting form is in a sense more optimal. It is the central ideas of this thesis that this “path-shortening” explains a large body of grammatical phenomena.

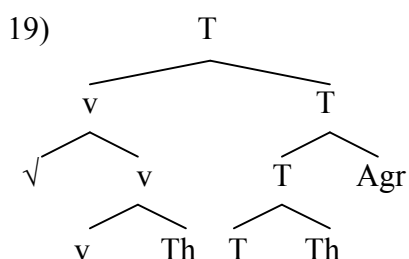
2.5 *The bushiness of natural language expressions*

The basic premise of this work is that the ideal syntactic structure, all else equal, is a Bush rather than a Spine. This seemingly conflicts with the consensus that the underlying base structure is a Spine (see the next section for more on this issue). However, if the burden of c-command based computation is incurred in the processes which “read” syntactic structure at the interfaces with non-linguistic mental components, then what matters is not the base, but the structure delivered to the interfaces (reflecting movement). Furthermore, and of central importance in this work, the very same cartographic literature mentioned above, following Kayne (1994), identifies a host of leftward movements involved in deriving the surface order of many languages, often a tangle of such radically transforming the base order. Each such movement creates a left branch, deviating from the Spine. Thus, at least in some languages, movement serves to pack the tree structure into something close to the ideal of the Bush. The following structure for the clause structure of Niuean, adapted from Kahnemuyipour and Massam (2006), illustrates:

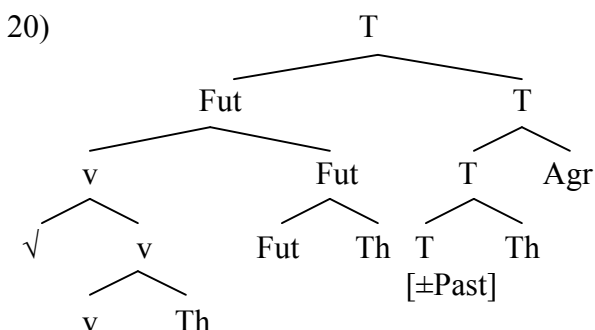


From a strongly right-branching base tree, the movements found in Niuean create a very bushy tree. In fact, depending on the structural details of the categories here, the surface form above may actually be as close to the Bush as one can get, a matter of considerable interest if true.⁸

At the level of the word the tree forms we find may in fact be quite bushy, cf. Oltra-Massuet and Arregi (2005) on Spanish. Inferring syntactic structure from phonological facts, they give the following analysis of the form of verbs in Spanish (their 8a, p 46):⁹



They give the following structure for future and conditional verbs (their (16), p 53):

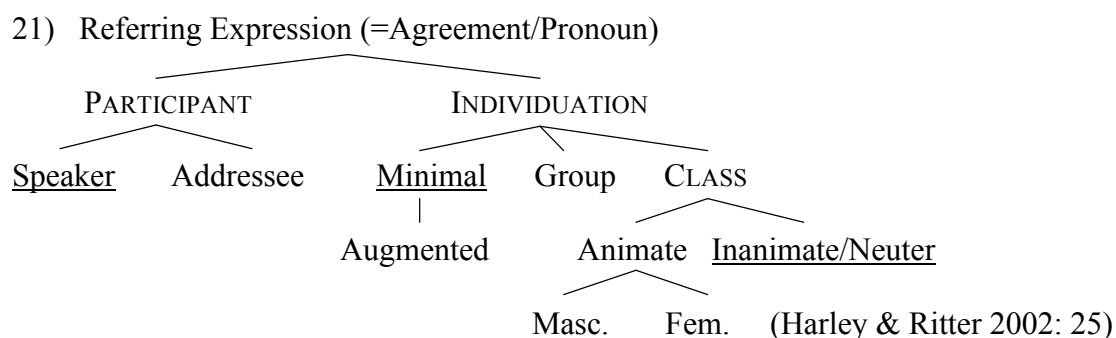


⁸ Difficult questions arise as to how to reconcile such an idea with undeniable cross-linguistic variation (not all languages have the same movements). One widely-discussed possibility is that the universal structure revealed by the cartographers is only partially represented in particular languages (see the next chapter for relevant discussion). If so, it is possible that each language has the best movements for its particular “reduction” of the universal structure. In light of the complexity involved in deriving testable predictions from this idea, I leave an exploration to future work (though see Chapter 4 for further relevant comments).

⁹ “Th” here stands for a theme vowel.

In this language, the syntactic form of verbs, as revealed by foot structure, is rather bushy, strongly reducing total c-command/dominance relations compared to a spinal (right-branching) arrangement of the same terminals. It is possible to quantify how “bushy” the structure in (20) is. Recall that, for 8 terminals as in this tree, there will be between 34 and 56 total c-command relations. In (20), there are 36 total c-command relations, one step away¹⁰ from absolute optimization in this sense. It is the point of this dissertation to argue that this sort of surface tree shape is neither exceptional nor accidental; language is this way for a reason.

When we examine the internal structure of phi features—and recent work indicates that the details of this structure are ‘visible’ for c-command, for example entering piecemeal into agreement relations with distinct controllers (see e.g. Bejar and Rezac 2009)—we likewise seem to find somewhat bushy structure. For example, Harley and Ritter (2002: 25) propose the following universal feature geometry for phi-features:



Of course, Harley and Ritter have in mind an implicational structure, like the feature geometries of phonology, representing logical dependency in the cross-linguistic distribution of phi features (if a node in this geometry is present in the morphology of a

¹⁰ Note that the number of c-command relations in a binary-branching tree must be an even number.

language, all dominating nodes must be as well). It is a further step to claim that this is effectively *syntactic* structure (and note the ternary branching present under the Individuation node), though Harley (p.c.) indicates that this is a reasonable interpretation.¹¹

All of this is just to stave off rejection out of hand: the branching forms of language reviewed above are rather closer to the Bush than to the Spine. Excepting the feature geometry, intended to be universal, readers may object that these examples are ‘cherry-picked’: not all languages have radical movement like Niuean, or bushy words like Spanish.

What about languages like English, say, where the phrase structure is thought to surface in something closer to the base order, and word-internal structure is generally impoverished? It is not clear that that is an accurate characterization; as reviewed in the next section, there is some evidence for movement low in the English verb phrase on the basis of ordering facts. And English uncontroversially has prolific movement of other kinds, including movement of nominals for Case; as I argue later in much more detail, such movements seem to be well-chosen to balance the resulting tree.

2.6 *On the spinality of the base*

An important question immediately arises, at this point. Why should the base structure effectively be a Spine, as seems to be the consensus, when that structure is the worst possible in terms of minimizing long-distance dependencies?

¹¹ See also Déchaine & Wiltschko (2002), who advance a less articulated (but specifically syntactic) proposal for the internal structure of nominals, distinguishing DPs, PhiPs, and NPs.

There is every reason to think that the base structure does not survive to the surface in any language, maybe not even in any domain (e.g., phase). Take English as a case in point: the gross nominal order, as examined in Chapter 5, reflects the supposed base order, Dem > Num > Adj > Noun (Cinque 1996, 2005, Abels & Neeleman 2009). But a closer look suggests that quite significant movement occurs even inside English DPs, as the examples below illustrate:

22) [the very same thing [that you saw]]

23) [the drawing [of the killer] [that you made]]

Relative clauses follow the noun they modify. If the noun is at the bottom of the nominal tree, and relative clauses are Merged higher up (Kayne 2008, Cinque 2003), then this is evidence for leftward movement of the noun. An analysis of relative clauses as complements of nouns is inconsistent with examples like the second above, where a theme-like *of*-phrase intervenes between the head noun and the relative clause.

24) the picture [on the wall]

Of course, PPs may follow the noun as well.

25) the drawing-s [of the killer]

The appearance of plural as a suffix on the noun is also interpreted as arising via movement of the noun to the left of that position (Julien 2002, Svenonius 2007). That understanding of suffixation as movement also argues for a non-complement position even for *of*-phrases. Indeed, Kayne (2008) argues that nouns cannot have complements in principle, being self-Merged (cf. Guimarães 2000) at the beginning of the derivation of a DP.

26) your dog/those dog-s of your-s

The last example shows that possession may also be expressed post-nominally, dissociated from the pronominal demonstrative. All of the post-nominal material in the examples above, save the plural, might well collapse as forms of (sometimes reduced) relative clauses, including as well some post-nominal adjectives (*a man alone*, etc.). Even so, we have evidence then for at least two post-nominal positions, and so at least one move carrying N(+) leftward.

Consider next the clausal domain, again keeping to English. The basic order of elements in the clausal spine seems, at first glance, to remain undisrupted in English. But that misses the important fact that in this language, DP arguments move, prolifically and obligatorily, along that Spine. There is good evidence for a “Raising to Object” move of object DPs to a specifier position just below the landing site of the verb (Koizumi 1993, Johnson 1991), with the verb taken to move to “little *v*” (Kratzer 1996). In other words, an English *vP* has a structure schematically like this:

27) [$\sqrt{\text{Verb}}$ *v* [ObjDP ... [... *t*_{ObjDP} ... *t*_{Verb}].

Higher up, English subjects undergo the still-mysterious EPP raising to (a specifier near) TP, reinterpreting somewhat the Extended Projection Principle of Chomsky (1981) in light of Koopman & Sportiche (1991). And *wh*-phrases and quantifier phrases raise still further into CP (Chomsky 1986), sometimes covertly. Within the spine itself, the verbal morphology exhibits Affix-Hopping (Chomsky 1957), another deviation from the expected base order.

The overall picture is that quite radical reordering occurs even in a language like English, often thought to basically preserve the base order in its surface form. More generally, in any language, any instance of non-head-finality indicates movement, according to Kayne (1994). So already in the relative order of subject, object, and verb, we must admit movement in the majority of languages (i.e., in all but one base order: SVO if the object is introduced as complement of the verb, as traditionally assumed but lately questioned; or SOV if the object, like all arguments, is introduced in a specifier position within an extended projection of the verb).

In light of these remarks, another crucial point to be made is that the concern of this thesis – economy of command, the hypothesized bias to minimize the burden of long-distance dependencies in linguistic expressions – is construed as realizational. To put it simply, what matters is economy of command in the representations presented to the interfaces. In particular, such objects generally do not reflect the base form, but rather the transformed, post-movement result.

Chomsky (2001) suggests that the very dichotomy between merge and movement is a mistake, rather interpreting movement as merge applying to an object and a proper subpart thereof. On that view, we may say that displacement is a natural part of structure-building; conversely, the structure that would result from pure external merge (the spinal base) is not a naturally-occurring syntactic structure. Seen this way, the base is really the abstract input to the structure-building system, an idealization that never appears in surface form as such.

Looking at the issues the other way round, so to speak, much recent work finds conceptual motivation for spinal, head-complement structure as the ideal form of syntax. This idea appears, in various forms, in Narita (2010), in Uriagereka's (1999) "derivational cascades", in Starke's (2004) proposal to eliminate specifiers (conflating them with heads, such that head-complement configurations exhaust syntactic structure), in Jayaseelan's (2008) "linear" syntax, in Chomsky's arguments based on minimal search in labeling, related to Moro's (2000) ideas on Dynamic Antisymmetry, and in Uriagereka & Hornstein's (2002) "reprojection" analysis of quantifiers.¹²

Another line of attack on understanding the spinal nature of the base is to note that the hierarchical links it instantiates are shorthand for selection relations, and selection seems to be basically linear (limited to complements)¹³. Recalling again the concern to minimize long-distance dependencies, the relations that form the spinal base are already maximally local. If only the shortest possible paths carry computation (here, selection of an appropriate complement for a head), then no preference for bushy structures is motivated at that level; it is only when truly long-distance dependencies come into play that structure makes a difference for computational efficiency.

Given that strict head-complement structure finds robust conceptual motivation in the above-mentioned works, we may take it that the spinal nature of the basis has its roots in such concerns. What is surprising, and requires explanation, is the bald fact that

¹² Their analysis is of considerable interest from the present perspective, in that it argues that elements combine with multiple arguments (for a quantifier, its restriction and its scope) piecemeal, derivationally (in effect, through two distinct stages of head-complement structure).

¹³ Thanks to Andrew Carnie for discussion on this point. Note that selected arguments seem to involve specifier selection, though perhaps indirectly, through selection of an argument-introducing functional head.

surface structure deviates from that ideal. In this regard, the present thesis has much to offer: namely, a compelling reason to transform the simplest kind of hierarchical structure into a more complicated—but in present terms more optimal—form.

2.7 *Summary*

This chapter has reviewed the notion of c-command. I briefly explored the mathematical terrain of c-command and dominance relations. I described two extremes of branching structure (maximally deep, and maximally shallow), and derived expressions for how the total number of c-command (and dominance) relations scale with tree size.

I discussed some examples that arguably exhibit signs of the structural preference argued for here (including Niuean clause structure, Spanish word structure, and morphological feature geometry). I pointed to the well-known property of Minimality/locality and certain agreement asymmetry facts as lending some credence to the hypothesis that minimization of c-command relations plays a role in determining syntactic conditions.

Much Minimalist work has been concerned with the notion of derivational economy. Economy of command is closely related to these concerns; however, especially in terms of motivating movement, it is better described as a matter of interpretational economy. That is, the benefits of movement obtain, not within narrow syntax, but in the mapping to interface interpretations of syntactic form as sound and meaning, proceeding by phase. It is only in post-syntactic interpretation that movement reduces the search space. In this way, economy of command relates also to what has been described as “bare

output conditions”, another central concern of Minimalist theorizing. In effect, economy of command is a matter of minimizing the cost of reading syntactic forms at the interfaces.

In the next chapter, I take up further relevant empirical considerations. In that chapter, I argue that minimization of c-command (and dominance) ‘links’ matters most in post-syntactic structure, after movement has applied. Leaving the details for later, there is reason to think that a large core of c-command-based linguistic computations (I have in mind linearization, agreement, and binding, and perhaps further relations) applies only to the output of movement. In that case, transformation of a spinal base structure into a bushy surface form can be understood in terms of minimization of the total number of c-command relations, as read by post-syntactic processes.

CHAPTER 3: UNIFORM STRUCTURE AND CYCLIC INTERPRETATION

3.0 Syntactic assumptions

In this chapter, I outline the assumptions I adopt about the architecture of the grammar. There are two broad issues to be addressed here. The first concerns the amount of structure present in the expressions of natural language, and whether and how that structure can vary between languages, and between expressions in a single language. The second issue revolves around cyclic interpretation, the notion that the syntactic engine interfaces with external systems by handing off representations at designated intervals (phases, in Chomsky's 2000 terms).

In regards to the first issue, structure and variation, I argue that syntactic structure is nearly uniform across all languages, and across expressions. There are two caveats I will discuss; the first concerns variable extent of projection, and the second is about variation in the number and location of nodes associated with agreement and negation. Variation of the first kind will be considered in Chapter 4, where I will argue that it plays a crucial role in triggering or failing to trigger certain movements. The second kind of variation will be largely set aside in this work.

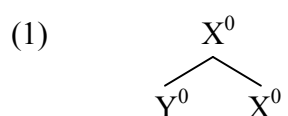
With respect to cyclic interpretation, my goal is to justify the approach to movement developed in chapters 4 and 5, where I will argue that movement is driven by economy of command. The reasoning here can be summarized as follows. First, the c-command relations that seem to "count" in linguistic relations hold at the interfaces, where a chain of copies formed by movement collapses onto a single position, generally the highest in the chain (lower copies are effectively invisible). Thus, the c-command

relations that enter into the computation of linear order, agreement, (some) binding, and nuclear stress are fed by movement. This means that movement can, in principle, reduce the number of c-command relations effectively present (by transforming a spinal structure into a more bushy one, before it is “read”). I show explicitly how bushier structures present less of a computational burden at the interfaces.

3.1 *Articulated syntax*

Let me first address my assumption that ‘terminal in the tree’ should not be identified with ‘word in the surface structure’ (in favor of multiplying the number of the former taken to correspond to the latter). It is an old idea that some word-internal, ‘morphological’ structure just is syntactic structure, a more articulated cartography obscured by higher-order, secondary groupings. This idea has gained prominence in recent work, with much of what was once thought to be the domain of morphology taken over by syntax (Halle & Marantz 1993, 1994, Marantz 1997).

Julien (2002) gives an extreme version of this view, supposing that words are in effect something like constellations, a perceptual illusion caused by accidents of syntax, and that the individual terminals that end up forming a ‘word’ need have no consistent structural relationship, other than being linearized adjacent to each other. In particular, Julien rejects head-adjunction structures like the following as the only permitted syntactic structure of a complex word:



That is, for Julien word-internal structure need not indicate recursion of X^0 (head adjunction), as for Travis (1984), Baker (1988), and others. Instead, many complex words are to be reanalyzed as arising from movements rearranging a fixed base order, perhaps quite radically. This work aligns with radical movement analyses seeking to replace head-movement with phrasal movement, including Müller (1998), Koopman & Szabolcsi (2000), Mahajan (2000), Julien (2002), Svenonius (2007), and many more.

This entails, clearly, a denial of the so-called Lexicalist Hypothesis, the view that words have a privileged syntactic status as indivisible atoms of the computation, with any internal structure formed by a distinct word-formation system, and opaque to the syntactic computation (Chomsky 1970, Williams & DiSciullo 1987, Anderson 1992, Ackema & Neeleman 2004, among many others). The issue is too large to tackle here, but I note that the results I derive below resonate strongly with the general view of a many-to-one correspondence of syntactic terminals to words, and radical movement (to derive the “word-internal order” of smaller syntactic units).

I draw, instead, upon the tradition of comparative micro-syntax or cartography. The beginnings of this project, expanding the range of functional categories admitted within a single language, while limiting or eliminating variation among languages in the number and order of the same, can be seen in Chomsky’s (1986) expansion of the X-bar formalism from the lexical categories to functional categories associated with the sentence (CP and IP). Abney (1987) added DPs to the inventory of functional elements; Pollock (1989) “split” IP into multiple projections to account for cross-linguistic variation in verb-movement possibilities. Larson’s (1988) VP-shell analysis of

ditransitives implied an expanded cartography of VP, a point of view in line with the work of Hale and Keyser (1993), while Kratzer (1996) proposed adding a “little *v*” above the verb. The project developed with Rizzi’s (1997) proposals for an articulated “left periphery” (CP), while Cinque (1996, 1999) produced detailed cartographic maps on the basis of intricate facts of adjective and adverb ordering. See the works collected in Cinque (2002), Rizzi (2004), and Belletti (2004) as a starting point for this enterprise.

Cinque & Rizzi (2010) discuss successes for the cartographic project, including evidence in favor of the universality of the structures proposed:

“[...] subtle evidence for the presence of a DP projection in languages like Serbo-Croatian, Russian, and Japanese, which lack overt determiners (Progovac 1998, Pereltsvaig 2007, Furuya 2008); or the indirect evidence discussed in Kayne (2003,219) and Cinque (2006b) for the presence of numeral classifiers in languages like English and Italian, which are traditionally taken not to be numeral classifier languages.”

3.2 Uniformity of structure

For the purposes of Chapter 5, I will assume a rigidly fixed universal structure for the shape of the DP. This is partly a methodological choice, as it makes for the cleanest, most restrictive predictions about what that shape must look like. It is also the simplest assumption one could make, a consideration that carries some weight, given the goals of this work.

However, I think that a more reasonable assessment of the facts would provide for some degree of variation in structure, of at least two kinds. First, there is the matter of more or less extended projections. Second, there is a class of elements that appears not to

have a fixed position with respect to the rest of the cartographic hierarchy: agreement and negation. I consider each of these in turn.

3.2.1 Variation I: Extent of projection

The nominal domain illustrates: we have bare nouns, PhiPs (Déchaine & Wiltschko 2002), DPs, (and perhaps PPs are a further extended projection of nominals, cf. Grimshaw 1991). Each arguably represents a further elaboration on the next smaller structure. The size of the extended projection varies between constructions in a single language: so in English, sometimes nominals are bare nouns, sometimes they have richer structure. The extent of projection might also vary between languages, with some languages maximally projecting only a portion of the full structure.

This view of structural variation as involving variation in the extent of projection finds some support in acquisition facts. Cinque (2004: 684) points out that distinctions in aspect are robustly acquired before distinctions in tense, citing Antinucci and Miller (1976), Weist (1986), and Schlyter (1990); moreover, acquisition of aspectual adverbs precedes that of temporal adverbs. Ouhalla (1991) goes further, suggesting a fixed order of maturation of functional categories. The basic picture is of a universal hierarchy that “grows” bottom-up as the child acquires a language. On this view, a possible source for differences among languages is variation in the upper limit of extended projection. This

calls to mind the debate about whether some languages lack determiners (the idea being that the nominal projection in such languages is impoverished at the top).¹

3.2.2 *Variation II: Agreement and negation*

There is a second kind of structural variation, revolving around a class of elements (negation and agreement) which, as even Cinque (1999) admits, do not occupy a unique place in the hierarchy.

“In the recent literature it is sometimes claimed that the relative order of functional heads is subject to parametric variation across languages. Interestingly, the cases which are brought up to support this conclusion all involve, in one way or another, the position of negation or of agreement with regard to other functional heads, especially Tense[.]”
(Cinque 1999: 136).

He goes on to note that “[T]he position of negation and of agreement [...] can vary even within the same language.” (1999: 137) On the other hand, he argues against the idea that other functional heads can be optionally present, even when they receive no overt morphological expression. Instead, even when functional heads have a default interpretation, often with no overt morphological realization, they are still structurally present (Cinque 1999: 131). He offers the following pair of examples, indicating that despite the considerable difference in overt material present, they both plausibly have the same full functional structure.²

¹ The question is not whether all languages overtly express the determiner category—some clearly do not—but rather whether the abstract structure is still present or not.

² There are two issues here. The first is whether an element *b*, hierarchically between *a* and *c* ($a > b > c$), can be omitted if *a* and *c* are present. The second is whether, given some portion of a domain (say, a clause), the complete structure must always be projected (i.e., for $a > b$, whether the presence of *b* entails the presence of *a*), or if there is an option to “truncate” the top of the structure (this is implicit in the discussion of the status of ECM clauses as TPs or CPs, cf Bošković 1995). I assume that “inner omission”

- (2) a. Prices rise.
 b. Prices must not have been being raised. (Cinque 1999: 131, his (2a-b))

We might think of the variation in the number and location of agreement and negation nodes as “insertions,” extending the universal base structure with further nodes at variable internal locations.³

Supposing that semantic uniformity is informative with respect to syntactic uniformity, Croft’s Semantic Map Connectivity Hypothesis is highly relevant to the discussion here: “any relevant language-specific and/or construction-specific category should map onto a *connected region* in conceptual space.” (Croft 2001: 96)

The claim is that the background conceptual space is universal, with variation among languages in their semantic categories arising from a kind of syncretism: some contiguous portion of the articulated universal structure is “collapsed” onto a specific category, in a particular language.

It is not too large a leap, I think, to identify this conceptual structure as syntactic structure, namely the “base” (the spine of functional categories). If so, then the Semantic Map Connectivity Hypothesis would translate into a statement about how languages vary in their underlying syntactic structure. For each language, there is a many-to-one map of the universal set of categories onto the language-particular set, preserving the relevant hierarchical relations (if C_1 and C_2 are distinct categories in some language, and $C_1 > C_2$,

is ruled out, reading Cinque’s remarks quoted in the text as agreeing with this. On the other hand, variable extent of projection (of a fixed structure) is a matter I leave open, beyond the schematic remarks here and in Chapter 4.

³ Note that, insofar as agreement nodes are not a part of the underlying universal structure, we can make intuitive sense of the “parasitic” nature of Agreement: having no innately-specified content, they must acquire their content through other means (by copying features from some appropriate nearby source). These comments do not extend to negation, of course.

then for each C_i , C_j categories from the universal structure mapped into C_1 and C_2 , respectively, $C_i > C_j$). This is in effect an extension of the variation admitted above with respect to agreement and negation, whereby structures vary not just at the top, so to speak (in their extent of projection), but also vary in internal structure.

3.2.3 Consideration of variation in this work

In Chapter 4, I consider at least the first kind of variation, suggesting that differences in movement relate to more or less extended projections. I consider this kind of variation at a cross-linguistic level, suggesting an account of the connection between rich agreement and movement in these terms (for example with respect to the presence of agreement and the status of P as a postposition or a preposition, taken to reflect movement of its nominal complement or not, respectively). This kind of structural variation can also hold within a single language, as for Object Shift, which I claim to reflect whether or not the syntactic layer of nominal phrases associated with definiteness is projected or not; and for the different landing sites of different kinds of nominals, e.g. strong vs. weak pronouns (Cardinaletti 2004). See that chapter for details.

However, I set aside “internal” variation, for example with respect to negation and agreement. This is largely a methodological choice, as ignoring this kind of variation leads to a simpler mathematical investigation. In future work, I hope to return to this issue; for now, I merely note that strong predictions can be derived about how differences in structure should correlate with differences in patterns of movement.

3.2.4 *Interim summary: Assumptions about structure adopted in this work*

For concreteness, in what follows, I tentatively adopt the assumptions below, in part because they afford a convenient and restrictive idealization for the present investigation. Assumptions (a)-(d) follow Cinque (1999) and Kayne (1994); assumption (e) aligns with phrasal movement analyses of supposed head movement cited above.

- a) There is a single, universal hierarchy of syntactic categories.
- b) This structure is identical across languages and expressions, up to variable extent of projection.
- c) This structure, without movement, would be linearized in a consistent highest-left order (Kayne 1994), what I will call the base order.
- d) Any deviations from the base order are derived by leftward movement.
- e) Movement may not affect single terminals; apparent cases of head movement reflect one or both of the following cases: (i) the ‘head’ has an articulated internal syntactic structure of several terminals, or (ii) the element in question has been rearranged as part of a phrasal movement.

3.3 *Cyclic Interpretation*

The ideas developed below (especially, that movement serves to reduce c-command relations) depend on a certain hypothesis about the relationship between syntax and the interpretive interfaces that “read” the syntactic form as instructions for computing meanings and pronouncing sentences. In particular, for the claims advanced here to go through, there must be “levels of representation”, i.e. constrained bottlenecks of access to

the syntactic derivation by the interpretive interfaces. This is fairly standard (for a recent articulation, see Chomsky's (2000 et seq) theory of phases), but it should be noted that some authors have advanced proposals that give interpretive access to the derivation directly (see for example Epstein et al 1998).

What is at issue is whether interpretation proceeds in lockstep with each step of the derivation, or whether derivations proceed for non-trivial stretches before periodically handing off a partial representation. Only in the latter case, where interpretation reads the results of a derivation, does movement as tree-balancing appear sensible.

To draw this out as clearly as possible, take a simple case of movement, which by hypothesis reduces c-command relations in the post-movement as compared to the pre-movement tree. If the transformed structure is what is "seen" by the interpretive components, then its reduced c-command total can be of benefit: c-command appears to describe the pathways that are "read" by interpretive processes, so presenting fewer and shorter such relations simplifies interpretation. If, however, interpretation proceeds in lockstep with the derivation, the same movement will not be of benefit. In effect, the "reward" for movement consists of erasing c-command into the lower copy of the chain (for some processes at least), but if each step is interpreted, the cost of c-command into the lower copy has already been incurred.⁴

⁴ Conceivably, there could still be a benefit to movement in a lockstep-interpretation scenario: it shortens c-command relations into the moved object from higher up the tree. In effect, it can serve as a way to "bump to the top of the list" an item that will enter into further computation. However, that in itself is not sufficient to motivate the short steps of movement, because those short movements themselves require substantive search relations to implement. It is not at all clear that just searching once, along a longer pathway, from the higher position is any worse than multiple short searches.

3.3.1 *On copies*

At first blush, there might seem to be a tension between the ideas here and the widely adopted copy theory of movement (Chomsky 1995b, Nunes 1995, Bošković 2001, among others). If movement is literally copying, within narrow syntax such an operation should be ruled out for the very concerns sketched here: it always increases the number of vertical relations, making “worse” trees. However, there is ample evidence that these vertical relations matter not just within syntax (say, for probe-goal relations), but also in the cyclic mapping to the interfaces. What is important is that, with respect to interpretation at the interface, multiple copies generally collapse onto a single location. To take one clear example, only one copy of a moved item is linearized (moreover, the highest copy, a fact which is also directly predicted here). Much the same seems to be true for semantic interpretation as well:

“Although chains have been used to account for various processes involving scope and binding, particularly in the context of reconstruction, it is, to my knowledge, never the case that multiple occurrences of a given element are interpreted. The use of chains on the meaning side amounts to allowing different *portions* of an element to be interpreted in different positions as in reconstruction effects (e.g., the operator part of a wh-word is interpreted in SpecCP, but its restriction is interpreted in a lower, ‘reconstructed’ position...)” (Boeckx 2008:47)

Consider operator-variable chains like that postulated in movement of wh-phrases (Chomsky 1986): the point is that even though parts of the phrase may be distributed over distinct chain positions, each part is pronounced and interpreted only once.

- (3) Which man did you see [~~which~~ ~~man~~]? PF
- (4) [Which ~~man~~] did you see [~~which~~ man]? LF

In fact, the present account makes predictions about which copies in a chain should be pronounced. Now, economy conditions of a very general sort militate against interpretation of every occurrence in a chain:

“Failure to pronounce all but one occurrence follows from third-factor considerations of efficient computation, since it reduces the burden of repeated application of the rules that transform internal structures to phonetic form – a heavy burden when we consider real cases.” (Chomsky 2009: 28)

However, that motivates only the fact that a single copy is pronounced; it says nothing about why it is usually the highest copy in a chain that is pronounced.⁵ On the present account, that fact falls out as well, and for the same reason: pronouncing the copy (of a sufficiently large object) in the highest occurrence usually minimizes the total number of c-command and dominance relations (compared to other choices of copy to pronounce), here postulated to be another important aspect of “the heavy burden” of realization at the interfaces.

Recall the concerns of Yngve (1960), reviewed in Chapter 1. In the absence of other factors, if the choice is to be made by considerations of efficient computation, explicitly rooted in the mapping to a surface form, we might expect that Yngve’s “save the hardest for last” would decide the issue. If only one copy is to be pronounced, it should, all else equal, be the deepest, rightmost one, as that choice least taxes memory resources. Instead, it is consistently the highest, furthest left copy in a chain that is pronounced; the present account allows us to understand this curious fact.

⁵ Nunes (1995) constructs a Minimalist explanation for pronouncing the highest copy in a chain, in terms of economy and/or convergence. He argues that lower copies contain more unreadable “junk” in the form of unchecked features, which must either be deleted (creating “extra work” at the interface) or cause the derivation to crash.

3.3.2 *Linearization*

It is quite straightforward to find conceptual motivation for economy of command in terms of linearization. As is particularly clear, what is relevant for linearization is the structure produced by movement, by hypothesis one transformed into a bushier structure (with fewer c-command relations). In other words, movement is overt; elements get pronounced in the position where movement has deposited them. According to Kayne's (1994) Linear Correspondence Axiom, linear order is read from (asymmetric) c-command relations. Thus, I will argue, a bushier tree minimizes the work of "reading" a linear order from c-command relations.

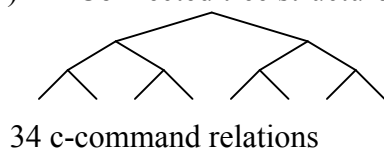
This requires immediate justification. Kayne's theory is quite literally about Antisymmetry, yet the present account supposes that maximally symmetric structure is optimal. However, a look at the details shows that there is no contradiction here, and that bushier trees create less burden for the antisymmetric computation of linear order.

A crucial ingredient to the reconciliation of economy of command and Antisymmetry is the opacity of left branches. For Kayne (1994), this property is a consequence of defining c-command relations with respect to categories rather than segments. Chomsky (1995a) achieves a similar result by limiting c-command to maximal categories (phrases), with the effect that phrasal adjuncts (specifiers) c-command the object they are merged with, but not vice-versa. Uriagereka (1999) proposes a theory of Multiple Spell Out, whereby whenever two complex objects are Merged one or the other must be "spelled out" first, frozen into a giant compound (which, with respect to

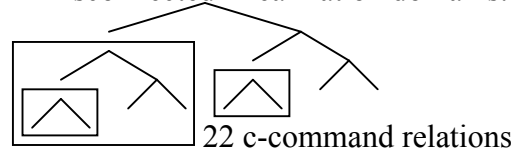
linearization, is treated as a simplex object, and so precedes its sister). The result is that the relations relevant to determining linear order in the Merge of two complex objects are one-sided. We might say that this splits the tree into multiple independent spines.

Assume that something like this is correct. The effect on c-command relations is dramatic: by “packing” the elements into a bushy tree, which will undergo linearization by multiple spell-out, far fewer c-command relations enter into the computation of linear order. The abstract example below makes this explicit; I indicate both the traditional tree structure for a bush with 8 terminals, and the structure as visible to linearization with Multiple Spell Out. The boxes in the tree on the right indicate complex left branches that are opaque to the structures that embed them. There are 34 c-command relations in the whole tree, but only 22 that are relevant to antisymmetric linearization.

(5) Connected tree structure:

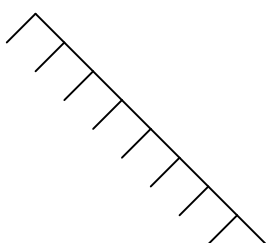


Disconnected linearization domains:



Compare this with a spine structure: instead of packaging the tree up into maximally mutually opaque substructures, the entire tree is visible, and a c-command relation exists between each pair of elements. Far more c-command relations exist in such a structure (for eight terminals, there are a total of 56 c-command relations in the spine).

(6)



If determining the linear order of the terminals in the tree involves computation of c-command relations, as in Kayne (1994), then (given some mechanism to make left branches opaque) the task is simpler for bushier trees. Economy of command eases the burden for the processes that “read” c-command relations, including linearization. In the next section, I argue that agreement presents a similar picture, in that what matters for agreement are the c-command relations that exist after syntactic movement has applied.

3.3.3 Agreement

Bobaljik (2008) argues that agreement (his concern is subject-verb agreement in Germanic) is a post-syntactic operation, sensitive to configurations after movement, independent of any additional syntactic licensing relation among agree-er and agreed-with (the target and controller of agreement). In particular, he defends this claim (his (3)), for instances of agreement with a single NP: “The controller of agreement on the finite verbal complex (Infl+V) is the *highest accessible* NP in the *domain* of Infl + V.” (Bobaljik 2008: 296, emphasis in original)

Accessibility is determined by m[orphological]-case, itself determined post-syntactically. Importantly, “an NP need bear no relation to a verb other than satisfying morphological accessibility and locality in order to trigger agreement on that verb. This contrasts with the proposal in Chomsky (2001) under which agreement is a reflection of core-licensing (feature-checking) relations in the syntax.” (Bobaljik 2008: 297)

If agreement is fed by movement, then movement can affect the structural complexity of the representations involved. In particular, agreement is computed over c-

command relations as they exist after movement. In general, the fewer and shorter the c-command relations involved, the simpler the computation. In the next section, I argue that similar conclusions hold for at least some instances of binding.

3.3.4 Binding

Turning next to binding, we can motivate the bias for balanced trees with respect to Condition C, especially. Condition C effects are sensitive to c-command relations of arbitrary length. Now, Condition C stands out in this regard; the conditions on anaphors and pronouns apply strictly within local domains (phases). So, when separated from its antecedent by two levels of embedding, a coreferring pronoun is grammatical (indicating the distance is beyond the reach of Condition B).

- (7) He_i suspected [_{CP} that they already knew [_{CP} that he_i /*John_i was a spy]].

These effects extend even into NP Islands (cf Ross 1967), in a way that long-distance wh-movement cannot.⁶ That is, movement of a wh-phrase is permitted to escape from an embedded CP, but not from an NP, the latter forming an island.

- (8) What did they believe [_{CP} that Sally had witnessed <what>]?

- (9) *What did they believe [_{NP} the claim [_{CP} that Sally had witnessed <what>]]?

Long-distance extraction itself involves dependencies over c-command pathways, unbounded in length but achieved through a series of short intermediate steps, and sensitive to islands. Condition C effects, preventing a full NP (John) from being c-

⁶ Note that wh-movement cannot access the downstairs position involved in the above example either, an instance of the so-called that-trace effect.

commanded by a co-referring pronoun at any remove, peer through even NP Island boundaries, as illustrated below:

- (10) He_i believed the claim that Sally had loved him_i/*John_i.

This is a relation of considerable scope. It is, moreover, of a peculiar nature, a “nowhere” condition on R-expression binding. Rather than positively establishing coreference (or not), it enforces a strict ban, a kind of anti-binding that is “active” throughout a sentence of multiple phases. But regardless of its peculiar “strength”, and whatever its ultimate source, it is clearly syntactic in substance. Specifically, the condition “sees” the syntactic relation of c-command. This can be brought out by minimally embedding the coreferring pronoun within another NP, from which position it does not c-command the R-expression. With this minimal adjustment, Condition C effects do not obtain; the full NP can appear downstairs:

- (11) [His_i mother] believed (the claim) that Sally had loved him_i/John_i.

All of this is familiar, and it is not my goal here to make any new contribution to the theory or description of binding. Rather, I mean to point out that the conditions of binding, as already revealed by decades of intensive research, are such that their computation is minimal when applied to bushier trees as opposed to spines. With respect to Condition C, in particular, c-command relations spanning the entire tree are “visible”. Within a bushier tree, the relevant relations are shorter and fewer in number; from the point of view of an arbitrary node in the tree, as much as possible of the rest of the tree is hidden, out of the range of c-command. Bushy structure presents a lesser overall burden

for search-like hierarchical processes sensitive to c-command relations, including the computation underlying Condition C.

I have little to say here about Conditions A and B. Certainly reconstruction effects indicate that some binding computations “see” lower copies.

- (12) Who_i did they say that Sally had thought <who_i> was already shaving himself_i?

In the example above, the anaphor is bound locally in its domain, apparently from a reconstructed position of the moving *wh*-phrase, indicated in angle brackets. In this case, the c-command relation that is relevant is one involving a lower copy of a moved element.⁷ On the other hand, examples like the following indicate that anaphoric binding can be fed by movement as well. That is, the anaphor below is bound only after movement of *who* into the matrix clause; it is not bound by this element in its base position in the embedded clause (again indicated with angle brackets).

- (13) Who_i seems to himself_i <who_i> to be the clear winner?

The facts are intricate, and I leave a more careful investigation of binding to future work. Condition C at least provides a clear case where economy of command ought to matter, in the sense that it appears to “read” c-command relations of extreme length. The burden of the computations implicit in (Condition C) binding effects is minimized in bushier trees (those with fewer and shorter c-command relations). I have not tried to establish the

⁷ That does not mean, however, that what is relevant is the base position, i.e. the tail of the chain. It is possible that the relevant binding takes place at the head of an intermediate chain at the phase level, after one or more steps of movement in an embedded CP or vP, but “before” movement carries the moving object to its final surface position. This matter of derivational timing of binding relations is an area of active research; see for instance Barss (2003) and references therein. Insofar as Binding Conditions A and B apply at the phase level, then what is seen is the structure after movement, and binding is established over the c-command relations in that structure.

same claim for the binding of pronouns and anaphors, noting the complexities involved. However, if these are computed at the phase level, then they ought to be sensitive to (intermediate) post-movement configurations, where movement (I will argue) produces a structure with a sparser scaffolding of c-command relations. In the next section, I construct a parallel argument for the assignment of nuclear stress (where dominance rather than c-command is the relevant relation).

3.3.5 *Nuclear Stress Rule*

We can illustrate how the concerns of economy of command arise in terms of phrasal stress, another canonical case of “cyclic interpretation” in the relevant sense. As I show below, the Nuclear Stress Rule of Chomsky & Halle (1968) is a simpler computation when applied to a Bush as compared to a Spine with an equal number of nodes. This is one example of how the structural preference I argue for arises naturally from the kinds of long-distance dependencies found in natural language.

Phrasal stress is a topic of perennial interest, and it is not my purpose to review the vast literature here (see Kratzer & Selkirk 2007 for a recent view). Instead, I focus on a familiar, if somewhat dated description of the phenomenon; the goal is simply to get a feeling for how the structural difference at issue is important for linguistic conditions.

The basic idea is that stress at the phrasal level closely tracks syntactic structure, with stress computed cyclically (Chomsky & Halle 1968, Bresnan 1971, 1972, Cinque 1993). This cyclic computation involves long-distance hierarchical relations of a sort (in this case, described in terms of dominance relations, rather than c-command relations).

As I will show, for a tree of a fixed number of nodes, computing the phrasal stress of a Bush structure involves fewer operations than computing the phrasal stress of a Spine with the same number of nodes. This is one example of what I claim to be a widespread pattern: the computations of natural language are governed by a natural principle of economy of command. The kinds of computations we find in natural language (in this case, the reading of hierarchy as phrasal stress contour) are such that the burden their computation induces is least for a maximally balanced, shallow tree.

Consider the details of phrasal stress assignment according to Chomsky & Halle's (1968) Nuclear Stress Rule. Items come with default 1 stress, and are cyclically "demoted" in an inside-out computation, from most to least-embedded. The simple derivation below illustrates. Here, the lexical items are entirely abstract; I indicate only their stress level. 1 is highest stress; then 2, 3, etc. Where demotion of stress occurs, I indicate the affected element in bold.

(14) [1 1]: **2** 1 Starting with two elements (initially 1s), one is demoted (to a 2).

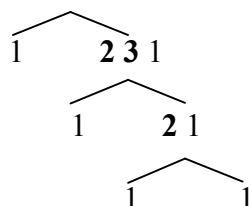
[1 [2 1]]: **2** **3** 1 Another element is added; two stress demotions occur.

[1 [2 3 1]]: **2** **3** **4** 1 Another element is added, and three demotions occur.

In the diagrams below, I explicitly count the number of "demotion"/stress adjustment steps. If stress assignment is operationalized as Chomsky & Halle envision it, this provides a measure of the complexity of the computation of phrasal stress; each demotion represents an operation of accessing a memory location storing the stress level on an individual item, and rewriting its contents. Each node is annotated with the stress contour as computed up to that point in the derivation, with stress levels adjusted at that level in

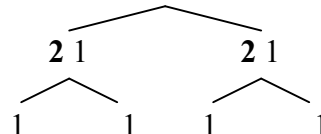
bold (e.g., each terminal is a 1; the first pairing produces **2** 1 as the stress on the first element is “demoted”, etc.).

(15) a. **2 3 4 1**



6 total stress adjustments

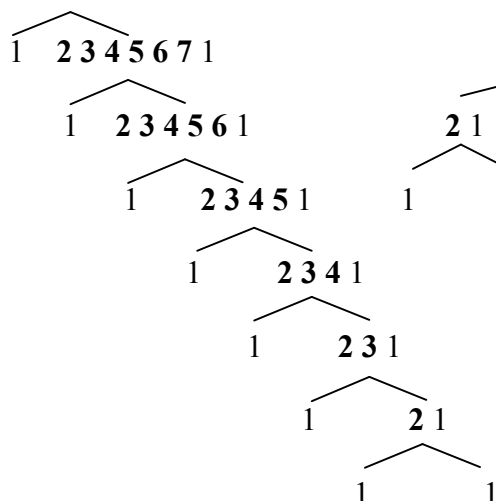
b. **3 2 3 1**



5 total stress adjustments

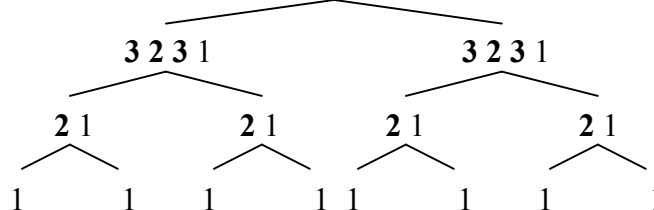
The difference is real, but hardly dramatic for such small pieces of structure. Let us examine how the totals diverge for larger trees:

(16) a. **2 3 4 5 6 7 8 1**



28 total stress adjustments

b. **4 3 4 2 4 3 4 1**



17 total stress adjustments

Here, the worst case involves more than 60% more operations (of over-writing the stress level on individual lexical items stored in memory). Thus, the application of the Nuclear Stress Rule involves less computational “action” for bushier trees than for more right-branching ones.^{8,9}

3.4. *Conclusions*

In this chapter, I have laid out the theoretical commitments that underpin the rest of this work. I discussed the cartographic project, whose vision of a richly articulated and nearly uniform syntactic structure for all languages I adopt. I discussed two kinds of variation in this structure (the extent of projection, and the variability of agreement and negation). The first kind of variation will be an important element in the next chapter, where I will argue that differences in movement within and across languages can in part be reduced to this variation, in conjunction with the view that movement is a mechanism to reduce the number of c-command relations in syntactic representations.

⁸ What about more modern metrical grid theories (Lieberman 1975, Halle & Vergnaud 1987)? In such cases a preference for bushier over spindlier trees is harder to motivate in terms of simplifying stress assignment, since the number of operations of “project a metrical head” is just equal to the number of non-terminal nodes, hence identical for any binary-branching arrangement of a fixed number of terminals. But notice that metrical grid theories do not directly provide output stress levels in the way that the NSR above does. That is, the proper stress level on an individual output item can simply be read directly off the representation produced by Chomsky & Halle’s NSR (e.g., main stress is a “1”, in any context). In metrical grid theories, one finds, locally, only information on relative prominence – grid marks projected on a level higher than immediate neighbors, or not. Some further computation is required to “translate” such a representation into fully-specified instructions for articulation. In particular, it appears, as an empirical matter, that a main stress is a main stress, regardless of the absolute size of the domain it occurs in (a 1 is a 1, so to speak), but that may be a stack of two metrical grid marks, or eight; local inspection of the metrical grid form yields only the local shape of the stress contour, which must be “normalized” to actual stress levels by a global computation. To put it simply, actual pronunciation requires absolute prominence, not just relative prominence. If that normalization procedure is cyclic in the desired sense, then the preference for bushier tree forms would again hold.

⁹ For the claim that phrasal stress reflects surface (i.e. post-movement) configurations—obviously a key ingredient in using considerations of the complexity of the computation of phrasal stress to motivate movement—see especially Cinque (1993).

I claimed that for many of the linguistic relations that depend on c-command, the c-command relations that matter are those that exist after movement, in cyclic interpretation at the interfaces. I discussed the copy theory of movement, and how it relates to present concerns (even noting that the general preference for the highest copy in the chain to be interpreted can be rationalized in terms of economy of command). I considered linearization, agreement, binding, and nuclear stress, arguing that these computations are fed by movement, and are simpler when applied to structures with fewer c-command relations (i.e., bushier trees).¹⁰

This conception of cyclic interpretation, wherein only the highest copy in the chain formed by movement is “visible” to c-command based interpretation, forms the basis for the next two chapters. In chapter 4, I explore how we can explain numerous empirical properties of movement in terms of an understanding of it as a mechanism to reduce c-command relations. In chapter 5, I further develop this view of movement into an analysis of word order in DPs.

¹⁰ I have limited the discussion to these four kinds of c-command or dominance-based computations. However, it is clear that the argument could be extended to include at least some cases of scope, as well as perhaps other phenomena (such as Case). Scope is another linguistic relation that tracks over c-command relations; in particular, element α takes scope over another element β if α c-commands β (May 1985). This suggests that interpreting scope relations involves “reading” c-command relations; again, this is arguably a simpler process in a tree with fewer and shorter c-command relations. The facts are complicated by the possibility of reconstruction (as with binding, see above), and cross-linguistic differences (e.g., some languages like Hungarian are described as “surface scope” languages, while English and others have scope relations involving covert LF movement and/or reconstruction). I leave consideration of the complexities of scope to future work.

CHAPTER 4: MOVEMENT AS TREE-BALANCING

4.0 Introduction

One of the enduring puzzles of human language is the ubiquitous property of syntactic movement. This pervasive feature of language seems particularly strange in light of the hypothesis that the computation of syntactic form is perfect or optimal in some sense (the Minimalist thesis of Chomsky 1995b *et seq*). On the face of it, movement is an “extra” operation, over and above what is required to create the basic phrase structure in which it is found. The mystery only deepens when we consider the structure-preserving nature of movement (Emonds 1970): movement does not really extend the phrase structural possibilities at all.

Moreover, differences in patterns of displacement are one of the central loci of language variation. Indeed, for Kayne (1994), and subsequent work in cartography (Cinque 1999, 2002, Rizzi 2004, Belletti 2004, among many others), movement is the sole source of cross-linguistic variation in word order (up to (non-) pronunciation). And it is striking that all languages apparently have displacement in some form or another: the phenomenon appears in widely varying forms, but is nevertheless universal. We would like to know why.

Chomsky (2000) provides a reason to expect movement, in the form of Internal Merge. As he argues, the simplest assumptions about Merge make Internal Merge a possibility (with further details, like selecting which copy in a chain to interpret, left to other principles). But, once enabled by Internal Merge, where and why should movement actually occur?

In this chapter, I propose that syntactic movement is a mechanism for reducing the number of c-command and dominance relations in interface forms. In other words, movement exists to balance trees. This is orthogonal to familiar Minimalist explanations of movement in terms of licensing and feature-checking (discussed only briefly below), but nevertheless the intuition fits squarely within the Minimalist paradigm. Here, movement is not seen as a “Last Resort” (cf Chomsky 1986); rather, it is directly part of an optimal structural solution to conditions of efficient computation.

The picture that emerges from these concerns alone (i.e., minimizing c-command and dominance) aligns rather well with current descriptions of natural language displacement. Optimistically, these concerns alone might go a long way towards achieving an empirically accurate description of where movement can and cannot occur.

Note, however, that the predictions made here are highly abstract, and we must be extremely cautious in thinking about how the concerns here could play out in the cognitive structures implicated in the multifactorial and profoundly complicated phenomena of real human language. Finding a method to evaluate the match or mismatch between a model incorporating the present concerns and empirical observables is the most meaningful and serious evaluation of this work; for one step in this direction, see Chapter 5.¹ Syntactic movement is a dauntingly vast empirical topic, and I cannot hope to demonstrate convincingly that all movement phenomena, in all languages, can definitively be accounted for in these terms. The goal of this chapter is to establish a

¹ The very austerity of the account is its primary weakness. That is, the predictions are framed in terms of undecorated tree structure; there is no direct role for the identity of individual heads. Then the task is to independently identify the underlying hierarchy of heads, as Cinque (1999) has done for the adverb space, and see if this hierarchy, realized as a syntactic structure, is such as to motivate the movements that are cross-linguistically instantiated for the elements of that hierarchy. See Chapter 5 for extensive discussion.

modicum of plausibility for the idea, in a handful of core cases. The next chapter develops a detailed application of this analysis to word order within nominals, with an eye to providing more rigorous testing of the analysis.

This chapter is organized as follows: In section 4.1, I outline some of the previous attempts at explaining movement. In section 4.2, I look at the question of the effect of movement on syntactic structures. In section 4.3, I state the Fundamental Movement Condition, an algebraic expression of the conditions under which movement reduces the number of c-command relations in a tree. The main thesis to be explored in this chapter (and the next) is that this condition governs syntactic movement: movement is allowed only if it reduces the number of c-command relations in the tree.

In sections 4.4 and 4.5, I point out two core predictions that follow from the Fundamental Movement Condition: a form of Anti-locality, and a size threshold effect. To put it simply, if movement is “for” tree-balancing, then it cannot be too local, nor it can it move too little material. This derives an empirically well-motivated ban on movement from complement to specifier of the same phrase, and a more controversial but not unprecedented ban on head movement. In either case, the restrictions are a matter of geometry, a condition on the configuration of nodes rather than their contents. Especially with respect to size threshold effects, I explore the consequences for understanding movement phenomena that have previously been treated in terms of requirements on interpretation (e.g., Object Shift), suggesting that interpretation is a byproduct of narrowly syntactic principles.

I turn in section 4.6 to a prediction about island effects. We predict that particularly well-balanced portions of the tree should act as islands. In particular, in the case where a mother node dominates two equal-sized daughter nodes (a point of symmetry), we predict that the daughter nodes cannot move individually. I argue that this provides an explanation for the Coordinate Structure Constraint of Ross (1967); however, it conflicts with some recent treatments of small clauses, where it is argued that movement is obligatory to destroy a point of symmetry (see especially Moro 2000). I discuss the issues that arise, and a possible solution in terms of predicate phrases (PredPs).

In section 4.7, I turn to an extended discussion of so-called roll-up movement, focusing on the Malagasy language. As I discuss there, this pattern of extremely local iterated movement, so problematic for interpretation-based accounts of movement, falls out readily from the hypothesis that movement is a matter of balancing trees.

In section 4.8, I consider a handful of types of movement predicted by this account, and how they behave under iteration. These include an analogue of roll-up movement, which I show to be subject to a form of positive feedback (driving each iteration more strongly than the last); an extremely local Spec-to-Spec movement, subject to negative feedback; and (relatively) long-distance A-bar-type movement to the edge of phases, showing that it achieves a stable equilibrium (each step of movement is driven as strongly as the last). I consider also feeding/bleeding interactions between the various types of movement. Section 4.9 concludes the chapter.

4.1 *Previous treatment of syntactic movement*

A step in a derivation is legitimate only if it is necessary for convergence.

(Chomsky 1995b: 201).

Chomsky (1986) proposed to view movement as a “Last Resort”, forbidden unless necessary, an idea that remains enormously influential. That is one straightforward way of addressing the seemingly wasteful nature of movement: whatever its cost, movement is expected if the derivation would fail to converge without it. Pursuing this intuition, much recent work adopts the hypothesis that movement is a matter of *licensing* (feature-checking). The empirical burden for such a claim is then to identify the lexical features involved in driving movement. To a certain extent, this has proven a fruitful line of attack, especially with respect to movement linked to Case and/or agreement; the working hypothesis in that project is that morphology (possibly abstract, i.e. null) is to blame for movement. This fits in a pleasingly natural way with Borer’s (1984) suggestion that syntactic variation among languages reduces to variation in the properties of individual lexical items.

A related hypothesis is that movement exists to enhance the expressive power of natural language. For instance, in recent work Chomsky has referred to the role of a “duality of semantics” in (certain instances of) movement (see for example Chomsky 2008: 140-141), the idea being that movement creates otherwise-unavailable interpretations (topic/focus, etc.; basically, discourse-informational effects, while core licensing relations – the domain of theta theory – is the province of External Merge). In

this case, movement is still seen as a matter of licensing, broadly speaking – in this case, of interpretations, not of lexical features.²

At the same time, a body of research following Kayne (1994) posits a great deal more movement in natural language than previously supposed. One of the strong predictions of Kayne's antisymmetry is rigid specifier-head-complement order. As a consequence, finding, say, complement-head order indicates that some movement has applied to disrupt the expected head-complement order. The so-called cartographic project (see Cinque 1994, Rizzi 1997, Cinque 1999, and much subsequent work) takes this one step further, supposing that the inventory and hierarchy of lexical items is also universal; in the limit, every category is present in every sentence of every language. Taken together with some version of antisymmetry, this leads to the identification of even more movements in deriving the surface orders of natural languages.

However, this radical proliferation of inferred movements has created considerable problems for licensing accounts of movement (of either stripe, lexical or interpretational). That is, it is generally the case that overt morphological differences between languages do not match, in number or richness, the proposed differences in movement patterns. This is especially so for analyses positing so-called roll-up or snowballing movement, where a large number of quite-local movements are identified, often only on the basis of rather subtle facts about, say, adverb ordering. Thus, if lexical feature-checking is to blame, the relevant features are at best obscure; moreover, the feature-based point of view offers

² Note, though, that this approach goes hand in hand with Chomsky's (2001) proposal that movement is "Internal Merge", freely available in a system based on Merge. This represents almost a return to the "Move alpha" conception of movement of Government and Binding theory; movement is not an imperfection used only as a last resort; instead, "its absence would be an imperfection." (Chomsky 2001: 8)

little insight into the stringing together of multiple movements that is the hallmark of roll-up movement.³ On the other hand, the view that movement exists to license interpretations would seem to face even greater difficulties in accounting for the cross-linguistic differences in movement claimed within this body of work. To put it simply, a VOS sentence (say, in Malagasy) means pretty much what its SVO equivalent (say, in English) means.⁴ What interpretational drive, then, can there be for the complicated series of movements deriving the Malagasy surface form? Moreover, pointing to interpretation as the driving force in movement would, in light of the massive variation in movement patterns, indicate massive variation in interpretation as well. That violates the intuition of semantic uniformity: “In the absence of compelling evidence to the contrary, assume languages to be uniform, with variety restricted to easily detectable properties of utterances.” (Chomsky 2001: 2) Particularly with respect to logical form (cf May 1985), it has often been claimed that there is very little variation among languages; semantic properties then seem an unlikely source of the apparently radical differences in movement among languages.

While the checking of features on lexical items provides a possible mechanism for driving movement, it would be rather unsatisfying if explanation went no deeper than

³ One exception in this regard is the Final-Over-Final Constraint (FOFC) of Biberauer, Holmberg, & Roberts (2007). They claim that head-finality is ‘inherited’ from a phase head via agreement, and use that to account for certain ordering facts. See that work for details. Insofar as it is correct, they provide a compelling explanation for it. However, their account appears to be incompatible with the DP ordering facts examined in Chapter 5 of the present work.

⁴ This is not quite true. For example, the external argument in Malagasy has rather different properties than English subjects. In the literature on this language (and related Philippine-type Austronesian languages), this position is described as the “trigger” rather than the subject, and appears to have a different information-structural status (Pearson 2007). While that fact might be taken as encouraging, the empirical challenge in associating differences in movement with differences in interpretation remains to motivate each step of movement – particularly difficult for the series of roll-up movements identified within the vP. See Rackowski (1998), Rackowski & Travis (2000) for discussion.

this, because the relevant movement-driving features are apparently subject to significant cross-linguistic variation. If explanation stops with features as primitives, we are left with a picture in which patterns of displacement are brute accidents of the lexicon.

All of this is not to deny that lexical variation is real, and perhaps an important contributor to cross-linguistic variation in syntactic properties. But perhaps we can explain movement-inducing lexical features themselves, as organized (synchronically and diachronically) by computational biases favoring certain kinds of displacement patterns. That could offer principled explanation for features themselves, and predictions about their distribution and stability.

In what follows, I propose a very different way of viewing movement. Rather than viewing movement as strictly necessary (as it is in the standard feature-checking view), I propose that the function of movement is to reduce the number of c-command and dominance relations in the syntactic representation. Thus, movement, despite the inherent cost of an additional Merge operation, nevertheless serves to simplify the syntactic computation. This makes movement a decidedly natural thing to find in human language. Moreover, since what is at stake is nothing more than tree geometry, very precise predictions can be derived and tested against known facts.

The view developed in this chapter bears rather directly on the autonomy of the syntax, and the relationship between syntactic operations and semantic interpretation. The present work fits well with the perspective of Hinzen (2011), who questions the idea that “what the syntax qua computational system of language produces has to match what is

independently there, in the hypothesized pre-linguistic conceptual – intentional (CI) systems. The semantics, in short, requires, and the syntax satisfies.” (Hinzen 2011: 423)

In light of the program to reduce what is attributed to language-specific capacities, and promote explanations in terms of domain-general principles (the third factor), Hinzen suggests the following alternative view of the relationship between syntax and semantics:

“We might think of this program as beginning from minimal codes (structures generated by Merge, say) that are as such entirely unspecified for the output achieved by them. The most minimal principles prove rich enough. This program is not consistent with the idea of setting an output that is to be matched. Indeed, as I will here contend, there is no independent task or output to be matched; the task accomplished by grammar arises with the grammar itself: syntax formats human thought rather than expressing it [...] Syntax, therefore, is not task-specific either, as there was no task before syntax was there. Syntax provides underspecified codes.” (Hinzen 2011: 423-424)

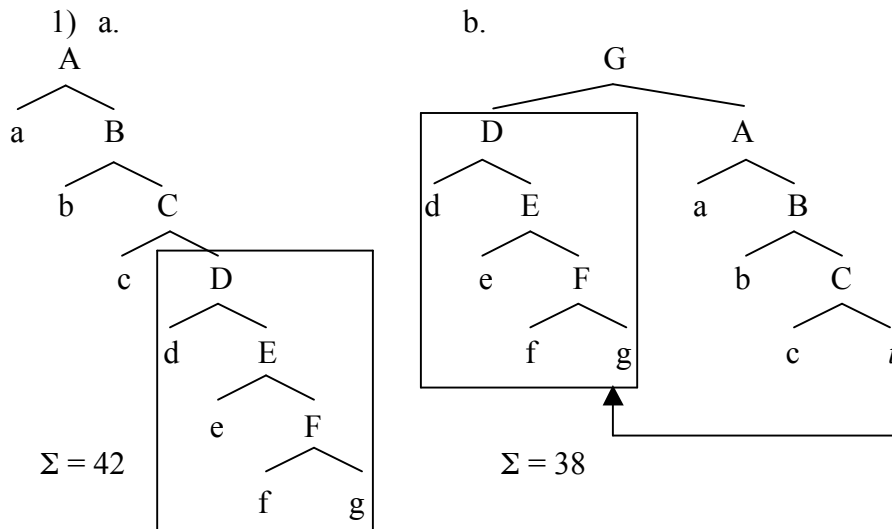
If movement is, at its root, a matter of tree-balancing, then it is not driven, ultimately, by the need to create interpretations, nor to check formal features. This is not to deny that movement has semantic consequences; likewise, there is reason to temper this strong view of features somewhat. I will suggest below that features offer a simplifying heuristic to implement movement “blindly”, but that their distribution is itself determined by whether or not the movements they trigger help balance the tree. In the next section, I illustrate how syntactic movement might serve as a tree-balancing mechanism.

4.2 *What does movement do?*

The point of departure for this chapter is the claim that c-command and dominance relations “count” primarily in interface representations. That is, the relevant c-command and dominance-based phenomena that implicate costly computation are sensitive to the

configurations produced after movement. If so, then movement can help, by easing the burden of computing over interface forms. In effect, movement has the effect of shrinking the search space for the (c-command/dominance-based) mapping procedures that apply post-movement.

Insofar as the computations that “read” c-command relations see post-movement configurations, movement that transforms a configuration with many c-command and dominance relations into a configuration with fewer such relations eases the burden for these interface mapping computations. So long as only the displaced surface position is detected by these processes, movements may serve to transform a costly configuration into a more optimal arrangement, as in the transformation of (1a) into (1b) by movement (leaving a ‘trace’, marked with *t*).⁵



⁵ For the calculations here, I assume that ‘traces’ are effectively dummy terminals, participating normally in c-command relations. If traces are ‘invisible’ to c-command or dominance, then movement is even more easily motivated, and slightly different conditions would apply.

As indicated above, in the pre-movement configuration the number of c-command relations sums to 42; after movement, the sum of c-command relations is 38. It is the thesis of this chapter that syntactic movement can be explained as a mechanism to achieve this kind of reduction.

In the remainder of this chapter, I explore the mathematical fundamentals of a theory of syntactic movement as a mechanism for tree-balancing. The principle theme is that structure determines movement; the geometry of a syntactic tree dictates what movements are possible or impossible, and which are “better” than others.

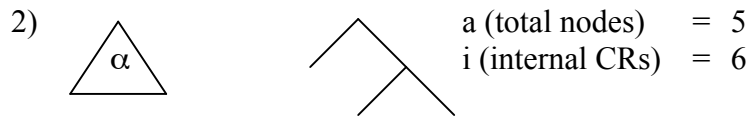
4.3 The Fundamental Movement Condition

Below, I derive the conditions under which movement of a single category α balances the tree, i.e. reduces the number of c-command and dominance relations present. For the purposes of these calculations, I assume that lower copies collapse into a single terminal position, effectively a “trace”, as in earlier formulations of generative grammar. Note that this is equivalent to the structure proposed in the hybrid theory of dependencies of Koster (2007).⁶

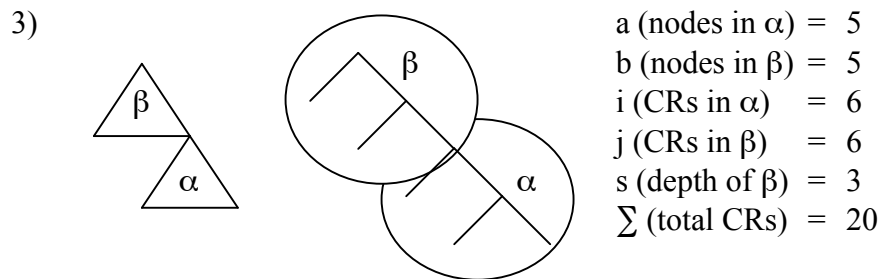
In the diagrams below, each category is labeled with a Greek letter, and has a pair of numbers accompanying it (e.g., (a, i) for α). The first number is the number of nodes in that category, while the second is the number of c-command (or containment) relations

⁶ Koster argues against the internal Merge conception of movement, in part on the basis of Emonds’ (1970) notion of structure preservation. Koster proposes instead to fold traces into a general theory of “‘empty’ elements, which I take as incomplete lexical elements, with categorial features, but without the full range of identifying semantic and phonetic features.”(Koster 2007:193) In other words, Koster assumes the bottom of a chain is occupied by a single terminal, consistent with (8).

internal to that category. The category β is also annotated with the number s , indicating its depth (the number of nodes along its spine, from its root to the root node of α , inclusive). C-command (equivalently, dominance) totals for each configuration can be expressed in terms of these variables, yielding an inequality that expresses the minimal structural condition for displacement to reduce the total number of c-command (equivalently, dominance) relations present in the tree.⁷



An arbitrary category α contains a nodes and i internal CRs. Supposing that α happened to represent the tree to its right, here $a = 5$ and $i = 6$. In some cases we will be concerned with the spinal depth of categories; for α above this is 3.



I will depict hierarchies of partitions as above. Note that, by convention, the partitions overlap (α and β above share a node). We can readily write an expression for the sum of

⁷ The notational conventions used here are as follows:

CRs c-command (equivalently, irreflexive containment) relations

$\alpha, \beta, \gamma, \dots$ label partitions of the tree (i.e. subtrees).

a, b, c, \dots are the total number of nodes in these tree partitions.

i, j, k, \dots are the total number of CRs (c-command, or equivalently containment, relations) internal to the partitions.

s, t, u, \dots are the spinal depths of non-bottom partitions, i.e. the number of nodes within a partition from its root to the root of the lower partition, including the shared node itself in this count.

$\Sigma_{1,2,3,\dots}$ are the sum of CRs (c-command or containment relations) in a composite figure, expressed in terms of the a -, i -, and s -series variables above.

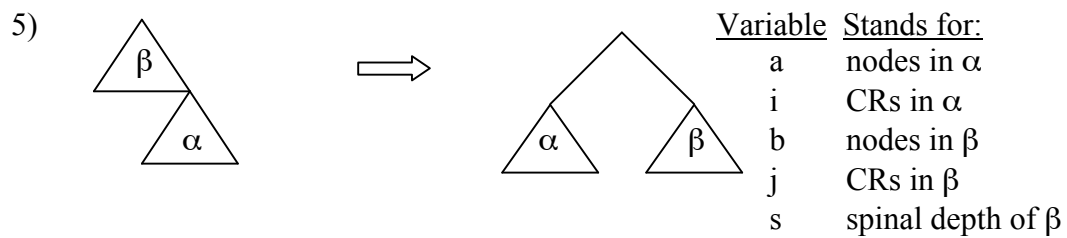
CRs in the whole figure as a function of the properties of the partitions α and β ; in the example under consideration this turns out to be:

$$4) \quad \Sigma = (i+j) + (s-1)(a-1) \quad \text{Where } i, j, \text{ are the CRs internal to } \alpha \text{ and } \beta; a \text{ is the number of nodes in } \alpha; \text{ and } s \text{ is the spinal depth of } \beta.$$

So, in (3), suppose α and β are as in the example to the right. Here $i=j=6$, $a=5$, and $s=3$, so the sum is $(6+6) + (3-1)(5-1) = 20$.

We will be concerned, in what follows, with comparing the total number of c-command (equivalently, dominance) relations in pre-movement configurations to the total number of such relations in post-movement configurations. Below, I give a general formula for the total CRs after movement of arbitrary syntactic object α to the edge of β .

The following example illustrates my assumptions about movement. I take movement to leave a trace (in interface forms);⁸ equivalently, base-generated “displaced” objects are associated with an underspecified terminal in the position of interpretation, a representational version of trace theory (cf. Koster 2007).⁹



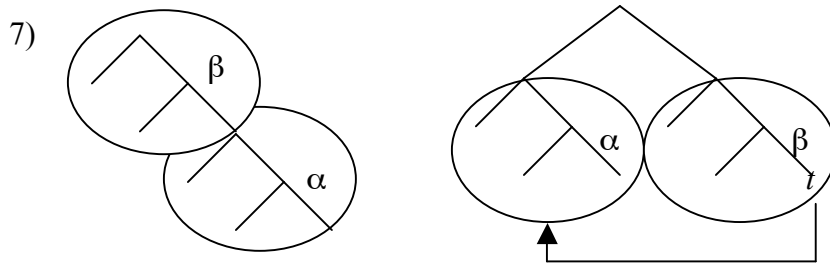
⁸ To repeat, this is not to deny the conception of movement as Internal Merge (the copy theory of movement), within narrow syntax. Here, we are considering the configurations from the point of view of interface processes pronouncing/interpreting only the highest copy in a chain, the most typical case (see the previous section).

⁹ These two treatments (of movement leaving traces, vs. a dummy terminal merged in the base position, with the “displaced” category only ever in its surface position) could differ in their predictions with respect to successive cyclic movement: in intermediate positions, we expect to find traces, but perhaps not Koster’s place-holders (with consequences, for, say, binding phenomena), a matter I leave aside here.

Here again, we can write a general expression for the sum of CRs in the post-movement configuration, in terms of the structural properties of the partitions. The correct form turns out to be:

$$6) \quad \Sigma = (i+j) + a + b$$

The example below illustrates; here $i=j=6$; $a=b=5$, so the sum is $(6+6)+5+5 = 22$.



In this tree, movement has resulted in a structure with more, not fewer, CRs.

For movement to be motivated by tree-balancing, the post-movement CR sum Σ_2 must be less than the pre-movement CR sum Σ_1 . This is the sole condition governing movement, on this account.

8) Fundamental Movement Condition (FMC), preliminary version:

$$\text{Move } \alpha \text{ only if } \Sigma_1 > \Sigma_2$$

Substituting into (11) the expressions derived above for Σ_1 and Σ_2 , we obtain (after a bit of algebra), this more formal version:

9) Fundamental Movement Condition (FMC):

$$\text{Move } \alpha \text{ only if } (a-1)(s-2) > b+1$$

Very roughly, the FMC amounts to saying that the size of the moved category (a), times the distance it moves (s), must exceed the size of the non-moving part of the tree (b). This constitutes an instantaneous evaluation at the derivational stage at which

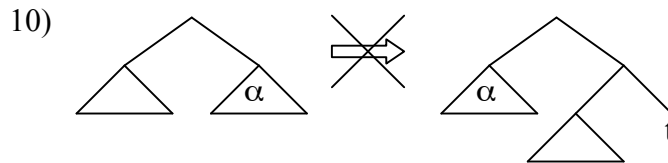
movement occurs; it might be more appropriate to consider, say, final results of movement in a completed phase as the standard for comparison. But the FMC as formulated above is the simplest and strongest claim; I choose to pursue this version, expecting that it will be easiest to falsify.

It is important to note that the FMC is a *minimal* condition indicating when movement might be of benefit (in the sense of creating a post-movement configuration with fewer “vertical” – c-command and dominance – relations than the pre-movement configuration). In other words, only if the FMC is satisfied could movement possibly be of benefit; in practice, of course, that may not be *sufficient* to motivate movement, with presumed additional “costs” (within narrow syntax, under a copy theory of movement, and at the level of performance). This gives us a starting point for investigating whether tree-balancing is the reason for movement: the more we find that attested movements in natural language satisfy the FMC, the more confidence we may have that this is the right kind of explanation. But much remains to be worked out for such an account.

In the next two sections, I point out two basic predictions that follow from the Fundamental Movement Condition derived above. These are a form of anti-locality, discussed in 4.4, and a size threshold effect, discussed in 4.5. The overarching theme is that structure determines movement; this theme runs throughout the chapter, but finds its simplest and clearest form here.

4.4 *Antilocality*

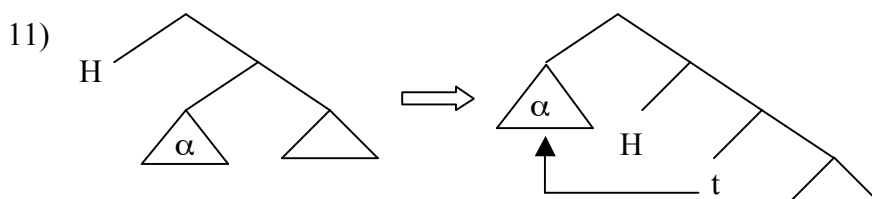
We can directly derive a form of antilocality from the FMC. Consider the possible values for s (recall that this represents the depth of the moving category in the tree). We know that the right hand side of the inequality in (9) must be strictly positive, because b is a positive integer. If $s = 1$, we would have a zero value on the left hand side of (9), an impossibility. Thus, s must be 2 or greater, ruling out movement of α in a configuration like (10), where it is immediately dominated by the root.



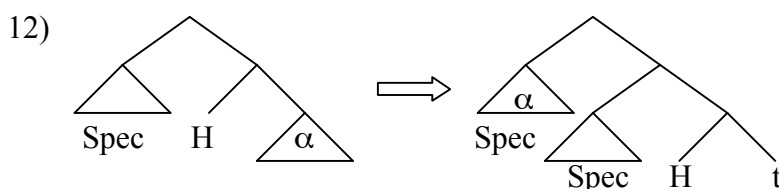
This effectively rules out movement of a category from the complement position to the (first) specifier position of the same head. A ban on exactly this kind of movement has been proposed by a number of authors (Bošković 1997, Abels 2003, Aboh 2004, Grohmann 2000, Kayne 2005, Boeckx 2008), on other grounds.¹⁰ Here, we derive this restriction directly from the structural conditions to which movement is, by hypothesis, a response. No conceptual argument is being invoked here (such as the idea that such movement could not serve any feature-checking purpose, as the head-complement relation is already the maximally local feature-checking configuration). Instead, this prediction falls out directly from the FMC.

¹⁰ The consensus of those works is that complements cannot move to another position within the projection of the head that selects them. This is somewhat different in content from the present claim, for which projection properties are not considered. For example, the present account would permit movement from complement to second specifier, as noted.

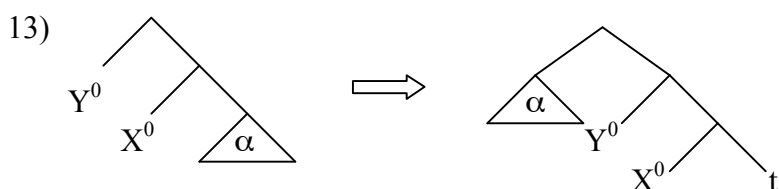
How local can movement be, on this conception? First let us consider a rather general case, with structural properties of the sister of α left open for now. Then $s = 2$ is sufficient for movement to satisfy the FMC:



If independent conditions on phrase structure permit multiple specifiers (as in the theory of Bare Phrase Structure (Chomsky 1995a), but contrary to Kayne (1994)), this means that movement from complement of a head to its second (or beyond) specifier ought to be permitted:



On the other hand, if multiple specifiers or multiple phrasal adjunction is ruled out for phrase-structural reasons (as in most versions of Antisymmetry, cf Kayne 1994), then the most local kind of movement possible would be from the complement of a phrase XP to the specifier of the immediately dominating phrase YP:



If a (the number of nodes in α) is large enough, the present account leads us to expect movement to sometimes be as local as this.

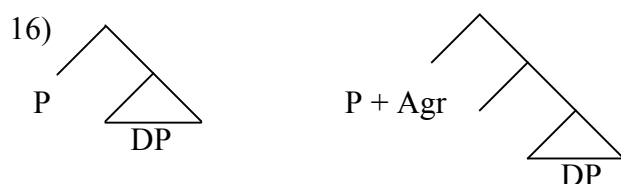
Antilocality as described above can offer insight into the correlation between agreement on P elements and whether those Ps are prepositions or postpositions. Kayne's Antisymmetry forces us to the conclusion that postpositions arise through movement of their nominal complement to the left. It has often been noted that there is a positive correlation between a P element displaying agreement with the nominal, and moving it to the left. Kayne (1994: 49) observes that postpositional languages may have agreement on the P, while prepositional languages never do. Indeed, in Hungarian, a language with both prepositions and postpositions, only non-agreeing Ps are prepositions (Marác 1989: 326).

- 14) a. én-möggött-em
 I-behind-POSS.I. SG
 b. *möggött-em én
 behind-POSS.I. SG I
 'behind me'

- 15) a. *a hídon át
 the bridge.SUP over
 b. át a hídon
 over the bridge.SUP
 'over the bridge'

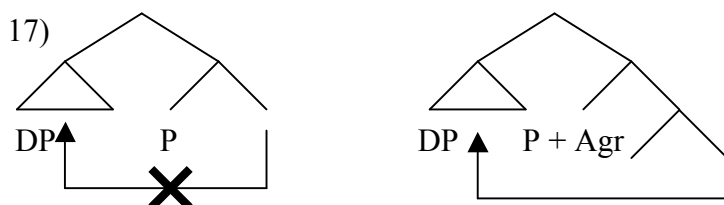
(Hornstein et al 2005: 125-126, citing data provided by Anikó Lipták)

Let us suppose that the presence of agreement indicates the presence of an additional syntactic position. Then PPs may have either of the following structures:



The movement facts fall out from tree-balancing in just the right way. The DP complement of a bare P cannot move, creating a prepositional structure. On the other

hand, with an additional layer of structure hosting agreement, movement of the DP complement becomes possible.



Thus, if we interpret the presence of agreement on P elements to indicate the projection of additional structure (say, an Agr head not present in the context of non-agreeing P), we predict, correctly, that agreeing P can trigger movement of its DP complement while non-agreeing P cannot.

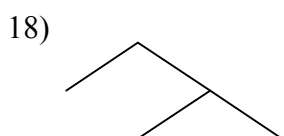
In the next section, I take up another straightforward prediction, of a size threshold effect on potential moving categories.

4.5 *Size threshold*

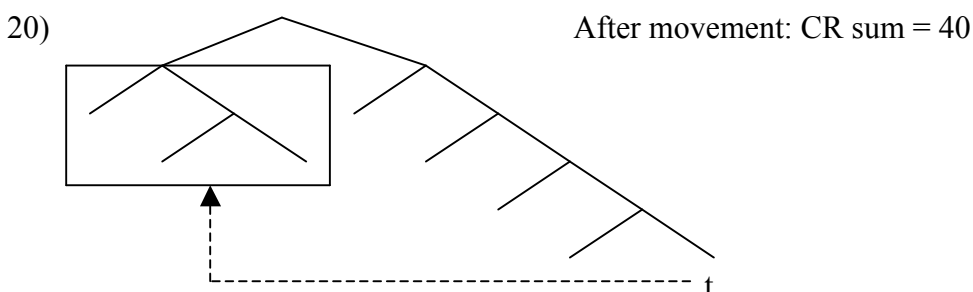
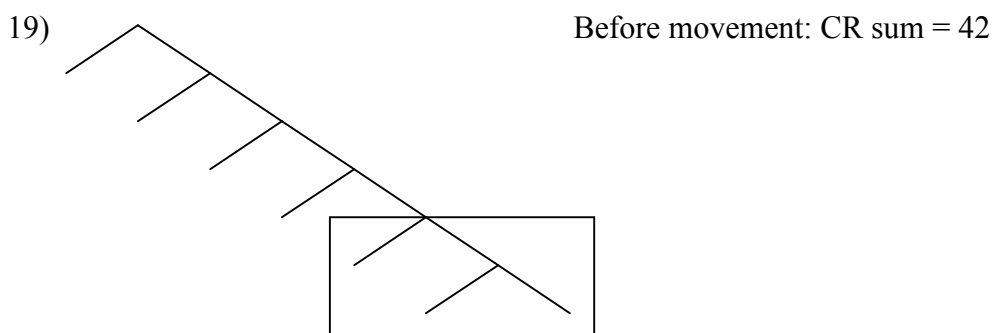
Another prediction which falls out directly from the FMC is that head movement as such—understood as moving a single terminal node—should never occur.¹¹ It turns out that $a = 5$ (3 terminals) is the minimal condition for movement to satisfy (9). This might seem problematic, in light of a rich tradition of head-movement analyses. However, ‘snowballing’ analyses (see below) can in many cases derive the same morpheme orders with XP movement alone (see, for instance, Mahajan 2000, Koopman and Szabolcsi 2000, and much subsequent work); this work endorses those analyses.

¹¹ Note that this does not rule out movement of *complexes* of heads, though we can exclude movement of an object consisting just of a pair of heads, for example the pairing of an acategorial lexical root and a category-determining functional head, as proposed by Marantz (1997): [f √]. In this case, $a = 3$, and we have $2(s-1)$ on the left hand side of the FMC. However, the right hand side is at least $2s$.

Five nodes (three terminals), as in (18) below, is the smallest piece of structure whose movement can result in a reduction of the number of c-command and dominance relations. This is an absolute size threshold; regardless of other details of the tree, nothing smaller than this should ever move, if movement obeys the FMC.¹²



The following diagrams illustrate a minimal context in which movement of (18) achieves such a reduction.



¹² As Massimo Piatelli-Palmarini points out (p.c.), there is a pleasing convergence between this prediction and the topic of X-bar phrase structure, considered elsewhere in this dissertation. The minimal piece of movable structure is a treelet matching the X-bar schema of specifier, head, and complement. One wonders whether phrases, which for mysterious reasons seem to have a privileged status with respect to various syntactic processes, might emerge from structural divisions made by tree-balancing movement. I leave this matter for future work.

As should be clear from (9), holding everything else constant, as α (the number of nodes in the moving category) is increased, the structural improvement also increases. Thus, one of the immediate implications of the explanation for displacement offered here is that there should be size threshold effects. That is, holding the rest of the details constant but varying the make-up of a category that is a candidate for movement (α in (5)), I predict that categories exceeding a certain size threshold will move, while smaller categories will stay in place.

Consider, in this light, the case of Object Shift in the Germanic languages. The basic observation is that definite/specific DPs must escape the VP, while indefinites/non-specifics stay low. Since Diesing (1992), the account of this fact has been that the latter type of nominal must remain in the scope of existential closure. However, it seems to be the case that exactly those categories which move have one or more layers of additional functional super-structure, such that definites are effectively ‘larger’ than indefinites.¹³ If so, differences in their movement may be a matter of a size threshold effect.

This is exactly the kind of threshold effect predicted: holding the rest of the structure constant, we find that a small category stays in situ, while a slightly larger category in the same configuration undergoes movement. If this is on the right track, it suggests a reversal of the direction of explanation. Here, it is not that movement serves the needs of interpretation; rather, movement is a blindly structural effect, though with

¹³ However, coordinated DPs—‘twice as big’ as a standard DP, one would think—resist Object Shift (Thráinsson 2001). On the face of it, this is exactly counter to what is expected. One way to proceed here is to appeal to the internal feature structure of lexical items as effectively syntactic structure ‘counting’ for c-complexity. Then conjoined DPs would count as ‘small’ if they have less feature structure than their individual conjuncts; something like this seems to be true. That is, the features of a conjunction are typically less marked than either individual conjunct; following Harley and Ritter (2002), we may take this to mean that the relevant feature geometry is truncated.

interpretational consequences. This would then be an example of Uriagereka's idea that it is "as if syntax carved the path that interpretation must blindly follow." (2002:64)

However, matters are complicated by a closer look at the facts: in (most) Scandinavian languages only pronouns obligatorily undergo Object Shift (OS), while OS of full DPs is either optional (as in Icelandic) or ruled out completely (as in the Mainland Scandinavian dialects) (Thr  insson 2001). If what is at stake is the application (or not) of a single movement of a DP, that would seem to contradict the predictions here: the DP, as the larger category, ought to be more strongly driven to move.¹⁴

I can think of several responses to these facts.¹⁵ But one clear way forward is to develop a more nuanced view of pronominal structure. While pronouns may be phonologically 'small', they tend to be maximally rich in phi-features, which are at the center of the agreement system, a prototypical (but certainly not the only) linguistic computation sensitive to c-command. For example, in English neuter gender appears only in the pronominal system (*it*), as does overt case marking (*he/him*). In terms of the c-command computations for agreement, then, regardless of reduced phonological size, pronouns are likely to be prominent indeed. This might solve the puzzle of the coordinated DPs as well; relative to phi-features, coordinated DPs are generally less

¹⁴ Thanks to Andrew Carnie for pointing this issue out to me.

¹⁵ For example, one possibility is that the rather simplistic analysis just sketched is incorrect, and in fact apparent OS arises through remnant VP movement. Then the larger DPs might be expected to preferentially raise to some intermediate specifier position prior to remnant movement, ultimately stranding them low (while pronouns, too small to escape on their own from the VP earlier, get carried along for the ride with the remnant). Such an analysis, of course, aligns very well with what has come to be known as "Holmberg's generalization", that OS is contingent on V movement (Holmberg 1986). To put it another way, OS does not disrupt VO linear order, which is at least consistent with the sort of remnant movement analysis under discussion (but see Pesetsky & Fox 2005 for a very different approach to such facts). Note, though, that Scrambling – similar in other ways to OS – is not subject to any such restriction (see Thr  insson 2001 for discussion).

specified than either of their conjuncts (e.g., for mismatching gender features, the default gender is chosen; see fn. 13).

Consider in this light the agreement properties in Irish, e.g. in verbs and prepositions, as analyzed by McCloskey & Hale (1984): they argue forcefully that null *pro* controls agreement. But the language evinces no agreement with overt DPs.¹⁶ On the present conception, despite its minimal overt realization, silent ‘little *pro*’ induces more real burden for c-command-based agreement computations than large overt DPs in this language. See also the extensive discussion in Cardinaletti & Starke (1999), who argue that little *pro* has all the syntactic and semantic properties of other pronouns.

Another apparent counterexample to the predictions made here are clitics, which are plausibly “small” categories (smaller than other pronouns, and full DPs) yet typically end up farther to the left in the surface structure. In some Scandinavian languages, pronouns undergo Object Shift while full DPs do not. On the other hand, exceptionally large DPs undergo Heavy NP shift, ending up at the right edge of the noun phrase. On the face of it, all of these phenomena go in the wrong direction with respect to the size threshold prediction. That is, all else equal, we predict that a large category will move, ending up on the left, while a small category will stay in place, ending up on the right.

The problem is that we may be misanalyzing the movements involved. If we insist that what is at stake is a single movement of the nominal, then the counterexample stands. However, it may well be that the apparent movements are not the real movements, which might, say, be rearranging the larger verbal structure around the nominals in a way

¹⁶ This is not quite true; a few dialects allow an overt 3rd person plural subject to co-occur with agreement. (McCloskey & Hale 1984)

consistent with the present predictions. Well-known analyses of these phenomena assume exactly the kind of movements required. Leftward movement analyses of Heavy NP Shift include Larson's (1988, 1990) light predicate raising analysis, or the treatment of the phenomenon in Kayne (1994: 71-74) and den Dikken (1995). Kayne (2000) provides the following schematic derivation of a heavy NP shift structure, wherein apparent rightward movement of a heavy constituent is achieved by the composition of two leftward movements: one extracting the heavy nominal, the other moving the remnant verbal category.

- 21) ... likes the type... too
 ... [the type...] _i likes t _i too
 ...[likes t _i too] _j [the type...] _i t _j (Kayne 2000: 46)

As for the problematic positioning of clitics farther to the left than other nominals, Kayne's (2002) analysis of clitic doubling exemplifies the kind of higher-order reshuffling required for the present account to succeed. The example below illustrates a complicated "leapfrogging" that first fronts the doubling constituent containing both the clitic and its full DP double, then extracts the latter, finally moving the clitic as part of a remnant together with the verbal material to the left again.

- 22) doy un libro [Juan le]
 [Juan le] _i doy un libro t _i
 Juan _j [t _j le] _i doy un libro t _i
 a Juan _j [t _j le] _i doy un libro t _i
 [[t _j le] _i doy un libro t _i] _k a Juan _j t _k (Kayne 2002: 135, his example 5)

This pattern of movement, in which the clitic is never moved by itself, is what is required to reconcile the present theory with the facts. The crucial point is that surface orders do not wear their derivations on their sleeves – what can superficially be described as

leftward displacement of one element (e.g., of the clitic in the example above) may arise through movements that do not move that element by itself at any point (here, movements of first *Juan*, and then a constituent containing the clitic and verb phrase, achieve the superficial leftward displacement of the clitic element *le*).

There is yet another effect we can assimilate to these predictions: the differential behavior of “weak” pronouns, analyzed by Cardinaletti as occupying a slightly lower subject position. For example, strong pronouns can be separated from the verb by parentheticals, while weak pronouns cannot:

23) John/he (as you know) is a nice guy.

24) It (*? as you know) costs too much. (Cardinaletti 2004: 137, her 80a-b)

As Cardinaletti (2004: 138) points out, this is not plausibly the result of a phonological constraint. This is brought out strikingly in German, where a single element *er* ‘he/it’ behaves differently based on its interpretation as human or non-human:

25) Hans/Er (soweit ich weiss) kommt morgen.
Hans/he (as far as I know) comes tomorrow

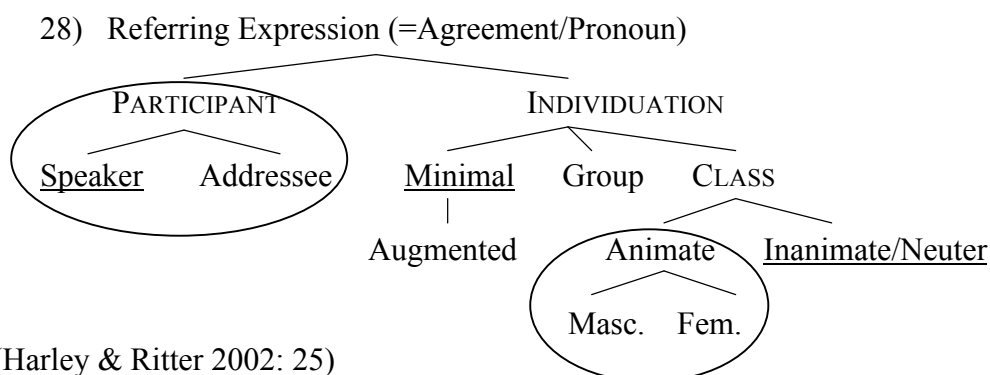
26) Es/Er (soweit ich weiss) kostet zuviel.
it/it (as far as I know) costs too much (Cardinaletti 2004: 137, her example 82b)

In Cardinaletti’s analysis, “weak” pronouns occur in a lower position (AgrSP) than strong subjects, which move on to a higher position (SubjP) above the attachment site of parentheticals. We can represent this proposal schematically as below:

27) [_{SubjP} strong subject [...parenthetical... [_{AgrSP} weak pronoun [...]]...]]

We may describe the relevant weak pronouns as those lacking the animate and participant portion of the phi-feature geometry; by hypothesis, these are then smaller objects. It

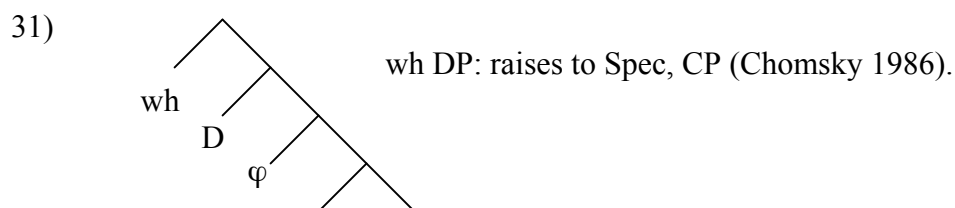
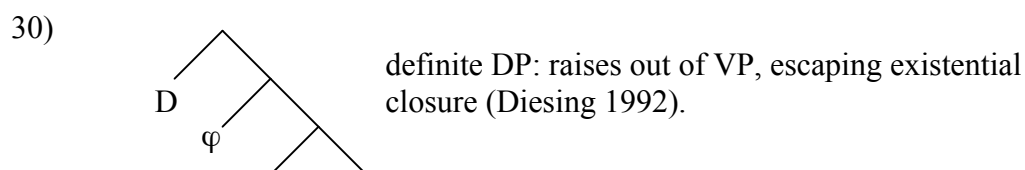
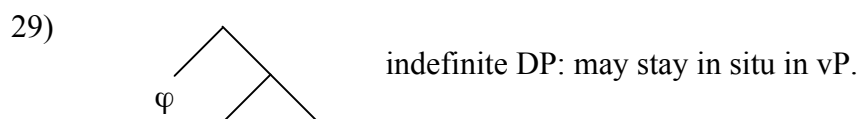
follows that they should move in fewer instances than larger objects, such as pronouns with a richer feature geometry. I repeat below the phi-feature geometry given by Harley & Ritter (2002); the nodes missing from the weak pronouns are circled.



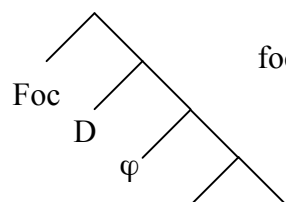
(Harley & Ritter 2002: 25)

If syntactic object α moves in a subset of the instances in which another object β of the same type (e.g., nominal) may move, in general we expect α not to move as high in the tree as β . That prediction matches well with the movement facts Cardinaletti reports.

Finally, consider the effect of constituent size on the distance moved by various types of phrases. Take the following different kinds of DPs: indefinite, definite, and +wh/Focus, respectively. Note that the smaller the constituent, the less it moves.



32)



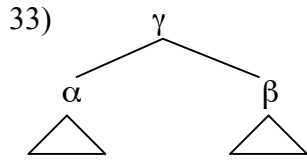
focused DP: raises to Spec, CP (Chomsky 1986).

In particular, as the tree grows, a fixed piece of structure will be smaller, relative to the whole. In general, we expect a pattern whereby small enough objects are immobile, while slightly larger objects in the same position may move a short distance, and larger objects can move further still.

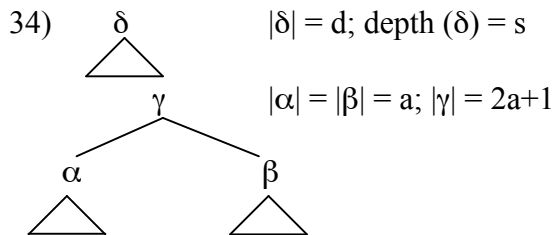
Summing up, this section has suggested that the different movement possibilities for nominal phrases may simply reflect differences in the relative size of those objects, as seen from the embedding clause. This view of things predicts that large objects will move often and far, smaller objects will move less often or not as far, and small enough objects should not move at all. I have sketched how we might apply these ideas to apparently semantically motivated movement to the left periphery, such as *wh*- and focus movement, taking such maximal movement to reflect the presence of a maximally projected nominal phrase. I argued as well that the movement of strong and weak pronouns to different subject positions follows what we predict based on their differing richness of ϕ -feature geometry.

4.6 Islands of symmetry

Certain kinds of island effects fall out as theorems of tree-balancing; for example, a version of the coordinate structure constraint. Consider the case below:



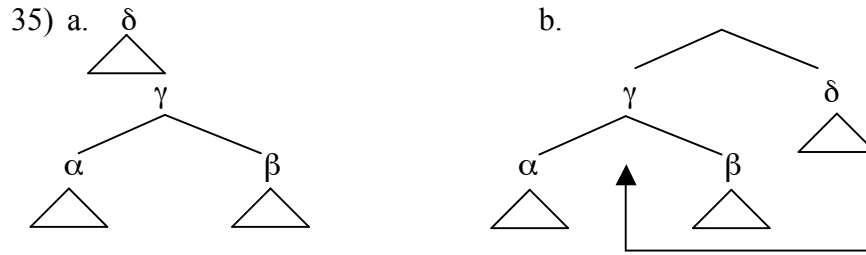
Here, a coordination is taken to consist of two equal conjuncts; idealizing, let γ be comprised of objects α and β , of equal size (say, a nodes)¹⁷. It can be shown that if γ cannot move in a certain configuration, neither can α or β .¹⁸ In configurations permitting movement, the reduction in the number of c-command relations achieved by moving γ is always greater than the reductions (if any) achieved by moving α or β . Moving γ is also a shorter move than moving either of its daughters. On purely structural grounds then, we derive a version of the Coordinate Structure Constraint: moving the full coordination is always preferable to moving an individual conjunct.



We are concerned with comparing two possible movements, once the coordinated structure is embedded within a further object δ : movement of the complete coordination γ , or movement of a single conjunct (say, α).

¹⁷ Depending on the structure assumed, a slight imbalance could be introduced by asymmetric structure introducing the conjuncts as specifiers and complements of an XP headed by the conjunction; one or the other of α or β will be slightly larger. If the conjuncts are large enough, this will hardly matter for blocking the movement of the conjuncts, as desired. However, one consequence of this account of the Coordinate Structure Constraint is that it should hold only of the full conjuncts themselves; subextraction from within a conjunct may still be possible.

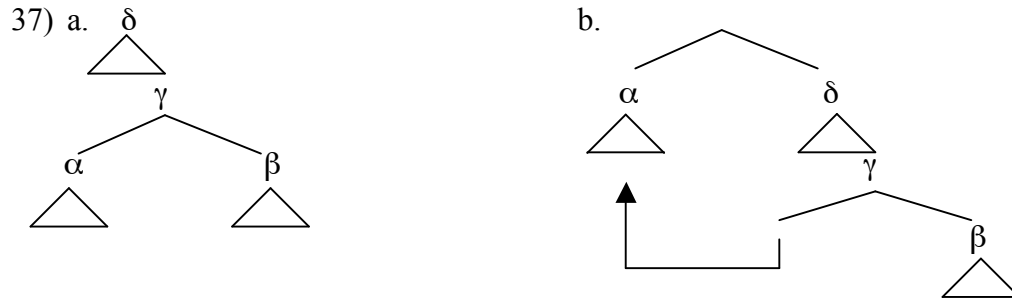
¹⁸ If γ cannot move because it is immediately dominated by the root (antilocality), the condition for α or β to move is that they must contain more nodes than the unmoving part of the tree $(a-1) > (b+1)$; since they are less than half of γ already ($|\gamma| = 2|\alpha| + 1$), this is impossible.



The condition for moving γ as diagrammed above is this:

$$36) (2a)(s-2) > d+1$$

Now consider moving just a single conjunct (arbitrarily, α) instead.



The condition governing this movement of α is this:

$$38) (a-1)(s-1) > d + a + 2$$

To bring out the desired comparison, I introduce the notion of the total reduction induced by a movement: this is simply the difference between the number of c-command relations in the pre-movement structure and the number of such relations in the post-movement structure (if the movement satisfies the FMC, this difference is positive). Let us label this quantity Δ , for a general movement. For movement of the full coordination γ , the difference is $\Delta(\gamma)$; for movement of α , we have $\Delta(\alpha)$.

$$39) \Delta(\gamma) = (2a)(s-2) - d - 1$$

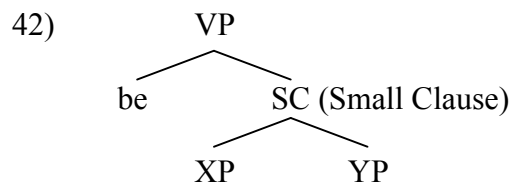
$$40) \Delta(\alpha) = (a-1)(s-2) - d - 3$$

We can then express the difference $\Delta(\gamma) - \Delta(\alpha)$, representing how much greater a reduction is achieved by moving γ instead of α :

$$41) \Delta(\gamma) - \Delta(\alpha) = (a+1)(s-2)+2$$

Note that a is strictly positive, as is $(s-2)$,¹⁹ so $\Delta(\gamma) - \Delta(\alpha) > 0$. This is the essential result: moving the full coordination always results in greater improvement than moving a single conjunct. Insofar as a movement that is shorter and more optimal blocks a longer, less optimal move (there is reason to expect both conditions to matter), we arrive at a structural analogue of the Coordinate Structure Constraint.

It should be pointed out immediately that what is really predicted here is a matter of structure. Rather than calling it a “Coordinate Structure Constraint”, it should properly be called a Symmetric Structure Constraint. Seen that way, an immediate challenge is the analysis of structure and movement in the copula. Since Stowell (1981), and argued in particular by Moro (2000), Pereltsvaig (2006), among others, the copula is assumed to involve a structure like the following:



From this underlying structure, one or the other of XP or YP must move. For Moro, this is motivated by Dynamic Antisymmetry, breaking up the unlinearizable small clause [XP YP]. If this structure is correct, the small clause should be an island for movement of its

¹⁹ Rather, $(s-2)$ is non-negative. For the special case of $s = 2$, note that γ cannot move because it is immediately dominated by the root (the antilocality condition). In that configuration, the condition for α or β to move is that they must contain more nodes than the unmoving part of the tree; since they are less than half of γ already ($|\gamma| = 2|\alpha| + 1$), this is impossible.

immediate daughters, if XP and YP are of equal size. For example, an example like this should be impossible:

- 43) [The morning star]_i is [_{t_i} [the evening star]].

That is, the relevant movement could pass the FMC threshold, but is nevertheless both a worse and longer move than moving the whole double. At first sight, it seems we cannot have our cake and eat it too: if the Symmetric Structure Constraint is to have the desired consequences for coordination islands, it should apply to make the small clause complement of the copula an island, apparently incorrectly.

However, any asymmetry in size between XP and YP can tip the balance and break the island effect. I think appealing to Case features (a difference in size of the structural bundles associated with Nominative vs other features), though a possibility, is the wrong move. Bowers (1993), Svenonius (1994), Starke (1995), among others, propose that small clauses are asymmetric PredPs, taking XP and YP as specifier and complement, respectively. If that is correct, then perhaps we can maintain that the Coordinate Structure Constraint follows from the Symmetric Structure Constraint, while small clauses, as instances of sufficiently asymmetric structure (e.g., PredPs), do not constitute a counterexample.

The larger lesson here is that the calculus of tree-balancing creates regions from which optimal movement cannot escape; the suggestion is that these are natural “islands”, in the linguists’ sense (see Ross 1967). We can make sense of what is going on here: if the goal is to transform structure to be bushier, there will be thresholds of local structure which prevent said structure from being broken apart by movement. A sufficiently bushy

local subtree can be rearranged within a larger structure, but will prove resistant to subextraction of its parts, in the terms outlined above. I leave a fuller exploration of island effects to future research, noting only the general prediction that islands can be understood in terms of tree-balancing insofar as they correspond to well-balanced structures (possibly structure created by movement).²⁰

4.7 Roll-up movement and Malagasy

In this section, I discuss some empirical properties of so-called roll-up movement,²¹ focusing on Malagasy as an illustration. I suggest here that this pattern receives a natural understanding if movement is a mechanism for reducing the number of c-command relations in syntactic trees (at least, the trees that interface interpretation sees, where copies are collapsed). This is of particular interest, because roll-up movement, while empirically well-supported, is mysterious from the point of view of theories that look to features as the explanation for movement. On the present account, such movement creates a positive feedback loop, a decidedly natural phenomenon.

4.7.1 The basic pattern

Deviations from the universal linear order that would follow from a universal hierarchy (Cinque 1999) in conjunction with Kayne's (1994) LCA do not appear to be random, but

²⁰ It is tempting to think of phases in these terms, given the claim that phases are associated with “edge” features (e.g., Chomsky 2007), thus with movement.

²¹ In the literature, this kind of very-local movement deriving head-final ordering is also called snowballing, intraposition, and onion-skin movement.

tend to involve strict mirror-order reversal of the expected order over some continuous sequence of the hierarchy. This pattern is extraordinarily widespread; examples include:

- a) All ‘mirror principle’ (Baker 1985) effects, on a strong interpretation of the relation between morphology and syntax;
- b) Word order in Dutch and German non-finite clauses (Zwart 1994, Müller 1998, Koopman and Szabolcsi 2000);
- c) Hungarian verbal complexes (Koopman & Szabolcsi 2000);
- d) Malagasy clauses (Rackowski 1998, Rackowski and Travis 2000, Pearson 2000, 2007, Svenonius 2008),
- e) Both clauses and DPs in Niuean (Kahnemuyupour and Massam 2006);
- f) Many others; see already Kayne (1994) for further examples.

On an account where displacement is caused by lexical features, there is no immediate explanation for why these strictly reversed sequences should be observed.²² Why something like this should be true on such a systematic and cross-linguistically widespread basis is not well understood, to say the least. Clearly other languages lack the relevant movement-driving features in those locations (because some languages exhibit the expected base order there). Then it is a matter of free variation among languages; but if so, why do we not find languages with some, but not all, of the movement-driving features in these regions (resulting in, for example, ‘long’ reversals, i.e. reversal of the relative order of non-trivial strings)? If there were nothing more than random variation in features at work, surely such patterns would be expected to vastly outnumber the incidences of strict reversal. But they do not; for example, examining some cross-

²² But see Svenonius (2007) for an attempt to implicate acquisition effects, and relevant discussion. Biberauer et al (2007) propose a Final Over Final Constraint (FOFC) to capture the facts at issue, accounting for strictly reversed sequences in terms of agreement for movement-inducing features obligatorily propagating down the tree. A full discussion of the FOFC is beyond the scope of this work, though I will point out here that it appears to be too strong. For example, the FOFC would seem to rule out many of the rare but attested DP orders discussed in the next chapter. Their constraint also runs afoul of two important facts of Malagasy discussed in more detail below: although vP adverbs appear in mirror order, in conformity with the FOFC, the verb and object do not; more puzzling still from the point of view of the FOFC, the two deepest adverbs (*tsara* and *tanteraka*) may optionally appear in uninverted order. See below.

linguistic data on the frequency of word order within nominals, the strictly rolled-up mirror order is on a par with the base order, comprising the two most common orders by far (see Cinque 2005, and Chapter 5). Moreover, overt manifestation of the morphology believed to drive these movements is often in short supply. For instance, Niuean, with snowballing movement in both DPs and clauses, displays robustly isolating morphology (Kahnemuyipour & Massam 2006), casting doubt on any putative need to move for affixation or other morphological reasons. Something else seems to be at work here.

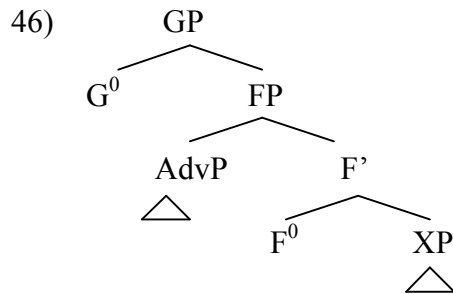
4.7.2 *Malagasy facts*

Malagasy is one language displaying such a pattern (Rackowski 1998, Rackowski and Travis 2000, Pearson 2000, 2007, Svenonius 2007). Importantly for us, “[w]hile the order of preverbal adverbial elements in Malagasy conforms to Cinque’s universal hierarchy, postverbal adverbials are in the mirror order.” (Rackowski and Travis 2000: 120) The examples below illustrate:

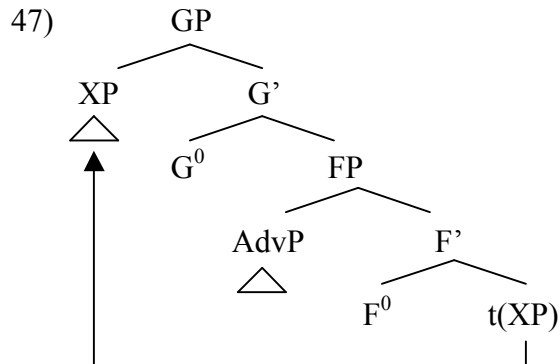
- 44) M- an- asa lamba tsara foana Rakoto
Pres-AT wash clothes well always Rakoto
 “Rakoto always washes clothes well.” (Rackowski 1998: 7)
- 45) Tsy manasa lamba tsara intsony mihitsy Rakoto.
NEG PRES.AT.wash clothes well anymore at-all Rakoto
 “Rakoto does not wash clothes well anymore at all.” (Rackowski 1998: 18)

The post-verbal adverbs in Malagasy appear in the reverse of the expected order. Let us suppose, with Cinque (2005), that roll-up movement proceeds in short, but not too-short steps: from the complement of one phrase to the specifier of the next. I follow Svenonius (2007) in labeling the alternating heads that do and do not induce movement as G and F,

respectively. The pattern in the lower portion of the Malagasy clause (where we find post-verbal adverbs in reverse order) is derived by iterating the pattern below. Before roll-up movement, the local configuration looks like this:



Roll-up movement is diagrammed below:



The pattern iterates, with the GP at the top of the diagram playing the role of XP in the next step of roll-up movement (thus, moving over another FP with an AdvP specifier).

We can apply the FMC to this configuration. Letting a represent the number of nodes in AdvP,²³ and x the number of nodes in XP, the movement results in fewer c-command and containment relations if the following inequality holds.

²³ Although I include a term for the interior size of the AdvP here, as a way of being agnostic, various considerations suggest that this value should be 1, regardless of the actual lexical contents of the AdvP. This reflects the idea that Externally Merged left branches are opaque to the structure embedding them; cf. Uriagereka (1999). In fact, the post-verbal adverbials of Malagasy seem not to permit additional structure anyway. Rackowski and Massam (2000) analyze the lower adverbs as heads, not specifiers, noting for

$$48) \quad 2x > a + 9$$

Simple counting tells the tale here. Specifically, as the pattern of roll-up movement continues, the moving “snowball” gets ever larger, while the local configuration crossed by movement remains the same. In terms of the inequality above, the quantity *a* is constant for each step of roll-up movement, but the quantity *x* is larger each time. In present terms, this can be interpreted as a stronger motivation for each subsequent step of roll-up movement.

4.7.3 *Dynamics of roll-up movement*

This view of the dynamics of roll-up movement has two consequences. First, we expect that the earliest stages of roll-up movement should be the most weakly motivated. If the optionality of a particular movement reflects relatively weaker motivation for it (i.e., the movements that are optional are those that produce the least reduction of c-command relations), then we expect that the first stages of roll-up movement, in the deepest part of the tree, should be most subject to optionality. Second, we expect that once roll-up movement gets underway, it should continue unless derailed in some way.

The first prediction aligns nicely with the facts. In particular, Rackowski (1998) reports that the inversion of the two most deeply embedded adverbs (*tsara* and *tanteraka*) is optional. I interpret that as indicating that the first step of roll-up movement in the Malagasy vP is optional. That is problematic for an approach which looks only to

instance that intensification by *tena* ‘very’ is impossible for these items, unlike other adverbs, suggesting they may not be phrasal. I set this possibility aside.

lexical features, but it makes good sense if what is at stake is structural optimization, since the benefit of moving is least on the first step.

What about the second prediction? On the face of it, we would seem to predict that roll-up movement should go “all the way up,” driven ever more strongly on each step. I leave a fuller exploration to future work, merely mentioning here two ways of explaining the fact that roll-up in Malagasy is limited to the lower (post-verbal) adverbials.

First, note that Chomsky (2000) takes vP to be a phase boundary. We might then suppose that roll-up movement deposits the moving snowball within the Transfer domain of vP. If so, it will be effectively invisible to material in the higher phase, and so roll-up movement should stop there. That seems promising, in that the voice morphology is an immediate prefix to the verb in Malagasy. That is consistent with the roll-up snowball landing in a specifier just below the voice head, which heads a phase, there being rendered invisible to the higher (CP) phase.

Another possible explanation is that roll-up movement transitions into an even more local form of movement, called “skipping”, which does not result in reversed word order; instead, skipping movement has results that are superficially indistinguishable from simple “long” movement. In this regard too, the facts of Malagasy are suggestive. In particular, Malagasy is a VOS language; some movement applies to bring the rolled-up region to the left of the subject.²⁴

²⁴ This requires further comment. The “snowball” constituent does not seem, on the face of it, to undergo long movement over a large stretch of structure to the left of the subject. Rather, preverbal adverbials, negation, and Tense also precede the vP snowball. This actually is consistent with skipping, if a further movement brings the skipped-over stack of heads to the left independently.

4.7.4 Rightward Object Shift in Malagasy: Against head directionality

It is interesting, and potentially important, to note that when the object of the verb is definite, it can be displaced to a position amidst the rolled-up post-verbal adverbial material, as the examples below illustrate:

- 49) a. Tsy manasa **lamba** mihitsy ve Rakoto?
 NEG PRES.AT.wash clothes at-all Q Rakoto
- b. *Tsy manasa mihitsy **lamba** ve Rakoto?
- c. Tsy manasa mihitsy **ny lamba** ve Rakoto?
 NEG PRES.AT.wash at-all DET clothes Q Rakoto
- ‘Does Rakoto not wash clothes at all?’

(Rackowski & Travis 2000: 120)

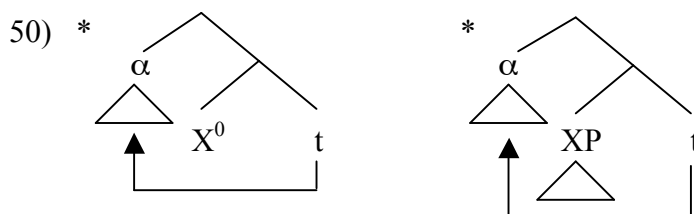
The generalization to be drawn here is that indefinite objects are obligatorily right-adjacent to the verb, while definites may be separated from the verb by adverbials. This looks very much like a form of Object Shift (interacting with roll-up movement).

This is an interesting fact. It tells us, for one, that whatever conditions govern Object Shift in other languages may be operative here as well. It would also seem to provide evidence against a Head Directionality Parameter approach to the word orders here analyzed as arising from roll-up movement, whereby the rightward heads simply take their complements on the left, without movement (see Abels & Neeleman 2009 for a recent articulation). If so, we would have to countenance rightward movement to get the object further right of the verb (or, perhaps, rightward specifiers).

4.7.5 Reconciling Antilocality and strict reversal

There is an issue that should be addressed at this point. I have suggested that the pattern of movement described here might underlie the phenomenon of roll-up movement. The essential characteristic of that pattern of movement is that the relevant reversals are “short”: in surface terms, the pattern results in a single overt position being reversed.

On the face of it, that ordering pattern seems to conflict with the predictions made here. It is an unavoidable commitment of this view of movement that it must be at least minimally antilocal: it must skip at least two tree positions. Put another way, movement that crosses only a single position (whether a head X^0 , or a phrase XP, hangs off the crossed branch) is predicted to be systematically impossible. The trees below illustrate the offending flavor of movement.



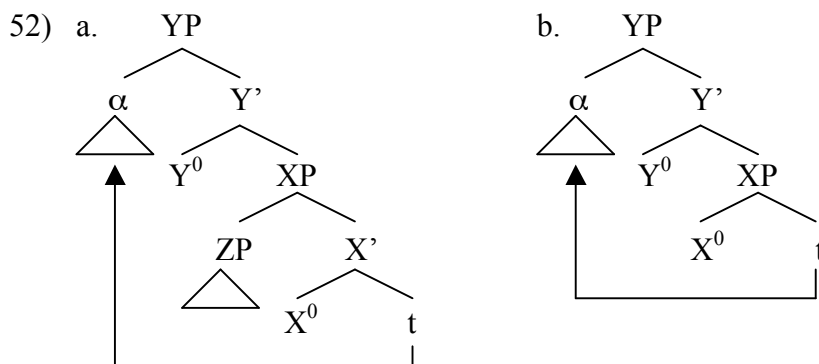
The prediction that at least two structural positions must intervene between launching and landing sites of movement appears to conflict with the empirical picture of roll-up movement, involving inversion around a single overt element. On the face of it, this analysis would seem to predict that roll-up movement would produce, in the surface order, a series of short but non-trivial reversals, with subsequences of (at least) two elements (the two or more positions necessarily skipped by roll-up) surviving in their native (head-initial) order. In other words, for an abstract underlying sequence 12345678..., this pattern of movement ought to produce something like ...-78-56-34-12.

However, there is a simple way to reconcile the analysis with the facts. Koopman (1996) proposes a Generalized Doubly-Filled Comp Filter (GDFCF):

51) A single XP cannot have both an overt head, and an overt specifier.

It follows as a consequence of the GDFCF that minimally antilocal movement, of the sort predicted here, must appear on the surface to involve order reversal of a single overt position.²⁵ To see why, let us consider the options. By hypothesis, the relevant kind of movement proceeds from the complement of one phrase, to the specifier of the next higher phrase. We do not know if the phrase embedding the rolled-up constituent as its complement also takes a specifier, or not.

This means that the configuration for roll-up movement, under the present analysis, is one of the two possibilities diagrammed below. Here, α is the moving category, which begins as a complement within XP, and ends up as the specifier of a higher phrase YP.



Per the GDFCF, within XP and YP, the specifier or the head may be overt, but not both.

YP takes overt α as its specifier; therefore, its head Y^0 must be silent. Within XP, at most

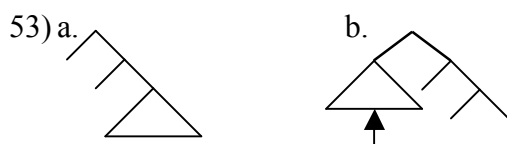
²⁵ The discussion here assumes that head and specifier are still distinct structural positions; put another way, the GDFCF is a restriction on pronunciation. See Starke (2004) for a more radical interpretation; he proposes that the GDFCF holds because specifiers and heads are never simultaneously present, structurally. In effect, having a specifier “counts as” having the appropriate head. If so, then the reasoning in this section no longer holds up.

one of X^0 (head) and ZP (specifier) is overt. This yields the desired result: there will be one overt position (namely, the head or the specifier of XP) in the stretch of phrase structure crossed by a single step of roll-up movement.

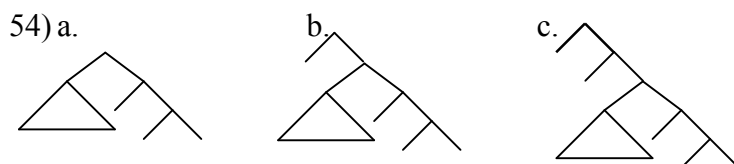
4.8 Iterating Patterns of Movement

To begin to understand how movement might unfold in a derivation guided by concerns of economy of command, we can frame an important question given the different basic forms of movement predicted above: how does *iteration* of the pattern affect the conditions for movement? Does the movement strengthen, weaken, or leave unchanged the motivation for undertaking an identical movement as the moved-across configuration repeats, later in the derivation? Which terms in the mathematical expression regulating the *next* step of that pattern change, and in which direction?

In this section, I focus on the kind of movement known as Roll-up movement, discussed in the last section and schematized below.

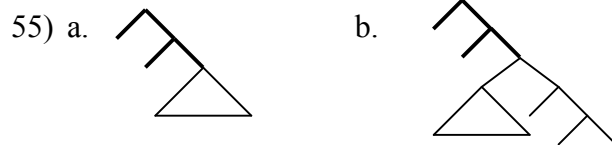


More structure accumulates via Merge, until the configuration repeats (right):

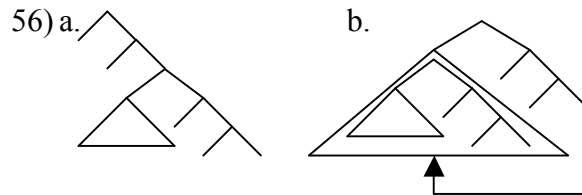


Notice the similarity between the first stage (a) of (53) and the last stage (c) of (54); the same configuration (buried beneath a stack of two heads) recurs. Just to make this

completely clear, I am drawing attention to the local isomorphism between the bolded portions in (55a) and (55b):



Now movement occurs in the same configuration, analogous to the transformation of (53a) into (53b):

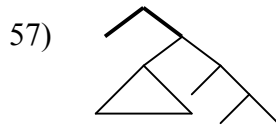


Recall that, according to the Fundamental Movement Condition, the relevant comparison is $(a-1)(s-2) > b+1$. Since the non-moving part of the tree looks the same for both the first and the next step of roll-up movement, the factors b and s are constant (they represent the node count and depth of this region). Only the factor a has changed; it has grown by 6 units (i.e., the moving category has 6 more nodes).

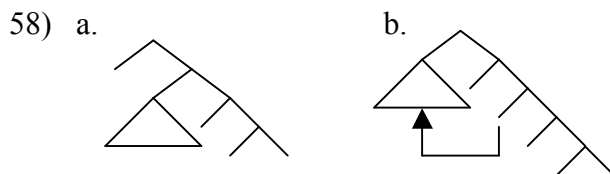
If the size of the moving alpha is within a certain range, this pattern feeds itself, exhibiting positive feedback. That is, if the conditions governing this type of roll-up movement were met on the last iteration, they will be met even more strongly the next time (after two more layers of structure have been added to the top of the tree). Notice also that, while it quickly becomes the case that a more evenly-balancing movement could be found by moving something smaller, deeper in the tree, the ‘snowball’ is the nearest object to the root whose movement is permitted. Given the pervasive evidence for locality/minimal search throughout the organization of human grammar, it is a small leap

to wonder whether it is operative here. If so, we directly predict a locality constraint on movement of the required sort, such that the shortest enabled (FMC-satisfying) movement is chosen. In these terms, we can rationalize a pattern in which snowballing movement ‘takes off’, a self-reinforcing, increasingly-good and first-available option as the derivation proceeds.

However, as alpha’s size (the ever-growing snowball) passes a certain threshold, non-snowballing, successive-cyclic move of the same object that moved on the last step is enabled, before the background configuration for roll-up is constructed. The relevant configuration arises after the very next Merge operation, adding a single additional layer of structure to the root (this minimal dominating structure is **bolded** below):



The roll-up pattern iterates (not sooner than) after a pair of additional Merges. Here, after just one additional Merge operation, a different movement can occur, but only if the moving category is large enough:

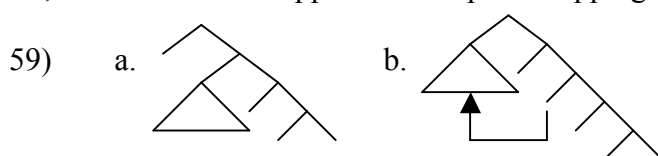


I suggest calling this predicted form of hyper-local, successive-cyclic²⁶ movement ‘*skipping*’. I argued just above that the snowballing pattern has positive feedback, such

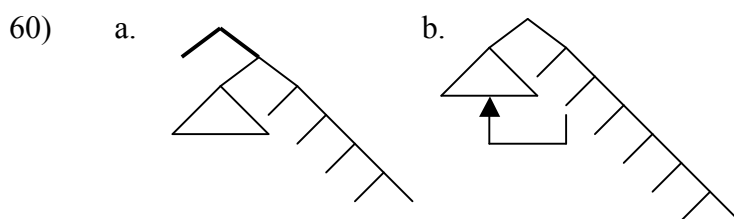
²⁶ By using ‘successive cyclic’ to describe this movement, I mean nothing more than that it affects the very same object that was affected by a previous movement. Snowballing or roll-up movement, as described above, is not successive-cyclic in the relevant sense; it affects a different, strictly ‘larger’ object than the last movement.

that as the configuration in which it occurs is built anew, each next step of movement achieves a greater reduction in the total number of c-command and dominance relations.

What about ‘skipping’ – once one step of such movement occurs, are further steps of identical movements more or less motivated in terms of balancing the tree? As it turns out, this form of movement is subject to negative feedback. Let us walk through the details, one more time. Suppose one step of ‘skipping’ has occurred, as below:



Now the configuration for skipping recurs immediately, after the very next head is Merged to this structure (new Merge **bolded**):

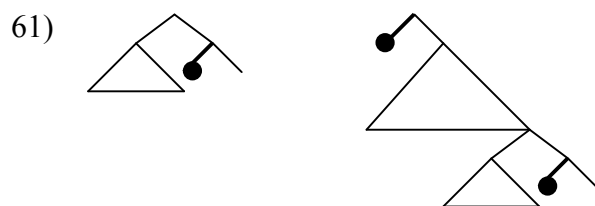


Comparing the configurations, we see a picture opposite to what we saw with the roll-up pattern. In this pattern of movement, the moving object is the same on both steps, as is its depth of embedding (a and s are constant), but the number of nodes in the non-moving part of the tree has grown (b is greater). Recalling the FMC $((a-1)(s-2) > (b+1))$, the terms in the product on the left are fixed, while the standard of comparison on the right grows; there is then negative feedback in iteration. To put it another way, this movement cannot feed itself indefinitely.²⁷

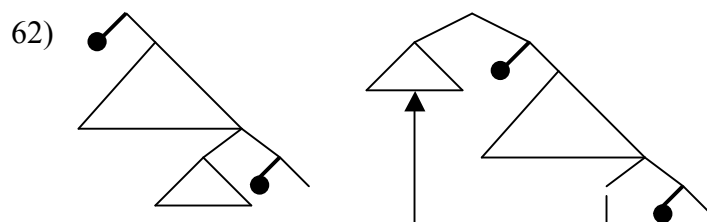
²⁷ It would seem that the two types of movement might intertwine, with roll-up movement derivationally preceding the ‘skipping’ pattern described above, perhaps transitioning back to roll-up movement when negative feedback ‘snuffs out’ skipping. I leave further investigation of these matters to future work.

To complete the picture, I turn to a structural analogue of A-bar movement, predicted with crucial reference to a notion of syntactic cycle akin to Chomsky's phase, where some material from a previous cycle remains 'live' in the derivation of the next cycle (see Chapter 8 for further speculative remarks about this pattern of movement). If cycles themselves are of a fixed size, hence essentially identical, then from one step of 'long' successive-cyclic movement to the next, the conditions are exactly the same: it is a kind of 'equilibrium' movement, stable once established.

Concretely, consider a typical application of successive-cyclic movement, carrying a moving category from the edge of one phase to the edge of the next higher phase. The complement of the phase head (this head is indicated by a black dot at the end of a bolded branch) is subject to Spell Out/Transfer. So after Transfer, the structure on the left is all that remains. Then another cycle is constructed atop the remnant of the last phase, with the new phase head the last item merged, as indicated (right):



The configuration is transformed by a step of movement, diagrammed below:



Then Transfer applies again, and the process repeats at higher cycles, if any. Consider the series of cyclic wh-movements in an English example like (63), indicated schematically with arrows:

63) What do you think t that Sarah believes t that John bought t ?



Here, the wh-moving object, with its source in the most deeply embedded clause, achieves a surface position at the ‘top’ of the matrix question. This is so despite the lack of any obvious semantic, at least, motivation for intermediate steps in movement (in this case, across *that Sarah believes*).²⁸ In simplistic terms, this kind of movement represents something like a stable solution. If one iteration of that kind of movement satisfied the FMC, then the next such step of movement will as well, and by the same numerical margin (under the simplifying idealization that cycles are identical to each other in relevant structural details – in this case, their node count and depth).

4.9 Conclusions

In this chapter, I have explored the idea that syntactic displacement is motivated by tree-balancing. I formulated the Fundamental Movement Condition, repeated below:

64) $(a-1)(s-2) > b+1$

This expression relates the number of nodes in the moving part of the tree (a), in the non-moving part of the tree (b), and the depth of embedding of the former within the latter (s).

I derive from this two predictions of minimal distance and minimal size:

65) A category immediately contained by the root cannot move.

²⁸ See McCloskey (2002) for a particularly illuminating look at the workings of the complementizer system and A-bar movement, especially with respect to intermediate stages of long-distance movement.

66) The moving category must consist of at least 5 nodes.

Antilocality as derived here is empirically well-motivated. The size threshold effects (both the “hard” prediction of a minimal size for a moving object, and the “soft” prediction that larger objects should move more and farther) are novel, but I provided several empirical domains where they seem to get the facts right.

This conception of movement also predicts that sufficiently symmetric structures should resist movement. I argued that this provides a naturalistic explanation of the Coordinate Structure Constraint of Ross (1967), though noting a conflict with the treatment of small clauses offered by Moro (2000).

Next, I considered the pattern of very-local inversion known as roll-up movement, focusing on some facts from Malagasy. As I showed, the data support, first of all, the conclusion that it is indeed movement, not a choice in the relative order of complements and heads, that derives the Malagasy word order. There is also suggestive support of the structurally-driven account of roll-up movement I provide, especially in the fact that reversal is optional for the deepest pair of adverbs.

I finally considered the possibility of derivational canalization and the effects of iterating certain patterns of movement. I suggested that the cross-linguistically very frequent pattern of snowballing or roll-up movement can be understood in terms of a runaway positive feedback loop, where each step of movement creates an increasingly strong pressure for the next movement of the pattern to occur (modulo the remarks about the interaction of ‘skipping’-type movement with roll-up).

In this chapter, I have focused on the mathematical predictions that follow from supposing that movement is for balancing trees, with only schematic remarks about how some familiar movement patterns might receive a natural explanation in these terms. Any attempt at a comprehensive survey of the relevant phenomenology is far beyond the scope of this chapter. But in an attempt to move forward, in the next chapter I focus on a smaller, well-studied body of phenomena, involving word order patterns observed within nominal phrases across the world's languages. I show there that the possible and impossible orderings can be explained by supposing that movement is constrained by the Fundamental Movement Condition, so long as the shape of the affected tree structure falls within certain bounds.

CHAPTER 5: TREE-BALANCING IN THE DP AND UNIVERSAL 20

5.0. Introduction

This chapter applies the ideas of the last chapter to the relative order of elements within the DP. Recall that chapter 4 argues that syntactic movement is a tree-balancing mechanism, i.e. a way to reduce the total number of c-command relations (equivalently, irreflexive dominance relations) in syntactic trees. While orthogonal to familiar Minimalist explanations of movement in terms of checking features or licensing interpretations, the intuition fits squarely within the Minimalist paradigm. Here, movement is not seen as a “Last Resort” (Chomsky 1986); instead I claim it is a structural response to conditions of efficient computation.

The purpose of this chapter is to confront this treatment of movement with some well-established and reasonably nuanced empirical facts. While the last chapter developed the idea that structure determines movement possibilities, here I pursue the related idea that movement reveals structure. I argue that observed movement patterns in a syntactic domain (here, the DP) diagnose the underlying structure of that domain (supposing that underlying structure to be identical across languages).

In this case, I will focus on the cross-linguistic ordering of demonstrative, numeral, adjective, and noun (the topic of Greenberg’s (1963) Universal 20). Of the 24 logically possible orders of these elements, Cinque (2005) reports that only 14 are attested as neutral/unmarked orders. I hypothesize that there is a single universal base DP tree, and that all and only the attested orders arise via instances of movement that reduce the number of c-command relations in this tree (movement always improves tree balance).

This strongly structural account demonstrably succeeds, so long as the base DP tree falls within structural limits detailed here. That is, we can find a single underlying base structure, such that each of the 14 attested relative orders (of Dem, Num, Adj, and N) has at least one derivation in which every step of movement reduces the number of c-command relations in the tree, while none of the 10 unattested orders has such a (monotonically tree-balancing) derivation. In fact, there are many possible underlying trees that meet this condition. The possible underlying structures with this property (found by computer-assisted search, detailed in the Appendix) include good matches to recent cartographic proposals, including in particular a number of strictly right-branching spines (considered a likely candidate for the shape of the base tree). If the structural predictions made here turn out to be correct, tree-balancing may be the deep “why” behind the nuanced facts of possible and impossible DP orders (and perhaps syntactic movement more generally).

5.0.1 DP orders: A brief sketch

The typological facts at issue were first described by Greenberg’s Universal 20:

“When any or all of the items (demonstrative, numeral, and descriptive adjective) precede the noun, they are always found in that order. If they follow, the order is either the same or its exact opposite.”
(Greenberg 1963: 87)

Cinque (2005) and Abels & Neeleman (2009) update this description; Figure 1 summarizes their findings (orders in grey cells are unattested)¹. See also Hawkins (1983), Lu (1998), Dryer (2009), among others.

a. DMAN	b. DMNA	c. DNMA	d. NDMA
e. *MDAN	f. *MDNA	g. *MNDA ◇	h. *NMDA
i. *ADMN	j. *ADNM	k. ANDM	l. NADM †
m. *DAMN	n. DANM	o. DNAM	p. NDAM †
q. *MADN	r. MAND	s. MNAD	t. NMAD †
u. *AMDN	v. *AMND	w. ANMD	x. NAMD

Table 3: Attested and unattested DP orders.

Relative orders of (D)emonstrative, Nu(M)eral, (A)djective, and (N)oun.

◇: *Greenberg's formulation incorrectly allows this order.*

†: *Greenberg's formulation incorrectly excludes these orders.*

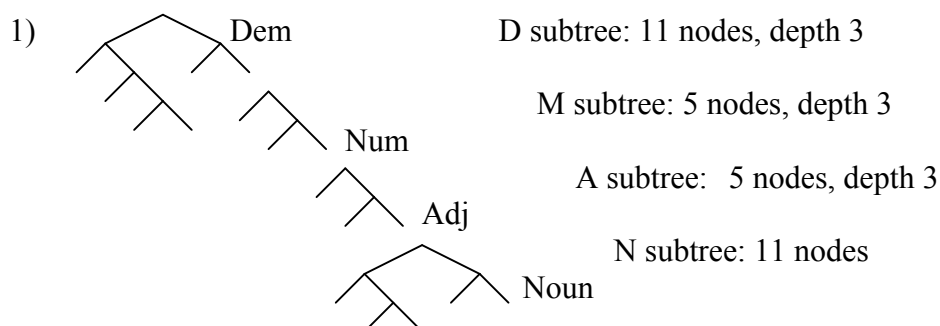
Cinque (2005), further developing earlier (1996, 2000) proposals of his, argues that the attested orders are derived by movement from a base order DMAN (e.g. English *these three blind mice*) by phrasal movement affecting the N(P), or something containing it. I leave a discussion of this proposed constraint, and its theoretical treatment, to section 5.1.

5.0.2 Tree-balancing is a sufficient explanation

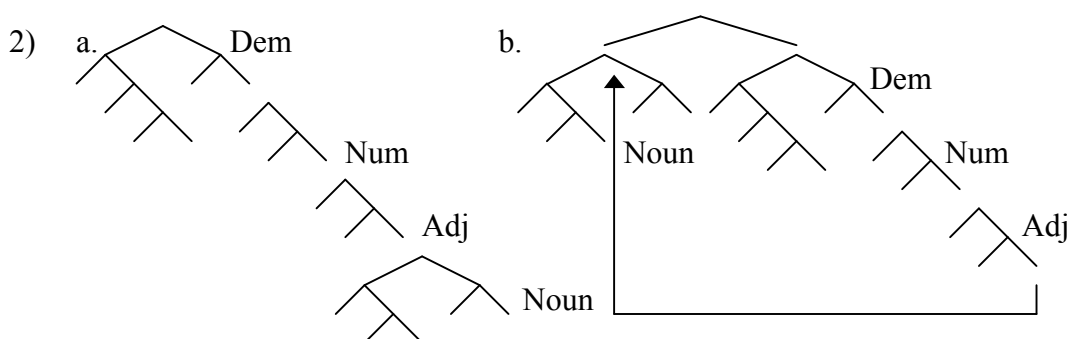
In what follows, I show that tree-balancing can account for the facts summarized in Figure 1. That is, there are possible shapes of a universal base DP tree such that each movement in the derivation of each attested order decreases the number of c-command relations in the tree, while each unattested order involves one or more movements which

¹ I abbreviate demonstrative as D, numeral as M, adjective as A, and noun as N. The lettering scheme here matches Cinque's (2005: 319-320) 6(a)-6(x), for ease of reference.

would increase (or leave unchanged) the number of c-command relations. The tree below is one example of a possible base structure meeting this condition:²



For example, the attested order NDMA is derived by a single movement (of the N subtree). As shown below, this move improves tree balance (i.e. reduces the number of c-command relations in the tree), and so is correctly predicted.



On the other hand, the last step of movement in the derivation of unattested order ADNMA from attested order DNMA, depicted below, would increase the number of c-command relations in the tree. It is thus correctly excluded as a possible pattern of movement.³

² This is just one possible shape fulfilling the condition; there are other possibilities. Moreover, note that what is predicted by this account is only the gross tree shape, not the identities of the nodes. The goal is to use the empirically observed ordering behavior of fixed points in the structure (here, Dem, Num, Adj, Noun) to track the transformations of a finer-grained underlying structure.

cartographic proposals. It would seem too much to attribute to coincidence if the “real” DP tree, as revealed by cartographic research, turns out to be one of these trees; they really are sparsely distributed in the space of possibilities. It remains to see whether the map of the DP, still something of a moving target, meets these conditions or not. But it is encouraging that the shapes predicted here look at least plausible as hypotheses about the cartography. If the real map does fit these conditions, syntactic movement may find a deep explanation going beyond lexical features,⁴ in purely structural terms of tree-balancing (by hypothesis, for structural/computational optimization).

Moreover, the featural account of Cinque’s generalization faces challenges in light of the study of Abels (2011), who looks at the typology, among Germanic varieties, of relative order of Modal, Auxiliary, Verb, and Particle. Abels reports that if one takes the relevant hierarchy to be $\text{Mod} > \text{Aux} > \text{V} > \text{Prt}$, the typology reproduces the DP typology exactly, *mutatis mutandis*. In other words, the attested orders are those that follow Cinque’s Generalization for this hierarchy: the only permitted movements affect Prt, or something properly containing it.

Notice, however, that the licensing account Cinque sketches, motivating movement within the DP for reasons related to its status as the extended projection of the noun, does not extend comfortably to Abels’ data. That is, while it is standard to take the particle to be at the bottom of this portion of the tree, the clausal domain is not thought of as an

⁴ See chapter 4 for clarification of my stance on the role of features in driving movement. To recap briefly, we probably still need features to explain why movement patterns for individual languages often tend to be quite rigid. But the present account might provide a deeper, principled explanation for the existence and distribution of movement-driving features themselves; by hypothesis, they are not brute accidents of cross-linguistic lexical variation, but rather are such as to drive beneficial tree-balancing movements. This might be studied from a diachronic perspective (say, looking for a “cline” in language change towards movements which provide better tree balance), but I cannot pursue the matter here.

extended projection of the particle. Rather, the familiar intuition is that the clause is an extended projection of the verb. But structurally, the verb is a higher category in the base.⁵ It seems unlikely that movement in this domain is motivated by a need for projections to check particle features, for example. Instead, what remains is a structural generalization: the deepest part of the base structure must move.

If this can be made to work, it allows a unification of two of the basic activities of syntax: on the one hand, drawing maps of syntactic structure, and on the other, determining the conditions on movement. If movement is driven by tree-balancing, then each instance of movement tells us something about the structures affected and produced (namely, that the latter must support fewer and shorter long-distance dependencies than the former). In other words, movement diagnoses structure, and structure determines possible movements. The present chapter can be viewed as an extended diagnosis of DP structure, on the basis of the movements observed in that domain across languages.

5.0.4 Structure of this chapter

The rest of this chapter is structured as follows. In section 5.1, I describe the empirical facts at issue in greater detail. In 5.2, I present my assumptions and methodology. Section 5.3 provides an overview of the numerical predictions I make. Section 5.4 comprises a discussion of these predictions and a comparison with cartographic proposals

⁵ This must be true for Abels' assimilation of the ordering facts to Cinque's Generalization to hold. One could not, for example, simply suppose that the base order was Mod > Aux > Prt > V; for then movement of the Prt without the verb, robustly attested, would constitute remnant movement.

in the literature. Lastly, 5.5 concludes the chapter, summarizing what I have established and exploring issues for future research.

Further relevant material is included in Appendix A. This includes a much more technical and thorough discussion of the methodology I employ, a table of numerical results and explicit solution lists, and the simple program used to explore the relevant structural conditions.

5.1. DP orders: Facts & analysis

As Cinque argues, the distribution of orders, as seen in Figure 1 above, is strongly consistent with the view that the possible orders are derived by movement from a common Dem Num Adj Noun (DMAN) base order.⁶ This regularity is strong evidence for “cartographic” proposals; it is unclear how one could explain the absence of the unattested orders in a “non-configurational” approach (cf. Hale 1983).

Cinque (2005) noted the striking fact that apparently only movements affecting the NP, or something properly containing it, are found. I will hereafter refer to this as Cinque’s Generalization:

- 4) Cinque’s Generalization: In the derivation of basic DP orders, movement affects the NP, or an XP properly containing NP.

That is, there is no evidence of so-called remnant movement (den Besten & Webelhuth 1987, Müller 1998, Hiraiwa 2002, among others) within the DP, though remnant

⁶ Properly speaking, they are all derived from a common syntactic hierarchy which, if no movement applied, would be linearized as DMAN; much current work takes it that linear order is determined after syntax, with syntax-internal forms unordered.

movement in other domains may well be real. Most telling is that the orders that could *only* be derived by remnant movement (e.g. ADN_M) are systematically absent. I return to this issue below; for now, I assert that the availability of remnant movement elsewhere, as well as its unavailability within the DP, could both be explained wholly in terms of the tree-balancing concerns pursued here. On the present account, patterns of movement are governed by the geometrical structure of the tree, so it is unsurprising to find different patterns in different syntactic domains.

5.1.1 Cinque's Generalization as Harmonic Bounding of left alignment

Steddy & Samek-Lodovici (2011) propose an Optimality-Theoretic account of Cinque's Generalization. In their view, Cinque's Generalization falls out naturally due to harmonic bounding (Samek-Lodovici 1992, Prince & Smolensky 1993) of Align-Left constraints applying to the individual nominal elements. Crucially, their Align-Left is penalized by intervening traces; this has the effect of ruling out remnant movement, deriving the desired typology.

There is a close relationship between the ideas pursued in this chapter and the mechanics of Steddy & Samek-Lodovici's account. Note that, in light of Kayne's (1994) LCA, there is an inextricable link between linear position and hierarchical structure; in crude terms, elements higher in the tree linearly precede lower elements (more precisely, if an element A asymmetrically c-commands another element B, the terminals that A dominates precede the terminals that B dominates). The present account favors tree forms with fewer c-command relations; in terms of a given portion of tree-structure, this

translates as a penalty for depth in the tree, with deeper elements entering into more c-command relations. Put another way, relative to each piece of structure there is a preference to have that structure as high in the tree as possible. But, again due to the LCA, movement higher in the tree amounts to movement leftward in the string. As a consequence, a preference for tree-balancing in hierarchical structure appears as a preference for leftward alignment in linear order. I leave to future work a more careful comparison with their approach, including ways to empirically distinguish between these two explanations.⁷

5.1.2 *Cinque's Generalization in Artificial Language Learning*

The work of Culbertson *et al* (2012) on DP ordering in an artificial language learning paradigm provides a powerful new way to study the nominal ordering facts at issue. Presented with a conflicting mix of (fragments of) nominal orders, their subjects alter the frequency distributions in their own outputs to boost the cross-linguistically most common orders, while typologically rare orders – and especially, unattested orders – are avoided, even when they dominate the input. This effect appears not to be an artifact of prior foreign language exposure, and it is measured after only a single hour-long training session (see those works for details).

⁷ The present account, as far as I can see, depends more sensitively on the exact syntactic structure involved. It also allows, in principle, remnant movement. In a sense, then, it is less restrictive than the account put forward by Steddy & Samek-Lodovici (2011). However, note that the conditions here are stated uniformly over nodes of all types; it is thus a more general explanation than they pursue, where the Align-left constraints specify the elements they target (namely, Demonstrative, Numeral, Adjective, and Noun).

It is hard to avoid the conclusion that some deep cognitive bias is at work both in their experiments, and “in the wild”, leading to the distribution of nominal orders we find. In particular, in this case we should not appeal to the (undeniably real) effects of evolution of the learned portion of language, where small biases can over time lead to stark categories and emergent behavior. The experimental results discussed above strongly suggest that we should look within a single speaker, at linguistic cognition of an innate and universal nature.

5.1.3. *Cinque’s Generalization beyond the DP*

Another important development is the finding by Abels (2011) that the ordering options found for DPs are reproduced exactly in the options for order in verb clusters across Germanic varieties. Explicitly, for a base hierarchy Mod (modal verb) > Aux (Auxiliary verb) > Verb > Prt (Particle), we find the same 14 orders. Put another way, a form of Cinque’s Generalization carries over to these verbal sequences: the allowed movements affect the Prt, or something properly containing it.

This is a challenge for the understanding proposed by Cinque, that his Generalization arises as a consequence of feature checking. The DP is considered to be the “extended projection” of the noun, in Grimshaw’s (1991, 2000) sense. Cinque suggests that the movements rearranging the noun are driven by licensing concerns; elements of the extended nominal projection must be licensed by establishing a relationship with the noun head, through agreement or movement.

But as Abels (2011) points out, the real generalization is, we might say, that the bottom of the tree is what moves. In particular, to get the verb cluster ordering facts to work out correctly, the particle *Prt* must be treated as the head of the extended clausal projection. That conflicts with the usual intuition, that the clause is a projection of the verb. Surely movement within verb clusters is not driven by the need to check *Prt* features? Abels accepts the implication that the particle is indeed a verbal head in the required sense.

Here again, I think we are missing the trees for the leaves. The important part of the generalization appears to be structural: movement affects the original bottom of the structure, the deepest part of the tree. If that can be explained for purely geometric reasons of tree-balancing (and it can), we need not invoke explanations in terms of the properties of the rearranged items themselves.⁸ Given some scheme for assigning identities to the terminal nodes, they will be “carried along for the ride”, reordered in some blindly structural way. Insofar as patterns of movement in different domains look alike, despite their different meanings, and moreover look like what concerns of tree-balancing would produce, we may suspect that the present account is on the right track.

⁸ It is worth pointing again to the proposal by Steddy & Samek-Lodovici (2011) in this regard; it is clear that their account would carry over directly to predicting this distribution of orders, given Align-left constraints targeting the relevant positions. Note however a kind of specificity: for their kind of account to go through, exactly these categories must be picked out for alignment optimization, not other categories interspersed in the hierarchy. The present account, while in principle allowing different distributions of orders for different structures, applies in general to any kind of tree structure. We expect parallel patterns of movement if parallel structures are affected (however, distinct structures may yield the same pattern of movements).

5.1.4. *On head-complement order and restrictiveness of analysis*

Abels & Neeleman (2009) propose an account of the DP ordering facts that permits both head-complement and complement-head order without movement, reviving the traditional “Head Directionality Parameter”. This follows a long tradition of assuming that the order in which sister nodes may be linearized can be parametrized (see Chomsky 1981, Stowell 1981, Travis 1984, Koopman 1984, among others). Given that assumption, only 6 of the 14 of the attested orders discussed by Cinque (2005) are necessarily derived by movement in their system; the remaining 8 reflect the base hierarchy with various choices of head-complement ordering (3 binary choices yield $2^3 = 8$ possibilities).

These authors argue that their own system is to be preferred, on the basis of restrictiveness. Despite the rigid specifier-head-complement ordering imposed by Kayne’s (1994) Linear Correspondence Axiom (LCA), in practice the analyses that exploit the LCA require prolific movement, but not too much: enough to derive all of the possibilities, but not so much that anything at all can be generated (by a series of leftward moves, one can derive any ordering at all). They argue that their own system is more restrictive, in that fewer movements are motivated, and in fewer orders.

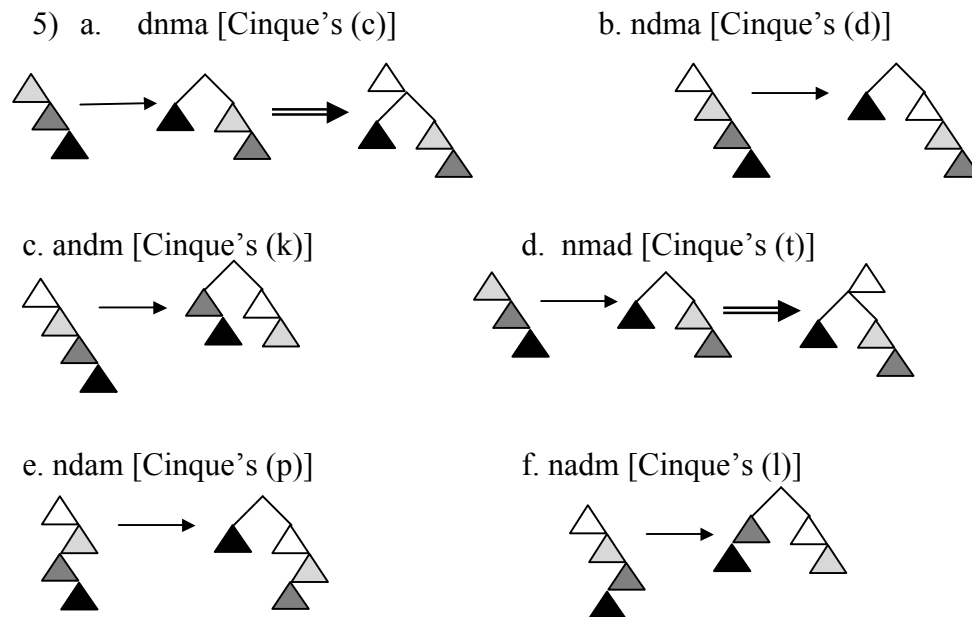
However, there are several intersecting issues here that must be teased apart. While leaving head-complement order free, as they do, makes available movement-eschewing analyses of certain orders, their argument that it restricts analyses does not seem to go through. If anything, the derivational possibilities that might be explored are multiplied: now reorderings can be achieved either through leftward movement, as for an LCA account, but also through free head-complement ordering. Put another way, all of the

derivations that might be countenanced under the LCA are available in principle under Abels & Neeleman's system, plus others besides (those furthermore exploiting complement-head order)

The assumption that is required for their account to be more restrictive is that analyses with fewer movements are to be preferred. But it is important to point out that the preferences that obtain for a theoretical linguist need not align with preferences in acquisition. That is, while preferring the simplest derivations makes the task easier for linguists, we should be cautious about attributing the same preference to children acquiring the language. They are not, after all, little linguists, building theories of language that are of a kind with those built by professional linguists. Rather, they are, by hypothesis, pursuing some biological process, unfolding according to its own design in conjunction with environmental input, which guides its development but perhaps not in the superficially simplest ways.

In terms of the present work, recall again the point that movement diagnoses structure: movement, on this account, is not just structure-dependent, in the traditional sense, but wholly structure-determined. Thus, the sparer invocation of movement by Abels & Neeleman in deriving the array of attested orders places weaker restrictions on the possible underlying structure. I have chosen to spend most of my effort on showing that the richer derivations Cinque proposes can all be made to work in terms of tree-balancing, mostly because that is harder to do, hence a more interesting result. But it is worth going through the effort of showing that the current account can succeed at the easier task of motivating the smaller set of movements entertained by Abels & Neeleman.

As Abels & Neeleman (2009) show, 8 of the 14 orders can be obtained without movement at all, simply by permuting choices for relative head-complement order among the elements of the base hierarchy. I show below only the 6 orders that crucially require movement under their account.⁹



These movements – really just three distinct movements, of NP to the edge of NumP or DemP, or of [Adj [NP]] to the edge of DemP – can readily be shown to follow from an appropriate tree. First, I show the conditions that must hold for the indicated movements to balance the tree:

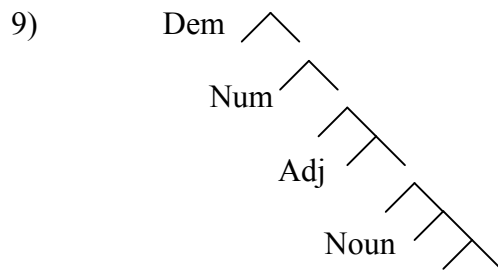
$$6) (n-1)(s+t-3) > a+m \quad \text{for (a), (d) above.}$$

⁹ I show the simplest, single-movement derivations, with accompanying tree diagrams shaded according to base depth (N is darkest, D lightest). As they point out, other orders might be derived by movement, but these six must be. Other structures, supporting different derivational spaces, could still map to this array. For example, the trees that support Cinque's derivations (see below) do not generate further orders under the assumption of free head-complement order. This turns Abels & Neelman's argument about restrictiveness of analysis on its head: depending on the shape of the base structure, the assumption of free head-complement order might allow more possible derivations corresponding to a given surface order than under an LCA-based account, presumably making the task of the child acquiring the language harder.

7) $(n-1)(s+t+u-4) > a+m+d-1$ for (b), (e) above.

8) $(a+n-2)(t+u-3) > d+m$ for (c), (f) above.

This small set of movements exhausts the permitted possibilities for a certain class of base trees. The following structure is an example of such a tree, for which all and only the three movements indicated above are motivated by tree-balancing. In this case, Dem and Num represent a single layer of embedding (e.g., a single functional head), while the Adj subtree contains two positions, and the Noun, four.



The following figure maps out the complete set of movements affecting the structure above that satisfy the Fundamental Movement Condition. In other words, it shows what we might think of as the “growth set”: all of the tree forms that are accessible via tree-balancing movements. In this figure, I show heads to the left of their complements for convenience; in the analysis of Abels & Neeleman (2009), the relative order of head and complement is left free, so in general each tree will correspond to multiple surface orders. Only the three movements Abels & Neeleman describe are found here; this means that, allowing free head-complement order, all and only the 14 orders documented by Cinque (2005) are “grown” by tree-balancing affecting this particular base structure.

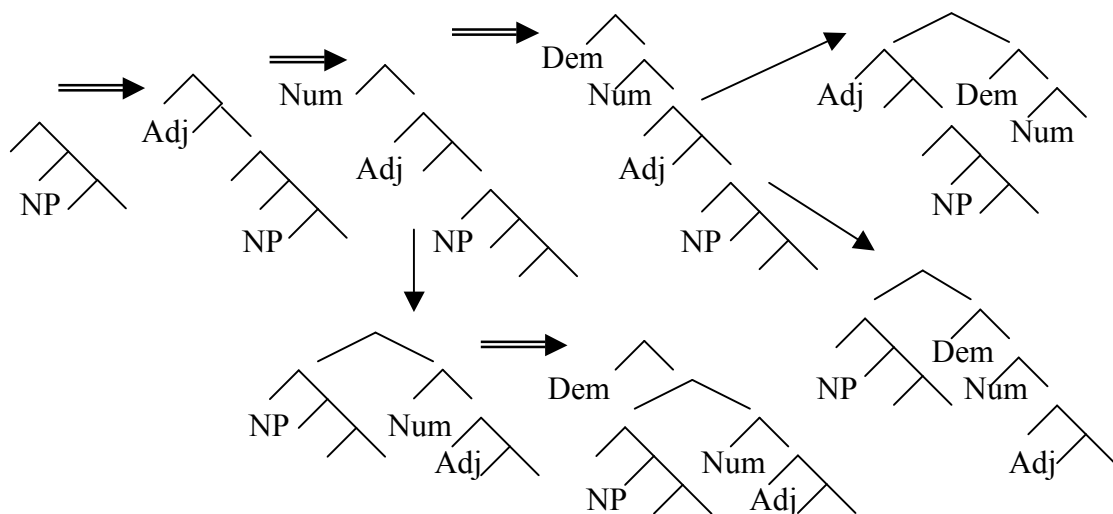


Figure 1: Derivations for minimal tree with free head-complement order.

All FMC-obeying derivations affecting the indicated structure. Only three movements are permitted (single arrows); double arrows indicate merge of the next higher category.

As indicated, it is rather straightforward to find such trees. In what follows, I take on the far more demanding task of showing that Cinque's orders can be "grown" as above from some suitable base structure, even with heads strictly preceding their complements, as assumed by Cinque (2005), following Kayne (1994).

Given that both flavors of analysis can be accommodated under this account, it seems that the present theory does not stand or fall on the basis of assumptions about the linearization of heads and complements. The crucial evidence will come from cartography, where the universal base structure as revealed in that study either will or will not be one of the tree forms supporting a tree-balancing account of nominal ordering. If the cartographers' tree matches that shown above (or another form in the same class), then we can maintain a tree-balancing approach to movement only in conjunction with the assumption that head-complement order is not universally fixed. On the other hand, if the real DP tree happens to be one of those that supports the richer array of movements

required to generate the DP typology with strict head-complement order, then a tree-balancing account is viable under either strict head-complement ordering or free ordering. That is, head-complement reorderings of the structures derived below for the LCA-compliant derivations do not yield unattested orders; only remnant movement can do that.

5.2. *Assumptions and methodology*

As a first foray into this domain, I have chosen to pursue the strongest interpretation of this treatment of movement. Specifically, following much cartographic work, I assume that all languages share a common universal “base” syntactic hierarchy¹⁰. I furthermore assume that every instance of movement must instantaneously reduce the number of c-command relations present in the tree.

The account I will suggest is something like a constrained Move Alpha (cf Lasnik & Saito 1992), recalling the anything-goes conception of movement of Government & Binding Theory: any movement that improves tree balance is possible. By possible, I mean that it is an option entertained by the human mind, a possible learnable language.¹¹

I make the strong simplifying assumption that all possible orders are attested, and show

¹⁰ That is not the only tree-balancing analysis of variation in DP order one might consider. Another possibility is that cross-linguistic differences in movement can be tied to cross-linguistic differences in the amount of structure present (much early cartographic work pursued this idea, and see the discussion of structural variation in Chapters 3 and 4). But I reject that option here for the simple reason that it is harder to falsify than the uniform-base alternative.

¹¹ A word about acquisition here: the account is cast in synchronic terms of a fully-formed grammar. But of course children must learn the particular language of their environment, which in present terms reduces to deducing the movements that have transformed some base structure, available to them in advance of experience in some way. Insofar as movement is optimized in some sense, the structural analyses available are correspondingly reduced. Moreover, we may expect that variation – observing a spectrum of possible orderings within a single language – may be crucial to converging on the proper derivation, with alternatives representing “nearby” options in the derivational space. See, in this regard, the comments on the range of variation of verbal orderings in Germanic varieties, and the striking implicational relations that hold at the construction level. I leave the exploration of those facts to future research.

that the observed array of orders can follow from an appropriate base geometry transformed to improve tree balance.

5.2.1 *Analytical assumptions*

I follow Cinque (2005) in assuming LCA-based linearization (contra Abels & Neeleman (2009), who treat the same ordering facts with underdetermined linearization). Cinque's analysis makes for stronger, more easily falsifiable conditions, since it has more movements.¹² Note, incidentally, that the present account, if correct, affords us a ready reply to Abels & Neeleman's point that accounts like Cinque's LCA-based system overgenerate. Treating movement as tree-balancing allows us to rule in all attested orders and rule out all unattested orders in purely structural terms, as I show. In this formulation, an elaboration of Cinque's LCA-based system with the hypothesis that movement must obey the Fundamental Movement Condition

Simplifying somewhat for tractability, I assume that the only possible intermediate landing sites occur at the boundaries of the overt categories (e.g., movement may deposit the N subtree between M and A, but not 'in the middle' of M). I do not make any *a priori* assumptions about how much tree structure is present corresponding to each overt category. Instead, the goal is to use the observed patterns of movement to find out how much structure is present. At the very least, we can discover whether an account along

¹² “[...] Cinque's theory requires movement in 13 of the 14 licit derivations, while our alternative does so only in six. In each of those no more than a single movement is required, while Cinque's derivations require up to three movements.” (Abels & Neeleman 2009: 67) That may well be an advantage of their theory, but not from the present perspective, where the more movements, the tighter the restrictions on a possible base tree which would be balanced by the movements underlying all of the attested, but none of the unattested orders.

these lines is even tenable: Can the conditions required to motivate all attested orders, and to rule out all unattested orders, be simultaneously satisfied by a single base structure? If so, (and, crucially, if the required conditions actually hold of the real DP hierarchy), a very strong tree-balancing account of syntactic movement in this domain is supported.

A crucial assumption here is that movement is always optional.¹³ Then any intermediate derivational stage could survive to the surface, and so we may rule out any that would lead to unattested orders without further movement (e.g. AMN). I also make the simplifying assumption that there are no ‘accidental gaps’ in the cross-linguistic data, i.e. no unattested orders that would arise through motivated movements. Unattested orders are assumed to be ‘actively’ ruled out, in the sense that I assume they do NOT arise through motivated movements (at least one step in the derivation of such an order must fail to improve tree balance/reduce c-command and containment totals). The list below summarizes these assumptions:

- 10) a. Universality: the DP orders of every language represent transformations of a single, common base structure.
- b. Coherence: Take the D, M, A, N to correspond to coherent partitions of the tree that cannot be broken apart further by movement.
- c. Monotonicity: All movement within the DP must satisfy the FMC.
- d. Continuity: Every monotonic derivation corresponds to an attested order.

¹³ This requires some clarification: the optionality is at the level of choosing among languages (rather, the derivations instantiated in a particular language). Within an individual language, of course, the pattern of movements is far more constrained – a separate matter from the present cross-linguistic investigation.

Within these rather rigid boundaries, the task is to find a shape of base tree (taking the base order to be fixed $D > M > A > N$) that can fit the typological pattern. It is clear that not just any imaginable typology (construed as a choice of some set of attested orders from the 24 logically possible relative orders of four elements) could be described in this way. Consider the trivial counterexample below; this typology of two attested forms (mirror images of each other) does not admit a tree-balancing account:

11) {DMAN, NAMD}

If the set of attested languages consisted of just two forms, mirror images of each other, a tree-balancing account satisfying all four conditions would be impossible. This is so because the mirror ordering arises through no less than three movements. Then, under (10c), derivations with just one or two of these three movements would be predicted to survive to the surface as attested orders in some languages. An example of a typology that could include the mirror order, and at least in principle prove compatible with a tree-balancing account, is given below:

12) {DMAN, DMNA, DNAM, NAMD}

Here, the intermediate configurations in a snowballing derivation of the mirror order need not experience further movement. That is, the first step of snowballing movement inverts AN to NA; that can be embedded without further movement, producing DMNA. Likewise, the second step turns MNA to NAM; this step need not be succeeded by the final inversion, instead surviving as DNAM. It remains to check (i) & (ii), but at least the surface typology does not reveal gaps corresponding to hypothesized derivational stages.

5.2.2 *Philosophical background*

The present account recalls Boeckx's (2008) arguments for a parameter-free UG. There is variation in this system, but it follows from under- rather than over-specification (a la GB's parameters) of the linguistic faculty. We are seeing the stochastic action of a single beast, a uniform but unstable stack which may fold in a number of complicated ways. The conditions here are taken to hold at a rather abstract ontological level of choosing among possible ("downhill") and impossible ("uphill") derivations, or equivalently attested surface forms in the cross-linguistic data.

On this view, syntactic movement has the flavor of a naturalistic process like an avalanche. Consider, say, pouring sand grains into a pile (see, e.g., Ball 1999). The grains will accumulate until a critical slope angle is reached, and at some point thereafter as more sand is poured on, an avalanche may be triggered, always ending with a pile with a lower slope than before the avalanche (a more stable form, with less gravitational potential energy). In our analogy, the accumulation of structure is provided by Merge, and once the conditions for movement can be met (the analogue of the critical slope angle), the tree may 'avalanche' into a more balanced form (here c-command totals stand in as the analogue of 'potential energy').

5.2.3 *Why look at DP orders?*

The DP makes an ideal choice for this investigation, on several grounds. First, there is a large amount of literature available surveying the cross-linguistic evidence, with a reasonably clear view of what orderings are and are not attested in the world's languages.

Thus, the empirical terrain is rather well mapped out. Second, focusing on just the relative orders of four elements (demonstrative, numeral, adjective, noun) constrains and simplifies the analytical problem, while drawing also on the body of literature following up on and revising Greenberg's (1963) Universal 20.





Finally, there is a theory-internal motivation for this choice: with respect to the phase architecture (Chomsky 2000 *et seq.*), DPs are effectively at the “bottom” of the tree, such that we are seeing the syntactic system operating without phase effects muddying the picture¹⁴. By focusing on the DP we “factor out” phase considerations, allowing us to set aside difficult questions about what happens at phase boundaries. Is the spell-out domain of completed phases forgotten immediately (Chomsky 2000), or does it remain accessible during the computation of the next phase (Chomsky 2001)? Because DPs typically do not contain further sub-phases (but see fn. 2), we can avoid answering these issues. The ordering patterns observed within DP should give us the clearest evidence of (whether, and) how structure influences movement, free from phase-based complications that would arise in treating ordering within *v*P or CP.

5.2.4 *Inferring movements from surface orders*

Leaving the details to Appendix A, Figure 2 below illustrates the derivations I take to underlie the attested DP orders. In order to fit all of the trees into the diagram, I adopt the expedient of shading the subtrees according to their base depth: nouns are black,

¹⁴ Except in cases where the DP contains, say, a relative clause, a possibility I ignore. I likewise abstract away from possessor constructions, and other recursion within the DP. Surely that is an option instantiated in natural language, but the intuition is that such DP-internal recursion is not typical, nor relevant at the level of the present considerations.

adjectives, dark grey; numerals, light grey; and demonstratives, white. Example (3) below illustrates, with the tree for the base order DMAN, corresponding to a derivation without movement.

13)		<u>Total nodes:</u>	<u>Depth:</u>
<u>Dem</u>		<i>d</i>	<i>u</i>
Nu <u>M</u>		<i>m</i>	<i>t</i>
<u>A</u> dj		<i>a</i>	<i>s</i>
<u>N</u> oun		<i>n</i>	--

For each of the subtrees represented by a triangle above (D, M, A, N), we need a variable to represent the total number of nodes, here *d*, *m*, *a*, *n*. For each category other than N, we also need to track “spinal depth”, i.e. the depth of embedding within that category of the next lower category; for *d*, *m*, *a* these are *u*, *t*, *s*, respectively.

In Figure 4, derivational pathways are indicated with arrows. Large, double arrows represent (external) Merge operations; smaller arrows indicate Move (internal Merge). Note that some surface orders (notably (x), NAMD) have several possible derivations in this figure. For such orders, I insist that at least one derivation be motivated, in present terms. That is, I remain agnostic on exactly which syntactic configuration(s) are actually present corresponding to these orders.

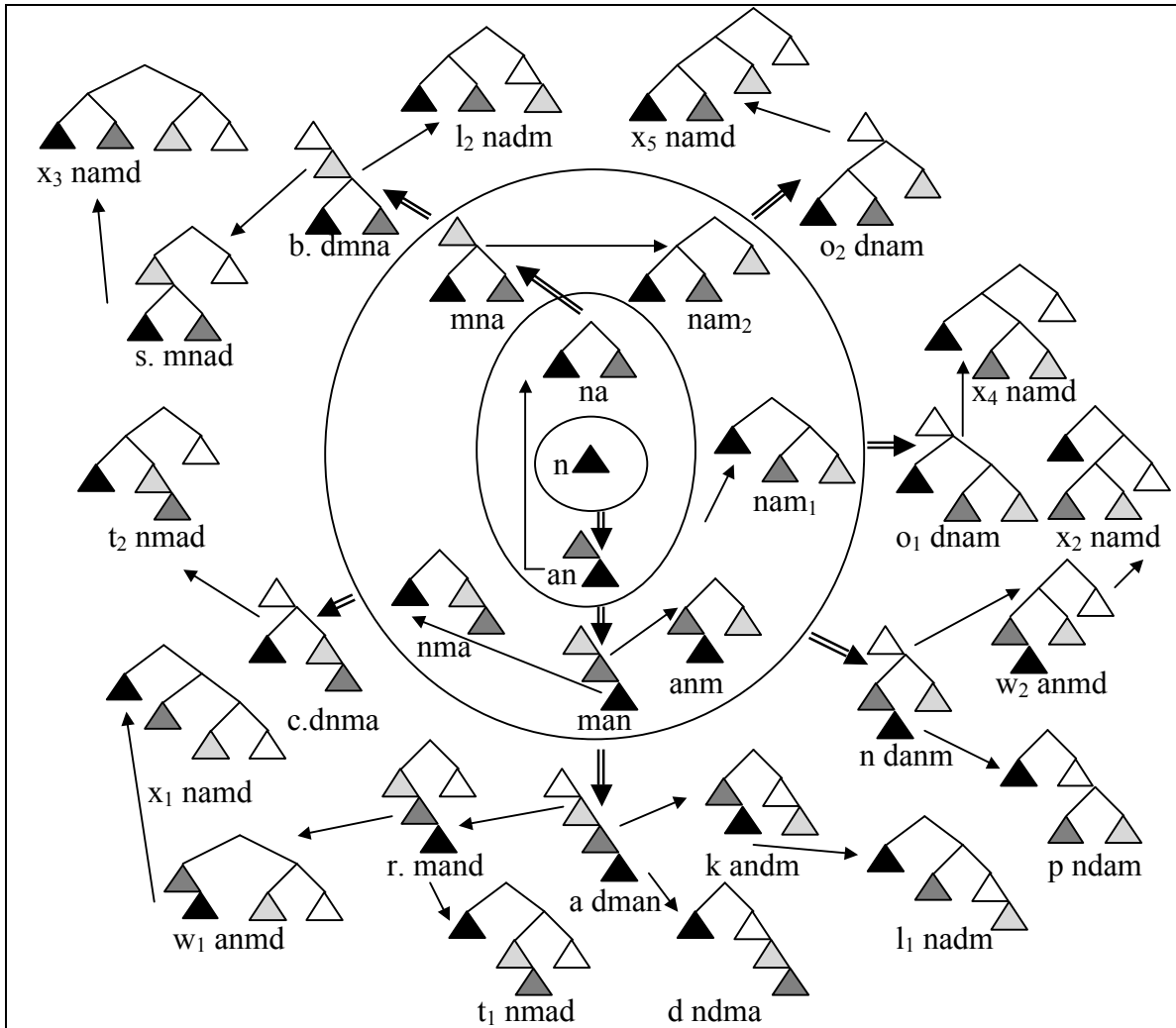


Figure 2: Derivations of attested orders considered in this work.

I leave the detailed treatment of unattested orders to the Appendix, simply noting here that they require more careful consideration than the attested orders. This is so because rather than trying to rule in at least one derivation, as for the attested orders, for the unattested orders we must rule out all derivations. For attested orders we need consider only ‘locally best’ derivations; for the unattested orders we must, in principle, consider more ‘exotic’, sub-optimal derivations (though even here we can reduce the combinatoric complexity significantly).

For each derivation above, we can apply the Fundamental Movement Condition to each movement involved, yielding a new inequality that must be satisfied. Thus, each order induces a set of restrictions on the possible underlying base tree.

5.2.5 The DP Condition

The goal here is to discover whether there can be a single base DP tree, such that all attested orders, and no unattested orders, arise through movements that reduce c-command totals at the point where they apply (and what such a tree must look like). As we have seen, some orders can be derived in several distinct ways. In such cases, if the order in question is attested, I assume at least one of its derivations of must monotonically reduce c-command totals. For unattested orders, I assume that none of the pathways to that order satisfy that condition. Conjoining (or disjoining, where appropriate), we obtain the complicated condition below (see Appendix A for details).¹⁵

$$\begin{aligned}
 14) & ((a+1 < (s-2)(n-1)) \& (m+a < (s+t+3)(n-1)) \& (a+m+d-1 < (s+t+u-4)(n-1)) \& (m+d < \\
 & (t+u-3)(a+n-2)) \& ((a+m+d+1 < (s-1)(n-1)) \mid (m+d < (t+u-3)(n+a))) \& (m+1 < (t-2) \\
 & (a+n-2)) \& ((a+m+2 < (s-1)(n-1)) \mid (m+1 < (t-2)(n+a))) \& (a+m+d+1 < (s+u-2)(n-1)) \\
 & \& (d+1 < (u-2)(a+n+m-3)) \& (d+1 < (u-2)(a+n+m-1)) \& ((a+m+d-1 < (s+t-2)(n-1)) \mid \\
 & (d+1 < (u(a-1)+(u-2)(n+m))) \& ((m+d+2 < (t-1)(a+n-2)) \mid (d+1 < (u(n-1)+(u-2)(a+m))) \\
 & \& (((m+d+2 < (t-1)(a+n-2)) \& (a+m+d+3 < (s-1)(n-1))) \mid ((d+1 < (u(n-1)+(u-2)(a+m)) \\
 & \& (a+m+d+3 < (s-2)(n-1))) \mid ((d+1 < (u-2)(a+n+m-1)) \& (m+d+2 < (t-1)(a+n))) \mid ((a+ \\
 & m+2 < (s-1)(n-1) \& (d+1 < (u-2)(a+n+m+1))) \mid ((m+1 < (t-2)(n+a)) \& (d+1 < (u-2) \\
 & (a+n+m+1)))) \& (2(a+n)+d+2 > (u-1)(m-1)) \& (a+n+d+4 > (u-1)(m-1)) \& (a+n+d \\
 & +4 > (u(m-1)) \& (n+m+d+2 > (t+u-2)(a-1)) \& (n+m+d+4 > (u(a-1)) \& (n+m+3 > (t-1) \\
 & (a-1)) \& (n+d+3 > (u-1)(a+m-2)) \& (n+d+3 > (u-1)(a+m)) \& \neg((d+m > a+n+4) \& \\
 & ((m+n+a-3)(u-2) > d+1) \& ((a+n-2)(t-1) > d+m+2) \& ((n-1)(s-1) > d+m+a+3)) \&
 \end{aligned}$$

¹⁵ To be clear, the DP Condition is not to be understood as a proposal about the content of Universal Grammar; i.e., I am not claiming that something as complicated as (14) is directly encoded as part of a native speaker's knowledge of language. Instead, the condition is an expression of the "structural diagnosis" obtained by supposing that all movements in the DP improve tree balance in a single, common base. The base must meet the conditions in (14) for this to hold. Thus, (14) is an artifact of linguistic analysis, a way to go from observed surface orders to underlying structure and derivations.

$$\begin{aligned}
& \neg(((m-1)3 > d+a+n+7) \mid ((a-1)2 > d+m+n+7) \mid ((m+1)2 > d+a+n+5) \mid (a+m+2 > n+d \\
& +4) \mid ((n-1) > d+m+a+7) \ \& \ (((a+m+2)2 > d+n+6) \mid ((m+1)3 > d+a+n+7) \mid ((m-1)4 > \\
& d+a+n+9)))) \ \& \ ((d+1)2 > n+a+m+5) \ \& \ ((m+n+a-3)(u-2) > d+1) \ \& \ ((a+n-2)(t-1) > \\
& d+m+2) \ \& \ ((n-1)(s-1) > d+m+a+3)) \ \& \ \neg(((n-1)(s-1) > a+m+2) \ \& \ ((n+a-2)(u-2) > \\
& m+1) \ \& \ (((a+m)(u-1) > d+n+2) \mid ((m-1)u > d+a+n+3) \mid ((a-1)u > d+m+n+3) \mid (((n-1) \\
& (u-1) > d+m+a+3) \ \& \ (((a+m+2)(u-1) > d+n+2) \mid ((a+m)u > d+n+4) \mid ((m-1)(u+1) > \\
& d+a+n+5)))))) \ \& \ \neg((a+m > d+n+4) \ \& \ ((n-1)s > d+m+a+3) \ \& \ ((n+a+m-1)(u-2) > d+1) \\
& \ \& \ ((n+a-2)(u-2) > m+1)) \ \& \ \neg(((n+a-2)(u-2) > m+1) \ \& \ (((m-1)(u-1) > d+a+n+1) \mid \\
& (((n-1)(s+u-2) > d+a+m+1) \ \& \ (((a+m)(u-1) > d+n+2) \mid ((m-1)u > d+a+n+3)))))) \ \& \ \\
& \neg(((d+m > n+a+4) \ \& \ ((n-1)(s-1) > d+m+a+3) \ \& \ ((a+n-2)(u-1) > d+m+2) \ \& \ ((n+a-2) \\
& (u-2) > m+1)) \ \& \ ((a+1 > d+m+n+5) \mid ((a-1)2 > d+m+n+7) \mid ((m+1)(u-1) > d+a+n+5) \mid \\
& ((m-1)u > d+a+n+7))) \ \& \ \neg(((n-1)(s+t-3) > a+m) \ \& \ (((a+m-2)(u-1) > d+n+2) \mid ((a-1) \\
& (t+u-2) > d+m+n+1))) \ \& \ \neg((((n-1)(t-1) > a+m+2) \ \& \ ((n-1)(s-2) > a+1)) \ \& \ (((a+m)(u-1) \\
& > d+n+2) \mid ((a+1)(t+u-2) > d+m+n+1) \mid ((a-1)(t-1) > m+n+2))) \ \& \ \neg(((a-1)(t-1) > \\
& m+n+2) \ \& \ ((n-1)(s-2) > a+1)) \ \& \ \neg(((n-1)(s-2) > a+1) \ \& \ (((a-1)(t+u-2) > d+m+n+1) \mid \\
& ((a-1)t > d+m+n+3))) \ \& \ \neg(((n-1)(s-2) > a+1) \ \& \ ((n+a)(t-2) > m+1) \ \& \ (((m-1)(u-1) > \\
& d+a+n+3) \mid ((a-1)u > d+m+n+3) \mid (((n-1)u > d+m+a+3) \ \& \ (((a+m+2)(u-1) > d+n+2) \mid \\
& ((m-1)u > d+a+n+5) \mid ((a+1)u > d+m+n+3) \mid ((a-1)(u+1) > d+m+n+5)))))) \ \& \ \neg(((m-1) \\
& (u-1) > d+a+n+5) \ \& \ (d+m-2 > n+a-4) \ \& \ ((n-1)(s-1) > d+m+a+1) \ \& \ ((a+n)(t+u-3) > \\
& d+m)) \ \& \ \neg((n-1 > a+m-4) \ \& \ ((n+a)(t-2) > m+1) \ \& \ ((n-1)(s-2) > a+1) \ \& \ (((a+m-2)(u-1) \\
& > d+n+2) \mid ((m-1)u > d+n+a+5) \mid ((a+1)u > d+m+n+3) \mid ((a-1)(u+1) > d+m+n+5) \mid \\
& ((n-1 > d+m+a+7) \ \& \ ((n+a+m+3)(u-2) > d+1) \ \& \ ((a+m+2)2 > d+n+6)) \mid (((n-1)(u-1) > \\
& d+m+a+5) \ \& \ (((a+m+4)(u-1) > d+n+2) \mid ((a+m)u > d+n+4) \mid ((m-1)(u+1) > d+m+n+7) \\
&)))) \ \& \ \neg((a+m+2 > d+n+4) \ \& \ ((n-1)2 > d+m+a+5) \ \& \ ((n+a+m+1)(u-2) > d+1) \ \& \ \\
& ((n+a)(t-2) > m+1) \ \& \ ((n-1)(s-2) > a+1)) \ \& \ \neg(((a+n-2)(t-2) > m+1) \ \& \ ((n-1)(s-1) > \\
& m+3) \ \& \ ((n-1)(u-1) > d+m+a+3) \ \& \ (((a+m+2)(u-1) > n+d+2) \mid ((a+m)u > n+d+4))) \ \& \ \\
& \neg(((a+1)(t-1) > n+m+2) \ \& \ ((n-1)(t-1) > m+a+2) \ \& \ ((n-1)(s-2) > a+1) \ \& \ (((a+1)(u-1) \\
& > d+m+n+3) \mid ((a-1)u > d+m+n+5) \mid ((n+m)(u-1) > d+a+4) \mid ((m-1)u > d+a+n+5))))
\end{aligned}$$

The payoff of going through all this work is that now we can simply “plug in” to this formula choices of values for the variables n , a , m , etc. If the condition holds for a particular choice of values, a base DP tree with those structural parameters would be “improved” by each step of movement in each attested order, but would be “worsened” (or at least, not improved) by at least one step in the derivation of each unattested order.

It is not at all obvious that the set of “solutions” to (14) above should be non-empty. Indeed, if the present account turns out to be misguided, and syntactic movement

in fact has nothing to do with tree-balancing, it would be quite surprising to find that a tree-balancing account of the facts is actually possible; that would seem a remarkable coincidence. So finding that the solution set is non-empty is a non-trivial result, and by itself lends some plausibility to the basic approach adopted here.

In practice, the conditions I derive are extremely complicated, far too complex to explore by hand. To investigate this matter, I have written a computer program to check which tree shapes would motivate the movements deriving all attested orders, and no unattested ones, in terms of c-command reduction. I include a table of results from this program, and the code itself, in Appendix A.

5.2.6. Direct demonstration for smallest solution tree

In this section, I demonstrate by brute force that this analysis succeeds. For the smallest overall tree consistent with the algebraic DP Condition (14) above, I explicitly show all movements respecting Coherence and satisfying the FMC. As claimed, all and only the attested orders are produced.

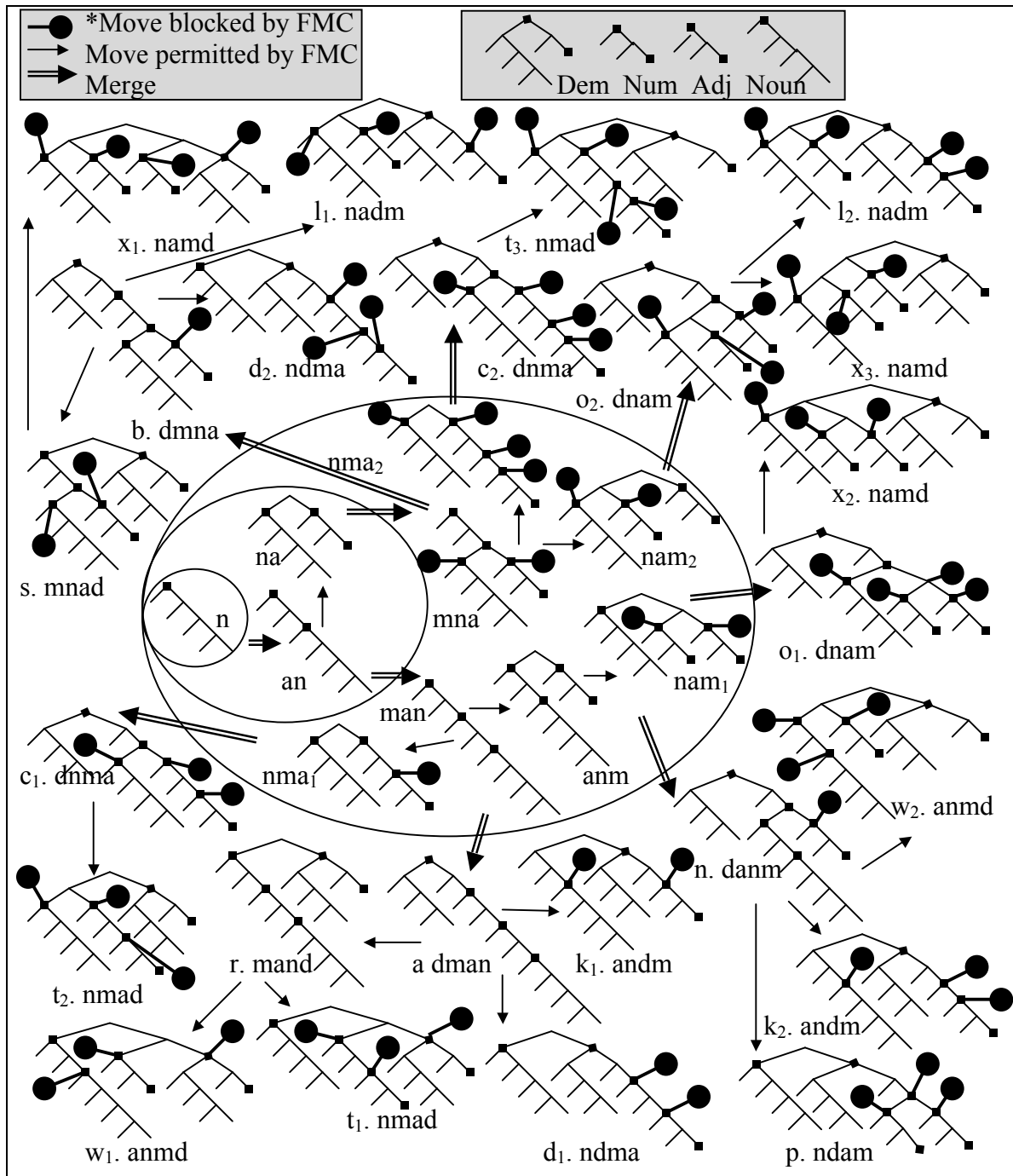


Figure 3: Derivations for minimal tree.

All movements respecting coherence are considered: the result is shown here if the movement obeys the FMC; non-FMC-obeying movements are marked with black circles.

5.3 Overview of numerical results

In this section, I summarize the numerical results provided in Appendix A.6, obtained by running the program in Appendix A.7. The program runs through all permutations of possible values for the node counts and depths of the D, M, A, N subtrees, within limits input by the user, and ‘plugs in’ those values to the DP condition, recording which values satisfy it. In other words, the program is designed to find and describe the possible base DP trees for which tree-balancing motivates all the attested orders seen in figure 1, and none of the unattested orders. Leaving a list of explicit solutions to later, here I draw some broad conclusions about the permitted shape of the base DP tree.

5.3.1 Antilocality

The upper regions of the tree (D, M, A, containing the demonstrative, numeral, and adjective) must each have a spinal depth of at least 3 (and hence, at least 5 nodes). This follows independently from the version of Antilocality predicted by this theory (see chapter 6). That is, we observe movements in the DP which “skip” just one of these categories (e.g., carrying the MAN complex to the left of D, yielding MAND order from base DMAN). Recall the Fundamental Movement Condition (FMC), repeated below:

15) Fundamental Movement Condition (FMC):

$$\text{Move } \alpha \text{ only if } (a-1)(s-2) > b+1$$

$(a = \text{nodes in } \alpha, b = \text{nodes in } \beta, \text{ and } s = \text{depth of } \alpha \text{ in } \beta)$

Here, we can see immediately that s , the spinal depth of the skipped-over category, must be at least 3 to obtain a positive value on the greater, left hand side of the inequality.

5.3.2 *N is big*

It is also evident from the table that there are many solutions when the value of n (the size of N , the region containing NP) is very large (notice that the term n appears on the larger, right hand side of each of the inequalities above). This also makes sense in light of the FMC, and the observation that the only movements we seem to find move the NP , or something containing it. Then the factor a in the FMC (the size of the moving category) includes n as a term; the larger a is, the easier it is to satisfy the FMC. We see that apparently 9 is the minimum possible value for the number of nodes in N .

5.3.3 *D can be big too, and bushy*

That said, it is not the case that N must have the highest node count of any of the tree regions; region D , in particular, can be bigger. The table of results reveals further subtleties: region D also permits very “bushy” structures, something of a surprise. Consider again the FMC; for $s \geq 3$, we can divide to isolate the a term on the left hand side of the inequality, as below:

$$16) \quad a-1 > (b+1)/(s-2) \quad (\text{for } s \geq 3)$$

For sufficiently large β , the right hand side approaches b/s . Note that b and s are both structural properties of β (its size and depth, respectively); b/s may be described as something like a “bushiness factor” of β . In general, for fixed moving category α , the less bushy the region it moves across, the better. It is surprising, then, to find a large but shallow structure (D) at the top of the tree permitting movement to cross it. It is precisely

the value of this numerical exploration to bring to light surprising conclusions like this, where intuitions based on broad predictions might lead us astray.

5.3.4 Spinal cartography

A number of considerations suggest that we should pay particular attention to base structures that are spines. In fact, this characterization of structure is strongly motivated at a derivational level, as argued for instance by Uriagereka (1999) and Narita (2010). For those authors, what appears on the surface as a “specifier” or complex left branch is treated, within a structure that embeds it, as effectively a single terminal object. Moreover, the same conclusion is virtually forced by the particular articulation of cartography that underpins the typological investigation in this chapter.

It is hardly disputable that the individual positions within a cartographic hierarchy can be filled by more or less “internal” material. So, for example, an adverbial position may be filled by a single characteristic adverb, or the same adverb with further modification (e.g., an intensifier). By hypothesis, the same position might host a prepositional phrase; very often, it will receive a default interpretation with no overt expression at all. If such variation within embedded (left) branches is visible to the embedding cartography, movement triggered by structure should be chaotically unpredictable, depending on fine details of individual expressions to such a degree that slightly different structures would be likely to produce wildly different results. That, patently, is not the general rule.

However, if we adopt the arguments put forward by Uriagereka, Narita, and others, then in fact we expect that the cartography will look the same regardless of the “size” of objects that are embedded within that fixed hierarchy. In fact, uniformity of structure is complete if we adopt the further claims of Starke (2004) and Jayaseelan (2008), among others, who take the apparent complementary distribution of overt heads and specifiers (cf. Koopman’s (1996) Generalized Doubly Filled Comp Filter) to indicate that specifiers just are heads, in effect. In that case, the syntactic structure is effectively identical whether the relevant position is “discharged” by a complex object (phrase) or a simple one (a head, necessarily present syntactically but perhaps without any overt reflex).

The crucial exception to this view of strictly spinal structure is induced by movement. In the present work I suppose that complex left branches formed by movement remain visible, at least within the phase in which they move. We can motivate this view of things derivationally. Recall Uriagereka’s (1999) explicitly derivational concerns in proposing Multiple Spell-Out: a complex left branch is built in a separate workspace, by hypothesis inaccessible to the right-branching “derivational cascade” that embeds it. However, movement within a single right-branching cascade does not go beyond a single workspace: the relevant left-branch structure was not assembled in a distinct cascade; it is built as a part of the same cascade within which it moves.

If we consider only spines as possible base structures, we arrive at a far more restrictive description of the base. Explicitly, using the program in the Appendix, the following Spines are detected as solutions to the DP condition within the following parameters: the noun may have up to 31 nodes, and Dem, Num, and Adj may each have

up to 11 nodes, in any configuration. Thinking in terms of terminal positions along the spine, we are considering a range of structures with up to 16 positions within the noun, and up to 5 positions within each of the other categories. Here are the distinct spinal structures satisfying this condition, within this size range:

17) Maximum values for variables $\langle n, a/s, m/t, d/u \rangle = \langle 31, 11/6, 11/6, 11/6 \rangle$

$n = 21, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$

$n = 23, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$

$n = 25, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$

$n = 27, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$

$n = 27, a = 5, s = 3, m = 7, t = 4, d = 5, u = 3$

$n = 27, a = 7, s = 4, m = 5, t = 3, d = 5, u = 3$

$n = 29, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$

$n = 29, a = 5, s = 3, m = 7, t = 4, d = 5, u = 3$

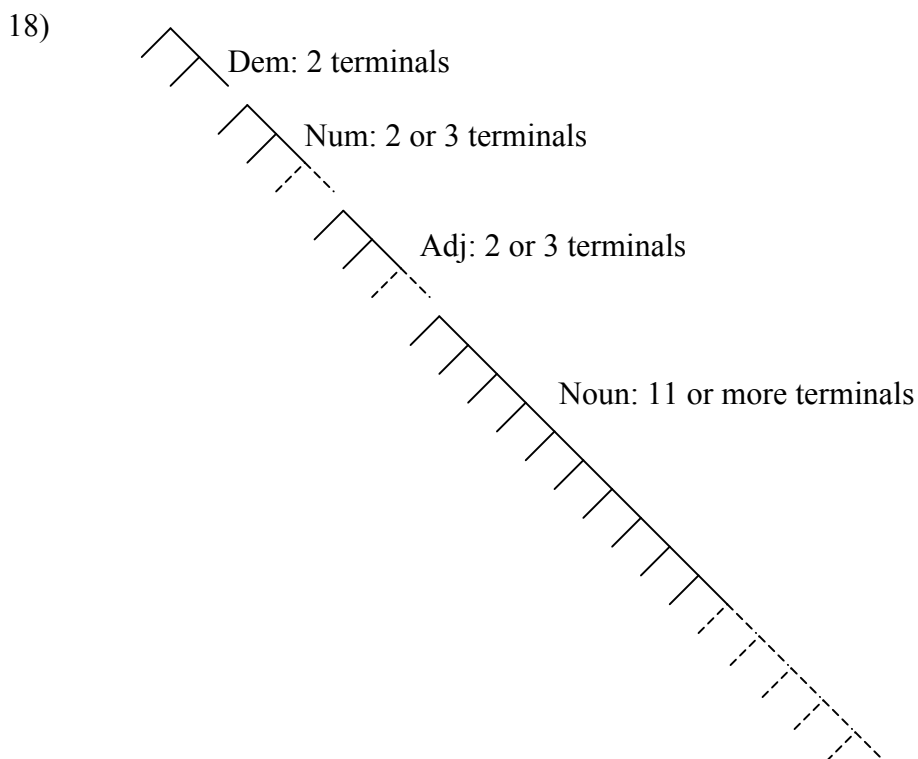
$n = 29, a = 7, s = 4, m = 5, t = 3, d = 5, u = 3$

$n = 31, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$

$n = 31, a = 5, s = 3, m = 7, t = 4, d = 5, u = 3$

$n = 31, a = 7, s = 4, m = 5, t = 3, d = 5, u = 3$

We can represent this as below:



The trend here is fairly clear. We can foresee that as the ‘beast in the basement’ (the Noun region) is allowed to be larger, slightly larger Num and Adj regions become possible, and eventually Dem as well may contain 3 terminals, though the number of nodes must become quite large to push it beyond that.

5.4. Discussion

In this section, I compare the results obtained numerically from the present account with some recent cartographic proposals. Consider the following hierarchies inferred for the DP, by Cinque (2005) and Svenonius (2008):¹⁶

19) [Q_{univ} . . . [Dem . . . [Num_{ord} . . . [RC . . . [Num_{card} . . . [Cl . . . [A . . . NP]]]]]]]
(Cinque 2005: 328, his (11))¹⁷

20) Dem > Art > Num > UNIT > Pl/SORT > Adj > *n* > N
(Svenonius 2008: 27, his (19))¹⁸

Leaving aside the structure internal to the part of the tree labeled ‘N’ above, these are, in fact, generally consistent with the structural predictions made here. In particular, note that the Antilocality requirement, a fundamental prediction of the present approach, is confirmed by these authors, who intersperse other categories between D, M, A, and N. That much extra structure insures that the D, M, A, and N subtrees, in present terms, do

¹⁶ See also Cinque (2004), Scott (2002), among many others.

¹⁷ This includes positions for universal quantifiers (Q_{univ}), ordinal and cardinal numerals (Num_{ord}, Num_{card}), relative clauses (RC), and classifiers (Cl). Cinque also proposes to have additional Agreement phrases interspersed among the categories, in part for theory-internal reasons, though he also notes (Cinque 2005: 321-322, fn 24), following Shlonsky (2004), some evidence for real agreement in these positions.

¹⁸ This adds, to demonstrative, numeral, adjective, and noun, positions for articles (Art), and numeral classifiers (UNIT), plural markers (Pl) or sortal classifiers (SORT), and noun classifiers, identified with little *n* (Svenonius 2008: 23).

indeed have spinal depth at least 3, as predicted (see 5.3.1).

Recall from 5.3.2 that a basic result of the present account is that region N, containing the NP, must contain at least 9 nodes. What evidence is there that NP is internally complex? I can think of a number of concerns that suggest this conclusion. Apart from some degree of functional structure present there including a “little *n*” category-determining functional head as well as the noun root itself (Marantz 1997), the noun itself presumably has a large feature structure drawn from the lexicon: phi-features (including at least inherent gender or class), a mark distinguishing count nouns from mass nouns, as well as any inflectional or derivational morphology on the noun head.¹⁹ Svenonius (2008) argues that there is an attachment point for idiomatic adjectives below *nP*, thus within N in our terms. Note further that NP seems to be a crucial “recursion point”, allowing introduction of a possessor in Spec, NP, and a (typically prepositional) complement as well,²⁰ and the noun could itself be a several-stage compound noun, such that at the very least it has a large potential size.

¹⁹ Note that, for our purposes, such morphology may well reflect further layers of syntactic structure, so long as the generalization holds that any such morphology on the noun head “comes along for the ride” in the relevant NP movement. This is true, at least for the cases I am aware of.

²⁰ Though Cinque (2005: 327, fn 34) points out that complements of N are typically stranded by movements of the NP within DP. Kayne (2000) suggests that these PPs are not true complements, but are Merged higher than N. This fact is somewhat surprising from the present point of view, as we would generally predict that another DP within a DP, at least doubling its node count, ought to be carried along for the ride for optimal tree-balancing. But if either DPs and/or PPs are phases, as suggested in much recent work, then such structure may be effectively invisible to the higher DP, in which case its movement would be irrelevant for c-command minimization at that level. I leave the issue aside here; it was precisely the ability to factor out phase effects that made the DP an appealing choice for the present analysis.

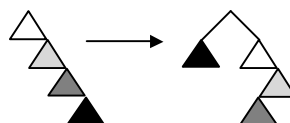
5.4.1 *On order (p) NDAM*

As mentioned above, I have sided with Abels & Neeleman (2009) in taking the order (p) NDAM to be real; Cinque (2005: 323, fn 27) includes it among the attested orders but suggests the order may be “spurious”. The issue is that, given Cinque’s (and my) assumptions, (p) is derived through subextraction: A and N move together to the left of M, then N moves by itself to the left of D, stranding A. This is troubling from the point of view of accounting for movement with invariant principles, since such movement is not otherwise well-attested, as Abels & Neeleman note, citing Postal (1972). Thus, Cinque’s account seems to undergenerate (if indeed (p) is real).

It is interesting to note, in this regard, that almost all possible base DP trees, which would be balanced by the movement in all attested orders excluding (p), are also balanced by the movements involved in deriving (p). To put it another way, the present treatment of movement predicts, just on the basis of the other attested orders, that order (p) should be ruled in as well. That seems to be the correct result.

Note, finally, that this order requires, within an LCA-based account at least,²¹ that movement must not necessarily create opaque left branches. The subextraction at issue is precisely a matter of accessibility within a complex left branch, namely movement of N after movement of the [Adj - N] constituent to the left of Num. However, we do not run afoul of the motivation for External Merge to create opaque left branches; Internal Merge does not as such entail a separate derivational “workspace”, in the sense of Collins or

²¹ Compare (5b) above, corresponding to order NDMA. For Abels and Neeleman, this order can be derived with a single “well-behaved” movement of N alone, as in (5b), and head-final ordering of the Num-Adj portion of the tree. The tree at right illustrates the relevant structure:



Uriagereka. In particular, if the structure produced by External Merge is spinal, Internal Merge affects two portions of the same “derivational cascade.” Indeed it seems it must be, as discussed in 5.3.4 above. This leads us to a particularly predictable kind of structure, and characteristic transformations thereof. This is a positive development, in light of the overall goal to sharpen the ideas here into testable predictions, and to develop a restrictive account of the distribution of movement in natural language expressions, within and across languages.

5.4.2 On remnant movement

Abels & Neeleman point out that Cinque’s account might overgenerate as well. In particular, an LCA-based system can, in principle, derive any relative order of the elements with successive leftward movements, including ones whose simplest derivation would involve rightward movement. “Consequently, proponents of antisymmetry will still need to make a stipulation banning apparent rightward movement (that is, structures that are the LCA-compatible equivalent of rightward movement).” (Abels & Neeleman 2009: 67)

For Cinque, the required stipulation comes in the form of a restriction on movement within DP to only affecting the noun, or something properly containing it. Cinque suggests that this might relate to a “[...] presumable need for the various phrases that make up the “extended” projection of the NP (in Grimshaw’s (1991) sense) to be licensed.” (Cinque 2005: 325) That rules out so-called “remnant movement” (den Besten & Webelhuth 1987, Müller 1998, among others), apparently a problem, since such

movement is analytically motivated (for LCA-based theories, anyway) in other domains:

“The requirement that every movement pied-pipe the lexical head does not seem to have a counterpart in the extended projections of other lexical categories. In fact, [...remnant movement...] is not just a hypothetical possibility, it is a widely used analytical tool[...]" (Abels & Neeleman 2009: 71).

My account makes no reference to a feature-based need for NP to raise for licensing reasons – itself a curious notion, since agreement may achieve the same result. Instead, the motivation is purely structural: NP is (by hypothesis) a large category buried deep in the tree, and a mechanism of tree-balancing would be expected to raise it.

As I show, the unattested movements can be ruled out, directly, by the very same tree-balancing concerns that rule in the full set of attested orders. Note that my theory does not rule out remnant movement everywhere; such movements are expected just in case the relevant structural conditions are met. The DP seems to resist remnant movements, a fact which follows wholly from tree-balancing if its structure is as required – but structures other than the DP may meet the conditions.

5.4.3 A simpler account, with messier predictions

All of this, I think, demonstrates what may be an advantage of my account: it does not rely on interacting, inviolable “principles” in the way Cinque’s account does. That is, no reference is made here to postulated constraints on derivations such as a ban on remnant movement, a freezing principle, or the like. Instead, each instance of movement is subject to a single, uniform structural condition regulating whether movement may or may not occur (which may be met, or not, depending on details of the tree). Thus, while

the superficial typology of movement may be “wilder” under such a theory, movement itself is very narrowly constrained (and so are its outputs).

Cinque states that “[i]deally, all and only the attested orders should follow from the conditions on Merge and the conditions on Move of the type discussed above.” (Cinque 2005: 328) In practice, there are many mysteries of DP ordering which Cinque must leave unaddressed, including an apparent preference for certain kinds of pied-piping within DP (p. 326), and the fact that partial movement of the NP is more marked than movement “all the way up” (p. 325).²²

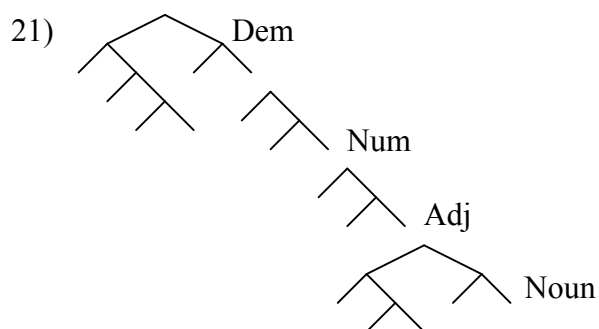
Compare this with the present account, where all and only the attested orders can demonstrably be made to follow from a single condition on movement, namely that it must always produce a more balanced tree (one with fewer ‘vertical’—c-command/dominance—relations).

That said, a note of caution is in order about what has really been established here. The fact that one could make a base DP tree motivating all observed transformations, and no unobserved ones, in terms of tree-balancing, is only significant if the required shape of DP tree is plausible. That is, showing that this account *could* work is not the same as showing that it *does* work; the real test is to check the tree shapes in the solution set against detailed cartographic analyses.

²² Both of those properties plausibly fall out from tree-balancing concerns as well, given an appropriate DP shape, though I do not pursue the matter here.

5.5. *Conclusions and direction for future research*

Let us step back for a moment and take stock. I have argued that syntactic movement is “for” reducing c-command (equivalently, dominance) relations in the representations it transforms. I have shown that a strong account of possible and impossible DP orders is tenable in these terms, and derived the conditions that must hold of the base DP tree for this to be true. Appendix A provides further numerical results on tree-shapes satisfying these conditions; in the introduction, I provided one such solution, pointing out that the required DP structure is very close to current cartographic proposals. I repeat this sample solution here:

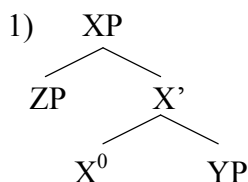


Lest this be overinterpreted, I have not shown that the present account is really “the” explanation of the movement facts in this domain. Instead, what has been established here is that a complete explanation of possible and impossible orders is available in terms of tree-balancing, so long as the “real” structure of the DP falls within the bounds given here. Much more careful empirical work would be required to verify that the DP in fact has the required shape.

CHAPTER 6: ECONOMY OF COMMAND IN PHRASE STRUCTURE¹

6.0 *Optimal phrasal shape*

I propose that the characteristic shape of human phrases, as captured by the X-bar schema and similar forms, constitutes what we might think of as an optimal packing solution or optimal growth mode. I show that constraining Merge to minimize the number of c-command and containment relations in growing syntactic representations leads to ‘projective’ phrasal shapes, exhibiting (tree-structural correlates of) endocentricity and maximality of non-head daughters. Thus, a tendency for syntactic structures to pattern according to the branching geometry of the X-bar schema (or other “projective” shapes) can be explained as an epiphenomenon of economy of command. By “X-bar schema”, I mean (1):



In (1), a phrase XP is comprised of a head X^0 composed with one phrase (its complement, YP), the resulting object composed with another phrase (its specifier ZP) I take this result to be interesting in light of the very widespread endorsement of a rather strict X-bar schema in much recent descriptive work, especially within the cartographic project; see discussion in Chapter 1. Shlonsky (2010) summarizes the situation:

¹ This chapter is a revised and shortened version of Medeiros (2008). Some text remains the same, but much has been removed, and some new material appears here as well.

“Arguably, this configurational schema, known as X-bar theory, is the only kind of structure that syntactic representations exploit. Other structural options, such as adjuncts to phrases, multiple specifiers of a single head, etc., have been experimented with in various ways but Cartographic research has, for the most part, eschewed these options, retaining only the core structures afforded by the X-bar schema. Indeed, Cinque (1999) argues forcefully against the adjunction of adverbials, as reviewed below. The core structural relations defined by X-bar theory seem to be not only necessary, but sufficient to characterize syntactic structure.” (Shlonsky 2010: 2)

Enumerating all possible recursive templates and counting c-command/containment relations in the trees they generate, I show that the best templates (those that grow trees with the fewest number of c-command and containment relations) have the shape of generalized X-bar projections, in the sense clarified below. The best phrasal template places a unique terminal at the bottom of the phrasal template, with ‘slots’ for several more objects of the same shape as the full phrase.

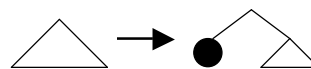
In the next section, I provide some illustration of what the predicted phrase structural forms look like, and examine forms that should be excluded by the present account.

6.0.1 Generalized X-bar phrases

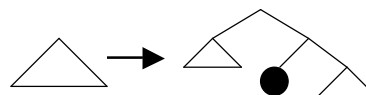
The term “generalized X-bar phrase” is intended as shorthand for a class of optimal patterns, differing among themselves in how many self-similar ‘slots’ (phrases) they permit. This includes (2), (3), and (4): In familiar terms, (2) corresponds to the geometry of the head-complement pattern, (3) to the specifier-head-complement pattern of the X-bar schema, and (4) to a pattern in which every ‘phrase’ may have two ‘specifiers’.

I include an abstract branching diagram at right, with triangles standing in for phrases and filled circles for terminals.

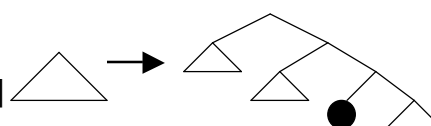
2) Phrase = [terminal Phrase]



3) Phrase = [Phrase [terminal Phrase]]

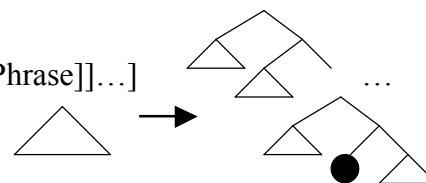


4) Phrase = [Phrase [Phrase [terminal Phrase]]]



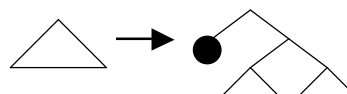
I will describe these phrasal shapes as “projective”. The generalized format of a projective phrase is represented in (5).

5) Phrase = [Phrase [Phrase ... [Phrase [terminal Phrase]] ...]]



By contrast, (6) is not projective in this sense: the terminal element is not at the ‘bottom’.

6) Phrase = [terminal [Phrase Phrase]]



I have avoided the more familiar notation $XP = [ZP [X^0 YP]]$, as that encodes information beyond the corresponding $\text{Phrase} = [\text{Phrase} [\text{terminal Phrase}]]$. Specifically, the former, but not the latter, requires a notion of projection or labeling, such that XP and X^0 are explicitly identified: X^0 is not just any terminal, it is the terminal for XP , i.e. its head. What is explained in this chapter is only the geometric pattern, with no reference to categorial labeling or headedness per se. Yet there is a clear sense in which the bare recursive geometry in (1) is ‘endocentric’: each phrase has a single designated slot for a terminal, at the most embedded level of the repeating pattern. It is only this idealized

structural relation between a phrase and its characteristic terminal “slot” that I explain here.

In the traditional representation of the X-bar schema $XP = [ZP [X^0 YP]]$, YP and ZP are phrases built according to the same body plan as XP , but with their own heads Y^0 and Z^0 . In other words, all the off-branches between the root and terminal connect to structures isomorphic to the root; in more familiar terms, all non-head daughters are full phrases.² Understood as a structural property (an isomorphism of idealized shape), this holds of “projective” phrasal geometries more generally, as the phrase = [... phrase...] notation makes clear. Yet again, this is not enforced by the assumptions I adopt; I explicitly allow that sub-parts of the structure-building recipes I consider may link in any (deterministic) order. Structures with root-like off-branches everywhere on the line from root to terminal just happen to emerge as the winners with respect to economy of command, from a larger field of phrasal possibilities where in effect “anything goes”.

I believe this is surprising and significant. The options for structure-building allowed here are quite free; any finitely-defined scheme incorporating terminals into indefinitely recursive patterns is considered. Needless to say, only a small minority of these patterns ‘look like’ projections. Other possibilities have a repeating phrasal template which places terminals at (potentially many) designated locations other than the

² A word about adjuncts here: as Chametzky (2000) points out, the relatively theory-neutral notion of *adjunct* is not the same as the theory-internal mechanism of (Chomsky-) *adjunction* used to represent them. I side here with the views of Cinque (1999), who takes it that adjuncts, once thought of as loose add-ons, with little restriction, an exception to the base phrase structure, are none of those things; in fact are remarkably regular cartographic mileposts (and cf Pollock 1989). Moreover, I take it that say, adjectives and adverbs, prototypical adjuncts, are not introduced by an exotic adjunction operation somehow different from regular merge. Rather, I assume with Cinque that such elements are specifiers in a well-behaved one-specifier X-bar phrase.

‘bottom’, or recurse via units different than the ‘top’ of the template, and so on.³ The considerations which enter into the investigation are of a purely configurational, geometric nature; no notion of ‘head of a phrase’, ‘label’, or other elements of the theory of projection are built into the assumptions. Yet something akin to projection (more precisely, a structural basis which could readily be mapped to a projection scheme) emerges ‘for free’ as an optimal solution. This suggests that the property of projection may be an epiphenomenon of ‘blind’ structural optimization.

I argue that this geometric property may give rise to linguistic projection as an accidental consequence— a happy result, since labeling had been claimed to require irreducible complexification of the basic apparatus. Hornstein (2009) puts it like this:

“What of labeling? This is less obviously what we expect of computational operations. The labeling we see in FL leads to endocentric phrases (ones with heads). There is a lot of evidence to suggest that phrases in natural language are endocentric. Hence it is empirically reasonable to build this into the Merge operation that forms constituents by requiring that one of the inputs provide a label. However, there is little evidence that this kind of endocentric hierarchical structure is available outside FL. Nor is it obviously of computational benefit to have endocentric labeling for if it were we would expect to find it in other cognitive systems (which we don’t). This suggests that endocentric labeling is a feature of Merge that is FL specific.” (Hornstein 2009: 13)

The upshot is that projection is empirically motivated, but conceptually mysterious. I return to the issue of projection, and whether and to what extent it can be explained by the present account, below.

3 Of course, this invites the further question of whether those options are ‘linguistically reasonable’, or are ruled out for other reasons. I address this matter below.

6.1 *Local comparison: A first pass*

As a first look at the considerations to be explored here, suppose that a syntactic derivation has reached a stage where the following three objects remain to be combined:

7) X^0 , AP, BP

Here, X^0 is a bare lexical item, while AP and BP are internally complex objects constructed by Merge. For the purposes of this simplified example, let us ignore any distinction between AP and BP. The options for continuing the derivation are these:

8) [AP [X^0 BP]] (or [BP [X^0 AP]])

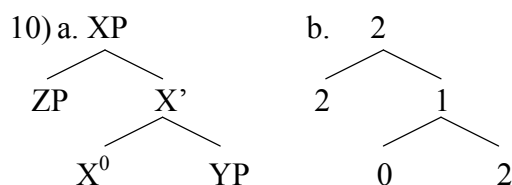
9) [X^0 [AP BP]]

Is there any basis for choosing between (8) and (9) in terms of their effects on c-command and containment relations? There is. Let a be the number of nodes in AP, and let b be the number of nodes in BP. Since AP and BP are internally complex, $a, b > 2$. When two objects Merge, the number of new c-command relations defined is simply the sum of the number of nodes in each; likewise, the operation also creates the same number of new containment relations (as the new mother node contains all of the nodes in each). Thus, creating (8) defines $(b + 1) + (a + b + 2) = a + 2b + 3$ new c-command and containment relations. Creating (9), on the other hand, allows $(a + b) + (a + b + 2) = 2a + 2b + 2$ new c-command and containment relations, which is strictly greater. Thus, fewer such relations are (potentially) computed at this stage if the derivation ‘grows’ according to (8) rather than (9). As argued in more detail below, this gives us good reason for preferring (8) over (9) in terms of efficient computation, all else being equal.

Needless to say, this departs from the usual way of thinking about these matters. For one thing, it is usually assumed that given some real example, only one of (8) or (9) could apply; the other choice would ‘crash’, failing to meet the requirements of the items involved. Moreover, only some of the c-command and containment relations defined would actually be exploited to carry real linguistic relations. I return to these issues in more detail later on. For now, the idea is that if we find as an empirical matter that the configuration in (8) tends to predominate as a structural pattern, while configurations matching (9) are relatively rare, we might be able to explain that fact in terms of this kind of comparison.

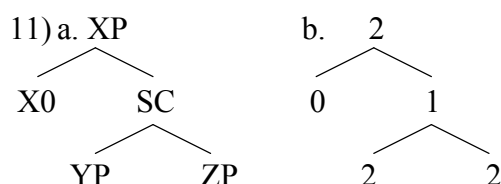
Note that (8) has the shape of an X-bar pattern of specifier, head, and complement, whereas (9) might correspond to a head taking a small clause complement, which seems to be a good deal less common (as an iterated pattern). What is at stake here has nothing to do with projection; questions such as whether X^0 is the ‘head’ of the construction do not enter into selecting one form over the other. Rather, the issue is one of branching form and its effects on c-command and containment relations.

In this light, consider the familiar X-bar schema in (10a). Setting aside the matter of projection (the fact that the complete syntactic object shares a lexical category label X with its head X^0), the relevant aspect for our purposes is that a complex syntactic object is formed by the particular recursive pattern in (10b).



At first, it looks like (10b) is just a matter of ‘bar-level’ notation: 0, 1, and 2 correspond to X^0 , X' , and XP respectively. But there is a way of thinking about (10b) which does not require reference to explicit ‘bar-level’ features (a grammatical device that has been discarded from Minimalist theory for good reasons). The objects in (10b) are merely a convenient notation for describing the particular recursive pattern embodied by the X-bar schema. That is, a 0 in (10b) is a terminal (a lexical atom), while 1 and 2 are defined recursively: A 1 is an object resulting from Merging a 0 and a 2, and a 2 is the result of Merging a 1 and a 2. This is a template for recursion, implicitly expandable ‘all the way down’.

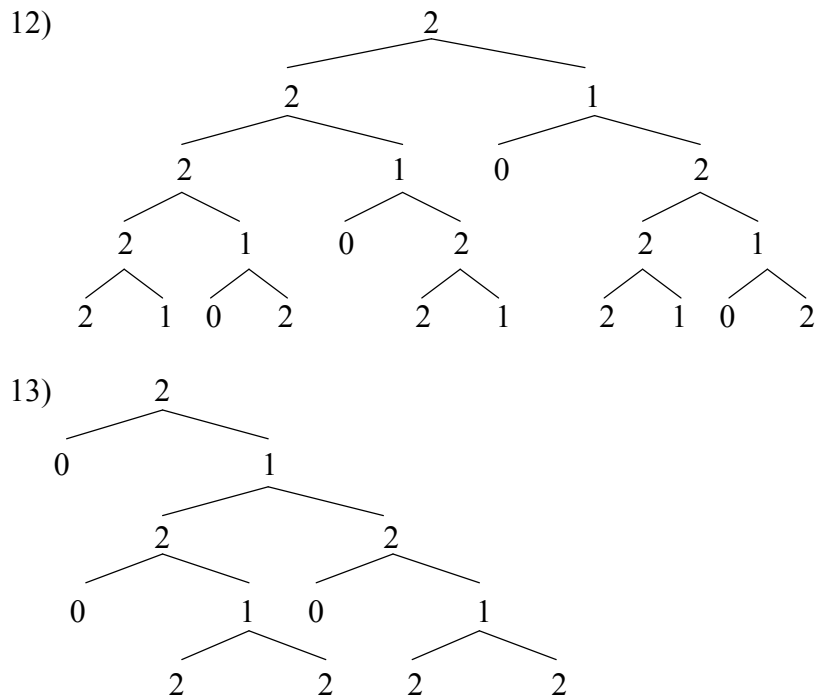
On the other hand, the option followed in (9) manifests a phrasal format distinct from the X-bar shape, as in (11). (11a) gives a familiar linguistic interpretation of the shape (a head taking a small clause complement, as in the analysis of the copula by Moro 2000). What is of interest for present purposes is the abstract recursive characterization of the shape in (11b).



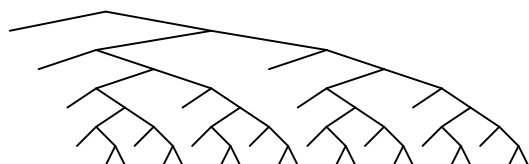
To be clear, I am not claiming by the representation in (11b) that small clauses are X' categories, or anything of the sort. Instead, the point is that this structure can be characterized in terms of three kinds of geometric object. One is a terminal, X^0 , labeled 0 in (11b). The other two objects (1 and 2) are distinguished by their recursive properties. The idea is that (11b) is an alternative to (10b) as a phrasal template. If this pattern

continued, the nodes labeled 2 at the lowest level of (11b) would themselves be head+small clause structures of the same shape as (11b), potentially ‘all the way down’. This would lead to different possible branching forms for linguistic structure.

I illustrate in (12) and (13) the results of recursively expanding the X-bar schema (10b) and the head+small clause pattern (11b). Expressions characterized by these patterns would fill some finite portion of these full branching spaces.



15) 9 generations of HH D-Bar (Phrase \rightarrow [terminal [Phrase Phrase]])



As is immediately clear, recursive expansion of the X-bar pattern creates a space of branching forms that is intuitively ‘denser’ than the space associated with the head+small clause pattern. This difference in ‘branching density’ turns out to be simply another aspect of the difference between (10b) and (11b), ultimately a part of the same fact underlying the local preference for (8) over (9). Put simply, the more densely the space of forms generated by a phrasal template branches, the better that phrasal template is for reducing the computational burden of c-command and containment relations. The relationship between recursive patterns (such as the X-bar format (10b) and the head+small clause format (11b)) and c-command and containment relations is the matter that will concern us in this chapter.

6.2 *Generalized phrases*

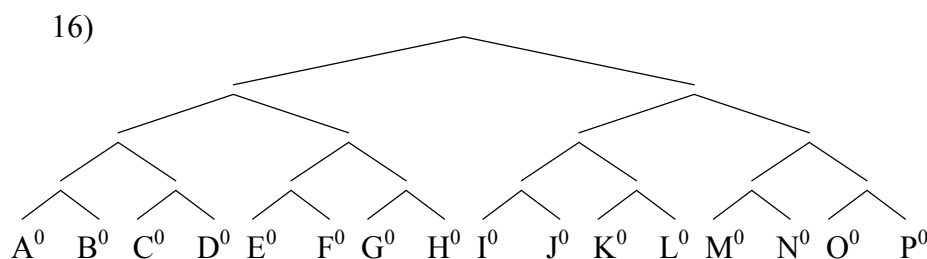
What I propose to investigate and compare below are phrase structural patterns, in the sense of characteristic aspects in the branching geometry formed by Merge applying recursively to lexical items and its own output. The hypothesis being entertained is that the forces that govern the process, in the sense of selecting some binary-branching structures over others, will give rise to identifiable and repeated tendencies (what might be thought of as ‘optimal growth modes’). To determine what tendencies we might expect, I generate all possible patterns that could be used as consistent ‘phrasal templates’

to build infinitely recursive structures from lexical atoms, and develop a technique to compare them to each other.

6.2.1. *A domain for terminals*

One condition that will need to be imposed is that the recursive templates include a characteristic place for terminal elements. This makes a good deal of sense on several levels. First, the objects are recursively defined, which requires some ‘base step’; it is hard to see what aspects of branching structure could provide this other than terminals. From another point of view, these are ultimately discrete, finite patterns, built bottom up from lexical items; they are ‘about’ structuring terminals into larger structure. Without terminals to ‘ground’ the patterns, there can be no distinctive shape, hence no ‘pattern’ at all; the only rule would then be ‘anything goes’.

The concern in this regard is structures like (16) below, which are ‘maximally balanced’, with all terminals at the same depth (or at two adjacent levels of depth). These structures provide absolute minimization of c-command and containment relations.



If economy of command really does ‘matter’ in the determination of structure, why do we not see such forms in natural language? If the only problem were optimizing at once the positioning of a full set of elements, we would indeed expect to see something like this.

But one guiding theme in minimalist work is the idea that syntactic forms are to be explained dynamically, by local (informationally limited) optimization at each step of a syntactic derivation. In these terms, the structure above looks decidedly unnatural. To actually derive such a form, Merge must apply as symmetrically as possible. This involves unbounded ‘vertical’ information flow at each step; the internal structure of syntactic objects must be accessible ‘all the way down’ so as to match objects (terminals, pairs of terminals, pairs of pairs of terminals, etc.) appropriately. But even this ‘local’ (i.e. one Merge operation at a time) matching of object structures is not enough. The derivation must be kept in appropriate synchrony across the entire set of parallel sub-derivations; if one process of merging terminals into ever-larger sets proceeds too many steps beyond other combinations occurring in parallel, we may be left with a final stage where only unmatched objects remain. Information must thus be shared ‘horizontally’ as well, in effect amounting to global pre-planning of the derivation.

We can identify a parallel situation in botanical growth, which proceeds by a local logic, where notions such as ‘final form’ have no power to shape the dynamics of growth. Similar concerns apply to the pattern of Fibonacci spirals in phyllotaxis: If the only problem were to pack at once a certain number of elements into a limited space, a hexagonal lattice structure would be best. But the observed patterns grow, with the result that what we in fact observe is not the best form, but the best growth pattern, a crucial distinction.

Given the dynamic view of syntax adopted here, similar constraints are expected to apply: The best configuration is ‘ungrowable’. Parallel to the phyllotactic case, we expect

to observe at best an optimal derivation, not an optimal final representation, because the dynamic system is limited by a fundamental locality. This is why (26) is not predicted here; no local pattern of growth can produce it.

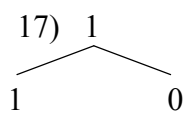
6.2.2 Possible growth modes

Such concerns lead us to expect that the considerations that enter into derivational choices will be limited by an informational horizon. Recall that one of the problems with (16) was that it required syntactic objects to be matched ‘all the way down’. Limiting this informational flow means that only some of the recursive structure of the operands of Merge is ‘visible’ to optimization concerns. For example, if one level of internal structure can be examined, then terminals can be distinguished from more complex objects. Allowing two layers of structure to be visible allows further distinctions, which allows more internal complexity in recursive patterns, and so on.

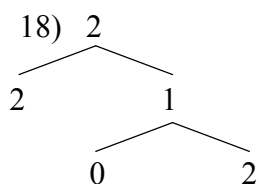
As an idealization to aid the investigation of these matters, I will suppose that whatever pattern might be found will be consistent (i.e. deterministic). A consistent recursive scheme carried out within a finite derivational window can be described by a finite number of distinct ‘types’ of syntactic object (terminals, or objects recursively defined as the result of Merging other terminals or recursively defined objects), which ‘loop’ into each other in a finite cycle.

6.2.3 Notational conventions

To allow the full range of recursive possibilities, let us simply use the natural numbers to represent the relevant distinctions among outputs of different Merge operations, reserving 0 for terminal elements. Let us furthermore use the largest number in a pattern to designate the root symbol (held constant, under the ‘top-down’ formulation discussed below). Here, we will take the appearance of the same number on two different nodes to mean that the structures so labeled have isomorphic recursive structure. In these terms, the simplest recursive pattern (both including terminals and allowing indefinite recursion) will be represented as below:



Likewise, in this formulation the X-bar specifier-head-complement pattern will have 0-level terminals marked as 0s, while ‘single-bar-level’ intermediate categories are 1s, and ‘phrases’ are 2s.



Thus, the numerical designations might be thought of as something like a generalization of conventional ‘bar-level’ notation. To be clear, this is not a proposal about reviving bar-level notation as an explicit grammatical device, thus violating Inclusiveness. Instead, the notation is a device for reasoning about possible derivational sequences; the relevant information is not to be understood as somehow reified in any way ‘on’ the node, but is a

matter of information that is in the way the derivation itself proceeds. If these patterns do characterize natural language, that fact presumably emerges from dynamic considerations, rather than being explicitly enforced by some mechanism like ‘bar-level features’.

Insofar as a pattern is consistent, its elements (other than 0) can be characterized by what amount to ‘rewrite rules’ (again, this is a matter of investigational convenience, not a proposal for a ‘real’ grammatical device). The general form of these descriptions of local binary-branching structure can be described in the following algebraic format:

$$19) \quad i \rightarrow j \ k \quad i \text{ in } \{1, 2, \dots, n\}; j, k \text{ in } \{0, 1, 2, \dots, n\}$$

That is, an arbitrary non-terminal element i is taken to consist of elements j and k , either non-terminals characterized in the same way or 0, i.e. terminal. The simplest structure (17) can be expressed as in (20), and the X-bar schema as in (21):

$$20) \quad 1 \rightarrow 1 \ 0$$

$$21) \quad \begin{array}{l} 2 \rightarrow 2 \ 1 \\ 1 \rightarrow 2 \ 0 \end{array}$$

6.2.4 *Generating all possibilities*

Let us now set to exploring the options systematically. If the ‘derivational window’ is as small as possible (i.e. the growth pattern is as simple as possible), then there is only one option for how to build recursive structure from terminals. I call this the ‘spine’, for obvious visual reasons (intuitively, it generates a uni-directionally branching tree); I will likewise use descriptive names for the other patterns for mnemonic convenience.

$$22) \quad 1 \rightarrow 1 \ 0 \text{ ('spine')}: \quad \begin{array}{c} 1 \\ \swarrow \quad \searrow \\ 1 \quad \quad 0 \end{array}$$

We obviously need at least this much structure to have recursion at all. Ignoring linear order (as I do throughout), and requiring the pattern to be built recursively from terminal elements and the output of Merge, for distinct objects 0, 1 the other combinations can be ruled out ($1 \rightarrow 1 \ 1$ is not built from terminals, while $1 \rightarrow 0 \ 0$ is not recursive).

Moving on to the next level of complexity in sequencing Merge, we consider patterns involving two types of non-terminals (equivalently, two-stage sequencing of Merge operations). Given the remarks above, we have at first pass $6^2 = 36$ distinct options for recursive patterns involving two order-irrelevant Merge rules (i.e. non-terminal characterizations) defined over three object types (0, 1, 2); for arbitrary n , there are $(n(n+1)/2)n-1$ options. Being a little more careful, we can restrict this further by ruling out the following types of characterizations (here, i is an arbitrary non-terminal, n is the root-type non-terminal, 0 is a terminal):

$$23) \quad * \quad i \rightarrow i \ i \quad \text{Does not terminate (DNT)}$$

$$* \quad n \rightarrow 0 \ 0 \quad \text{Is not recursive (INR)}$$

$$* \quad n \rightarrow n \ 0 \quad \text{Isomorphic to the Spine}$$

In words, any object which immediately contains two isomorphic copies of itself cannot be recursively constructed from terminals. If the root node (designated as the largest number n) consists of two terminals, recursion is impossible. Finally, if the root node consists of a terminal and an object isomorphic to the root, it is isomorphic to the spine (1

$\rightarrow 0\ 1$), hence is not really a member of the higher-order comparison set. The table below lists all options for the comparison set built from $\{0, 1, 2\}$; non-viable options are in grey.

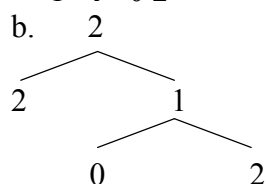
	$2 \rightarrow 2\ 1$	$2 \rightarrow 1\ 1$	$2 \rightarrow 1\ 0$
$1 \rightarrow 2\ 2$	DNT	DNT	high-headed D-bar
$1 \rightarrow 2\ 1$	DNT	DNT	high-headed X-bar
$1 \rightarrow 2\ 0$	X-bar	D-bar	(spine)
$1 \rightarrow 1\ 0$	spine of spines	pair of spines	(spine)
$1 \rightarrow 0\ 0$	double-headed spine	INR	INR

Table 4: Options for comparison set built with two non-terminals.

I have also greyed out the option described as a ‘pair of spines’, which, as the name is intended to suggest, consists of two spines merged at the root. It should be clear that this is not a repeating structure; the configuration at the root is unique, and thus it is not a growth pattern in the desired (basically, self-similar) sense. I illustrate the remaining options below, including their repeating ‘molecular’ structure as a partial tree diagram.

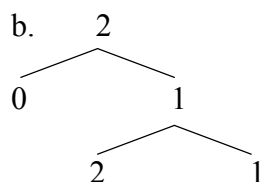
24) a. $2 \rightarrow 2\ 1$ (‘X-bar’)

$1 \rightarrow 0\ 2$



25) a. $2 \rightarrow 1\ 0$ (‘high-headed X-bar’)

$1 \rightarrow 2\ 1$

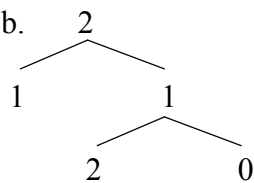


Options (24) and (25) form a natural pair, as do (26) and (27) below, in that the members of the pairs are really the same recursive cycle caught at different times, with a different selection of which non-terminal serves as the root. I call the member of each pair of patterns in which the terminal occurs nearer to the root ‘high-headed’. See the discussion in 4.3.3.2 below.

26) a. $2 \rightarrow 1\ 1$ (‘D-bar’)

$1 \rightarrow 2\ 0$

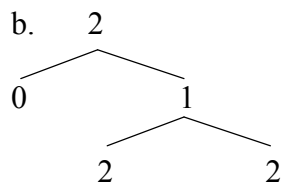
b.



27) a. $2 \rightarrow 1\ 0$ (‘high-headed D-bar’)

$1 \rightarrow 2\ 2$

b.

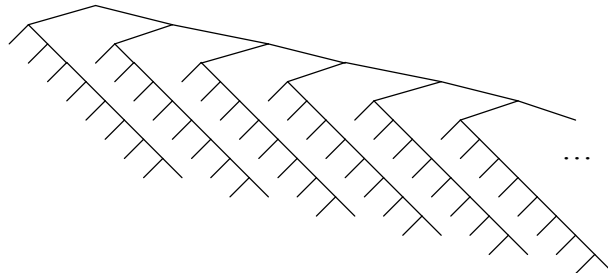
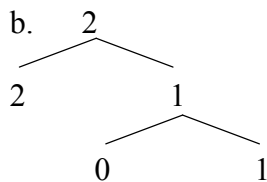


This pair (again, really different ‘snapshots’ of the same pattern) has a fundamental symmetry; the D in D-bar is meant to stand for ‘double’ for this reason.

28) a. $2 \rightarrow 2\ 1$ (‘spine of spines’)

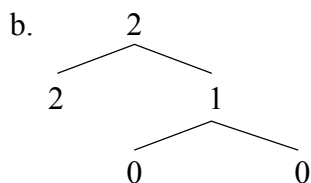
$1 \rightarrow 1\ 0$

b.



29) a. $2 \rightarrow 2\ 1$ (‘double-headed spine’)

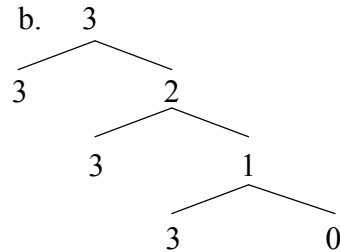
$1 \rightarrow 0\ 0$



Enumerating all of the options for further comparison sets (allowing three stage Merge sequences/three non-terminal types) would be a good deal more tedious. For illustrative purposes, I include just one of the options. This represents the ‘projective’ geometrical format, and thus is the optimal member of its class (for reasons discussed below, and proven in fuller generality in section 6.6). Intuitively, it corresponds to the structures described by Jackendoff’s (1977) ‘uniform three-level hypothesis’, an X-bar-like structure with two specifiers. In other words, it is a version of the X-bar schema utilizing three non-terminal types; hence, ‘3-bar’. Recall the convention that the highest number in the algebraic phrase structure rule representation (40a) is identified with the root. In other words, while 2 was in effect something like a ‘maximal category’ – again, with no reference to lexical projection or labeling – for X-bar and other patterns of the same complexity, a 3 is a root/maximal category for the class including 3-bar.

The X-bar class (built from two non-terminal types) has 6 viable phrase structure patterns, including 2 with degenerate subcycles (they contain subpatterns characterizable with fewer non-terminal types than the whole pattern). The 3-bar class has 57, (13 with degenerate subcycles, ie non-terminals dominating structures from the X-bar or Spine class (two or one non-terminal). The size of the next class (about 800 distinct patterns) and the attendant complexity means an end to practical investigation, with the present techniques at least.

30) a. $3 \rightarrow 3\ 2$ ('3-bar')
 $2 \rightarrow 3\ 1$
 $1 \rightarrow 3\ 0$



6.3. Comparing growth modes

Now that we have developed a way of enumerating the possibilities for recursive growth modes, we turn to the task of comparing them to each other. Recall the fundamental observation underlying this investigation, that building structure in some ways results in fewer c-command and containment relations than other options. I have argued that having fewer such relations lessens the computational burden for the derivation. The hypothesis is that this results in a preference for patterns in the application of Merge that will tend to reduce c-command and containment relations. Our goal in this section will be to develop a technique to compare the recursive options we have enumerated on the basis of their consequences for c-command and containment totals.

6.3.1. Comparison sets based on local complexity

Each of the recursive patterns we are considering is defined within the bounds of some fixed amount of sequential complexity. Some patterns have more or less internal structure than others: The spine is 'simpler' than the X-bar schema. The X-bar schema requires

more in the way of (relatively local) information flow to structure the derivation appropriately. Different choices of the size of the derivational window (i.e. the number of different types of object, or equivalently, the number of derivational steps in a characteristic cycle) will partition the possibilities into natural comparison sets. That is, we will compare recursive patterns of comparable complexity to each other. In present terms, we will be comparing patterns that can be specified with the same number of symbols, so that a comparison set will consist of all the recursive possibilities that can be described with numbers from 0 to some fixed n .

6.3.2. *Direct comparison redux*

How can one growth mode (recursive pattern) be compared to another? Sometimes the comparison can be made quite directly. Consider again the following example from the introduction. We are given the problem of combining the syntactic objects AP, BP, and X^0 via binary Merge. AP and BP are internally complex, while X^0 is a terminal. The options are these:

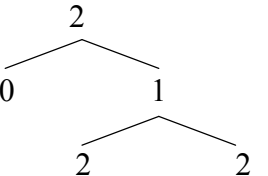
31) [AP [X^0 BP]] (or [BP [X^0 AP]])

32) [X^0 [AP BP]]

Again, given just the information that AP and BP are internally complex, the first option produces fewer c-command and containment relations than the second. Noticing the monotonic way in which c-command and containment relations accumulate in a derivation (i.e. additively), this local superiority gives us very good reason for preferring to apply the pattern manifested in the first option over the second more generally, if we

are forced to choose one or the other as a repeated format. Put another way, it motivates the choice of the growth mode (33) over (34):

33)  ('X-bar')

34)  ('high-headed D-bar')

However, this sort of direct comparison will not work for the full comparison set they belong to. Consider another member of that set:

35)  ('double-headed spine')

No local, direct comparison with the previous two patterns is possible, since they take different inputs (35 calls for two terminals); in general, where (33) and (34) can be applied, (43) cannot.

6.3.3. Indirect Comparison

To get around this problem, I will proceed as follows. First, it is an inescapable fact that these are discrete patterns, ultimately built from some finite number of terminal atoms. This suggests an alternative, slightly indirect way to compare different growth patterns: Compare the set of tree-forms they can generate for some constant number of terminals.

These patterns implicitly define a class of trees. For example, The Spine can be applied to generate (36); that unidirectionally branching structure belongs to the set of trees associated with the growth mode (such a tree can be ‘grown’ by the pattern). On the other hand, (37) does not belong to the class of trees associated with the Spine.

36) [W0 [X0 [Y0 Z0]]]

37) [[W0 X0] [Y0 Z0]]

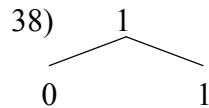
For a fixed number of terminals, there are many different binary-branching arrangements of that number of elements. Some of those branching structures will belong to the class of trees associated with a particular phrase-structure pattern, and some will not. These will typically differ in their number of c-command relations. However, for a fixed number of terminal elements and a particular recursive pattern, we can identify the best tree(s), which contain the fewest number of c-command relations of any of the trees associated with a particular pattern. These best trees for a number of terminals then serve as a basis for comparison among the patterns themselves (since, as it turns out, this comparison is monotonic: If a pattern allows a better tree for n terminals than any competing pattern, it also has a better tree for $n+m$ terminals).

6.3.4. The ‘bottom of the tree’ problem

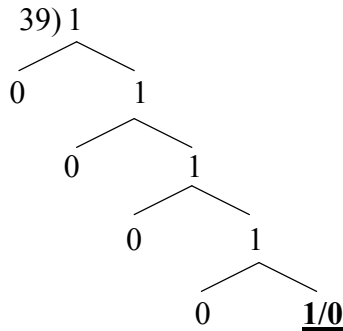
However, this requires some further clarification. The idea is to find some way to compare templates for infinite growth, by isolating them and seeing what happens when they are followed as faithfully as possible. The problem is that none of these rules can be followed completely faithfully. This is an inevitable consequence of insisting that they

allow for indefinite recursion: Any such growth pattern must contain ‘slots’ for other objects of indefinitely large size. Yet the objects that manifest these patterns must ultimately be finite, with nothing but terminal nodes at the bottom of the tree. As a result, some ‘slot’ that calls for a larger object must be filled with a terminal instead.

To illustrate, consider the simplest possible growth rule for combining terminals into an indefinitely large recursive structure:



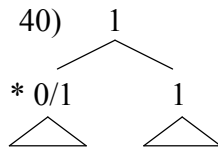
Even in this, the simplest pattern, the very first step in a derivation presents a problem, as it does not follow the rule. Any derivation whatsoever must begin by creating a structure of the form $[X^0 Y^0]$; there simply is no other option. So for a pattern like (38), we will accept a structure like (39) as manifesting it as faithfully as possible:



The notation 1/0 indicates where we have deviated from following the growth rule (necessarily, since the tree is finite), here including a terminal (0) where the rule calls for a complex object (1).

However, if we must allow some ‘fudging’ at the bottom of the tree, we can at least be faithful everywhere else. Keeping in mind that our ultimate goal is to find some basis

for comparing one growth mode to another, we reason that we do not want to ‘truncate’ the pattern encoded in the growth rule anywhere not required by the brute fact of discreteness. In particular, we will insist that the growth pattern be followed faithfully ‘in the middle’ of the derivation, so to speak. This amounts to the formal specification that the only deviation from the recursive pattern allowed will be replacing a called-for non-terminal with a terminal. We rule out non-terminal to non-terminal sequencing that violates the pattern, as in (40) below. Here, the notation *0/1 marks the illegitimate portion: A called-for terminal has been filled with a non-terminal instead.



6.3.5. *Top-down generation*

Note that we have imported a further complication by the convention of assuming that one of the non-terminal types (n, the highest of the numbers designating the non-terminal types) will be uniformly associated with the root. Formally, this amounts to generating the trees to be compared from the root down, allowing any branch to terminate. It is an important (if subtle) point that this is not a matter of committing to a top-down view of syntactic derivation, though it should be recognized that a Merge-based system need not be quite so literally bottom-up as often assumed:

Thus if X and Y are merged, each has to be available, possibly constructed by (sometimes) iterated Merge. [...] But a generative system involves no temporal dimension. In this respect, generation of expressions is similar to other recursive processes such as construction of formal proofs.

Intuitively, the proof ‘begins’ with axioms and each line is added to earlier lines by rules of inference or additional axioms. But this implies no temporal ordering. It is simply a description of the structural properties of the geometrical object ‘proof’. The actual construction of a proof may well begin with its last line, involve independently generated lemmas, etc. The choice of axioms might come last. (Chomsky 2007a: 6)

Regardless, in the present investigation top-down generation is an artifact of notational choices, rather than a substantive claim.⁴ Recall that the objects of interest are recursive cycles. Understood as time-neutral geometric patterns of recursion, these patterns do not properly have a ‘beginning’ or an ‘end’ (other than terminal elements, which can in principle appear anywhere in the looping structure as inputs to Merge, but not outputs). Their structure is a matter of how outputs from one step loop into the input to other steps. But we have kept to the familiar tree-diagram notation, assigning numerical designations to non-terminal types. The result is that certain patterns are multiply represented. For example, ‘X-bar’ and ‘high-headed X-bar’ are really the same recursive pattern, with a different choice for which non-terminal occurs at the root.

However, it turns out that a certain orientation of the pattern (fixing one or another of the non-terminal types at the root) will consistently provide better results than others. Thus for each looping object we can generate a set of alternate versions fixing one or another of the stages as the ‘top’, corresponding to the ‘root’ of a tree, and see which are

⁴ In light of this point, the claim made here about ‘projective structures’ needs to be clarified somewhat. Represented in the format $[\alpha [\beta \dots [\gamma [X0 \ \delta]] \dots]]$, the claim is a little too strong. What is motivated here is rather the recursive cycle underlying this format. Put another way, even universal strict adherence to such a growth mode in reality would not necessitate that the root node be maximal; the recursive cycle could be oriented differently at the root, thus showing up as one of the ‘high-headed’ alternatives (such a situation would look like a ‘small’ projection at the root embedding an otherwise well-behaved projective structure).

best. Since the basis for comparison is best performance, this should not present a problem in any way.

6.3.6. Some Results from Indirect Comparison

Figure 2 graphs the growth in c-command and containment relations for several recursive patterns. Recall that for each growth mode, there is an associated set of trees generated by adhering to the structural pattern consistently from the root down, allowing terminals to appear in ‘slots’ calling for non-terminals (required for finite trees). For a given number of terminals, a number of trees can be generated by a given pattern. These will differ in the number of c-command and containment relations they encode, but for each choice of growth mode and number of terminals, there will be a best tree (or set of such trees). A ‘best tree’ has the fewest possible c-command and containment relations that could be produced by that growth mode for that number of terminals. It is these totals that appear in Figure 2 (as a function of the number of terminals).

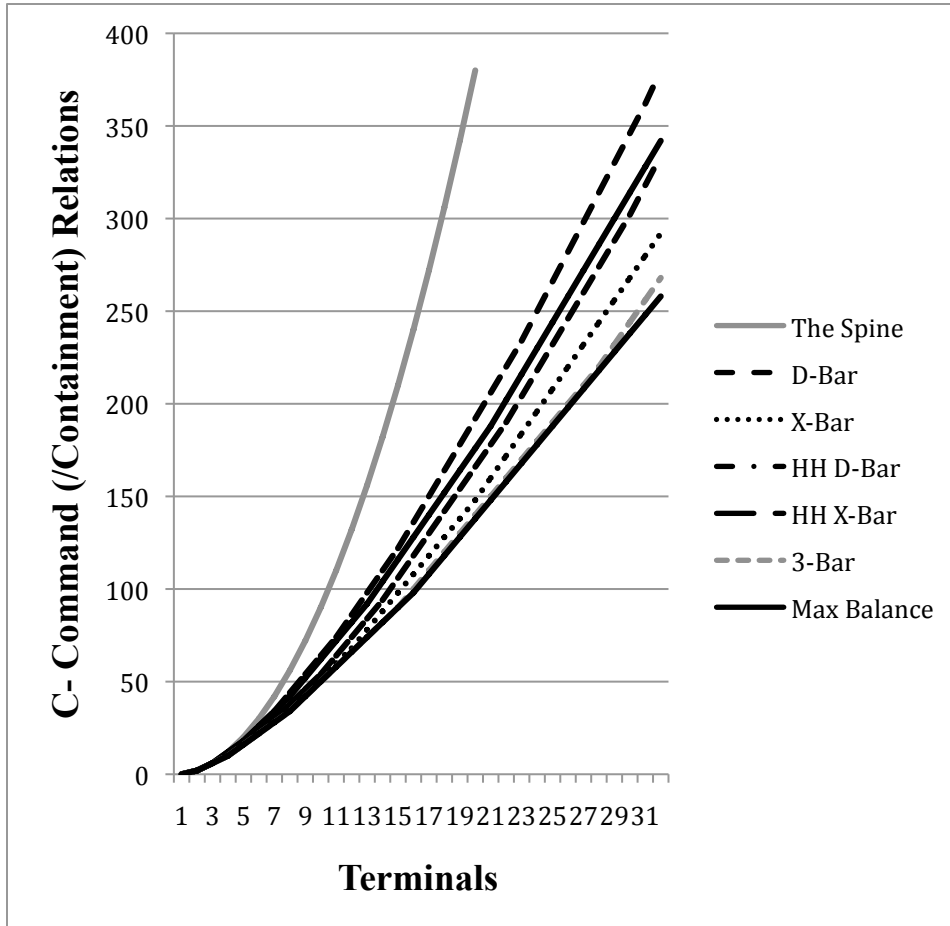


Table 5: C-command relations as a function of terminals in best trees.

I include in the figure ‘best trees’ in X-bar (34), as well as three other two-layered constructional schemes (35-37). I also include the best system utilizing a 4-way combinatorial distinction (40), which I call ‘3-bar’ (intuitively, an X-bar-like system with two types of intermediate category). The spine (32) forms the upper boundary curve; no growth pattern results in worse performance (in the sense of creating more c-command and containment relations for a given number of terminals). There is also a lower boundary curve, here labeled ‘Max Balance’. This is the number of c-command and containment relations in a maximally balanced tree like (26) from section 4.1; the pattern

is not the result of any finite growth pattern, but forms the boundary on best-case performance.

Among the growth modes in its comparison set, X-bar has the best performance: Its curve is closer to the best-case lower boundary ('Max Balance'). The optimal pattern from the next comparison class, '3-bar', has slightly better performance (the best trees that can be 'grown' by that pattern have fewer c-command and containment relations for the same number of terminals).

To be clear, the figure is meant as an illustration, not a proof. The general result that projective growth modes are best is established formally in section 6.6.

6.4. *Deriving projection*

As suggested by Figure 2, X-bar is the best growth mode that can be achieved by any two-stage scheme for constructing recursive structure from terminals via binary Merge. What I call '3-bar' is better still, though it requires more distinctions (more recursive complexity, more information flow) to construct. Generalizing, these are examples of the 'projective' format in (51), where X_0 is a terminal at the 'bottom' of the repeating structure, and α , β , and so on are objects themselves constructed according to (41).

$$41) \quad [\alpha [\beta \dots [\gamma [X_0 \delta]] \dots]]$$

The structural properties of (41) can be captured in our alternate notation as in (42), where 0 is a terminal, and n the non-terminal associated with the root.

$$42) \quad \begin{array}{l} n \rightarrow i \ n \\ i \rightarrow j \ n \\ \dots \\ k \rightarrow 0 \ n \end{array}$$

In words: every non-terminal type immediately contains a root-type non-terminal (n), and in the chain of generation from the root, through each other non-terminal type, the (unique) non-terminal type k that contains a terminal is last in line (i.e., the head comes at the bottom of the repeating phrasal molecule). The specifier-head-complement format of X-bar theory is one example of such a ‘projective structure’: Specifically, it is (42) with $n=2$. The more optimal ‘3-bar’ system of (40) is another example, this time with $n=3$. As I prove in section 6.6, this is the optimal format for $n+1$ (i.e. 0, 1, ... n) types of category (many other less optimal possibilities exist). Intuitively, the idea is as follows. The phrase structural possibilities are understood to be (partially) realized by finite expressions, built bottom-up by Merge. As such, every recursive pattern must include terminals (0s) as one of its structural types. Moreover, no categories are built solely from non-terminals ‘all the way down’.

Given these restrictions, and the determinacy of the structural characterizations assumed, any non-terminal node must dominate a terminal node within depth n , for $n+1$ types. The best kind of structure, following the format in (42), introduces terminals no closer to the root than forced by this. In essence, introducing terminals too close to the root ‘closes off’ branches, forcing complex structure to appear deeper in the tree, where it will induce more c-command and containment relations than if it were shallower. The format in (42) allows arbitrarily large structures to be as balanced as possible given the limitations resulting from finitely many structural distinctions.

Note two very interesting properties of (42):

- 43) a. Every non-terminal immediately dominates a root-type node.
- b. Terminal nodes and root-type nodes are associated one-to-one; a single terminal occurs at the lowest level of the chain of non-root-type nodes dominated by a root-type node.

Replace ‘root-type node’ with ‘maximal projection’, ‘terminal’ with ‘head’, and ‘chain of non-root types dominated by a root type’ with ‘projection chain’, and we have:

- 44) a. Every non-terminal immediately dominates a maximal projection.
- b. Heads and maximal projections are associated one-to-one; a single head occurs at the lowest level of the projection chain.

That is, the recursive scheme that best minimizes c-command and containment relations has geometric properties corresponding to (44a) the maximality of non-head daughters, and (44b) endocentricity. Such properties are the essence of the theory of projection. But the notions entering into (43) are purely structural ones. Does this ‘derive’ projection? Not in the sense of literally providing labels on non-terminal nodes. But it suggests a reason for syntactic objects to tend to take the form of structures which are ready-made to be read as projections, in that there is a natural one-to-one association in the optimal format between larger molecules of structure and unique terminals at their ‘bottom’. The head-phrase relation we think of as *projection* is recreated here in structural terms.

6.5. On projection

One issue in syntactic theory that has received considerable attention of late is the nature of *projection*. Consider, for example, the familiar pair in (1).

- 45) a. The enemy destroyed the city.
 b. the enemy's destruction of the city

These two expressions are virtually a minimal pair, at least in terms of their overt lexical contents (though many theorists attribute considerable amounts of covert structure to both). Likewise, there is an intuitive semantic similarity, in that both expressions are 'about' the same event in the world, roughly speaking. Yet they have very different properties as syntactic objects. To give the most trivial example, (45a) may occur in the frame (46a) while (45b) may not; conversely, (45b) may occur in the frame (46b), while (45a) may not.

- 46) a. John knew that _____.
 b. _____ was seen by all.

Earlier theories attributed this difference to the verbal character of *destroyed*, as opposed to the nominal character of *destruction*. On current understanding, the culprits are more properly the functional formatives *-ed* and *'s*, the former a T and the latter a D, determining that the expressions containing them are a TP and a DP, respectively (note that if further developments should point to some other elements as being the key to this difference, the basic picture would hardly be changed). Moreover, the contents of the expressions *other* than these key elements can be changed while leaving the distributional properties basically intact (modulo semantic coherence). Thus, (47a), like (45a) a TP containing *-ed*, behaves like (45a) in relevant respects (e.g., with respect to well-

formedness in the environments of (46)); and likewise (47b) parallels the behavior of (45b), with which it shares the ‘head’ element ‘s.

- 47) a. The hero completed his quest.
 b. the hero’s completion of his quest

Thus, syntactic expressions sort into distributional categories largely on the basis of some key lexical item they contain. We say, then, that the expressions are ‘projections’ of these ‘heads’. All of this is, of course, extremely well known; any adequate theory of syntax must capture this most basic fact in some way.

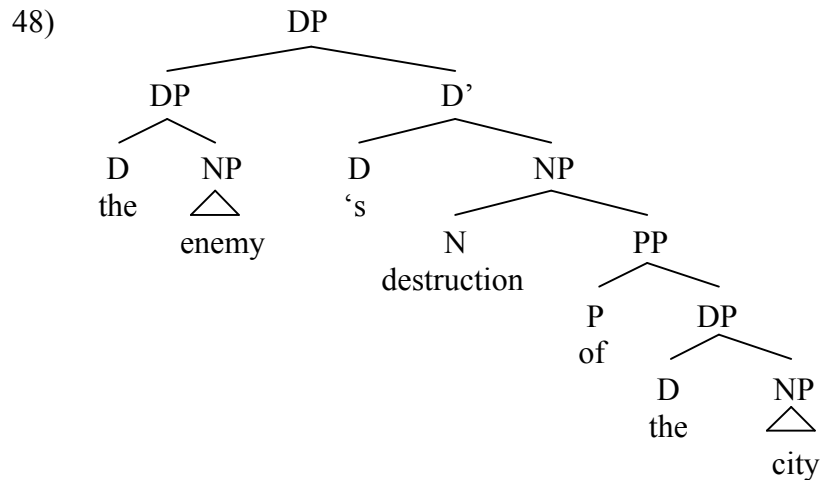
6.5.1. What is projection, and what does it do?

We may well ask several questions at this point.

Firstly, what exactly is the nature of projection within the syntactic system? That is, how is the property of projection encoded – is it read off of ‘labels’ present on syntactic ‘nodes’, or determined implicitly by some algorithm applying to inherently unlabeled structures, or does it arise in some other way? Secondly, why does human language have such a property at all?

Within the Minimalist Program, an explicit attempt is being made to reduce to a minimum the theoretical machinery deployed to capture the facts of human syntactic knowledge. Given this effort, questions arise as to how – and indeed, whether – to capture the facts of projection in terms of constituent structure.

In syntactic theories that assume a rich mathematical structure of trees as the basis for phrasal hierarchies, projection can be captured quite naturally by the device of labeling non-terminal nodes. Thus, an example like (45b) could be represented as in (4).



But such technology is not forced just by the need to capture hierarchies of constituency (perhaps the most basic fact of syntax); all that is needed to describe that aspect of phrase structure is ‘bare’ sets. If one takes Minimalism seriously, this may be seen as something of a problem. That is, we would prefer to find that the barest machinery required for recursively combining lexical items also suffice to explain the nature of projection; such a discovery would be a fulfillment of Minimalist expectations.

6.5.2. *Is projection Minimalist?*

On the surface at least, this hope simply fails. The simplest possible formalism for describing the kind of hierarchical combinations exhibited in linguistic constituency is the language of sets. By their nature, sets include no information beyond membership; in particular, they give no basis for any asymmetry among their members. But asymmetry

seems to be the very essence of projection: when one lexical item, or complex structure built from lexical items, combines with another, one object or the other is ‘more important’ in determining the properties of the composite formed by the combination. Of course, projection can be captured in a set-based system with some further assumptions and machinery, as in Chomsky’s (1995a) theory of Bare Phrase Structure. But the point is that on the barest assumptions, such further complications are unnecessary, thus a departure from minimal/perfect design.

In light of the lack of clear conceptual motivation for labeling, the results of this work look promising. Although the one-to-one association of phrases and terminals, and the phrasal character of non-head daughters, is only implicit in the optimal forms considered here, there is nonetheless a purely structural basis for projection on offer. If this mysterious property of language falls out from optimization of bare tree forms, then it needs no extra machinery to encode. That, I argue, is a good thing.

6.6 Proof of the optimality of generalized X-bar forms

In this final section of this chapter, I reproduce the informal proof given in Medeiros (2008) that generalized X-bar patterns (endocentric ones) build trees with fewer c-command and dominance relations than alternative phrasal arrangements of equivalent complexity.

Take a recursive pattern P to be defined as above over terminal type 0, nonterminal types 1, ... n , with properties of determinacy (every non-terminal i branches according to a unique rule $i \rightarrow j k$, with j, k in $\{0, 1, \dots, n\}$), and termination (no non-terminal

dominates only non-terminals ‘all the way down’).

The reasoning here will involve the infinite tree-space T generated by maximal iteration of a recursive pattern P . In such trees, every non-terminal node in the recursive pattern will be recursively expanded, and the non-terminals thus introduced will be expanded, and so on ‘all the way down’.

Now, we may consider mapping nodes in the tree-space T_1 generated by one pattern P_1 to nodes in the tree-space T_2 generated by another pattern P_2 . The idea is to find immediate-containment-preserving maps of sets of nodes in T_1 to sets of nodes in T_2 such that:

49) The image of the root node of T_1 is the root node of T_2 , and

50) If node α immediately contains node β in T_1 , the image of α immediately contains the image of β in T_2 .

Let us say that T_2 contains T_1 if there is some mapping of the set of all nodes in T_1 into nodes of T_2 meeting this condition, and that T_2 properly contains T_1 if T_2 contains T_1 but T_1 does not contain T_2 . (If T_1 contains T_2 and T_2 contains T_1 , then T_1 and T_2 are isomorphic, and so are P_1 and P_2 .)

We will also consider finite trees within these infinite trees, i.e. contained by them in the sense above. For notational clarity, we reserve T_i for infinite treespaces generated by maximal expansion of P_i . Clearly, if T_1 properly contains T_2 , every finite tree generable by P_2 can be generated by P_1 . We are interested in comparing the optimality, with respect to number of c-command and containment relations, of best finite trees (with equal numbers of nodes) generated by distinct recursive patterns P_1, P_2 . At the very least,

if every arrangement possible under P2 is also possible under P1, but there are arrangements generated by P1 more optimal than any arrangement of the same number of nodes under P2, we will judge P1 to be more optimal than P2.

51) Lemma 1: If T1 properly contains T2, P1 is more optimal than P2.

Clearly, every finite tree generable by P2 can be generated by P1. For proper containment to hold, T1 cannot be mapped to T2. The mapping from T1 to T2 fails first at some finite depth d (succeeding at all depths less than d); the maximal finite trees in T1 and T2 can be mapped to the other up to depth $d-1$. For the mapping to fail, T1 must have one or more non-terminals at depth $d-1$ that map to one or more terminals at the same depth in T2.

Then consider the maximal finite tree in T1 of depth d (all recursive options expanded to depth d , all non-terminals in T1 at depth d replaced with terminals). This tree has fewer c-command and containment relations than any tree in T2 with the same number of terminals. One or more of the non-terminals at depth $d-1$ that were expanded in T1 must terminate at that depth in T2. Then some number of nodes in T1 at depth d cannot be mapped to corresponding nodes in T2 at the same level, and the same number of nodes must appear at depth $d+1$ or greater in T2; all other nodes correspond. Since the number of c-command and containment relations induced by a node is equal to its depth in the tree, it follows that any tree in T2 containing the same number of nodes as the maximal finite tree of depth d in T1 must have strictly more c-command and containment relations.

Thus, if T1 properly contains T2, P1 is more optimal than P2: Every arrangement

possible under P2 is also possible under P1, but there are arrangements generated by P1 superior to any arrangement of the same number of nodes under P2.

52) Lemma 2: The infinite tree space T_p generated by the projective recursive pattern P_p defined over some number n of non-terminal types properly contains all treespaces T_i generated by distinct recursive patterns P_i defined over the same number of non-terminal types.

To see this, we will need one more concept, that of 'least path-to-terminal'. A 'path' leading from node α to node β is the set of nodes containing α , β , and all nodes dominating β which are also dominated by α . For any non-terminal node in a tree, we can identify the paths of nodes leading to terminals it dominates, and measure the depth of those paths. Among these paths, there will be one or more least paths-to-terminals (clearly, of depth at most n , for n non-terminal types).

Let us consider these paths under the sort of mapping described above. First, in T_p , the least path-to-terminal from the root node has length n . Let us call an 'off-branch' from this path a sub-tree whose root node is immediately dominated by a node on the path, but is not on the path itself. In T_p , the least path-to-terminal from the root of any off-branch is itself of length n (since any off-branch is isomorphic to the root node).

Now suppose T_i is a tree-space distinct from T_p defined over the same number n of non-terminal types. First, T_p contains T_i . For this to be false, there must be some finite depth d at which the mapping first fails. Find the shortest path-to-terminal from the root in T_i (or select one of them, if there are several of the same shortest length). Let us map the nodes in this path to nodes in the least path-to-terminal in T_p . This mapping succeeds,

because this path is of depth at most n , and the path-to-terminal in T_p is of depth n .

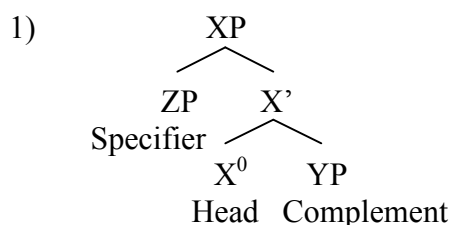
Now, for each off-branch from the path in T_i , we can map a least path-to-terminal successfully to the least path-to-terminal on the corresponding off-branch in T_p , which again is of the greatest possible depth n . And so on, for off-branches of off-branches; this exhausts the set of nodes in T_i , since (due to the termination requirement) every non-terminal lies on some least path-to-terminal. Thus, T_p contains T_i . It cannot be the case that T_i contains T_p , because we have supposed that T_p and T_i are distinct. Thus, T_p properly contains T_i . Then from Lemma 1 and Lemma 2, P_p is more optimal than P_i ; since P_i was an arbitrary recursive pattern distinct from P_p defined over the same number of non-terminal types, we conclude that the projective pattern is the most optimal.

CHAPTER 7: THE GOLDEN PHRASE

7.0 *Golden syntax*

There are several closely related mathematical objects called *golden*. These include the quantity variously described as the golden mean, the golden ratio, the golden section, or the golden number (this is Phi, τ to mathematicians, the value $(1+\sqrt{5})/2$, about 1.6180339...; sometimes we are interested in its reciprocal “little phi” 0.6180339...), and the golden string (a related binary string 10110101... with remarkable properties). The so-called golden angle (the angle of separation between successive growths in the dominant, Fibonacci-based patterns of phyllotaxis) is just the golden section measured out on the circumference of a circle.¹ Both the golden mean and the golden string are intimately related to the Fibonacci sequence (0, 1, 1, 2, 3, 5, 8, 13...).²

This chapter is about the “Golden Phrase”, the X-bar schema of specifier, head, and complement, widely agreed to be the universal ‘molecule’ of phrase-building, usually represented as in (1). As I will show, this object exhibits “golden” mathematical properties. To put it simply, (1) is the expression, in binary-branching sets, of the same theme underlying the Fibonacci numbers, the golden mean, and the golden string.



¹ By convention, we take the smaller of the arcs created by the golden division of the circle as the relevant angle (i.e. the golden angle is $(1 - 0.618...) \times 360^\circ$, about 137.5°).

² Indeed, the golden string is also known as the (infinite) Fibonacci word.

Golden may also, in a different sense, indicate some particularly favorable property – a “goldilocks” solution, as it were. I have this very much in mind as well. That is, this chapter is not just an exploration of obscure mathematical facts about the X-bar schema, purely for their own mathematical interest. Rather, I will claim in this chapter that the unique properties of the X-bar schema set it apart from other conceivable phrasal organizations as a uniquely useful or optimal choice for structuring natural language expressions, with respect to several (presumed) desiderata of language design.

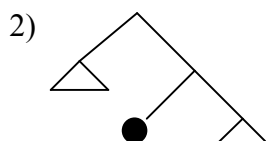
A priori reasoning about what natural language “should” look like is dangerous territory. A common – and quite reasonable – objection to all of this is that optimality is only meaningful once the problem to be solved is fully specified, with optimal solutions typically a compromise among ramified and detailed constraints on the system in question. Yet while that is surely true, at another level the study of complex systems confirms the Turing’s (1952) claim, echoing Thompson (1917), that “certain physical processes are of very general occurrence” – notably those involving Fibonacci-based “golden” forms, ubiquitous in nature. This adds immediate interest to the observation that the repeated structural motif in the human syntactic system (the X-bar schema) is likewise a “golden” form (Carnie & Medeiros 2005, Medeiros 2008), one of several Fibonacci patterns observed in the forms of natural language (Uriagereka 1998, Idsardi 2008, Idsardi & Uriagereka 2009), and leads us to inquire whether whatever is behind the natural ubiquity of such phenomena, in other domains, might possibly be at work here as well. If so, this peculiar aspect of human phrase structure would fall under Chomsky’s (2005) “third factor”, a fact about language explained by domain-general principles

beyond the organism. See especially Piatelli-Palmarini & Uriagereka (2008) for further relevant remarks about the biolinguistic significance of Fibonacci patterns in language.

In the context of purported explanation in terms of Chomsky’s “third factor”—what Massimo Piatelli-Palmarini describes as a search for “the physics of language”—considerations at this level of generality take on a particular importance. As Boeckx puts it, discussing Bejan’s (2000) abstract characterization of branching in flow systems, “Only the appeal to general laws lends a certain sense of inevitability to the explanation [...] a sign that it is not chance, but necessity alone that has fashioned organisms.” (Boeckx 2011: 57)

7.0.1 The X-bar schema as recursive template

What follows presupposes very little that should be controversial. In particular, the machinery of traditional X-bar theory (Chomsky 1970, Jackendoff 1977), including bar-level features, specifiers as conceptual primitives, and much else that has lately been judged suspicious (see Stowell 1981, Muysken 1982, Stuurman 1985, Speas 1990, Fukui 1995, Starke 2004, Narita 2010, among many others) will not be presupposed. Instead, the perspective adopted throughout is a view of (1) as a recursive template, a purely geometric object with no substance beyond its very shape. On this view, the X-bar schema dissolves into a simpler object, depicted below in (2):



In this depiction of the X-bar shape, the black dot is a terminal position; in traditional X-bar terms, this is the “head” of the phrase, determining its label (I make no such assumption here, where this is simply a structural position, no more). The triangles indicate further “phrases”, with the same shape as the whole. It is this label-free bare shape that is the topic of this chapter.

Many of the interesting properties of this object only appear when we consider recursion, i.e. (indefinite) self-embedding of the X-bar shape within itself. The X-bar schema encodes locations for copies of the whole shape at two interior points – the specifier and the complement. This scheme then implicitly defines an infinite shape, formed by indefinite iteration of self-embedding. It will be useful to have a name for this infinite shape; I will call this object “the maximal X-bar tree” in what follows.

Needless to say, that notion is an abstraction. We do not find the maximal X-bar tree manifested in any real natural language expression. Rather, natural language expressions partake, at best, of imperfect portions of this structure; they are finite, and so the non-terminal structure must “bottom out” with all terminals, of course. Moreover, at any level the full local structure of an X-bar phrase may not be present: we may find phrases with only heads and complements, and no specifiers.³

³ Though there is a way of mapping stacked head-complement configurations to sub-trees of the maximal X-bar tree (namely, by placing the complements of the stacked specifier-less phrases along what would normally be interpreted as the specifier line, with heads replacing the X' nodes and their contents). There may even be something insightful to this seeming trick, since it amounts to saying that the relationship between a head and its complement is not distinct from the relationship between an X' node and its specifier sister. If this convention is adopted then any tree built from phrases with 0 or 1 specifiers (and 1 complement) forms a subtree of the maximal X-bar tree. In that case, the study of the maximal X-bar tree is more directly relevant in understanding linguistic properties.

Even so, I suggest that it is worth examining the properties of the maximal X-bar tree. As I will show, some unique mathematical properties are present there as well as in the bare X-bar schema itself, and these carry over in remarkable properties of natural language expressions built around the X-bar schema.

7.0.2 Structure of this chapter

This chapter is structured as follows. In Section 7.1, I review some well-known mathematical properties of the golden mean, and related objects. As I show there, we can unify the description of the Fibonacci numbers, the (polynomial determining the) golden mean, and the string concatenation procedure generating the golden string, in terms of a fundamental “golden” recurrence relation.

In section 7.2, I show that the X-bar schema manifests the natural syntactic interpretation of the same fundamental recurrence relation underlying the golden mean and related objects. As a result, the maximal X-bar tree exhibits Fibonacci numbers of category types at successive levels of the tree. A number of further “golden” properties follow, which are taken up in later sections.

In section 7.3, I propose that the X-bar format represents a dynamic “minimax” solution to conflicting requirements on syntactic computation. On the one hand, as emphasized by much recent Minimalist work, there is reason to expect syntactic processes to be very simple and local; in practice, this translates to a preference for spinal, head-complement structure as the ideal syntactic form. On the other hand, this thesis motivates the idea that the opposite kind of structure (what I call the Bush) is ideal,

with respect to minimizing the burden of long-distance c-command and dominance-based computations. I point out that, within its phrasal horizon, the local form of the X-bar schema literally spans the gap between the spine and the bush; it is, we might say, the last spiny bush, and the last bushy spine. In the last part of this section, I observe that, among the set of generalized X-bar schemas, each globally best (providing the bushiest overall trees) among arrangements of matched complexity, the one-specifier X-bar schema is the last to also be locally best in its field.

In section 7.4, I turn to exploring the fractal properties of the X-bar schema. Interpreting binary branching as geometric halving of a line segment, a phrasal pattern can be interpreted as a recursive line-division algorithm. Interpreting the X-bar schema in this way, I show that it is the simplest kind of binary-branching scheme that produces a fractal image on the line; in fact it yields an asymmetric Cantor set, technically a multi-fractal. I remark that the dimensionality of this object is “golden” (specifically, the Hausdorff dimension is $\log_2(\Phi)$).

In section 7.5, I show that the X-bar form exhibits “golden” growth in yet another sense. I define a notion of “growth factor” for abstract phrase structural patterns, and show how to compute it directly for any conceivable recursive pattern. The growth factor of the X-bar form, in these terms, is the golden number Φ . Finding the growth factor requires us to formulate phrase structure patterns as matrices, a technique with considerable utility. I briefly explore the matrix forms for phrase structure classes of increasing complexity, and note some mathematical generalizations of interest.

Section 7.6 concludes the chapter. I review the terrain we have covered, and point out some overarching themes. I also return to the question of how this syntactic pattern might relate to broader phenomena in nature, including brain properties. Eventually, it might be hoped that this investigation might point the way to a deeper understanding of the mechanisms that underlie syntax, though this chapter goes no further than an exploration of the narrowly syntactic domain. Nevertheless, even at this isolated level of description, we can see that the X-bar form has a distinctly natural, perhaps even inevitable character.

7.1 A brief introduction to “golden” mathematics

In this section, I explore some simple mathematical properties of the family of “golden” objects. These include the Fibonacci numbers, the golden mean, and the Golden String. As I will show, these objects manifest a common kind of recurrence relation, though interpreted differently in each case. In the next section, I follow up by showing that the X-bar schema manifests the same “golden” recurrence, under a syntactic interpretation.

7.1.1 The golden mean

The Golden number (aka the golden ratio, the golden section, the golden mean) is the quantity $x > 1$, such that when one divides x into two smaller lengths, 1 and $x-1$, the ratio of the larger to the smaller portion is equal to the ratio of the whole to the larger section (see for example Schroeder 1997:5 and Livio 2002: 103). We can represent this geometrically as (3) below.

$$3) \quad x/1 = 1/(x-1) \quad \frac{x}{1} = \frac{1}{x-1}$$

By multiplying both sides by $(x-1)$, we get this:

$$4) \quad x^2 - x = 1$$

(4) can be rearranged to this standard polynomial form:

$$5) \quad x^2 - x - 1 = 0$$

This equation has two solutions:

$$6) \quad x = (1-\sqrt{5})/2, \text{ or } (1+\sqrt{5})/2$$

We are interested in the positive value, which is approximately 1.618, called Phi. Sometimes we will be interested in its reciprocal, called (little) phi, about 0.618; note that the identical series of digits after the decimal in Phi and its reciprocal phi is no accident, but an expression of the defining “golden” property.

Finally, note that we can rewrite the polynomial (5) defining the golden mean as below. This will be significant; as we will see, a parallel relation crops up in other guises, in the recurrence relations describing other “golden” objects.

$$7) \quad x^2 = x^1 + x^0$$

7.1.2 The Fibonacci numbers

As we will see, the golden mean is intimately linked with the Fibonacci sequence and the Golden String (also known as the Fibonacci word). In this section, I review the properties of the Fibonacci numbers that will be relevant below.

The Fibonacci numbers, named after Leonardo da Pisa, though known earlier, are the following sequence of integers:

$$8) \quad 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, \dots$$

One property of this sequence is that, as it continues, the ratio of adjacent elements approaches the golden mean (Schroeder 1997: 4). To draw this out, below I show the ratios of the first few numbers in the sequence, which (slowly⁴) converge to 1.618...

$$9) \quad 1/1, 2/1, 3/2, 5/3, 8/5, 13/8, 21/13, \dots$$

$$1, 2, 1.5, 1.666\dots, 1.6, 1.625, 1.615\dots, \dots$$

Besides just listing them, we can describe these numbers through a recurrence relation, as below. In words, each Fibonacci number (a_n) is simply the sum of the two previous numbers in the sequence (a_{n-1} and a_{n-2}).

$$10) \quad a_n = a_{n-1} + a_{n-2}$$

The Fibonacci numbers have a number of interesting mathematical properties, and appear in numerous domains in nature, most famously in the growth of plants. For example, pinecones and sunflowers typically show geometric arrangements of elements (e.g., scales on the pinecone, or florets in a seedhead), such that connecting adjacent elements produces two sets of spirals, with a Fibonacci number of clockwise spirals and an adjacent Fibonacci number of counterclockwise spirals. The process that leads to this pattern is now relatively well understood, from a mathematical point of view (see e.g.

⁴ The Fibonacci numbers provide successive best approximation to the golden mean for numbers their size, though that number is itself the hardest of all numbers to approximate with small fractions; it is in this sense the “most irrational” number (Schroeder 1997: 4). In the theory of continued fractions, the Fibonacci numbers are the convergents of the golden mean (Livio 2002: 103).

Levitov 1991, and much subsequent work), as well as at a physical level (see especially Douady & Couder 1992). This is not the place to expound on this fascinating topic, but see Uriagereka (1998) for relevant discussion.

As we can see, there are two components to this description of the Fibonacci numbers: the recurrence rule, and a particular choice of “seed values” to begin the sequence. Noting this, we might investigate sequences obtained by making a different choice of seed values, but applying the same recurrence relation. The Lucas numbers are one such series:

$$11)(2,) 1, 3, 4, 7, 11, 18, 29, \dots$$

This sequence also obeys the same recurrence relation as the Fibonacci sequence (i.e., for a_n a Lucas number, $a_n = a_{n-1} + a_{n-2}$). The Fibonacci sequence begins with initial values (0, 1,). Note that choosing (1,1) or (1,2), or any other adjacent pair of successive Fibonacci numbers, will generate (the remainder of) the same sequence. If one chooses different “seed” values, one can get different sequences. Lucas is the first/simplest variant of the pattern, “grown” from seed values of (2, 1) (or (1,3), (3,4), or (4, 7), or any other adjacent pair from the Lucas sequence), the smallest choice of seed values that does not produce the Fibonacci sequence. In phyllotaxis, patterns with Lucas numbers of spirals are the next most common spiral pattern after the Fibonacci patterns (Jean 1994).

As with the Fibonacci numbers, as the sequence continues the ratio of one Lucas number to the previous Lucas number converges on Phi (1.618...); in fact this is true for any choice of seed values for the recurrence relation above (other than (0,0), for which

the ratio of successive terms is undefined). The essence of the “golden” property here, it seems, lies in the recurrence rule, not in what it applies to.

7.1.3 The Golden String

Related to the golden mean and the Fibonacci numbers, there is a binary sequence called the Golden String.⁵ I produce the first portion of the sequence below:

12) 1011010110110...

The Golden String is built by a recurrence relation parallel to the golden polynomial, and the Fibonacci addition relation, but construed as concatenation:

$$13) s_{n+2} = s_{n+1} + s_{n+0}$$

That is, successively longer portions of the Golden String can be grown by concatenating shorter portions. In particular, we start with “seed” strings 1 and 10 as s_0 and s_1 , respectively, and build longer portions as described by the string recurrence rule above.⁶

I illustrate the construction of longer portions of the golden string by this method below:

⁵ This sequence is also called the Rabbit sequence (e.g., in Shroeder 1997:315); the Fibonacci word is often taken to be this sequence with 1s and 0s interchanged, the “binary complement” (i.e. 010010100...). These are sequences A005614 and A003849, respectively, in the online encyclopedia of integer sequences (<http://oeis.org>); as should be clear, what is important is the distribution of digits in this object, not the identities of those digits.

⁶ This line of thinking invites us to consider “Lucas strings”, built by different seeds subjected to the same concatenation rule. The Golden String is “grown” from adjacent seeds in the series 0, 1, 10, 101, 10110, etc. We can grow a distinct series if we start with, say, 1 and 101: 1, 101, 1011, 1011101, 10111011011, and so on. Note that, just as this kind of recurrent string-generating process grows Fibonacci length portions of the Golden String, with these seed values we get Lucas number lengths. The Golden String can be interpreted geometrically as a “cutting” sequence, specifying the order in which a line of constant slope (in this case, Φ) passes units on the y and x axes (<http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fibrab.html#gen2>). The Lucas string described here is also a cutting sequence in this sense, this time for a line with slope $1+\Phi = \Phi^2$.

14) 1, 10, 101, 10110, 10110101, 1011010110110,
 101101011011010110101, ...

Pleasingly, the strings built by this procedure themselves have lengths which are Fibonacci numbers. That is, in (13), the string lengths are 1, 2, 3, 5, 8, 13, 21, etc.

Before moving on, there is a further property of the golden string that is worth noting. Due to its self-similar construction, there are a number of self-generating procedures for the golden string (distinct from the iterated string concatenation shown above), such that reading the sequence left to right generates larger scale left-to-right structure. The pseudo-code algorithm below is one example (this can be found on Ron Knott's site cited above, or here: <http://www.worldofnumbers.com/em118.htm>).

15) {examine the value at a pointer.
 If val=1, append 10 to the end of the string.
 If val=0, append 1 to the end of the string.
 Move the pointer one space right.}
 Repeat.

Beginning with just the first two digits of the sequence (10), with the pointer on the second digit (0, pointer indicated by underline and bold), we have this:

16) 1 0

Since the pointer is at 0, according to (14) we add 1 to the end and move the pointer.

17) 1 0 1

Now the pointer is on 1, so we add 10 and move the pointer.

18) 1 0 1 1 0

And so on:

19) 1011010, 10110101, 1011010110, 10110101101, 1011010110110, etc.

The Golden String thus has a fascinating kind of ‘vertical’ self-similarity at many scales; portions of the sequence at a small scale describe the structure at a larger scale. As a result, we may say that a small portion of the sequence encodes the very procedure used to compute a larger portion of the sequence.

Although I do not pursue the matter here, I will remark that this might be significant in light of the double articulation of language noted since antiquity: its dual life as a linear outer form, and a hierarchically-structured inner form (in simple terms, a natural language expression is simultaneously a string and a tree). This object, in a sense, brings its own double articulation with it; the projection of a syntactic form from its sequence is inherently already there. In other word, there’s already a tree in this string. If it can be shown that the golden string itself in some way characterizes the surface forms produced by adhering to the X-bar schema⁷, we might gain some insight into the nature of the real double articulation of natural language. I set this aside in what follows, returning to the unifying theme of a golden recurrence relation.

⁷ Indeed this is true, in several respects. In Medeiros (2011) I noted that the golden string can be “read off” a maximal X-bar tree truncated at a fixed depth, in several ways. For example, the sequence of most-local specifiers and complements follows the golden string, as does the sequence of head positions, if the latter are marked for whether they occur on the bottom line of the tree or not. Finding the golden string in the “bottom-ness” of heads is particularly intriguing, in light of the proposal that deepest heads are phrasal stress peaks (Chomsky & Halle 1968, Bresnan 1971, 1972, Cinque 1993). In those terms, the golden string property should be visible in the phrasal stress contour, indicating hierarchical properties in easily-detectable string properties. One would then want to examine productions of the implied “golden grammar” (and alternatives) other than maximal truncated trees; preliminary investigation reported in Medeiros (2011) indicates that such a grammar has lowest ambiguity among alternative binary grammars of equivalent complexity. I leave further investigation of such matters to future work.

7.1.4 Golden recurrence

I repeat below the essential recurrence relations characterizing the polynomial specifying the Golden Number, the Fibonacci (and Lucas, etc.) numbers, and the Golden String.

$$20) x^2 = x^1 + x^0$$

$$21) a_{n+2} = a_{n+1} + a_{n+0}$$

$$22) s_{n+2} = s_{n+1} + s_{n+0}$$

To make the comparison complete, note that we can in fact write, for the “golden” polynomial (19) above, the matching general form below (because the factor x^n can be “divided out” from both sides).

$$23) x^{n+2} = x^{n+1} + x^{n+0}$$

Seeing this parallelism of form, we might expect a “golden” syntactic structure to obey the following syntactic recurrence relation, where SO_n is interpreted as the number of syntactic objects of a given type (e.g., heads, or XPs) at depth n in the tree.

$$24) SO_{n+2} = SO_{n+1} + SO_{n+0}$$

In fact, as the next section details, exactly this relation holds, in a tree built by maximal expansion of the X-bar schema. This gives us a pleasing and direct way of expressing the idea that the X-bar format is the “Golden Phrase”.

7.2 Golden recurrence in the X-bar schema

In this section, I point out that the syntactic recurrence relations in the X-bar schema are exactly those which manifest the “golden” recurrence relation discussed at the end of the

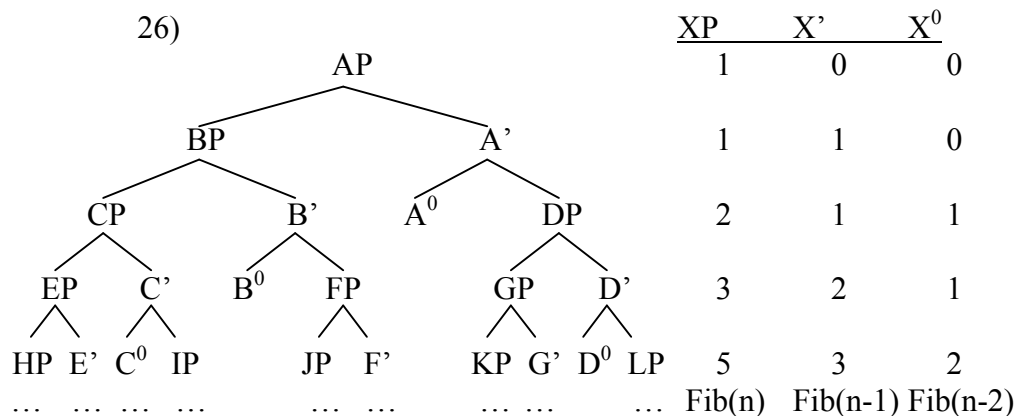
last section. In particular, for various kinds of syntactic objects as defined relative to the X-bar form, we will see that the relation below holds:

$$25) SO_{n+2} = SO_{n+1} + SO_{n+0}$$

That is, the number of objects of any type, on some level of the tree, is simply the sum of the number of such objects on the preceding two lines.

7.2.1 Fibonacci numbers of syntactic categories

As noted in Carnie & Medeiros (2005), the golden recurrence relates the number of X-bar type objects on one line of the tree to the number of such on previous lines. The X-bar schema incorporates three kinds of branching object: a terminal, and two distinct kinds of non-terminals (i.e., XPs and X's, in traditional terms). As indicated in the diagram below, with respect to each of these object types, relation (24) holds).



We can understand why the relevant recurrence relation holds, in the following terms. On any line of this expansion, any phrase is either a specifier or a complement of a higher phrase; there is no other source of new phrases. There is one specifier for each phrase on the previous line, and a complement for each phrase on the line previous to that; thus, the

number of phrases on level n is the sum of the number of such on levels $n-1$ and $n-2$, the desired relation. It follows that the number of X' and X^0 objects must follow the same sequence, since they track the number of phrases on the line above, or the line above that, respectively. Of course, not only do the category types fit the recurrence relation, they moreover fall into the Fibonacci sequence⁸. See Medeiros (2008) for further consequences.

Raising our sights a bit, this section has demonstrated a substantive sense in which the X-bar schema is a golden syntactic form. In particular, it instantiates the same recurrence rule that underlies the golden mean, the Fibonacci numbers, and the golden string, but this time understood as a syntactic recurrence relation. As we will see, there are further relationships between the X-bar phrasal organization and golden mathematics. In the next section, I discuss the idea that this syntactic format represents a kind of minimax solution to conflicting desiderata of phrase structure.

7.3 The X-bar schema as minimax solution

In this section, I return to the notion of two opposite “poles” of binary-branching syntactic form, the Spine and the Bush, that has run throughout this work. In this section, I suggest that the X-bar format is a kind of minimax resolution of irreconcilable requirements on syntactic form, favoring each of the two poles.

⁸ Juan Uriagereka (p.c.) points out that one could define “Lucas” syntactic forms, by analogy with the relationship of the Lucas numbers to the Fibonacci numbers. See section 6.5 below for some relevant remarks.

While the Bush is maximally symmetric, and the Spine is maximally asymmetric, the X-bar form is, in a sense, uniquely antisymmetric; Kayne (1994) motivates (a version of) the X-bar format⁹ in terms of antisymmetry of c-command. The Spine is achieved by maximally local determination of form; each step of the derivation looks just like the last, involving the Merge of a terminal with a complex syntactic object. The Bush can only be built by maximally costly processes sensitive to the global form (see below). The X-bar form is minimally antilocal; just “one step” more complex than the Spine. Between the two poles of monolithic regularity, it represents a kind of optimal compromise in the most irrational form (quite literally, via its connection with the golden mean, the most irrational number).

There is a sense in which the X-bar schema represents the best (perhaps even inevitable) dynamic resolution of fundamentally orthogonal desiderata for phrase structure¹⁰. This can be understood as a push and pull between local and global forces.

⁹ In Kayne’s system, the one specifier shape corresponds to a limit of one on phrasal adjunction. Note, however, that Kayne’s formulation also permits indefinitely deep head adjunction, not considered here, and moreover requires unary branching. See that work for details.

¹⁰ This recalls Binder’s (2008) suggestion that Fibonacci/golden patterns in nature typically reflect “dynamic frustration”, the situation of a system subject to fundamentally opposed tendencies. Consider the case of phyllotaxis, where we might say that the two opposing tendencies are (i) the repulsive dynamics between the individual elements on a local scale, and (ii) the growth along the axis of the meristem. When the stem growth (ii) dominates, the degenerate distichous pattern arises as the “polar” solution: each growth is only repelled by the very last, and so is placed 180 degrees away. This is the pattern of, say, a palm frond, with alternating elements forming a plane collinear with the meristem axis; note that plants with distichous growth exhibit significant spacing between adjacent growths. On the other hand, if there was essentially no stem growth, and a field of mutually repelling growths sprang at once from a uniform field of large extent (i), we would expect them to pattern according to a hexagonal lattice, the shape of a fly’s eye or a honeycomb. The Fibonacci spiral mode, the dominant form in the plant kingdom (cf Jean 1994), robustly arises when new growths appear one-by-one at the meristem fast enough, relative to growth of the stem, to be “repelled” by more than one previous growth (so-called whorled modes typically exhibit double Fibonacci (or Lucas) numbers of spirals, and form when two growths appear at once in the meristem; cf Snow & Snow 1962). The Fibonacci-based form appears to be a robust minimax solution to the “problem” of growth, essentially inevitable for a broad range of growth conditions (see especially Douady & Couder 1992).

At a local level, it is desirable for each step of the computation, examined individually, to be as “minimal” as possible – requiring the fewest sub-operations and abstract memory resources. But globally (in the cost of long-distance hierarchical computations), there is a “force” favoring bushier trees.

These two tendencies are irreconcilable. To build the best possible global tree requires the worst-possible complexity of local computation to achieve via bottom-up structure building. The essential problem is that perfectly bushy trees are perfectly symmetrical; to build a perfectly symmetrical tree bottom-up, one must merge objects of perfectly matched size. Matching size requires indefinitely deep search to decide each step of structure-building.

7.3.1 Local computation, but not too local

Maximally local and maximally simple computation keeps to the head-complement form $\{X^0, YP\}$; these concerns favor the Spine. A considerable amount of recent work converges on the idea that unidirectionally-branching “spinal” structure is the ideal form for syntax (see, e.g., Chomsky 2007, Narita 2010, among a large literature, and as discussed earlier, Yngve 1960).

The essential motivation for this preference is that such trees are as easy as possible to build. The structure-building recipe involved is as simple as possible to achieve discrete infinity; see section 6.5 below for further discussion. They also appear to be optimal for the problem of determining the label, if that is accomplished by search (cf. Cecchetto & Donati 2010): the search for a label is as shallow as possible, and

unambiguous beyond the first step. Uriagereka (1999) pursues a different advantage of spinal structure, namely simplicity of linearization. The reader is referred to those works, and references therein.

However, as noted throughout this work, there is a problem with the Spine: it is the worst possible global structure, with respect to the accumulation of long-distance relations. See Chapter 2 above for further discussion.

7.3.2 Optimal trees, but not too optimal

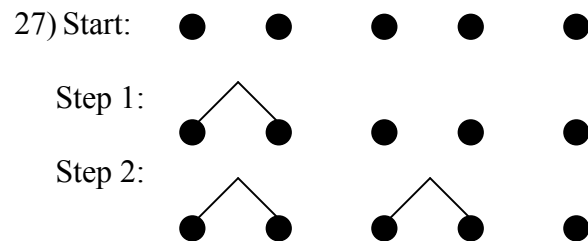
On the other hand, different binary-branching hierarchies incur different costs with respect to the computation of long-distance dependencies over those hierarchies. On that metric, the Bush is the most economical scaffolding, minimizing the number and length of c-command and dominance-based relations, the hierarchical pathways for computations implicit in crucial properties of expressions.

As noted above, the problem with this kind of structure arises when we actually try to build it. It is a maximally symmetric form; in terms of structure-building, that implicitly requires keeping track of the size of various objects that have already been built. Information about every previous stage of the derivation must remain visible in its entirety to every subsequent step.

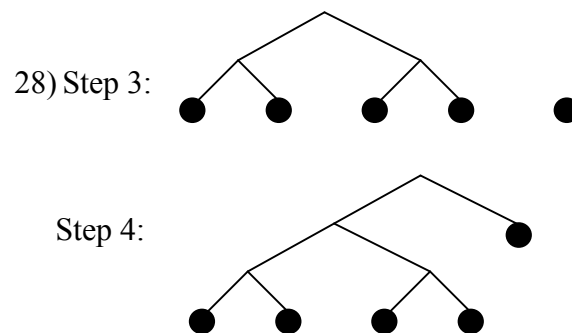
The structure-building problems involved are even more severe than they seem at first. The following simple example illustrates that not only must the interior structure of the operands be checked, but in fact there must be a kind of global coordination of sub-derivations as well. Suppose that the goal is to assemble an arbitrary number of terminal

nodes into a structure as close as possible to the ideal of the Bush. In this case, we are supplied with an input of five terminals.

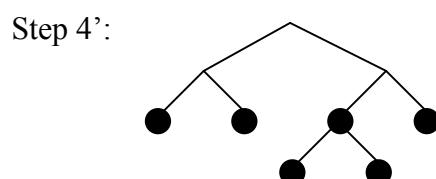
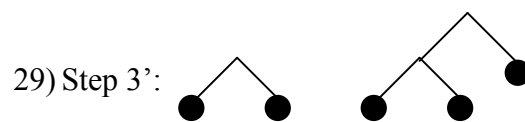
The first stage of the derivation must, of necessity, Merge two terminals. Suppose that the next step also Merged two terminals. The derivation to this point can be represented schematically as follows:



We must be careful in taking the next step. If we choose to Merge the two matching objects (the rule of thumb for building symmetric trees), the derivation proceeds as follows:



This results in a tree with 18 total c-command relations. The better option is to proceed as below, where we break the matching heuristic, Merging a terminal with a pair of terminals, rather than the merging the matching pairs of terminals. This creates the bushiest tree possible, with only 16 total c-command relations.

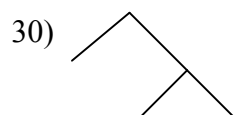


Building the bushiest possible tree for a given number of nodes involves a global coordination of derivational steps; in particular, the progress of various sub-derivations must be kept in appropriate synchrony. As should be clear, this is the most complicated kind of structure-building procedure. What is desirable, it seems, is a compromise that does not require unboundedly large resources to decide what next step to take, yet at the same time builds bushy trees.

7.3.3 *The last bushy spine, the last spiny bush*

Note what is special about the X-bar schema: within its phrasal horizon, it is the treelet with three terminals and two non-terminal nodes. This structure straddles the gap between bush and spine: it is the only possible structure with this amount of material. It is, we might say, the last spinal bush (or bushy spine).

Explicitly, the skeleton of the X-bar schema is the object below:



As should be clear, this is the largest piece of binary-branching structure that is simultaneously the best realization possible of the ideals of the bush and the spine. For this object, and smaller ones, the distinction between spine and bush does not exist. This object literally bridges the gap between the two poles of binary-branching structure.

7.3.4 The largest endocentric form which is still locally best

As remarked in Medeiros (2008), within any class of phrase structural patterns of equivalent local complexity, in the sense above, the pattern representing a generalized X-bar schema – one head, one complement, some fixed number of specifiers – provides the bushiest trees. As noted there, this in effect derives a structural basis for endocentricity: we expect trees to form according to an endocentric, generalized X-bar format because that provides the bushiest, most balanced trees.

However, we can perhaps go beyond that rather general claim, and motivate specifically the one-specifier X-bar schema as unique, “the” natural solution. Even in present terms of choosing one format or another among a class of matched-complexity alternatives (each a distinct “recipe” for structure-building), we can see something special in the one-specifier form.

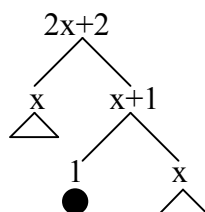
Among the set of generalized X-bar forms, the more complicated forms provide the bushiest trees. The Spine is the trivial generalized X-bar form with a single kind of non-terminal object, the only viable phrase structure pattern in its complexity class. So although the Spine is the worst kind of tree for minimizing c-command relations, it is also, as the only option in this class, trivially also the best. The one-specifier X-bar

schema provides globally better (bushier) trees than the Spine; the two-specifer generalized X-bar form, called “3-bar”, builds bushier trees still. In terms of global evaluation, over complete tree forms, the more complex the molecule, the better the tree that can be built (and within any class, this optimal form corresponds to a generalized X-bar format).

However, the X-bar schema, among the spectrum of globally optimal endocentric (generalized X-bar) structural formats, is the last such to also be locally best among its matched-complexity alternative set. As briefly noted in Medeiros (2008), extremely local changes in the total number of c-command relations, as each recipe is iterated, select X-bar over alternatives from the same complexity class. That is, as one example of an endocentric, generalized X-bar form, the X-bar schema falls in the spectrum of globally-best against a field of matched-complexity competing patterns. However, it is also locally best, taking the best next step at an extremely local scale (within a single iterable phrase structural “molecule”).

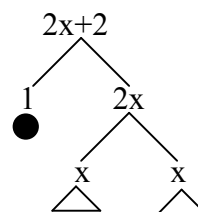
Compare how new c-command relations accumulate in the X-bar format, as opposed to High-headed D-bar, (one manifestation of) the only other non-degenerate pattern in its class (see below). In these diagrams, suppose that the triangles represent unknown but equal amounts of structure, with x nodes. The diagrams count how many new c-command relations are created, within the construction of the characteristic molecule of structure, as a function of x .

31) a. X-bar



3x+3 new c-command relations

b. High-headed D-bar



4x+2 new c-command relations

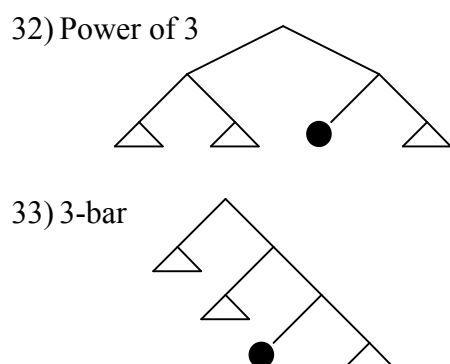
So long as the phrasal objects are internally complex ($x > 1$), fewer new c-command (and dominance) relations accumulate locally in the X-bar pattern than in the HH D-bar pattern, though they both apply to the same kind of input structures. This reflects local optimality, in the intended sense.¹¹

Interestingly, for the globally optimal generalized X-bar forms of higher complexity (corresponding to phrase structural patterns built around an X-bar molecule with one head, one complement phrase, and two or more specifier phrases), this is not the case. That is, while the generalized X-bar forms with two or more specifiers are still globally best, they are not locally best in the way that the one-specifier X-bar schema is. Some other format within their complexity class achieves a better balance within the local molecule, thus is more locally optimal.

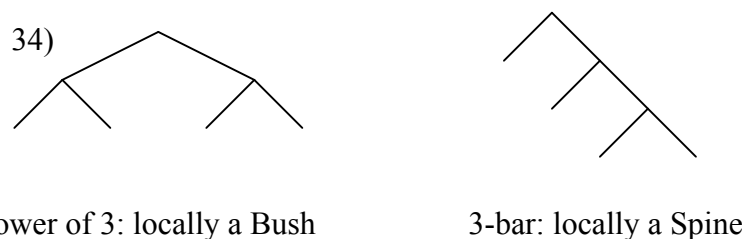
We can understand the issue, again, in terms of the Spine and Bush: a generalized X-bar form aligns the phrases it contains (complements and specifier(s)) into a stack above the head of the molecule. Within the horizon of a single phrase, this is effectively a spine. Within that horizon, a better balance is provided by locally bushy structure. In

¹¹ Note that no fully general local comparison among phrasal patterns, of the sort indicated above, is possible. This is so because different patterns may not take commensurable inputs. For example, both Power of 3 and 3-bar combine three phrasal objects of like shape with a single terminal; other members of their class might combine two terminals and two phrases, or other possibilities. See Chapter 6 for further discussion.

particular, the pattern I call “Power of 3” (see further discussion of this object in 6.5 below) is locally better than the generalized X-bar format called “3-bar”, of equivalent local complexity.

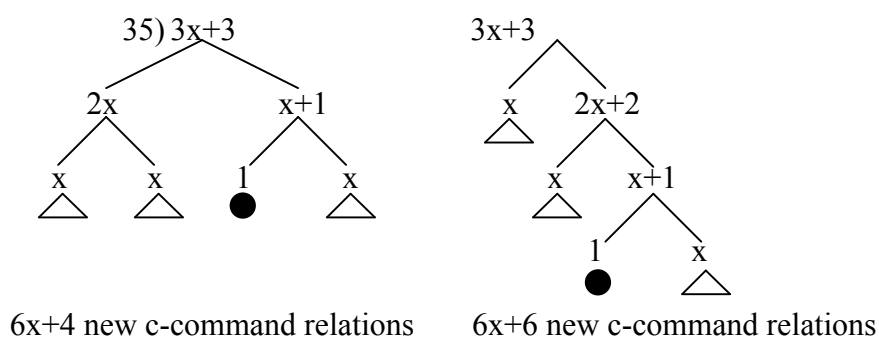


The “local” form of the Power of 3 system is bushy, while 3-bar is locally a Spine. This points to the tension between local and global optimality in the phrase structure systems of greater complexity than X-bar. Within its phrasal horizon, if phrasal off-branches are treated as having equal size, Power of 3 is actually a better choice than 3-bar: fewer new c-command relations are computed within its horizon if that pattern is followed, rather than 3-bar. 3-bar “wins out” only on a global scale, going beyond the phrasal horizon. Below, I depict the “skeleton” of each of the forms in question, within its own phrasal horizon.



Suppose that, within the derivational horizon of the assembly of one or the other of these phrasal molecules, the inputs are H, XP, YP, ZP, where H is a head, and XP, YP, ZP are

complex phrases. Simplifying, assume that XP, YP, and ZP are of equal size, say x nodes. Then by assembling these objects into a Power-of-3 molecule, one adds a total of $6x+10$ new c-command (or dominance) relations, while the 3-bar pattern adds $6x+12$ new vertical relations. The diagram below illustrates the accumulation of vertical relations within these two patterns¹²:



Let me point out immediately that this in no way contradicts the claim that 3-bar makes available the bushiest overall trees; it is still the globally-best system. However, minimization of c-command at a local, dynamic level favors, in this instance, the non-projective Power of 3 format.

To repeat, this is different from the case for X-bar, which is selected both globally (like each generalized X-bar format, within its own class of patterns of matched complexity), but also locally. It is a consequence of the fact that the local skeleton of the X-bar schema is simultaneously a bush and a spine that this holds true.

We see again that there is something special about the X-bar schema, related to the fact that it represents a kind of optimal compromise between concerns pushing toward the

¹² Note that the formulae given below do not compute the number of c-command relations in the whole tree, but rather just the new relations added at this stage. The sum for the full tree would add, to these formulae, a further term for the internal c-command sums of each of the phrasal objects.

Spine as the ideal syntactic structure, and concerns which favor the Bush. In the next section, I divert to explore a conception of this shape as a fractal, noting that it is in a sense the first binary-branching fractal (moreover, a multi-fractal), with “golden” properties expressed in terms of its dimensionality.

7.4 The X-bar schema is a golden fractal

In this section, I examine some geometric properties of a particular conception of the X-bar format as a line-division algorithm. I first discuss a conception of branching patterns as division schemes on a line, and relate that notion to the construction of the so-called Cantor set (see for example Schroeder 1997: 320), the simplest fractal shape of all. I then show that the X-bar schema generates an asymmetric variation on the Cantor set, called the two-scale Cantor set.

One intriguing property of the X-bar schema is that it is, in a sense defined below, the ‘first’ (or simplest) kind of binary-branching (multi-)fractal. Fractals are self-similar objects of often non-whole-number dimension. Fractal patterns in nature are well known, so it should not be that surprising to find such a property underlying the structures of natural language. The defining property of such objects is that their ‘size’ depends on the scale at which they are measured. One of the seminal papers in the study of fractals (Mandelbrot 1967) was entitled, “How long is the coast of Great Britain?” As observed there, the answer to this question depends on the scale at which one measures; as ever-finer structure is considered, the length computed increases without converging on a constant value. In this light, some might object to calling natural language fractal, for

surely real expressions are digital, ‘bottoming out’ with an invariant absolute size.¹³ This is nothing more than the familiar *discreteness* property of natural language. Regardless, the point here is one about the abstract (infinite) space of possible tree-forms that the X-bar schema generates, which is indeed fractal ‘all the way down’.

7.4.1 Phrasal patterns as line division algorithms

In defining a notion of dimensionality for phrase structure patterns, there is an immediate problem. In particular, it is not clear how to compute such a measure if we assume that one node in the tree is “as big as” any other. Instead, we wish to find a way of thinking about these patterns that incorporates a straightforward notion of scaling. I will suggest that the natural implementation is to interpret binary branching as geometric halving of a line segment. The intended correspondence can be easily understood with a simple example; in fact, the simplest conceivable example of a fractal, the Cantor Set.

7.4.2 The Cantor Set

Consider the following geometric construction. We begin with a line segment, corresponding to the interval $[0,1]$. We then remove the middle third, leaving us the two intervals $[0, 1/3]$ and $[2/3, 1]$. We then remove the middle third of those intervals, and so on, *ad infinitum*. The first few steps in this process are sketched below:

¹³ Note that recent developments in syntactic theory (notably, the micro-comparative syntax of cartography, and Starke’s so-called nano-syntax) suppose that the atoms of syntactic combination are much smaller than previously thought, probably smaller than morphemes. Even so, that amounts only to a revision of the granularity of syntactic structures; such approaches hardly deny the fundamentally digital character of natural language.

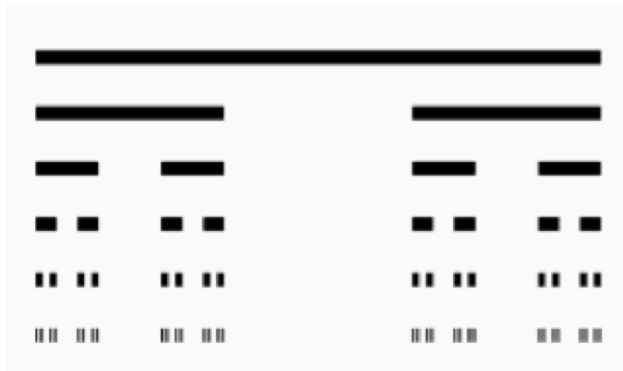


Figure 4: Iterative construction of the Cantor Set.

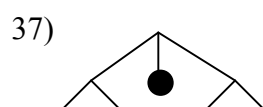
This process, in the limit, generates an infinite ‘dust’ of points. At each generation, we have two one-third scale copies of the whole figure; thus one can compute the so-called “box dimension” in the usual way, according to this formula:

$$36) \frac{\ln(\text{number of copies})}{\ln(\text{scale factor})}$$

According to the construction just described, at each generation we have 2 copies of the whole, with a scale factor of 3. The box dimension of the (one-scale) Cantor set is then $\ln(2)/\ln(3)$ (equivalently, $\log_3 2$), about .631 (Schroeder 1997: 320, 322). Thus, the figure has a dimensionality in between that of a point (dimension = 0) and a full line (dimension = 1), as expected.

Moreover, this is in a clear sense the “first” or simplest kind of fractal. That is, the background dimension cannot be lower than 1 (division of a zero-dimensional point does not make any sense), as it is here. Three is the smallest denominator for a division scheme that results in fractal structure. In particular, removal of halves of a line segment (i.e., division by 2) would generate, in the limit, a single point (dimension 0, not a fractal).

The iterative construction of the Cantor set discussed above invites a natural kind of “phrasal” analysis. In particular, we can describe the recursive construction of the Cantor set in terms of the following object:



That is, the removal of the middle third of a line segment corresponds to an iterated syntactic form with ternary branching, with two further phrases and a terminal in the middle. It is natural to identify syntactic terminals with removed segments in the geometric construction, because neither contains further structure subject to division. The syntactic non-terminals, on the other hand, are the pieces of structure that are still “live”, with internal structure relevant to the next iteration of the division process. Note also that the syntactic branching corresponds directly to geometric division; in this case, we have ternary branching in the phrasal form matching division by thirds on the line segment.

7.4.3 The image of X-bar structure is an Asymmetric Cantor Set

Continuing with the idea of a correspondence between iterated geometric division and syntactic patterns, let us mark out how the expanded X-bar schema divides the line segment. We perform this mapping in a natural way: just as we took ternary branching in the construction of the Cantor set to correspond to division of the line segment into thirds, we will take binary branching in the tree to correspond to halving in the line segment. The object at the root is mapped to the interval $[0, 1]$; the left daughter of the root is

mapped to $[0, \frac{1}{2}]$, and the right daughter to $[\frac{1}{2}, 1]$ (see fn 4 below). Further binary branching within each daughter corresponds to a further division in half of the branching object's interval. A syntactic terminal, in terms of branching geometry, is a branch permitting no further internal growth. In the interval map, we delete the intervals corresponding to head positions (terminals), and consider only the branching residue.

The figure below illustrates the intended mapping scheme applied to a single X-bar phrase. In this instance, the specifier ZP is mapped to $[0, \frac{1}{2}]$, while the complement YP is mapped to $[\frac{3}{4}, 1]$; the non-branching X^0 is mapped to $(\frac{1}{2}, \frac{3}{4})$.¹⁴

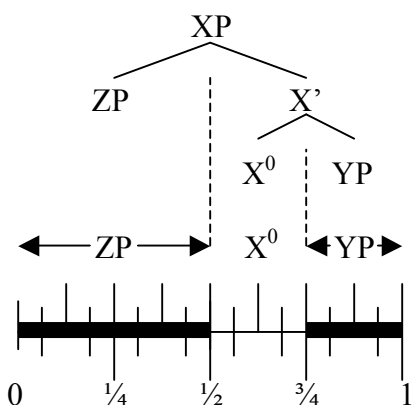


Figure 5: First stage of mapping X-bar form to line segment.

Of course, ZP and YP themselves have the same structure as the root XP:

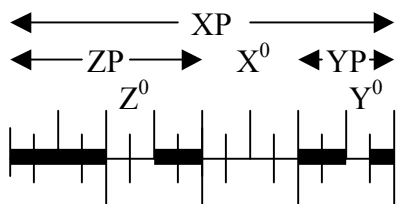


Figure 6: Second stage of mapping X-bar form to line segment.

¹⁴ Mapping branching categories to closed intervals $[x, y]$, and terminals to open categories (x, y) – given the way that the X-bar scheme places heads in the interior of their phrases – ensures that this scheme will not map a single point to non-overlapping syntactic categories (with the exception of “bar-level” categories). Put another way, if point x is in the interval corresponding to distinct syntactic objects X and Y , then either X dominates Y or Y dominates X . This ensures that the mapping is mathematically well-behaved.

Continuing this process *ad infinitum* generates a ‘two-scale’ Cantor set, below.



Figure 7: Iterative construction of two-scale Cantor set.

Examining the two-scale Cantor set, note that it is created by having two copies of the whole at different scales at each generation, with one half-scale and one quarter-scale copy. The figure can be generated by iteratively removing the third quarter of each line segment; in linguistic terms, ‘removing a segment’ means having a terminal in that position (preventing any further internal subdivision within its own boundaries). In other words, the half-scale copy of the whole is the specifier, the quarter-scale, the complement, and the removed segment, the head.

Now, technically any difference in scale between the two copies would result in a two-scale Cantor set. But there is something decidedly natural about the choice of one half-scale and one quarter-scale copy, especially in light of the background assumption of strict binary branching: there is just no simpler way of getting off-scale copies under that assumption.¹⁵

15 Going further, there is a sense in which this (X-bar/2-scale Cantor set) shape is the first non-trivial binary-branching fractal of any sort (not just the first multi-fractal). That is, it is the smallest kind of self-similar binary-branching object whose image on the line is neither the full line, nor a single point. For example, consider the options utilizing a smaller/more local recursive schema, i.e. whose characteristic structure can be fully described in terms of a single level of embedding. The options are: $XP \rightarrow [X^0 Y^0]$ (not a growth scheme, this shape projects no image on the line at all); $XP \rightarrow [X^0 YP]$ (iterated head-complement structure, what I have been calling the Spine: the image on the line of this shape is a single point, dimension 0); and $XP \rightarrow [YP ZP]$, a scheme determining no terminal locations at all, and whose

7.4.4 *The golden dimension of the X-bar form*

This figure, perhaps unsurprisingly, has Hausdorff dimension defined in terms of the golden mean. Specifically, its dimensionality is $\ln(\phi \sim 1.618...)/\ln(2)$, or equivalently, $\log_2(\phi)$, about .694 (see Tsang 1986: 1390).¹⁶ Technically, the figure is a multi-fractal; various ways of computing its dimension do not agree as they do for simple fractals, and the object properly speaking has a spectrum of dimensions. For example, the simple formula used to compute the box dimension of the Cantor set cannot be applied, because there is no single “scale factor” that describes the copies of the whole at each generation (see fn. 16).

I leave a further exploration of these properties to further work. For now, the point is that this gives us yet another way of expressing the intuition that the X-bar schema is the Golden Phrase. In the next section, I turn to a closer examination of other phrasal arrangements, defining a notion of “growth factor” in terms of a matrix formulation of phrasal recurrence relations. As one might already guess, the growth factor of the X-bar format is just ϕ , the golden mean. As we will see, the matrix formulation is of interest and considerable utility in itself, allowing us to draw connections, for example, between

image is the full line, dimension = 1. Allowing a recursive scheme with depth 2 provides access to X-bar organization, as well as “D-bar” organization (see chapter 4), e.g. $XP \rightarrow [[X^0 WP][Y^0 ZP]]$. Such a scheme, when mapped to the line segment as described above, produces a (fractal only, not multi-fractal) figure with two one-quarter-scale copies of the whole at each generation (so, with box dimension = .5).

¹⁶ Note that the box dimension cannot be computed here according to the formula above. However, one interpretation of the dimension of this object is to say that it has $\phi^{1/2}$ scale copies at each generation. This accords with the comments on “growth factor” in section 6.5 below, effectively the limit of the ratio of the number of items on one level of a tree to the number on the previous line. Successive lines of the tree correspond, in the line segment mapping, to half-scale copies; the fact that the X-bar form has a growth factor of ϕ is then another way of expressing what this dimensional measure states. Going further, I make the natural conjecture that for the non-terminal image on the line of an arbitrary non-degenerate phrasal pattern with growth factor G is $\log_2 G$.

phrasal patterns that compose simpler patterns, and factorization of polynomials associated with those patterns.

7.5 Golden growth in X-bar phrase structure

This section provides another way of formulating the deep mathematical connection between the X-bar schema and the golden mean. In particular, by describing various phrasal patterns in terms of matrices, we find a very simple and natural way of quantifying their growth properties, via a number associated with each matrix that I will call the “growth factor” of each syntactic pattern.

Along the way, this investigation of phrase structure allows us to see further interesting facts. For example, we will see that some superficially distinct phrasal patterns really reflect the same underlying pattern; I investigate a conception of phrasal growth as iterated linear transformation (i.e., matrix multiplication) applying to some “seed” vector, with different choices of seed vectors leading to superficial variety. We will also see a connection, for “degenerate” phrasal patterns that can be described as the composition of simpler patterns, between syntactic composition and the factorization of the associated polynomials.

7.5.1 Defining a notion of ‘growth factor’

The goal of this section is to find a way of quantifying the “growth” of distinct syntactic patterns. Intuitively, the desired notion of “growth factor” describes the ratio of the number of syntactic objects on one line of a maximally expanded tree built from some

phrasal pattern, to the number of syntactic objects on the preceding line.¹⁷ That is, we wish to define the growth factor as follows:

$$38) \lim_{n \rightarrow \infty} \frac{(\text{number of syntactic objects on line } n)}{(\text{number of syntactic objects on line } n-1)}$$

In the case of the X-bar schema, it turns out that we already have what we need to see that its growth factor will turn out to be the golden mean. Recall from section 6.2 above the observation that the number of each kind of X-bar object on successive lines of a maximal X-bar tree form the Fibonacci sequence. Knowing that the limit of the ratio of one Fibonacci number to the preceding one is Phi (see section 6.1.2 for discussion), it follows that the growth factor for the X-bar form is also Phi.

However, that is a bit unsatisfying. In particular, at this point we do not have a fully general notion of how to find the growth factor directly for phrasal arrangements other than the X-bar form. Such a notion would be useful in understanding the phrasal terrain a bit better. Put another way, since the goal of this chapter is to argue that the X-bar schema is in some sense the “best possible” structure, it will pay to think carefully about what else is possible, and to have some handle on the properties of every other kind of phrasal organization. As we will see, the key to capturing the relevant properties lies in formulating syntactic patterns as matrices.

¹⁷ Note that we are moving away from the geometric scaling discussed in the last section, necessary to find a usable notion of fractal dimension for syntactic patterns, back to a conception where every node counts equally.

7.5.2 Expressing phrase structure patterns as matrices

In this section, I discuss the description of phrasal recurrence relations in terms of matrices. To help orient us, it will be helpful to review the description of the X-bar schema that has been given previously. To this point, we have described the relevant pattern through the use of abstract phrase structure rules (see chapter 3), and through tree diagrams. I reproduce both of these descriptions of the X-bar pattern below.

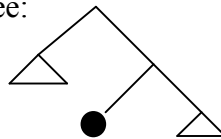
39) X-bar

PSRs:

$2 \rightarrow 2 \ 1$

$1 \rightarrow 0 \ 2$

Tree:



If we are willing to give up on representing linear order as part of our phrase structure description, there is a very natural way of translating these descriptions into matrix form.

40) X-bar

PSRs:

$2 \rightarrow 2 \ 1$

$1 \rightarrow 0 \ 2$

Matrix:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

The rows of the matrix match the phrase structure rule on the same line; both the rule and the row in the matrix describe what a given kind of non-terminal object (an input) dominates (its output). The columns correspond to non-terminals in the output, with the order among columns matching the order of the rows. However, the elements in the matrix themselves are the count of the number of each type of object produced.¹⁸ For the X-bar form, the first row in the matrix is 11; this corresponds to the phrase structure rule $2 \rightarrow 2 \ 1$. This means that when expanding the category characterized by the first row

¹⁸ Note that the matrix form does not encode the terminal positions explicitly. Instead, terminals appear as an absence; a non-terminal dominates a terminal if the entries in its corresponding row sum to less than 2.

(the root-type object), one gets 1 of the categories specified in the first row (2, a root-type object) and one of the categories in the second row (1). The second row of the X-bar matrix, 1 0, corresponds to the head-complement $1 \rightarrow 0\ 2$ phrase structure rule, with one of the first-row non-terminal objects (an XP), and zero second-row non-terminal objects (no X's). To avoid confusion, note that the numbers in the phrase structure rules are arbitrary labels for types of syntactic objects; the numbers in the matrix count the number of those objects.¹⁹ The numbers in the phrase structure rules correspond to positions (rows and columns) in the matrix.

7.5.3 *The growth factor is the characteristic root*

Having struck on this formulation, we are now in a position to use the tools of linear algebra to describe the syntactic patterns at issue. A first step in this direction is to note that, associated with each matrix, there is a polynomial, called the characteristic polynomial of the matrix.

In the case of the X-bar form, the characteristic polynomial is, unsurprisingly by this point, the “golden” polynomial describing the golden mean (see section 7.1 above):

¹⁹ This invites a straightforward characterization of the structural patterns at issue. Namely, the set of phrase structural systems of n non-terminal types is (a subset of) the set of $n \times n$ matrices with whole number elements, such that the sum of the elements in any row is at most 2 (up to 2, because of binary branching; or less, because a non-terminal may immediately dominate one or two terminals). To qualify as phrase structure systems in the relevant sense, a number of further conditions must be met, which can again readily be formulated as conditions on matrices. One condition is that it includes terminal positions; in terms of the matrix, at least one row must sum to less than 2. Another condition is that the root category must be the analogue of a “cyclic generator”, in the group-theoretic sense, of the set of non-terminal categories (including itself). That is, the root category must dominate (some category that dominates...) each non-terminal type; this rules out the case where there are, in effect, disjoint cycles, with the root leading only to some proper subset of the non-terminal categories specified by the phrase structure rules. Note that the “Pair of Spines”, specified by PSRs $2 \rightarrow 1\ 1$, $1 \rightarrow 0\ 1$, fails this condition: the configuration at the root does not occur anywhere else in the tree.

$$41) x^2 - x - 1$$

We can describe the desired notion of growth factor in terms of the characteristic polynomial. It is simply the largest real root of the characteristic polynomial (algebraically, the largest real k such that $(x - k)$ is a factor of the polynomial).²⁰ This quantity is called the characteristic root; it is identified with the dominant eigenvalue of the matrix (for discussion of these terms, the reader is referred to any basic textbook on linear algebra).

For the X-bar example, the largest positive root of the characteristic polynomial (its characteristic root, and the dominant eigenvalue) is the golden number Φ . Thus, it is quite correct to say that the X-bar format exhibits “golden” growth. Note that the growth factor for any binary-branching syntactic pattern is a number in $[1, 2)$: the bottom boundary 1 corresponds to the Spine (where each level in the tree is an exact copy of the level above it), and 2 to the Bush (where each level of the tree exactly doubles the material on the previous level).

7.5.4 *Growth factors by complexity class*

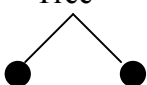
It is of some interest to consider what kinds of growth, as quantified by the growth factor just discussed, characterize other, less familiar phrasal patterns. In what follows, I examine the growth factor of systems simpler than the X-bar form, those of equivalent complexity, and those in the class “one step beyond” X-bar. I do not explore patterns any more complex than that in this work.

²⁰ Complications arise for “degenerate” systems, which may have some subparts that grow at different rates. See below for more discussion of such forms.

7.5.4.1 One non-terminal: pair, spine, or bush


The simplest kind of syntactic patterns are built around a single kind of non-terminal object. The choices here are quite constrained. Indeed, if we insist that a syntactic pattern must provide discrete infinity (understood in the present context as a requirement for designated positions for terminals, as well as indefinite embedding of further non-terminals), only one option in this class is viable, the one corresponding to the Spine. Nevertheless, I illustrate the two non-viable options as well. I present descriptions of these patterns in terms of abstract phrase structure rules, matrices, and trees, indicating the characteristic polynomial and growth factor (characteristic root) of each.

42) The Pair

PSR	Matrix	Tree
$1 \rightarrow 0 \ 0$	$\begin{bmatrix} 0 \end{bmatrix}$	

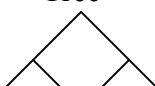
Characteristic polynomial: $x - 0$
 Growth factor: 0

43) The Spine

PSR	Matrix	Tree
$1 \rightarrow 0 \ 1$	$\begin{bmatrix} 1 \end{bmatrix}$	

Characteristic polynomial: $x - 1$
 Growth factor: 1

44) The Bush

PSR	Matrix	Tree
$1 \rightarrow 1 \ 1$	$\begin{bmatrix} 2 \end{bmatrix}$	

Characteristic polynomial: $x - 2$
 Growth factor: 2

The members in this class are quite simple. Note that the Pair and the Bush are not discrete infinite systems: the former does not allow for indefinite growth, while the latter does not provide designated locations for terminals.

7.5.4.2 Two non-terminals: The X-bar class

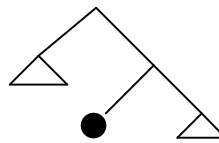
The structural possibilities become more interesting in the next class of phrasal patterns, defined over two non-terminal types. While the systems with one non-terminal type were described with rather dull one-element matrices (e.g., [1] for the Spine), this class is described by 2x2 matrices.

45) X-bar

PSRs: Matrix:

$$\begin{array}{l} 2 \rightarrow 2 \ 1 \\ 1 \rightarrow 0 \ 2 \end{array} \quad \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Characteristic polynomial: $x^2 - x - 1$
 Growth factor: $\varphi \sim 1.618$

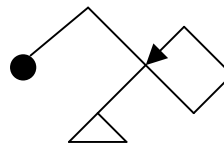


46) High-headed X-bar

PSRs: Matrix:

$$\begin{array}{l} 2 \rightarrow 0 \ 1 \\ 1 \rightarrow 1 \ 2 \end{array} \quad \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$$

Characteristic polynomial: $x^2 - x - 1$
 Growth factor: $\varphi \sim 1.618$

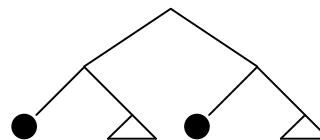


47) D-bar

PSRs: Matrix:

$$\begin{array}{l} 2 \rightarrow 1 \ 1 \\ 1 \rightarrow 0 \ 2 \end{array} \quad \begin{bmatrix} 0 & 2 \\ 1 & 0 \end{bmatrix}$$

Characteristic polynomial: $x^2 - 2$
 Growth factor: $\sqrt{2} \sim 1.414$

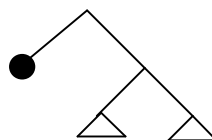


48) High-headed D-bar

PSRs: Matrix:

$$\begin{array}{l} 2 \rightarrow 0 \ 1 \\ 1 \rightarrow 2 \ 2 \end{array} \quad \begin{bmatrix} 0 & 1 \\ 2 & 0 \end{bmatrix}$$

Characteristic polynomial: $x^2 - 2$
 Growth factor: $\sqrt{2} \sim 1.414$



Notice that the four patterns above consist of two pairs of “siblings” (e.g., X-bar and high-headed X-bar are one pair of siblings), where the superficial form is different, but the polynomial and growth factor are identical. In fact these sibling patterns are really the same pattern, as discussed below.

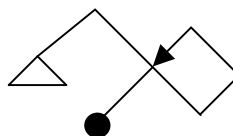
The last two patterns in this class are “degenerate”: they incorporate subtrees drawn from a simpler class of phrase structures. In this case, they are built by composing the Spine with itself, or with the Pair.

49) Spine of Spines

PSRs: Matrix:

$$\begin{array}{l} 2 \rightarrow 2 \ 1 \\ 1 \rightarrow 0 \ 1 \end{array} \quad \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

Characteristic polynomial: $x^2 - 2x + 1$
 Growth factor: 1

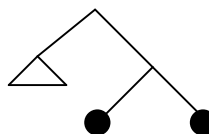


50) Spine of Pairs

PSRs: Matrix:

$$\begin{array}{l} 2 \rightarrow 2 \ 1 \\ 1 \rightarrow 0 \ 0 \end{array} \quad \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$$

Characteristic polynomial: $x^2 - x$
 Growth factor: 1



That exhausts the viable possibilities in this class. Note that a seventh possibility, a Pair of Spines, is ruled out by failing to be self-similar; the root does not dominate any further root-like categories.

7.5.4.3 Orientation families

In the representations of “sibling” systems given above, the matrix form is different (but the polynomial and growth factor are the same). However, we can recast things slightly to see that in effect these systems are really two views of the same pattern, merely oriented differently with respect to the root.

Top-down maximal growth of a given phrase structure system can be understood as repeated linear transformation of a “seed vector” representing whatever non-terminal category is placed at the root. This column vector is multiplied on the left by the phrase structure matrix, producing a new vector: the first vector counts the number of objects of each non-terminal type on one line of the tree, and the product with the matrix yields a vector counting non-terminal types on the succeeding line of the tree. Let us call this the “accumulation vector”. The general form is this, where A is the phrase structure matrix, and x_i is an accumulation vector:

$$51) Ax_i = x_{i+1}$$

In these terms, we can represent X-bar and High-headed X-bar with the very same matrix²¹, but different choices of seed vector (in effect, X-bar is this pattern grown from an XP at the root; high-headed X-bar is the same pattern with the X-bar non-terminal type at the root). This means that X-bar has as its seed this column vector:

²¹ In this case, I choose the X-bar matrix as the more basic form. We could as well take the matrix for the High-headed form as basic, and express X-bar as iterated linear transformations by that matrix of a seed vector (0,1). This reinforces the point that there is no reification of bar-levels as anything other than a notational convenience.

$$52) \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Meanwhile, High-headed X-bar has this seed vector:

$$53) \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

We can then understand the growth of these syntactic patterns, as the maximal tree is grown from the root, in terms of a series of linear transformations. The seed vector is multiplied by the phrase structure matrix (on the left); the output vector undergoes the same multiplication, etc., yielding a series of column vectors representing the number of non-terminal types at each level of the tree.

For example, the first iteration of “growth” of X-bar translates to this matrix multiplication:

$$54) \begin{array}{ccccc} \text{A} & & x_0 & = & x_1 \\ \text{Matrix} & \text{X} & \text{Seed vector} & = & \text{First Accumulation vector} \\ \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} & & \begin{bmatrix} 1 \\ 0 \end{bmatrix} & & \begin{bmatrix} 1 \\ 1 \end{bmatrix} \end{array}$$

Iterating this multiplication, we get the following sequence of accumulation vectors:

$$55) \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \end{bmatrix} \begin{bmatrix} 5 \\ 3 \end{bmatrix} \begin{bmatrix} 8 \\ 5 \end{bmatrix} \begin{bmatrix} 13 \\ 8 \end{bmatrix} \begin{bmatrix} 21 \\ 13 \end{bmatrix} \begin{bmatrix} 34 \\ 21 \end{bmatrix}$$

Thus, the accumulation vectors for the X-bar pattern take the form of two consecutive Fibonacci numbers.

On the other hand, as stated, high-headed X-bar starts with (0,1) as a column vector, creating the following sequence of accumulation vectors (each a sum of the number of each type of non-terminal syntactic object, on successive lines of the maximal tree grown by that pattern):

$$56) \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \end{bmatrix} \begin{bmatrix} 5 \\ 3 \end{bmatrix} \begin{bmatrix} 8 \\ 5 \end{bmatrix} \begin{bmatrix} 13 \\ 8 \end{bmatrix} \begin{bmatrix} 21 \\ 13 \end{bmatrix}$$

As a final note, observe that with different seed vectors (representing an anomalous configuration at the root of the tree), we can get accumulation vectors with numbers drawn from distinct Fibonacci-like sequences (those that share the same recurrence relation, but start with different seed values), for example the Lucas numbers. Thus, if near the root there is a configuration with one XP-type and two X'-type objects, corresponding to (1,2) as a seed vector, we get the following sequence of accumulation vectors, where the elements are adjacent Lucas numbers:

$$57) \begin{bmatrix} 1 \\ 2 \end{bmatrix} \begin{bmatrix} 3 \\ 1 \end{bmatrix} \begin{bmatrix} 4 \\ 3 \end{bmatrix} \begin{bmatrix} 7 \\ 4 \end{bmatrix} \begin{bmatrix} 11 \\ 7 \end{bmatrix} \begin{bmatrix} 18 \\ 11 \end{bmatrix} \begin{bmatrix} 29 \\ 18 \end{bmatrix} \begin{bmatrix} 47 \\ 29 \end{bmatrix} \begin{bmatrix} 76 \\ 47 \end{bmatrix}$$

Before concluding this section, I provide an intuitive demonstration of how and why the characteristic root (the dominant eigenvalue) gives the growth factor in the desired sense. To repeat, maximal iteration of a phrasal pattern resolves as iterated multiplication of a vector by the phrase structure matrix. We can understand these $n \times n$ matrices as linear transformations mapping \mathbb{R}^n to \mathbb{R}^n .²² In these terms, examining successive lines of the maximal expansion of a given pattern amounts to tracking the trajectory of an initial point (the configuration at the root) under iteration of the map. Understood this way, we can understand directly why the dominant eigenvalue describes the phrasal growth.

Start with an arbitrary vector with non-negative components. Recall that these components express the number of each non-terminal type at a given level of the tree. In

²² The entries in the matrices, and in the vectors, are always non-negative integers (one cannot grow “half a node”, for example). However, to understand the action of the linear transformation with respect to an eigenbasis, it will be useful to consider them to act over the reals rather than the integers; in particular, the eigenvectors are often expressed in terms of non-integers.

geometric terms, we may think of this as a linear combination of independent basis vectors x, y, \dots (one for each different type of non-terminal):

$$58) ax + by \dots$$

We may express these instead as a linear combination of eigenvectors (a standard technique), finding appropriate coefficients c, d , etc.:

$$59) ax + by \dots = cv_1 + dv_2 \dots$$

Suppose v_1 has eigenvalue λ_1 , eigenvector v_2 has eigenvalue λ_2 , etc. Then multiplication by the matrix n times has a particularly nice expression in terms of the eigenvector basis:

$$60) \lambda_1^n cv_1 + \lambda_2^n dv_2 \dots$$

Suppose λ_1 is the largest eigenvalue; then it is clear that as n increases, the sum of component vectors converges on $\lambda_1^n cv_1$ (for non-zero c).

I leave the matter here. In the next subsection, I discuss the phrase structural possibilities that arise once we allow three non-terminal types.

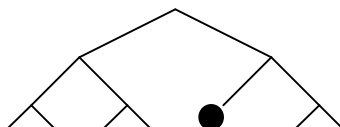
7.5.4.4 Three non-terminals

With an additional non-terminal object, considerably more patterns are available: 57 superficially distinct patterns. I illustrate a handful of these systems in this section, then present a table summarizing the characteristic polynomial and growth factor for each of the distinct underlying patterns. As might be expected, with three non-terminal types each underlying pattern can have up to three superficial manifestations, by permuting which non-terminal is placed at the root. These systems are furthermore identified with a catalogue number, corresponding to a numbering system in a different work. It is

included here as a way to cross-reference between the individual presentations of some of these patterns, and the fuller table at the end summarizing the growth factors of all of the non-degenerate patterns in this class.

61) 2 Power of 3

$$\begin{array}{l} 3 \rightarrow 1 \ 2 \\ 2 \rightarrow 0 \ 3 \\ 1 \rightarrow 3 \ 3 \end{array} \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 2 & 0 & 0 \end{bmatrix}$$

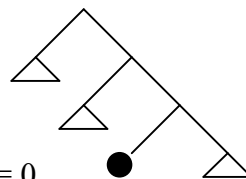


Characteristic polynomial: $x^3 - 3x = 0$

Growth factor: $\sqrt[3]{3} = 1.732\dots$

62) 9 3-bar (Generalized X-bar format with two specifiers)

$$\begin{array}{l} 3 \rightarrow 2 \ 3 \\ 2 \rightarrow 1 \ 3 \\ 1 \rightarrow 0 \ 3 \end{array} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

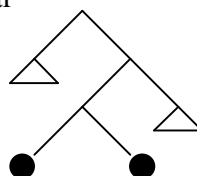


Characteristic polynomial: $x^3 - x^2 - x - 1 = 0$

Growth factor: the "Tribonacci" constant, $\sim 1.839\dots$

63) 14 Double-headed X-bar

$$\begin{array}{l} 3 \rightarrow 1 \ 3 \\ 2 \rightarrow 0 \ 0 \\ 1 \rightarrow 2 \ 3 \end{array} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

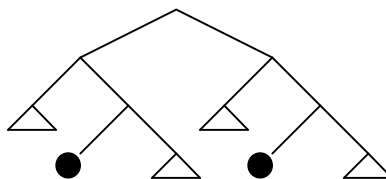


Characteristic polynomial: $x^3 - x^2 - x$

Growth factor: Phi, 1.618...

64) 24

$$\begin{array}{l} 3 \rightarrow 2 \ 2 \\ 2 \rightarrow 1 \ 3 \\ 1 \rightarrow 0 \ 3 \end{array} \begin{bmatrix} 0 & 2 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

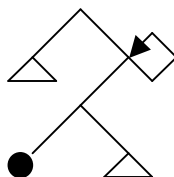


Characteristic polynomial: $x^3 - 2x - 2$

Growth factor: 1.769...

65) 26

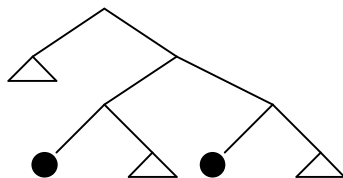
$$\begin{array}{l} 3 \rightarrow 2 \ 3 \\ 2 \rightarrow 1 \ 2 \\ 1 \rightarrow 0 \ 3 \end{array} \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$



Characteristic polynomial: $x^3 - 2x^2 + x - 1$
 Growth factor: $1.7548... (\rho^2)$

66) 29

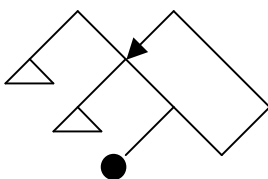
$$\begin{array}{l} 3 \rightarrow 2 \ 3 \\ 2 \rightarrow 1 \ 1 \\ 1 \rightarrow 0 \ 3 \end{array} \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 2 \\ 1 & 0 & 0 \end{bmatrix}$$



Characteristic polynomial: $x^3 - x^2 - 2$
 Growth Factor: $1.6956...$

67) 39

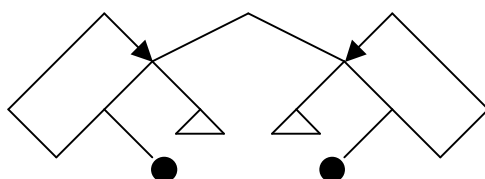
$$\begin{array}{l} 3 \rightarrow 2 \ 3 \\ 2 \rightarrow 1 \ 3 \\ 1 \rightarrow 0 \ 2 \end{array} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



Characteristic polynomial: $x^3 - x^2 - 2x + 1$
 Growth Factor: $1.8019...$

68) 40

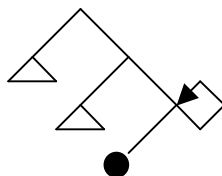
$$\begin{array}{l} 3 \rightarrow 2 \ 2 \\ 2 \rightarrow 1 \ 3 \\ 1 \rightarrow 0 \ 2 \end{array} \begin{bmatrix} 0 & 2 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



Characteristic polynomial: $x^3 - 3x$
 Growth factor: $\sqrt{3} = 1.732...$

69) H X-bar of Spines

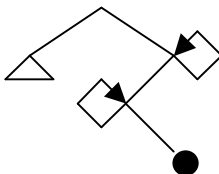
$$\begin{array}{l} 3 \rightarrow 2 \ 3 \\ 2 \rightarrow 1 \ 3 \\ 1 \rightarrow 0 \ 1 \end{array} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$



Characteristic polynomial: $x^3 - 2x^2 + 1$
 Growth factor: $\Phi = 1.618...$

70) K Spine of Spines of Spines

$$\begin{array}{l} 3 \rightarrow 2 \ 3 \\ 2 \rightarrow 1 \ 2 \\ 1 \rightarrow 0 \ 1 \end{array} \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$



Characteristic polynomial: $x^3 - 3x^2 + 3x - 1$

Growth factor: 1

I give below a table collecting the orientation families of the non-degenerate systems from this class. They are sorted by their growth factor; I also list the characteristic polynomial for these families, and note mathematical properties of the growth factor. Many, but not all of the growth factors are Pisot numbers (also known as Pisot-Vijayaraghavan or PV numbers), which are algebraic integers (real solutions of polynomials with integer coefficients) where the other roots of the polynomial are all of magnitude less than one (lying within the unit disk on the complex plane).

Note that the last entry, the tribonacci family including 3-bar, has the largest growth factor in this set. That is another way of expressing the “global optimality” of this system relative to alternative phrasal forms built with the same number of non-terminals, as discussed in Chapter 6.

Systems in family	Polynomial	Growth factor	Special notes
7, 32, 34	$x^3 - 2$	1.2599	$\sqrt[3]{2}$, non-Pisot
13, 25, 35	$x^3 - x - 1$	1.3247	Plastic number <i>rho</i> , the smallest Pisot # ²³
22, 28, 33	$x^3 - x^2 - 1$	1.4656	Pisot #
1, 31, 41	$x^3 - x - 2$	1.5214	Non-Pisot
5, 30, 37	$x^3 - 4$	1.5874	$\sqrt[3]{4}$
3, 20, 29	$x^3 - x^2 - 2$	1.6956	Non-Pisot
4, 18, 27	$x^3 - x^2 - 2$	1.6956	Non-Pisot ²⁴
2, 38, 40	$x^3 - 3x$	1.7321	$\sqrt{3}$, Non-Pisot
17, 19, 26	$x^3 - 2x^2 + x - 1$	1.7548	Pisot #; plastic number <i>rho</i> squared
6, 12, 24	$x^3 - 2x - 2$	1.7693	Non-Pisot #
11, 16, 39	$x^3 - x^2 - 2x + 1$	1.8019	Non-Pisot #, = $2 \cdot \cos(\pi/7)$; three distinct real roots.
9, 10, 21	$x^3 - x^2 - x - 1$	1.8393	Pisot #, the “tribonacci” constant

Table 6: Characteristic polynomials and growth factors for three non-terminals.

7.5.4.5 Factorization and composition in degenerate systems

There is a close relationship between phrase structural composition and factorization of the characteristic polynomial. We see this clearly in the degenerate systems, which can be described in terms of composing patterns of lesser complexity than the whole pattern.

In particular, if a pattern consists of the composition of two simpler patterns, its

²³ The plastic number *rho* is associated with the so-called Padovan sequence (1,1,1,2,2,3,4,5,7,9,12,16,21, 28,37...), in the same sense that the golden number is associated with the Fibonacci sequence. Note further that the polynomial for this orientation family, $x^3 - x - 1$, corresponds to the recurrence relation that generates the Padovan sequence, $a_n = a_{n-2} + a_{n-3}$.

²⁴ Here, distinct syntactic forms have the same growth factor. This can again be understood in terms of linear algebra: the matrices corresponding to members of a single orientation family are similar matrices, in the technical sense (A and B are similar matrices if there is an invertible matrix C such that $A = CBC^{-1}$). The fact that these distinct orientation families have the same growth factor reflects the well-known fact that, while similar matrices have the same characteristic polynomial (and, so, roots thereof), matrices with the same characteristic polynomial may not be similar.

polynomial will be the product of the polynomials associated with its component patterns.²⁵

For example, one degenerate system of interest, from the three-non-terminal-type class, is what I call “double headed X-bar”; as the name suggests, it is the form obtained from an X-bar tree by replacing all of the original terminals with a non-terminal dominating a pair of terminals. The characteristic polynomial of Double-headed X-bar is $x^3 - x^2 - x$; this is the X-bar polynomial ($x^2 - x - 1$) times an additional factor of x . This terminal subcycle (the Pair) corresponds to a linear factor of $(x - 0)$, i.e. a “zero-growth” portion. Another degenerate system from this class is “the X-bar of Spines”. It can be described as an X-bar pattern in which all of the original terminals have been replaced with spines. Its characteristic polynomial is $x^3 - 2x^2 + 1$. That is the X-bar polynomial times an additional linear factor of $(x - 1)$; this growth factor of 1 is diagnostic of spinal structure.

I leave the matter here for now, noting that further exploration of phrase structural properties in the terms laid out here may well yield further relevant insights.

7.6 *Conclusions*

This chapter has explored a number of properties of the X-bar phrasal organization, and shown how they are deeply related to “golden” mathematics. The discussion has gotten

²⁵ From this observation, it follows that the growth factor of a degenerate system will be equal to the largest growth factor among its components. This is so because no new roots are picked up by multiplying polynomials; factors of the product must be factors of one of the components.

quite abstract, for a putative work of linguistics. It is time to step back from the minutiae to get some perspective on what has been established here.

I reviewed the family of mathematical objects described as golden, including the golden mean, the Fibonacci numbers, and the golden string, and pointed out a unifying theme in the form of a common recurrence relation describing each of these objects. This took the form of a polynomial, an addition relation, and a string concatenation formula. I showed that the X-bar schema satisfied the natural syntactic manifestation of the same kind of “golden” recurrence:

$$71) SO_{n+2} = SO_{n+1} + SO_n$$

I suggested that the X-bar form represents a compromise between irreconcilable requirements on syntactic structure, favoring opposite poles of branching form. The local skeleton of the X-bar schema is literally both a Spine and a Bush, the largest object to span the gap between the opposite extremes of binary branching forms. Related to this fact, the X-bar schema is the last (most complex) of the generalized X-bar forms, each globally optimal among their class of alternatives of matched complexity, that is also locally optimal in its class.

Next, I suggested an interpretation of syntactic patterns in terms of divisions of a line segment. Under that understanding, the X-bar form is a “golden” fractal (in fact, a multi-fractal), the simplest kind of scheme inducing fractal structure on the line. As I showed there, its Hausdorff dimension is this value:

$$72) \text{Dim} = \log_2(\text{Phi})$$

Finally, I turned to another way of quantifying growth properties of various phrase structure systems. Writing the recurrence relations of a given phrasal pattern in matrix form, we see that limit of the ratio of the number of syntactic objects at successive generations of maximal expansion is the characteristic root of the polynomial associated with that matrix. In the case of the X-bar schema, this growth factor is Φ . In that section, I also explored further terrain of interest, listing growth factors for the “pure” (non-degenerate) systems one step beyond the X-bar class, and noting a connection between syntactic composition and polynomial factorization for degenerate systems.

Setting all of this in a broader context, the point of detailing the intricate relationship of the X-bar syntactic pattern to “golden” mathematics is to point the way to eventual deeper understanding that transcends the narrow facts of syntax. In this regard, it seems promising that other “golden” patterns in nature have an intriguingly robust character. This is clearest in the domain of plant growth (phyllotaxis), where Fibonacci spiral forms dominate the phenomenology. As shown especially by Douady & Couder (1992), this pattern in plants is virtually inevitable, given a broad range of growth conditions. The explanation for that pattern revolves around a self-organizing process, dynamically optimizing the spacing of elements at a very local level around the meristem.

Similar patterns can be found “closer to home”, so to speak, even in human biology. For example, Goldberger et al (1985) document asymmetry in bronchial branching, across many species of mammals (including humans), “consistent with a process of morphogenetic self-similarity described by Fibonacci scaling.” More intriguingly, golden mathematics has been found even in the functioning of brain tissue.

Roopun et al (2008) report that EEG rhythms in *in vitro* human cortex are spaced according to the golden mean; in their words, by “using phi as a common ratio between adjacent frequencies in the EEG spectrum, the neocortex appears to have found a way to pack as many, minimally interfering frequency bands as possible into the available frequency space.” (Roopun et al 2008)

In other words, the least rational character of the golden mean provides something like an “optimal packing solution” for cortical rhythms. In much the same way, golden angle spacing among successive primordia leads to Fibonacci spiral modes in phyllotaxis, a (dynamically) optimal packing solution in space. It is possible that the golden mathematics of the X-bar schema likewise represents a kind of optimal packing solution for syntax, a robust, perhaps even inevitable minimax solution.

CHAPTER 8: CONCLUSION

8.0 *Overview*

In this dissertation, I have proposed a notion of “economy of command,” a geometrical constraint on syntactic structures. I have argued that endocentricity and detailed patterns of movement fall out as simple reflexes of minimizing the number of c-command relations in binary-branching trees. That looks promising, as these properties are long-standing problems. Endocentricity, lately discussed in terms of projection or labeling, has long been a bugbear of syntactic theory, and finding real explanations for patterns of movement has always been a central concern. I have argued that these properties follow from economy of command. If that is on the right track, these properties of syntax may dissolve as a kind of “third factor” effect (Chomsky 2005), following not from genetic instructions, nor from aspects of the linguistic environment, but rather from a very general computational principle that is in some sense “beyond” the biological and historical particulars.

However, as I have emphasized throughout, this is only a preliminary investigation of these matters. Many central questions remain open; it is time to gather the dangling threads and say something about where to go from here.

8.1 *Minimizing c-command relations*

The cornerstone of this work is the idea that long-distance dependencies are costly in a way that matters to linguistic conditions, and that syntactic structure building should

minimize the number and length of these relations. This depends on a particular view of how to evaluate the cost associated with these relations.

In particular, I pursued the idea that syntactic structure-building proceeds by iterated computational cycles, called phases, and that after each phase is constructed, the resulting partial representation is subject to interpretation at the interfaces with non-syntactic systems. A crucial element of this cyclic interpretation is the “reading” of c-command (or dominance) relations into long-distance interpretive dependencies (including linear order, agreement, and binding).

Decades of syntactic research indicate that c-command and/or dominance relations describe the pathways for these dependencies. Moreover, there is robust evidence for a locality preference in these dependencies, such that the shortest available structural pathway is chosen to host the relevant dependency. Conceiving of the establishment of these dependencies as a kind of computation, it is natural to suppose that this computation should be minimal, with steps that are as simple and few in number as possible. This kind of minimization is achieved by structures with the fewest and shortest computable pathways, I have argued.

I briefly reviewed some reasons to think that it is in interface representations where the cost of c-command relations is incurred. This claim is required for the theory of movement developed here to be coherent. Current thinking has it that within syntax, movement produces a chain of copies of the moving object. If such chains are visible in full to the computation of long-distance dependencies, it would seem that movement only ever makes things worse, so to speak; it strictly increases the number and length of c-

command relations in the tree. However, it seems that with respect to interpretation, only one copy in the chain is visible, typically the highest; other copies collapse or are invisible. In that case, movement can indeed reduce the number of hierarchical relations present, and so (by hypothesis) make for a more easily interpretable structure.

8.2 Movement as tree-balancing

The crucial ingredients in the analysis of movement developed here are that (i) c-command relations “count” in interface forms, where (ii) movement has applied to reduce the overall number of such relations, given that (iii) the interface processes that “read” c-command relations typically see only the highest copy in a chain (the other copies collapsing as traces).

I hypothesized that syntactic movement is governed by the Fundamental Movement Condition (FMC), such that only movements that satisfy the FMC are possible movements in some human language. Taking a to be the number of nodes in the moving object, b to be the number of non-moving nodes, and s to be the number of nodes in the spine from the root of the non-moving object to the trace of the moved object (inclusive), the FMC can be stated as:

- 1) Move α only if $(a-1)(s-2) > b+1$

I showed how a number of constraints on syntactic movement could be derived as consequences of the FMC. The first such consequence is a form of Antilocality: if the FMC holds, objects immediately dominated by the root node cannot be moved. I pointed to a number of independent proposals in the literature that confirm this constraint on

empirical grounds. I also described a size threshold effect, such that a larger candidate for movement within a fixed embedding context will be more strongly driven to move than a smaller candidate object in the same position (and too small an object – a terminal, or a pair of terminals – cannot move at all). I suggested that the size threshold effect could explain the relationship between Object Shift and definiteness (taking definites to be more extended nominal projections than indefinites, hence larger objects), to the correlation between agreement on P and movement (creating postpositions), and to the different movement trajectories for different kinds of nominals (Cardinaletti 2004).

The FMC also predicts a Symmetric Island Condition that I suggested as a source for the Coordinate Structure Constraint of Ross (1967), though noting a conflict with the treatment of small clauses in Moro (2000). On the present analysis, points of symmetry (non-terminals with two daughters of equal size) should be islands, in that the symmetric daughters are always less favored to move than the object containing them.

Finally, I described the kinds of movement that would be expected to arise from this view of movement, considering how the conditions favoring various kinds of movement changed under iteration. I noted that one pattern of extremely-local movement, corresponding to roll-up or snowballing movement deriving mirror orders, was expected to “take off” under positive feedback (though eventually bled by another type of movement I called “skipping”). I explored roll-up movement in some detail, focusing on Malagasy, showing that the rather mysterious ordering facts in that language might fall out from tree-balancing movement driving a roll-up pattern. I noted, following Rackowski & Travis (2000), that the presence of a form of Object Shift interacting with

roll-up movement in this language seems incompatible with an analysis that describes the ordering facts in terms of variable head-complement order without movement. Another curious fact they note, that the deepest pair of adverbs may appear in un-rolled-up order, also is as expected on the present analysis. That is, if roll-up movement is driven by tree-balancing concerns, the first step in that pattern of movement (the one that would invert the order of the two adverbs in question) is the most weakly driven, hence the likeliest to be omitted or optional.

8.3 On Cinque's Generalization

In chapter 5, I outlined how the array of orders of demonstratives, numerals, adjectives, and nouns attested in the world's languages could follow from the view of movement developed here. I showed that we can find possible shapes for the universal base tree underlying nominal phrases, such that all and only the attested orders can be derived through zero or more movements that strictly obey the FMC. I indicated the range of possibilities admitted, and suggested that there was a reasonably good match between the trees allowed under this account, and the independent proposals of DP cartography, e.g. in Cinque (2005) and Svenonius (2008).

A critical issue for expanding the present work is to develop analytical techniques that dispense with the assumption of “coherence” adopted in chapter 5. Recall that that condition stipulated that, whatever the shape of the individual subtrees containing the four landmarks Dem, Num, Adj, and N, those subtrees had hard boundaries, such that movement never reached inside the subtrees, but only rearranged them with respect to

each other. This was an important analytical wedge to make the study tractable, but it is clear that ultimately we would like to dispense with this condition. Instead, we hope to find that the “seams” in the base structure where movement may occur arise naturally from minimization of the number and length of c-command relations. On this view, conditions of optimization cause the shape to fold in characteristic ways.

One troubling issue is that the explanation for Cinque’s Generalization offered here is structurally contingent. Put another way, this theory would allow violations of Cinque’s Generalization, if the base structure were different. This raises a number of questions, especially with respect to the apparent fact that Cinque’s Generalization holds for other domains as well (e.g., in verb clusters). This is expected on the present account only insofar as distinct domains fall into the same class of base tree structures; domains with different structure could in principle support distinct patterns of movement, including remnant movement.

Schlenker (1999) proposes a strong form of parallelism between domains quite explicitly, for the semantic level at least, in his Semantic Uniformity Hypothesis:

“Hypothesis of Semantic Uniformity: Universal Grammar uses the same distinctions (features) and the same interpretive procedures for reference to individuals, times, and possible worlds. Specifically: a. Every interpretable feature that exists in one domain can (in principle at least) exist in every other domain as well. b. The interpretive rules for those features are the same across sortal domains.” (Schlenker 1999: 11)

In a footnote, he expands on this: “There is a more general way of stating this hypothesis: ‘The interpretive system is domain-neutral’. Every interpretable feature and every interpretive rule exists in every sortal domain (individuals, times, worlds, and probably events as well).” (ibid.) Of course, Shlenker is proposing uniformity of

interpretation across distinct domains, a slightly different matter from structural uniformity. However, uniformity of interpretation is certainly compatible with uniform structure. In the next section, I delve into the issues regarding variation in the base structure in more detail.

8.4 The nature of cross-linguistic syntactic variation.

The most important issue to address, in my view, is the nature of cross-linguistic variation. Is there really a single base structure effectively present in all languages? If not, are the differences between the base structures of different languages constrained enough to admit mathematical exploration? For the sake of saying something concrete, I have adopted a very strict view of the nature of cross-linguistic syntactic variation (that there is none, except a different choice of possible movements). That gives a very spare account of variation (perhaps not a bad thing). But there remains something unsatisfying in the account, an element of random chance that some set of movements rather than another, is actually found in some language. Why not insist instead that the very best movement available (itself a notion that needs careful clarification) must be chosen, a form of strong optimization?

There is a fundamental tension between variation and optimization. One reasonable hypothesis is that structures do vary, and each language is finely optimized with respect to the particular base structure it selects. Such a view leads, naturally, to rich predictions at a diachronic level: given a change in structure (perhaps reflected in morphology, e.g. more or less articulated paradigms of agreement), how should

optimized movement adjust? Or perhaps the predictions could go the other way: finding evidence for some movement in the primary linguistic data, how does the inferred relevant structure get adjusted?

If we adopt the further restriction that possible base structures are spinal, we have a very restrictive “window” for cross-linguistic variation: this is a good thing, in that it makes narrow and testable predictions. In particular, we expect on this view that cross-linguistic differences can be explained, to a significant degree, by a simple counting measure, the number of positions in each portion of the tree. These may vary due to differential growth, or collapse of a more articulated universal structure into a smaller set of language-particular distinctions. We may treat both cases in terms of allowing a more or less extended spine in each category space.

Insofar as movement is driven by tree-balancing, and the base structure of any language is a more or less extended spine, then we might maintain stronger claims about optimization and uniformity of movement, with cross-linguistic variation reducing to, first, how large a spine is involved, and second, how universal categories are collapsed into the language-particular categories. We might, for instance, suppose that two languages with different word orders might actually have the same size base tree and the same abstract movements, though with corresponding categories more or less extended and so found on different sides of the rigidly predictable “cuts” made by movement.

Another hypothesis, more in line with the assumptions pursued in Chapter 5, is that the base structure is truly universal, and the typological frequency of various possible orders reflects their relative optimality compared to alternative derivations, and/or a kind

of “sum over histories” approach, with those orders that can be derived in more ways being more frequent. Then the question becomes one of “tuning” the assumed base structure to see how good a fit can be achieved to the data on cross-linguistic frequency of the different orders. Such a tuning would be a narrowing of the range of possible structures for the base, and so a sharpening of cartographic predictions.

8.5 Phrase structure

I argued in Chapters 6 and 7 that properties of phrase structure, as captured by X-bar theory and the notion of endocentricity, can be seen as following from more general principles. In particular, I suggested that generalized X-bar structures follow naturally in a syntactic system sensitive to economy of command, as those phrasal patterns build bushier trees than any competitors. In Chapter 7, I pointed out that the one-specifier X-bar form is particularly natural, in light of its connection to “golden” properties which show up so robustly in other domains.

It might seem that the endorsement of these structural patterns is at odds with the spinal nature of the base structure revealed by cartographic studies. However, there is no real conflict here. On the one hand, the Spine itself is, in fact, one of the optimal generalized X-bar forms. As such, finding such a form in natural language does not contradict the predictions about phrase structure made here; rather, such a finding is merely uninformative. What would contradict these predictions would be the discovery of some non-trivial phrasal pattern that was not a generalized X-bar form (for example, the phrasal format I called D-bar). On the other hand, we need not suppose that the

optimal phrasal patterns are realized in the base structure. Rather, given the claim defended here that what matters is the form of interface representations, we expect that optimal phrase structure can as well be achieved by movement as well as in the base (i.e., by Internal as well as External Merge). In a sense, then, the concerns about phrase structure here can be seen as the static, abstract version of the concerns about movement, which were framed derivationally. Before concluding this work, I turn to one last possible application of the view of movement as a form of structural optimization.

8.6 Optimizing movement by phase: A-movement and A-bar movement

Where can we go from here? One further development of central interest, which considerations of space prevented me from exploring, is a consideration of how movement would be expected to be optimized at the phase level. Taking the simplest case, what is the best single movement within a phase? The FMC provides a rather direct (though vague) answer: the best movement is the one that moves as large an object as far as possible. As noted, there is a fundamental tension between these two conditions. Consider the case of an object A immediately embedded inside another B: B is larger (it includes all the nodes in A, and more besides), but it is also closer to the root, hence A would move farther.

However, if we are optimizing movement at the level of phases, it is clear where the endpoint of the longest movement would be. Within a single phase, the longest possible movement is to the top of the tree, landing in the specifier of the Phase head (the edge). This is a simple conclusion, but the consequences may be profound: this is exactly

the kind of structural pattern that we see in A-bar movement. It suggests that A-bar movement can be understood in purely structural terms, as movement optimized at the phase level, a welcome result.

Movement to the phase edge is optimal if only a single phase is considered. However, phases are often embedded inside other phases, and there a problem arises. Precisely because the locally best movement reaches the edge, it remains “live” in the embedding context, where it may undergo successive-cyclic movement. Supposing that interpretation proceeds phase by phase, the moving object will enter into relations in each new phase it enters (cf Pesetsky & Fox 2005 for a treatment of linearization along these lines). By optimizing locally, we have “passed the buck” to the next higher phase, creating a problem that propagates at a global level.

If we were allowed to be a little more clever about it, we might settle on a slightly suboptimal movement within a phase that does not give rise to problems in further phases that embed it.¹ The best movement would then be one that moved as far as possible without reaching the Edge: such a movement would stop in the specifier of the phrase immediately dominated by the phase head.

This is a particularly exciting prediction, as it matches up with a set of phenomena that are empirically well-motivated, but theoretically mysterious. I am referring to the

¹ Of course, this leads us to expect that non-embedded phases might behave slightly differently. Specifically, they should permit more “excess” in the form of A-bar movement, since it will not be penalized within any higher phases. One wonders if an account of so-called Root phenomena (Emonds 1970) can be constructed in these terms. Indeed, something like V2 in German looks like a promising candidate for this approach. Simplifying crudely, matrix clauses exhibit V2 while embedded clauses usually do not. V2 is standardly analyzed as movement of the verb to the C position, with attendant movement of an XP to Spec, CP (Koster 1975, den Besten 1983). This is the predicted kind of root phenomenon, with extra movement to the Edge (only) in non-embedded structures.

EPP (Chomsky 1981), a requirement that the specifier of TP (or AgrSP, in some formulations) be filled (see also Chomsky 1995b, Lasnik 2001a), and to Raising-to-Object movement (first proposed by Postal 1974; see also Johnson 1991, Koizumi 1993, Lasnik & Saito 1991, Lasnik 2001b), often described as displacing the object to the specifier of AgrOP. What is significant, from the present perspective, is that the landing sites of these movements are just below the phase heads C and v .² In other words, EPP movement and Raising to Object are movements to just below the Edge of a phase.

I believe that such an account of these patterns of movement, as a form of structural optimization with respect to phases, would be an important development. In particular, this account could offer a way to understand the curious “promiscuity” of the EPP: Alexiadou & Anagnostopoulou (1998) argue that the EPP is not a feature requiring movement of a DP (as proposed by Chomsky 1995b); rather, the EPP can be satisfied by moving verbal rather than nominal categories (in so-called predicate-fronting languages). If the real motivation for the EPP is structural rather than featural, this pattern makes a good deal of sense; what matters is that a large piece of syntactic material be moved to this position, regardless of the contents and features thereof. However, concerns of space and time require that further development of this idea be left to later research.

² If Raising to Object movement is real, and English verbs move to little v , then the obligatory adjacency of verbs and their direct objects indicates that this description is correct (i.e., the object moves to the specifier immediately below the phase head little v).

8.7 Final remarks

This brings us to the end of this dissertation. I have attempted to construct explanations for some core syntactic phenomena in terms of economy of command. The explanations, while novel, draw on rather well-established pieces of syntactic theorizing: that c-command and/or dominance is a crucial relation in linguistic expressions, and that there is a preference, all else equal, for syntactic structures to be such as to minimize the burden of computation. This work can be seen as an application of two central ideas of the Minimalist Program: a concern for economy in representation or derivation, and an assimilation of syntactic conditions to constraints imposed by systems external to syntax (interface or bare output conditions).

However, as I have emphasized throughout, this work is but a first step towards building a new theory of syntax based on economy of command. Establishing a firmer foundation for these ideas, and testing their applications more rigorously, is next on the agenda. The project is promising, I think, but it has barely begun.

APPENDIX A: DERIVATION OF THE DP CONDITION

A.0 Overview

In this appendix, I provide the (quite involved) mathematical considerations underpinning the DP Condition provided in Chapter 5. This condition is a large conjunction of inequalities holding over variables tracking the relevant structural parameters of possible underlying base DP trees. The idea is that base structures lead to all and only the attested orders if their structural parameters satisfy the DP condition. The individual inequalities in the DP condition are individual applications of the Fundamental Movement Condition described in Chapter 4 to possible movements in the derivation of attested and unattested orders. For each attested order, we insist that there is at least one derivation such that each step of movement satisfies the FMC; for each unattested order, we insist that no derivation of the order satisfies the FMC.

The data we have in hand are the presence or absence of each of the 24 logically possible orders of demonstratives, numerals, adjective, and noun (henceforth D, M, A, N, respectively) within the DP, in a large sample of the world's languages. By hypothesis, each of those orders is derived by leftward movement affecting a base DMAN hierarchy. The task now is to infer the movements that derive the attested orders (and later, the movements deriving unattested orders), and determine the structural conditions that must be satisfied for each instance of movement to reduce c-command totals.

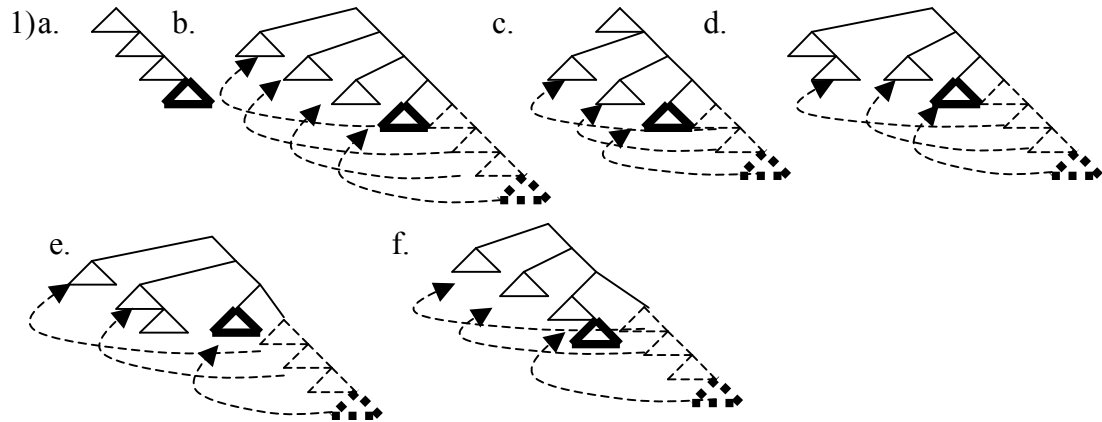
This is not as straightforward as it sounds, as all we have is evidence of the output of the transformations; we must recover the transformations themselves. In many cases, several derivational routes lead to the same surface order. Consider, for example NDMA

(noun demonstrative numeral adjective) order, by hypothesis derived by movement of NP by itself to the top of the tree. This could be derived by a single, “one fell swoop” move of NP. Alternatively, NP could have moved successive-cyclically through one or more intermediate positions, leaving a trace and an extra layer of structure in each, I assume.

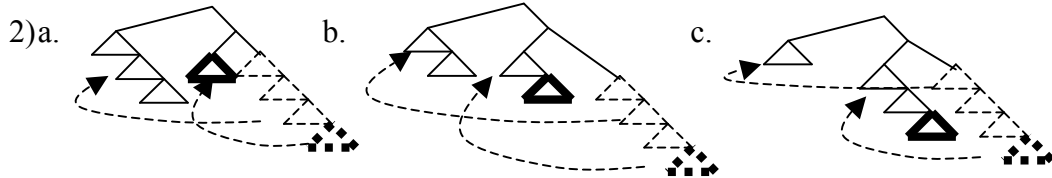
With just three coherent/indivisible pieces of structure (say, M, A, N, as in Greenberg’s Universal 18), the possibilities to consider are quite reasonable. However, once we allow four pieces (D, M, A, N), the possibilities explode. In practice, the complexity is just on the verge of tractability; below I develop some well-motivated heuristics to narrow the space of possibilities.

A.1 Leapfrogging

To get a first look at the combinatoric landscape here, consider just the ‘base’ order DMAN. There are (at least) 6 distinct derivations of this order worth considering, all but one involving a form of ‘leapfrogging’, destroying and then recreating the base linear order. This is, needless to say, a horrifying degree of complexity for what is usually taken to be a single derivation; if even the seemingly underived base can be derived by movement in six distinct ways, what of other orders?



Notice that we may rule out (via Antilocality) these configurations:



At first, it might seem like this could get completely out of control, potentially enabling infinite loops of movement. However, we can formulate a Squeezing Lemma demonstrating that there is a limit to such leap-frogging movements, under the crucial assumption that movement must always improve tree balance.

Suppose some fixed amount of structure has been Merged. Then there is some maximum number of c-command-reducing movements m that could apply to this structure before more material is Merged (and so, if the amount of material to be Merged later is finite, so is the number of movements enabled, still). We can formulate this in terms of ‘squeezing’: each instance of movement adds 2 nodes to the tree, strictly increasing the minimum total number of c-command relations in the tree. But movement is, by hypothesis, constrained to strictly reduce the number of c-command relations in the tree (in practice, always by at least 2). So, given a base tree structure with a total number of c-command relations somewhere in the spectrum between minimal (Bush) and

maximal (Spine), each movement must carry the c-command total an increment of at least two units down toward a monotonically-rising floor. Recall that for a given number of nodes, the possible number of c-command relations in the tree fall within a band between the maximum (associated with a uni-branching Spine) and minimum (associated with the maximally shallow, symmetric Bush) values. Then we can understand the Squeezing Lemma graphically, as below:

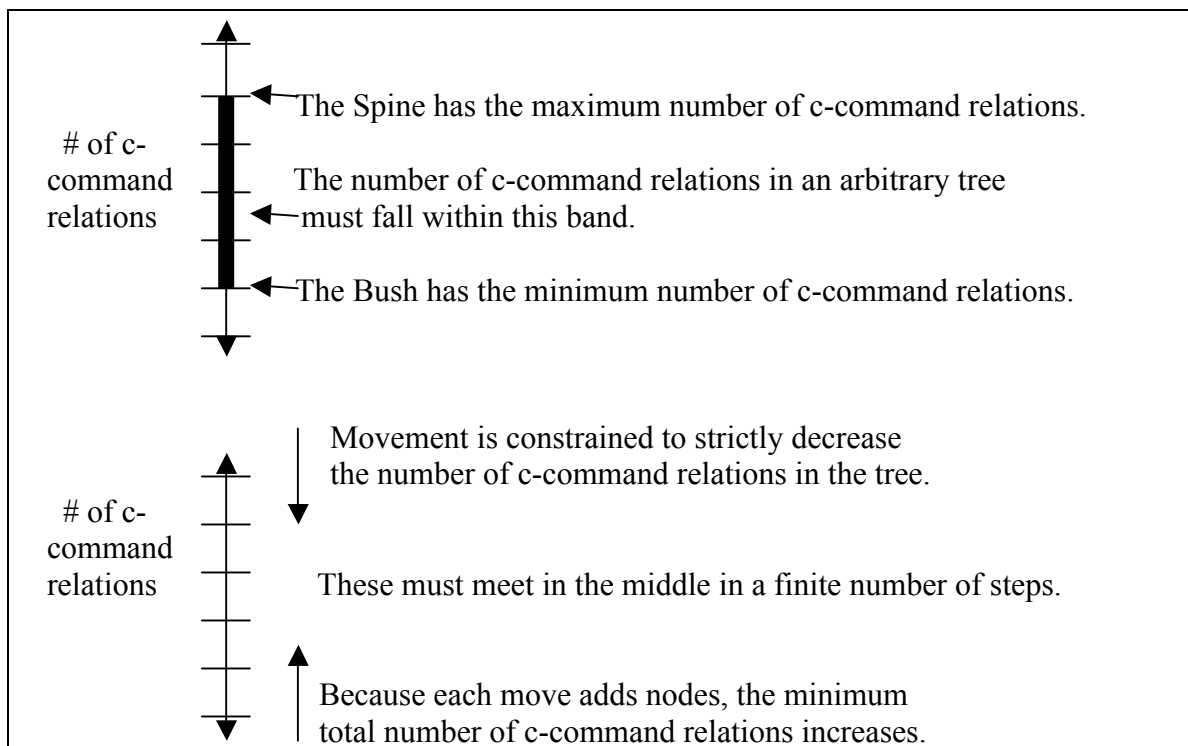
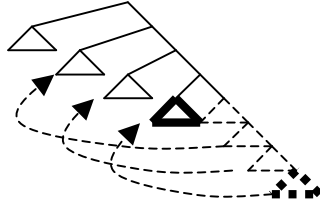


Figure 8: The squeezing lemma illustrated.

Consider in more detail the ‘leapfrogging’ movement at issue. Let us consider a simple case of three elements: once they all have been placed in left branches by movement, can further movements apply? There are two reasons why there cannot be another complete leapfrogging cycle.

3)



First, we can take a global perspective: the final (post-second-leapfrog) configuration has strictly more c-command relations than the initial (one leapfrog) configuration. Thus, clearly, it cannot result from a series of movements that strictly decrease the total number of c-command relations. Second, we can take a local perspective, and show algebraically that the inequalities motivating this set of movements cannot be simultaneously satisfied.

However, the first cycle of leapfrogging, starting with a stack and ending with layers of left branches, can in principle be motivated at each step (for an appropriate choice of structural variables n, a, m, d, s, t, u). For the solutions I have examined (to just the leapfrogging condition corresponding to (1b) above, not incorporating the wider set of inequalities based on the attested/unattested orders), it seems that n must be the smallest, and d the largest, of $\{n, a, m, d\}$. In particular a and m must be large enough to trigger movement of A and M, an option apparently ruled out in natural language.

In fact, if we are willing to say that A and/or M remnant movement is ruled out systematically, on the basis of empirical evidence (i.e. the lack of orders, such as ADN_M, that would arise via “remnant” movement of $[A \ t_N]$ in $[D \ [N \ [M \ [A \ t_N]]]]$), then the

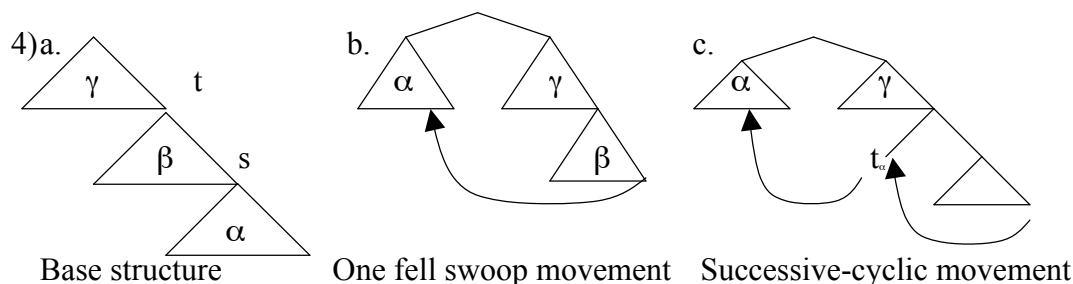
picture is simplified radically. In fact, for order DMAN we need consider *none* of the alternatives outlined in (1b-f), as they all involve movement of an A or M remnant.

I leave for future work a more detailed and rigorous investigation of such combinatoric complexity, here making use of well-motivated heuristics to trim down the possibility space, generally considering only the best/most reasonable derivations of the various orders. I develop and defend this methodology below.

A.2 *Excluding successive-cyclic movement*

With respect to the derivations of unattested orders, I systematically exclude derivations in which the same category moves successive-cyclically to multiple landing sites.¹ The reason for this is straightforward: we are concerned with finding only the weakest (most easily satisfied) structural conditions that can be motivated on the basis of the data on attested and unattested surface orders. It can be demonstrated that, for a particular moving category α , it is always easier to motivate a single one-fell swoop movement than a conjoined series of shorter movements. I illustrate the reasoning below.

Suppose that we find, on the surface, that category α has moved past two dominating categories, β and γ , with spinal depths s and t , respectively.



¹ To be clear, this in no way bears on the undoubted existence of such movement, at the very least at the CP level. The point here is purely a matter of mathematical logic.

We can apply the FMC $((a-1)(s-2) > b+1)$ to the single movement in the one-fell-swoop derivation, and the pair of shorter moves in the successive-cyclic derivation.

$$5) \text{ One fell swoop: (OFS) } (a-1)([s+t-1]-2) > [b+c-1]-1 \quad (a-1)(s+t-3) > b+c-2$$

6) Successive-cyclic:

$$\text{a. first step: (SC}_1\text{) } (a-1)(s-2) > b-1$$

$$\text{b. second step: (SC}_2\text{) } (a-1)([t+1]-2) > [b+c+1]-1 \quad (a-1)(t-1) > b+c$$

Crucially, to motivate the successive-cyclic derivation, both inequalities must be simultaneously satisfied. We proceed by showing that whenever both the successive-cyclic conditions can be met, the one-fell-swoop condition is necessarily satisfied as well, and further that the OFS condition can be met when the conjoined SC conditions cannot.

Compare the condition motivating the second step of movement in the successive-cyclic derivation (SC₂) to the one-fell-swoop (OFS) condition.

$$\begin{aligned} 7) \text{ OFS: } (a-1)(s+t-3) &> b+c-2 \\ &= (a-1)(t-1 (+s-2)) > b+c (-2) \end{aligned}$$

$$8) \text{ SC}_2: (a-1)(t-1) > b+c$$

Notice that in SC₂ we multiply the term $(a-1)$ by a factor $(s-2)$ units less than in OFS, and insist that the product is 2 units more than the product in OFS. But by Antilocality, we know that s is at least 3 (thus $(s-2)$ is at least 1); likewise by the minimal size condition, a is at least 5 (so $(a-1)$ is at least 4). The left hand side of the OFS condition is larger than the left hand side of SC₂, while the right hand side is smaller. Holding the rest of the variables constant and examining permitted values of a , we see that more values of a can satisfy OFS than can satisfy SC₂, and every value of a satisfying SC₂ also satisfies OFS. That entails that OFS is a weaker condition than conjoined SC (SC₁ and SC₂). Suppose

the conjoined SC is satisfied; then so is SC₂. Then the OFS is necessarily satisfied as well (conjoined SC \Rightarrow OFS). Suppose the OFS is satisfied; SC₂ need not be satisfied, and so the conjoined SC need not be (OFS \nRightarrow SC). This is the desired result.

In the present context, we insist that at least one derivation of any attested surface order must be motivated by tree-balancing. Clearly, then, we need not consider any derivations involving successive-cyclic movement. This ensures us that the figure above is not missing any derivations we should be considering; it contains all and only the non-successive-cyclic derivations of the possible surface orders (also meeting the ‘Cinque condition’ barring remnant movement, inductively motivated from this distribution of orders²). Notice that some orders have several possible derivations (notably x, with 5 distinct underlying derivations)³.

The same reasoning does not extend to the derivations we must consider for unattested orders. In that case, rather than trying to rule in the “best” routes to those orders, we must rule out even quite marginal derivations. Furthermore, the derivations in question are those that involve remnant movement. For the unattested orders, because I assume that remnant movement is not involved, successive-cyclic movement typically only adds material to the non-moving part of the tree⁴. However, once remnant

² That is, we must admit that some of the orders here—namely, those analyzed as involving ‘pied-piping’ movement (of N plus A and maybe M)—could in principle also be derived by remnant movement. However, it is telling that the orders that could *only* be derived by remnant movement (e.g., ADNM) are systematically absent. Simplifying, I assume such movement is ruled out across the board.

³ It is interesting to note that the derived order with the most underlying derivations here (namely, the rolled-up mirror image of the base, (x) NAMD) is also the most *common* derived order.

⁴ It might seem that this fails for attested orders that move a small category first, then a larger category containing the moved object and its trace. In that case, the extra movement matters, as it adds material, and so could add just enough material to enable movement that would be blocked without the extra previous step of movement. However, with four coherent subtrees, this comes up, as far as I can see, in only one order: NMAD. That order is derived by first moving N past M and A, then moving the whole past D. If N

movements are considered, possible derivations of unattested orders can in principle move the extra material associated with successive-cyclic movement. On the whole, we will have to consider a wider array of derivational choices with the unattested orders. Before getting to the complexities, I start with the somewhat more straightforward matter of ruling in all attested orders.

A.3 Attested orders

Recall from Chapter 5 the convention of using variables to track relevant parameters of the tree structure. These are d, m, a, n for the number of nodes in D (demonstrative), M (numeral), A (adjective), and N (noun). For D, M, and A we also need a variable for their depth, here u, t, s , respectively.

We can immediately set some bounds on the permitted values. First, the variables must be strictly positive integers:

$$9) a, n, m, d, s, t, u > 0$$

Moreover, a, n, m, d must be odd. There is a further relationship between the total nodes in a category, and the spinal depth of that category.

$$10) a > 2(s-1); m > 2(t-1); d > 2(u-1)$$

Next, I reproduce Figure 2 from Chapter 5, now numbering each step of movement.

moved successive cyclically past A before moving to the left of M, that would add nodes to the NMA object, making it more strongly driven to move. However, note that we find order MAND, which results from movement of the undisrupted MAN object past D. Since that is necessarily ruled in, we need not worry about making the moving object NMA bigger still; if the movement deriving MAND is motivated, so is the movement deriving NMAD, without successive-cyclic movement of N. NMAD could also be derived by moving N from the configuration in MAND, or MNAD. In the second case, NMAD is derived from MNAD, by first moving N to the left of A, then moving MNA past D, then finally extracting N. But this is more weakly motivated than extraction of N directly from MAND, derived by a single leftward movement past D; this derivation is considered explicitly. If these methods were to be extended to larger hierarchies, this issue would require more careful consideration.

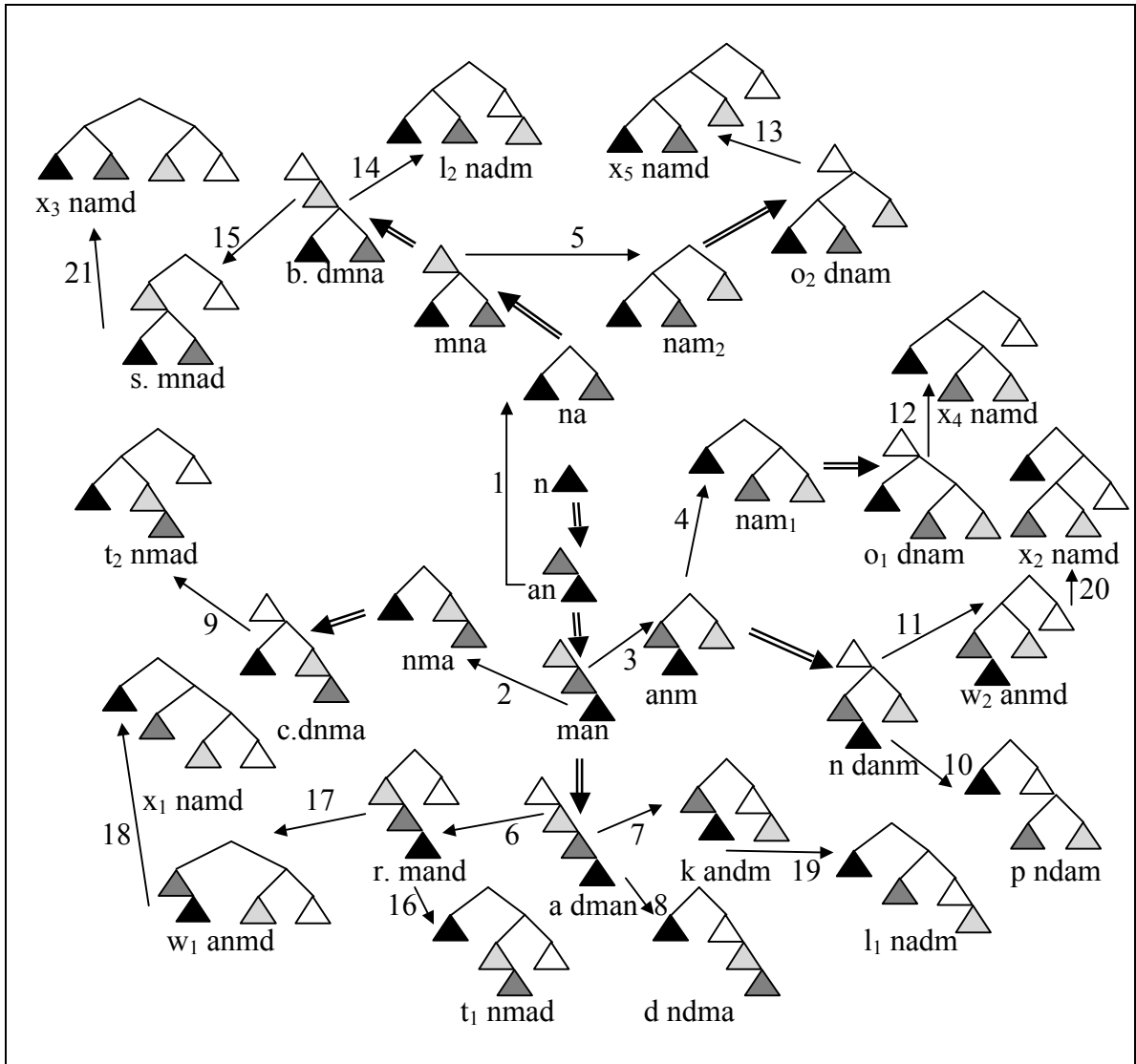


Figure 9: Derivations of attested orders considered in this work.

In the figure above, large double arrows represent External Merge operations, while the thin arrows represent Internal Merge (movement). The movement arrows are also numbered, as we will need to refer to the movements. First, we can reorganize the information in the figure above, listing for each attested order, which movement(s) were involved in its derivation.

Order	Movements
a	base order: no movement
b	1
c	2
d	8
k	7
l	l_1 : 7 , 19 l_2 : 4 , 14
n	3
o	o_1 : 3 , 4 o_2 : 4 , 5
p	3 , 10
r	6
s	4 , 15
t	t_1 : 6 , 16 t_2 : 2 , 9
w	w_1 : 6 , 17 w_2 : 3 , 11
x	x_1 : 6 , 17, 18 x_2 : 3 , 11, 20 x_3 : 4 , 15, 21 x_4 : 3 , 4, 12 x_5 : 4 , 5, 13

Table 7: Movements involved in deriving each attested order

In Table 7, some of the numbered steps are marked with strikethrough; this indicates that they occur higher in the list as the only way to derive some order, and so need not be listed again. Next, for each movement, we can write the relation among the structural variables that must hold for that movement to reduce the number of c-command relations in the tree.

Conditions:

1. $a+1 < (s-2)(n-1)$
2. $m+a < (s+t+3)(n-1)$
3. $m+1 < (t-2)(a+n-2)$
4. $a+m+2 < (s-1)(n-1)$
5. $m+1 < (t-2)(n+a)$
6. $d+1 < (u-2)(a+n+m-3)$
7. $m+d < (t+u-3)(a+n-2)$

8. $a+m+d-1 < (s+t+u-4)(n-1)$
9. $d+1 < u(a-1) + (u-2)(n+m)$
10. $a+m+d+1 < (s+u-2)(n-1)$
11. $d+1 < u(n-1) + (u-2)(a+m)$
12. $d+1 < (u-2)(a+n+m+1)$
13. $d+1 < (u-2)(a+n+m+1)$
14. $m+d < (t+u-3)(n+a)$
15. $d+1 < (u-2)(a+n+m-1)$
16. $a+m+d-1 < (s+t-2)(n-1)$
17. $m+d+2 < (t-1)(a+n-2)$
18. $a+m+d+3 < (s-1)(n-1)$
19. $a+m+d+1 < (s-1)(n-1)$
20. $a+m+d+3 < (s-2)(n-1)$
21. $m+d+2 < (t-1)(a+n)$

To motivate a single movement in these terms, the inequality given above must be met.

To motivate a series of movements, each individual movement must be motivated. For orders that have multiple possible derivations, I only insist that at least one derivation is motivated. So, eliminating redundancies, we have this:

- 11) $1 \ \& \ 2 \ \& \ 8 \ \& \ 7 \ \& \ (19 \mid 14) \ \& \ 3 \ \& \ (4 \mid 5) \ \& \ 10 \ \& \ 6 \ \& \ 15 \ \& \ (16 \mid 9) \ \& \ (17 \mid 11) \ \& \ ((17 \ \& \ 18) \mid (11 \ \& \ 20) \mid (15 \ \& \ 21) \mid (4 \ \& \ 12) \mid (5 \ \& \ 13))$

Filling this in with the indicated inequalities, we obtain the following expression.

- 12) $(a+1 < (s-2)(n-1)) \ \& \ (m+a < (s+t+3)(n-1)) \ \& \ (a+m+d-1 < (s+t+u-4)(n-1)) \ \& \ (m+d < (t+u-3)(a+n-2)) \ \& \ ((a+m+d+1 < (s-1)(n-1)) \mid (m+d < (t+u-3)(n+a))) \ \& \ (m+1 < (t-2)(a+n-2)) \ \& \ ((a+m+2 < (s-1)(n-1)) \mid (m+1 < (t-2)(n+a))) \ \& \ (a+m+d+1 < (s+u-2)(n-1)) \ \& \ (d+1 < (u-2)(a+n+m-3)) \ \& \ (d+1 < (u-2)(a+n+m-1)) \ \& \ ((a+m+d-1 < (s+t-2)(n-1)) \mid (d+1 < u(a-1) + (u-2)(n+m))) \ \& \ ((m+d+2 < (t-1)(a+n-2)) \mid (d+1 < u(n-1) + (u-2)(a+m))) \ \& \ (((m+d+2 < (t-1)(a+n-2)) \ \& \ (a+m+d+3 < (s-1)(n-1))) \mid ((d+1 < u(n-1) + (u-2)(a+m)) \ \& \ (a+m+d+3 < (s-2)(n-1))) \mid ((d+1 < (u-2)(a+n+m-1)) \ \& \ (m+d+2 < (t-1)(a+n))) \mid ((a+m+2 < (s-1)(n-1)) \ \& \ (d+1 < (u-2)(a+n+m+1))) \mid ((m+1 < (t-2)(n+a)) \ \& \ (d+1 < (u-2)(a+n+m+1))))$

This forms the portion of the DP Condition that rules in attested orders. In the next section, I turn to the task of ruling out unattested orders, a more complicated matter.

A.4 Unattested orders

The next step is to see if we can rule out unattested orders in the same terms. Let us examine the unattested orders, and suppose that the relevant movements involved in their derivations do NOT satisfy the FMC. That is, for an arbitrary excluded movement, we negate the FMC, supposing that $(a-1)(s-2) \leq b+1$. Adding these negative conditions to the positive conditions already derived from the attested orders, we will narrow in on precisely the universal base DP trees that would be consistent with a tree-balancing motivation both for the patterns of movement that we do find, and for the orders that are ruled out. Finding that the solution set is non-empty is intriguing enough, but the real test, to repeat, is to see if the structure independently revealed by cartographic studies matches up with the predictions here.

This is a tricky topic, since there are many routes to each surface order. This analytical problem is particularly acute for the unattested orders, as we must ensure that we are not missing “weird” derivations of these orders which do monotonically decrease c-command totals (while more obvious derivations do not). Potentially we would have to examine infinitely many derivations of each order, obviously impossible. In practice, this is ruled out by the present account; there cannot be infinite loops of tree-balancing movements involving finite material.

In what follows, I consider only the most “reasonable” derivations of the unattested orders, noting that further investigation is warranted. However, we may be confident that considering further derivations of the unattested orders will not leave us with an empty solution set. There is a basic conceptual reason for this, again revolving around the fact

that all the attested orders involve movement of NP or something containing it. Then any unattested order is derived by at least one step of movement not affecting NP or something containing it. In this light, recall the FMC: $(a-1)(s-2) > b+1$. Given the empirical observation at issue (no non-NP moves), note that n , the size of N, will be a term of a , part of the quantity on the greater, left hand side. On the other hand, for the ruled-out movements, N will not move, hence n will be a term of b , on the lesser, right hand side of the inequality. Thus if N is big enough, the non-NP-affecting movements that would derive unattested orders will be excluded.

As a first, manageable step towards mapping out the possibilities we need to consider, let's consider the invisible “kernel” of derivations before the Merge of the D portion of the tree – with just three elements in play (the subtrees (N, A, M)), the possibilities are much more constrained.

To make the tree diagrams more perspicuous, I adopt the convention of shading the triangles representing the subtrees D, M, A, N as white, light grey, dark grey, and black, respectively. Thus, each category is tagged with its native depth, deeper (in the base) objects appearing darker. For example, N(oun), the bottom of the base structure, is black. I use a double arrow to represent (External) Merge, a single arrow to indicate movements.

I indicate with letters A and B two things worth noting in this partial derivational space. These are discussed in turn below.

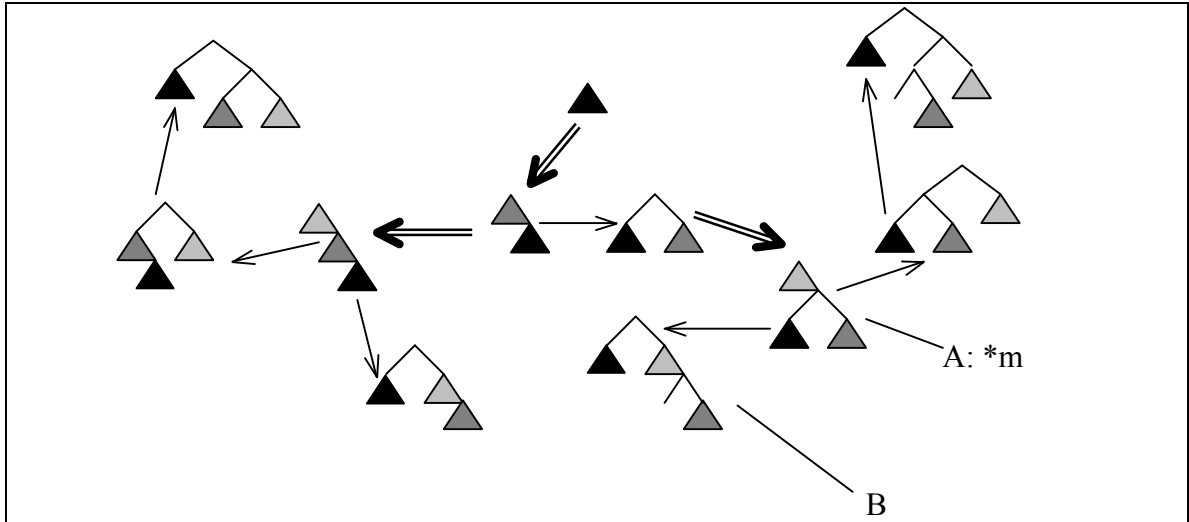
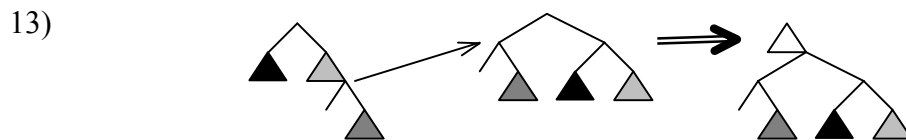


Figure 10: Derivational possibilities with N, A, M.

A: We can rule this movement out directly, as it would result in order AMN, unattested. Recall the assumption that movement is never forced; if this were a possibility, some language would be expected to permit it to survive to the surface under embedding with no further movement, yielding the unattested order m *DAMN.

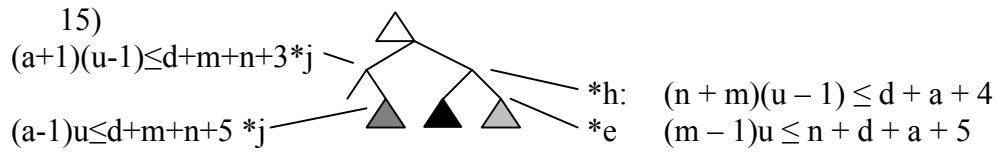
B. Let's look at one "exotic" branch, with movement of A following movement of N after embedding under M (this is an example of successive cyclic movement, which we could ignore when considering only attested orders, as discussed above). This results in a permitted relative order ((D)ANM):



The relevant condition for this movement is that

$$14) (a + 1)(t - 1) > n + m + 2$$

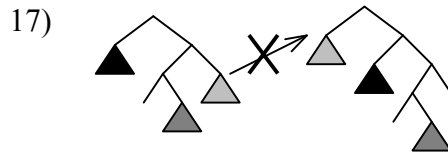
The resulting structure is a rich source of unattested orders.



Let's look more closely at ruling out movement of A and M. Consider the structure below.



The light grey (M) category cannot possibly move here.



To see why not, notice that on the previous step, N moved by itself from a minimally deep position in the tree. Along this derivational branch, we can then apply the FMC, concluding that the following holds.

$$18) n > a + m + 5$$

The move being ruled out would carry M from the same minimal depth; for this to occur, we would have to have this condition met:

$$19) *m > n + a + 7$$

Rearranging terms in the first expression, we have this.

$$20) m < n - a - 5$$

But the following holds because the variables are strictly positive.

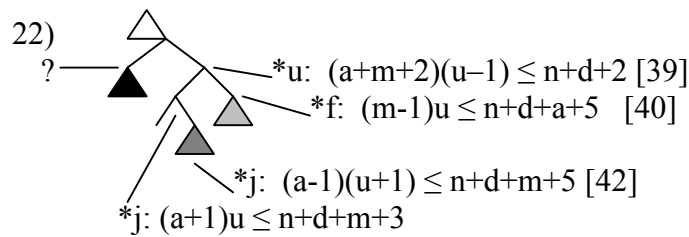
$$21) n - a - 5 < n + a + 7$$

We know $m < X$, $X < Y$, and want $m > Y$: clearly, this is impossible.

In fact, this result is quite general: if one category can undergo movement this local, no disjoint object can do the same. I call this the “LeapFrog” condition, and exploit it heavily. We must be careful to keep in mind that we can only apply this within a particular derivational trajectory, in terms of earlier movements along that trajectory. Surface orders might be derived in surprisingly convoluted ways, and we must “relativize” our conclusions to an individual branch of derivational history.

Now let’s look at the same structure considered in (16) above, after D is Merged.

Here again, the resulting structure is a rich source of unattested orders.



In general, it is these “worst-case”, prolific-movement derivations that will produce structures yielding the strongest constraints on structure (though at the cost of winnowing via the conditions for all the moves to get there).

For reasons of space, I do not list the individual movements leading to unattested orders, as I did above for attested orders. Instead, I present two figures indicating all of the structural sources of unattested orders considered here. Each black dot connects to a node in the tree whose movement is ruled out because it would lead to an unattested order. These are the source of the “negative conditions” in the DP condition (the bulk of the expression). In Figure 11, I collect derivations that do not involve movement of N past A as soon as it is Merged. Those derivations that do utilize that movement are diagrammed in Figure 12.

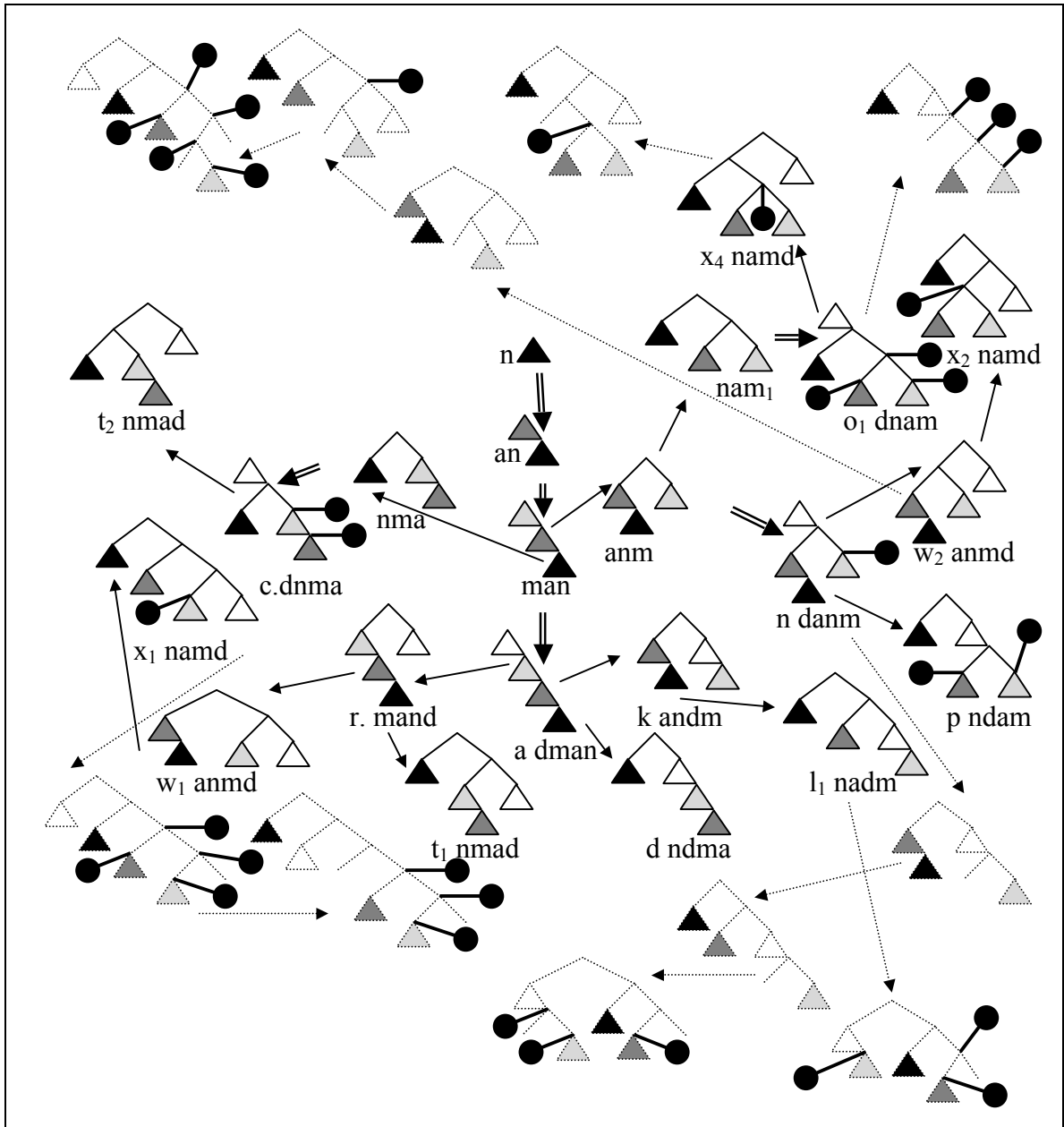


Figure 11: Excluded routes to unattested orders 1.

These derivations do not involve immediate inversion of N past A. Movements that would lead to unattested orders are indicated with black dots. Derivations outside the considered routes to attested orders (see Figure 9) are indicated with dashed lines.

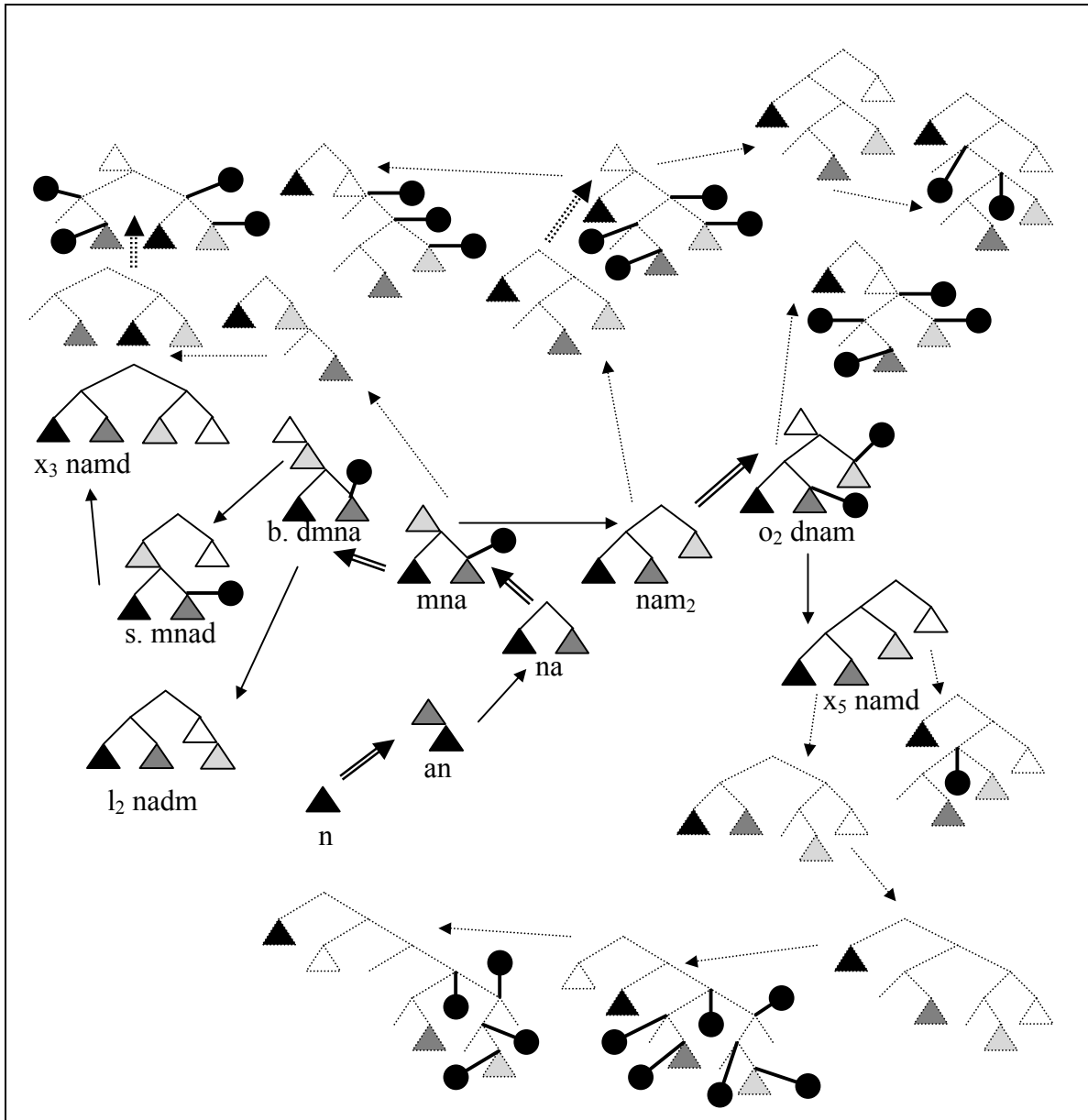


Figure 12: Excluded routes to unattested orders 2. These are derivations that involve inversion of N past A before anything else is Merged. Movements that would lead to unattested orders are shown with black dots. Derivations outside the considered routes to attested orders are indicated with dashed lines.

A.5 Putting it all together

We are now in a position to assemble these conditions into a larger structure. The reasoning is as follows. For each attested order, we want to ensure that at least one

derivation of that order “succeeds” (i.e., proceeds via a series of movements that each reduce the total number of c-command relations present). For each unattested order, we want the opposite condition: no derivation of that order can succeed (every possible derivation must involve at least one step of movement that increases the number of c-command relations present).

The DP Condition:

$$\begin{aligned}
 23) & ((a+1 < (s-2)(n-1)) \& (m+a < (s+t+3)(n-1)) \& (a+m+d-1 < (s+t+u-4)(n-1)) \& (m+d < \\
 & (t+u-3)(a+n-2)) \& ((a+m+d+1 < (s-1)(n-1)) \mid (m+d < (t+u-3)(n+a))) \& (m+1 < (t-2) \\
 & (a+n-2)) \& ((a+m+2 < (s-1)(n-1)) \mid (m+1 < (t-2)(n+a))) \& (a+m+d+1 < (s+u-2)(n-1)) \\
 & \& (d+1 < (u-2)(a+n+m-3)) \& (d+1 < (u-2)(a+n+m-1)) \& ((a+m+d-1 < (s+t-2)(n-1)) \mid \\
 & (d+1 < (u(a-1)+(u-2)(n+m))) \& ((m+d+2 < (t-1)(a+n-2)) \mid (d+1 < (u(n-1)+(u-2)(a+m))) \\
 & \& (((m+d+2 < (t-1)(a+n-2)) \& (a+m+d+3 < (s-1)(n-1))) \mid ((d+1 < (u(n-1)+(u-2)(a+m)) \\
 & \& (a+m+d+3 < (s-2)(n-1))) \mid ((d+1 < (u-2)(a+n+m-1)) \& (m+d+2 < (t-1)(a+n))) \mid ((a+m+2 < (s-1)(n-1) \\
 & \& (d+1 < (u-2)(a+n+m+1))) \mid ((m+1 < (t-2)(n+a)) \& (d+1 < (u-2) \\
 & (a+n+m+1)))) \& (2(a+n)+d+2 > (u-1)(m-1)) \& (a+n+d+4 > (u-1)(m-1)) \& (a+n+d+4 > (u(m-1)) \\
 & \& (n+m+d+2 > (t+u-2)(a-1)) \& (n+m+d+4 > u(a-1)) \& (n+m+3 > (t-1) \\
 & (a-1)) \& (n+d+3 > (u-1)(a+m-2)) \& (n+d+3 > (u-1)(a+m)) \& \neg((d+m > a+n+4) \& \\
 & ((m+n+a-3)(u-2) > d+1) \& ((a+n-2)(t-1) > d+m+2) \& ((n-1)(s-1) > d+m+a+3)) \& \\
 & \neg((((m-1)^3 > d+a+n+7) \mid ((a-1)^2 > d+m+n+7) \mid ((m+1)^2 > d+a+n+5) \mid (a+m+2 > n+d+4) \mid \\
 & ((n-1) > d+m+a+7) \& (((a+m+2)^2 > d+n+6) \mid ((m+1)^3 > d+a+n+7) \mid ((m-1)^4 > d+a+n+9)))) \& \\
 & ((d+1)^2 > n+a+m+5) \& ((m+n+a-3)(u-2) > d+1) \& ((a+n-2)(t-1) > d+m+2) \& ((n-1)(s-1) > d+m+a+3)) \& \\
 & \neg(((n-1)(s-1) > a+m+2) \& ((n+a-2)(u-2) > m+1) \& (((a+m)(u-1) > d+n+2) \mid ((m-1)u > d+a+n+3) \mid ((a-1)u > d+m+n+3) \mid \\
 & ((n-1)(u-1) > d+m+a+3) \& (((a+m+2)(u-1) > d+n+2) \mid ((a+m)u > d+n+4) \mid ((m-1)(u+1) > d+a+n+5)))) \& \\
 & \neg((a+m > d+n+4) \& ((n-1)s > d+m+a+3) \& ((n+a+m-1)(u-2) > d+1) \& ((n+a-2)(u-2) > m+1)) \& \\
 & \neg(((n+a-2)(u-2) > m+1) \& (((m-1)(u-1) > d+a+n+1) \mid (((n-1)(s+u-2) > d+a+m+1) \& (((a+m)(u-1) > d+n+2) \mid ((m-1)u > d+a+n+3)))) \& \\
 & \neg(((d+m > n+a+4) \& ((n-1)(s-1) > d+m+a+3) \& ((a+n-2)(u-1) > d+m+2) \& ((n+a-2)(u-2) > m+1)) \& \\
 & ((a+1 > d+m+n+5) \mid ((a-1)^2 > d+m+n+7) \mid ((m+1)(u-1) > d+a+n+5) \mid ((m-1)u > d+a+n+7))) \& \\
 & \neg(((n-1)(s+t-3) > a+m) \& (((a+m-2)(u-1) > d+n+2) \mid ((a-1)(t+u-2) > d+m+n+1))) \& \neg(((n-1)(t-1) > a+m+2) \& ((n-1)(s-2) > a+1)) \& \\
 & (((a+m)(u-1) > d+n+2) \mid ((a+1)(t+u-2) > d+m+n+1) \mid ((a-1)(t-1) > m+n+2))) \& \neg(((a-1)(t-1) > m+n+2) \& \\
 & ((n-1)(s-2) > a+1)) \& \neg(((n-1)(s-2) > a+1) \& ((a-1)(t+u-2) > d+m+n+1) \mid ((a-1)t > d+m+n+3))) \& \\
 & \neg(((n-1)(s-2) > a+1) \& ((n+a)(t-2) > m+1) \& (((m-1)(u-1) > d+a+n+3) \mid ((a-1)u > d+m+n+3) \mid ((n-1)u > d+m+a+3) \& (((a+m+2)(u-1) > d+n+2) \mid ((m-1)u > d+a+n+5) \mid ((a+1)u > d+m+n+3) \mid ((a-1)(u+1) > d+m+n+5)))) \& \neg(((m-1)(u-1) > d+a+n+5) \& (d+m-2 > n+a-4) \& ((n-1)(s-1) > d+m+a+1) \& ((a+n)(t+u-3) >
 \end{aligned}$$

$$\begin{aligned}
& d+m)) \ \& \ \neg((n-1>a+m-4) \ \& \ ((n+a)(t-2)>m+1) \ \& \ ((n-1)(s-2)>a+1) \ \& \ (((a+m-2)(u-1)>d+n+2) \ | \ ((m-1)u>d+n+a+5) \ | \ ((a+1)u>d+m+n+3) \ | \ ((a-1)(u+1)>d+m+n+5) \ | \\
& ((n-1>d+m+a+7) \ \& \ ((n+a+m+3)(u-2)>d+1) \ \& \ ((a+m+2)^2>d+n+6)) \ | \ (((n-1)(u-1)>d+m+a+5) \ \& \ (((a+m+4)(u-1)>d+n+2) \ | \ ((a+m)u>d+n+4) \ | \ ((m-1)(u+1)>d+m+n+7) \\
&)))) \ \& \ \neg((a+m+2>d+n+4) \ \& \ ((n-1)^2>d+m+a+5) \ \& \ ((n+a+m+1)(u-2)>d+1) \ \& \ ((n+a)(t-2)>m+1) \ \& \ ((n-1)(s-2)>a+1) \) \ \& \ \neg(((a+n-2)(t-2)>m+1) \ \& \ ((n-1)(s-1)>m+3) \ \& \ ((n-1)(u-1)>d+m+a+3) \ \& \ (((a+m+2)(u-1)>n+d+2) \ | \ ((a+m)u>n+d+4))) \ \& \\
& \neg(((a+1)(t-1) > n+m+2) \ \& \ ((n-1)(t-1)>m+a+2) \ \& \ ((n-1)(s-2)>a+1) \ \& \ (((a+1)(u-1)>d+m+n+3) \ | \ ((a-1)u>d+m+n+5) \ | \ ((n+m)(u-1)>d+a+4) \ | \ ((m-1)u>d+a+n+5))))
\end{aligned}$$

In the next portion of this appendix, I give the numerical results of the program built to explore solutions to the DP condition. The program itself is given explicitly after that, in the final section of this appendix.

A.6 Table of results and selected solutions

The table below includes the results of running the program given in Appendix A-3, with the listed choices for the structural parameters (n, a, m, d, s, t, u). Following the table, I include a brief list of selected sample solutions (these sample solutions were obtained by using the option in the program to print a list of solutions, within the user-specified parameters). The column headings are as follows:

NCap (max n): This is the maximum number of nodes allowed in NP for that iteration of the program; the program considers NPs of that size, and all smaller sizes.

AdjCap (max a): The maximum number of nodes that region A of the tree may contain.

AdjDep (max s): The maximum allowed spinal depth of the A region (s).

NumCap, DemCap: The maximum node counts to be considered in regions M and D, respectively.

NumDep, DemDep: The user-entered upper limits on the spinal depth of M and D.

Total: This represents the number of distinct choices of variable values that the program has considered.

Solutions: This is the number of distinct choices of variable values for which the DP condition, describing the shapes of base DP tree for which the all attested orders and no unattested orders are derived by strictly tree-balancing movements.

Rows in grey represent choices of tree parameters with no solutions to the DP Condition.

NCap (max n)	AdjCap (max a)	AdjDep (max s)	NumCap (max m)	NumDep (max t)	DemCap (max d)	DemDep (max u)	Total	Solutions
21	11	5	11	5	13	5	38808	42
21	11	5	13	5	11	5	38808	17
21	13	5	11	5	11	5	38808	17
21	11	5	11	5	11	5	30184	17
19	11	5	11	5	11	5	27440	9
17	11	5	11	5	11	5	24696	6
15	11	5	11	5	11	5	21952	4
13	11	5	11	5	11	5	19208	3
11	11	5	11	5	11	5	16464	2
9	11	5	11	5	11	5	13720	1
9	11	5	11	5	15	5	21560	6
9	13	5	13	5	15	5	35640	6
9	15	5	15	5	15	5	53240	6
7	11	5	11	5	11	5	10976	0
7	21	9	21	9	21	9	562432	0
9	9	5	11	5	11	5	9800	1
9	11	5	9	5	11	5	9800	1
9	9	5	9	5	11	5	7000	1
11	5	3	5	3	7	4	324	0
11	5	3	5	3	9	5	540	0
9	5	3	5	3	5	3	135	0
13	5	3	5	3	5	3	189	0
15	5	3	5	3	5	3	216	0
13	5	3	5	3	7	4	378	0
13	5	3	5	3	9	5	630	0
15	7	4	5	3	5	3	432	0
15	5	3	7	4	5	3	432	0
15	5	3	5	3	7	4	432	0
15	5	3	5	3	9	4	648	0
15	5	3	5	3	11	5	1008	4
17	5	3	5	3	5	3	243	0
17	5	3	5	3	7	4	486	0
17	5	3	5	3	9	5	810	1
17	5	3	5	3	11	6	1215	6
17	5	3	5	3	11	3	729	6
17	7	4	7	4	11	3	2916	6
35	5	3	5	3	5	3	486	8
35	5	3	5	3	7	4	972	19
35	5	3	7	4	5	3	972	18
35	7	4	5	3	5	3	972	18
35	5	3	7	4	7	4	1944	41
35	7	4	5	3	7	4	1944	41
35	7	4	7	4	5	3	1944	36

Table 8: Results of program.

I list below some sample solutions.

These were obtained by running the program given below, with these maximum values of $\langle n, a/s, m/t, d/u \rangle$: $\langle 31, 11/6, 11/6, 11/6 \rangle$.

Spines are in **bold underline**; solutions with $u > 3$ are in *bold italics*.

$n = 9, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 11, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 13, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 15, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 17, a = 5, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 17, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 19, a = 5, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 19, a = 5, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 19, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 21, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 21, a = 5, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 21, a = 5, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 21, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 21, a = 5, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 21, a = 5, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 21, a = 7, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 21, a = 7, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 23, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 23, a = 5, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 23, a = 5, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 23, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 23, a = 5, s = 3, m = 7, t = 3, d = 9, u = 3$
 $n = 23, a = 5, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 23, a = 5, s = 3, m = 7, t = 4, d = 9, u = 3$
 $n = 23, a = 5, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 23, a = 7, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 23, a = 7, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 23, a = 7, s = 4, m = 5, t = 3, d = 9, u = 3$
 $n = 23, a = 7, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 25, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 25, a = 5, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 25, a = 5, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 25, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 25, a = 5, s = 3, m = 7, t = 3, d = 7, u = 3$
 $n = 25, a = 5, s = 3, m = 7, t = 3, d = 9, u = 3$
 $n = 25, a = 5, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 25, a = 5, s = 3, m = 7, t = 4, d = 7, u = 3$
 $n = 25, a = 5, s = 3, m = 7, t = 4, d = 9, u = 3$
 $n = 25, a = 5, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 25, a = 7, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 25, a = 7, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 25, a = 7, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 25, a = 7, s = 4, m = 5, t = 3, d = 7, u = 3$
 $n = 25, a = 7, s = 4, m = 5, t = 3, d = 9, u = 3$

$n = 25, a = 7, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 27, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 27, a = 5, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 27, a = 5, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 27, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 27, a = 5, s = 3, m = 7, t = 3, d = 5, u = 3$
 $n = 27, a = 5, s = 3, m = 7, t = 3, d = 7, u = 3$
 $n = 27, a = 5, s = 3, m = 7, t = 3, d = 9, u = 3$
 $n = 27, a = 5, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 27, a = 5, s = 3, m = 9, t = 3, d = 11, u = 3$
 $n = 27, a = 5, s = 3, m = 7, t = 4, d = 5, u = 3$
 $n = 27, a = 5, s = 3, m = 7, t = 4, d = 7, u = 3$
 $n = 27, a = 5, s = 3, m = 7, t = 4, d = 9, u = 3$
 $n = 27, a = 5, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 27, a = 5, s = 3, m = 9, t = 4, d = 11, u = 3$
 $n = 27, a = 5, s = 3, m = 9, t = 5, d = 11, u = 3$
 $n = 27, a = 7, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 27, a = 7, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 27, a = 7, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 27, a = 7, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 27, a = 7, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 27, a = 7, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 27, a = 9, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 27, a = 7, s = 4, m = 5, t = 3, d = 5, u = 3$
 $n = 27, a = 7, s = 4, m = 5, t = 3, d = 7, u = 3$
 $n = 27, a = 7, s = 4, m = 5, t = 3, d = 9, u = 3$
 $n = 27, a = 7, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 27, a = 7, s = 4, m = 7, t = 3, d = 11, u = 3$
 $n = 27, a = 7, s = 4, m = 7, t = 4, d = 11, u = 3$
 $n = 27, a = 9, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 27, a = 9, s = 5, m = 5, t = 3, d = 11, u = 3$
 $n = 29, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 29, a = 5, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 29, a = 5, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 29, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 29, a = 5, s = 3, m = 5, t = 3, d = 11, u = 4$
 $n = 29, a = 5, s = 3, m = 7, t = 3, d = 5, u = 3$
 $n = 29, a = 5, s = 3, m = 7, t = 3, d = 7, u = 3$
 $n = 29, a = 5, s = 3, m = 7, t = 3, d = 9, u = 3$
 $n = 29, a = 5, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 29, a = 5, s = 3, m = 9, t = 3, d = 9, u = 3$
 $n = 29, a = 5, s = 3, m = 9, t = 3, d = 11, u = 3$
 $n = 29, a = 5, s = 3, m = 7, t = 4, d = 5, u = 3$
 $n = 29, a = 5, s = 3, m = 7, t = 4, d = 7, u = 3$

$n = 29, a = 5, s = 3, m = 7, t = 4, d = 9, u = 3$
 $n = 29, a = 5, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 29, a = 5, s = 3, m = 9, t = 4, d = 9, u = 3$
 $n = 29, a = 5, s = 3, m = 9, t = 4, d = 11, u = 3$
 $n = 29, a = 5, s = 3, m = 9, t = 5, d = 9, u = 3$
 $n = 29, a = 5, s = 3, m = 9, t = 5, d = 11, u = 3$
 $n = 29, a = 7, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 29, a = 7, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 29, a = 7, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 29, a = 7, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 29, a = 7, s = 3, m = 7, t = 3, d = 9, u = 3$
 $n = 29, a = 7, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 29, a = 7, s = 3, m = 7, t = 4, d = 9, u = 3$
 $n = 29, a = 7, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 29, a = 9, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 29, a = 9, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 29, a = 7, s = 4, m = 5, t = 3, d = 5, u = 3$
 $n = 29, a = 7, s = 4, m = 5, t = 3, d = 7, u = 3$
 $n = 29, a = 7, s = 4, m = 5, t = 3, d = 9, u = 3$
 $n = 29, a = 7, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 29, a = 7, s = 4, m = 7, t = 3, d = 9, u = 3$
 $n = 29, a = 7, s = 4, m = 7, t = 3, d = 11, u = 3$
 $n = 29, a = 7, s = 4, m = 7, t = 4, d = 9, u = 3$
 $n = 29, a = 7, s = 4, m = 7, t = 4, d = 11, u = 3$
 $n = 29, a = 9, s = 4, m = 5, t = 3, d = 9, u = 3$
 $n = 29, a = 9, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 29, a = 9, s = 5, m = 5, t = 3, d = 9, u = 3$
 $n = 29, a = 9, s = 5, m = 5, t = 3, d = 11, u = 3$
 $n = 31, a = 5, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 31, a = 5, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 31, a = 5, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 31, a = 5, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 31, a = 5, s = 3, m = 5, t = 3, d = 9, u = 4$
 $n = 31, a = 5, s = 3, m = 5, t = 3, d = 11, u = 4$
 $n = 31, a = 5, s = 3, m = 7, t = 3, d = 5, u = 3$
 $n = 31, a = 5, s = 3, m = 7, t = 3, d = 7, u = 3$
 $n = 31, a = 5, s = 3, m = 7, t = 3, d = 9, u = 3$
 $n = 31, a = 5, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 31, a = 5, s = 3, m = 9, t = 3, d = 7, u = 3$
 $n = 31, a = 5, s = 3, m = 9, t = 3, d = 9, u = 3$

$n = 31, a = 5, s = 3, m = 9, t = 3, d = 11, u = 3$
 $n = 31, a = 5, s = 3, m = 7, t = 4, d = 5, u = 3$
 $n = 31, a = 5, s = 3, m = 7, t = 4, d = 7, u = 3$
 $n = 31, a = 5, s = 3, m = 7, t = 4, d = 9, u = 3$
 $n = 31, a = 5, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 31, a = 5, s = 3, m = 9, t = 4, d = 7, u = 3$
 $n = 31, a = 5, s = 3, m = 9, t = 4, d = 9, u = 3$
 $n = 31, a = 5, s = 3, m = 9, t = 4, d = 11, u = 3$
 $n = 31, a = 5, s = 3, m = 9, t = 5, d = 7, u = 3$
 $n = 31, a = 5, s = 3, m = 9, t = 5, d = 9, u = 3$
 $n = 31, a = 5, s = 3, m = 9, t = 5, d = 11, u = 3$
 $n = 31, a = 7, s = 3, m = 5, t = 3, d = 5, u = 3$
 $n = 31, a = 7, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 31, a = 7, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 31, a = 7, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 31, a = 7, s = 3, m = 7, t = 3, d = 7, u = 3$
 $n = 31, a = 7, s = 3, m = 7, t = 3, d = 9, u = 3$
 $n = 31, a = 7, s = 3, m = 7, t = 3, d = 11, u = 3$
 $n = 31, a = 7, s = 3, m = 7, t = 4, d = 7, u = 3$
 $n = 31, a = 7, s = 3, m = 7, t = 4, d = 9, u = 3$
 $n = 31, a = 7, s = 3, m = 7, t = 4, d = 11, u = 3$
 $n = 31, a = 9, s = 3, m = 5, t = 3, d = 7, u = 3$
 $n = 31, a = 9, s = 3, m = 5, t = 3, d = 9, u = 3$
 $n = 31, a = 9, s = 3, m = 5, t = 3, d = 11, u = 3$
 $n = 31, a = 7, s = 4, m = 5, t = 3, d = 5, u = 3$
 $n = 31, a = 7, s = 4, m = 5, t = 3, d = 7, u = 3$
 $n = 31, a = 7, s = 4, m = 5, t = 3, d = 9, u = 3$
 $n = 31, a = 7, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 31, a = 7, s = 4, m = 7, t = 3, d = 7, u = 3$
 $n = 31, a = 7, s = 4, m = 7, t = 3, d = 9, u = 3$
 $n = 31, a = 7, s = 4, m = 7, t = 3, d = 11, u = 3$
 $n = 31, a = 7, s = 4, m = 7, t = 4, d = 7, u = 3$
 $n = 31, a = 7, s = 4, m = 7, t = 4, d = 9, u = 3$
 $n = 31, a = 7, s = 4, m = 7, t = 4, d = 11, u = 3$
 $n = 31, a = 9, s = 4, m = 5, t = 3, d = 7, u = 3$
 $n = 31, a = 9, s = 4, m = 5, t = 3, d = 9, u = 3$
 $n = 31, a = 9, s = 4, m = 5, t = 3, d = 11, u = 3$
 $n = 31, a = 9, s = 5, m = 5, t = 3, d = 7, u = 3$
 $n = 31, a = 9, s = 5, m = 5, t = 3, d = 9, u = 3$
 $n = 31, a = 9, s = 5, m = 5, t = 3, d = 11, u = 3$

A.7 Java program.

What follows is the code for the Java program used to produce the results reported throughout this chapter. Readers are invited to use (and modify) this code freely to further explore the properties of the solution space.

```

import java.io.*;
public class DPcondition {
    /*
    This class is designed to explore the solution space in which the movements underlying
    DP orders can be explained as reducing c-command totals.
    */
    public static void main(String[] args) throws IOException {

        int NumberOfPossibilities = 0;
        int NumberOfSolutions = 0;
        boolean PrintSol;

        System.out.println("Enter a maximum size for NPs:");
        BufferedReader bufferedreader = new BufferedReader(new
        InputStreamReader(System.in));
        String number1 = bufferedreader.readLine();
        int NounCap = Integer.parseInt(number1);

        System.out.println("Enter a maximum size for AdjPs:");
        String number2 = bufferedreader.readLine();
        int AdjCap = Integer.parseInt(number2);

        System.out.println("Enter a maximum depth for AdjPs:");
        String number3 = bufferedreader.readLine();
        int AdjDep = Integer.parseInt(number3);

        System.out.println("Enter a maximum size for NumPs:");
        String number4 = bufferedreader.readLine();
        int NumCap = Integer.parseInt(number4);

        System.out.println("Enter a maximum depth for NumPs:");
        String number5 = bufferedreader.readLine();
        int NumDep = Integer.parseInt(number5);

        System.out.println("Enter a maximum size for DemPs:");
        String number6 = bufferedreader.readLine();
        int DemCap = Integer.parseInt(number6);

        System.out.println("Enter a maximum depth for DemPs:");
        String number7 = bufferedreader.readLine();
        int DemDep = Integer.parseInt(number7);

        System.out.println("Print list of viable solutions (y), or not (n)? BEWARE, only do this
        after verifying solution set is small! ");
    }
}

```



```

String PrintSolChoice = bufferedreader.readLine();
if (PrintSolChoice.charAt(0)=='y')
{
PrintSol = true;
}
else PrintSol = false;

for (int n = 1; n < NounCap + 1; n = n + 2)
{
for (int s = 2; s < AdjDep + 1; s++)
{
for (int a = 2 * s - 1; a < AdjCap + 1; a = a + 2)
{
for (int t = 2; t < NumDep + 1; t++)
{
for (int m = 2 * t - 1; m < NumCap + 1; m = m + 2)
{
for (int u = 2; u < DemDep + 1; u++)
{
for (int d = 2 * u - 1; d < DemCap + 1; d = d + 2)
{
NumberOfPossibilities++;
if (((a+1 < (s-2)*(n-1)) && (m+a < (s+t+3)*(n-1)) && (a+m+d-1 < (s+t+u-4)*(n-1)) && (m+d < (t+u-3)*(a+n-2)) && ((a+m+d+1 < (s-1)*(n-1)) || (m+d < (t+u-3)*(n+a))) && (m+1 < (t-2)*(a+n-2)) && ((a+m+2 < (s-1)*(n-1)) || (m+1 < (t-2)*(n+a))) && (a+m+d+1 < (s+u-2)*(n-1)) && (d+1 < (u-2)*(a+n+m-3)) && (d+1 < (u-2)*(a+n+m-1)) && ((a+m+d-1 < (s+t-2)*(n-1)) || (d+1 < u*(a-1) + (u-2)*(n+m))) && ((m+d+2 < (t-1)*(a+n-2)) || (d+1 < u*(n-1) + (u-2)*(a+m))) && (((m+d+2 < (t-1)*(a+n-2)) && (a+m+d+3 < (s-1)*(n-1))) || ((d+1 < u*(n-1) + (u-2)*(a+m)) && (a+m+d+3 < (s-2)*(n-1))) || ((d+1 < (u-2)*(a+n+m-1)) && (m+d+2 < (t-1)*(a+n))) || ((a+m+2 < (s-1)*(n-1)) && (d+1 < (u-2)*(a+n+m+1))) || ((m+1 < (t-2)*(n+a)) && (d+1 < (u-2)*(a+n+m+1)))) && (2*(a+n)+d+2 > (u-1)*(m-1)) && (a+n+d+4 > (u-1)*(m-1)) && (a+n+d+4 > u*(m-1)) && (n+m+d+2 > (t+u-2)*(a-1)) && (n+m+d+4 > u*(a-1)) && (n+m+3 > (t-1)*(a-1)) && (n+d+3 > (u-1)*(a+m-2)) && (n+d+3 > (u-1)*(a+m)) && !((d+m > a+n+4) && ((m+n+a-3)*(u-2)>d+1)&&((a+n-2)*(t-1)>d+m+2)&&((n-1)*(s-1)>d+m+a+3)) && !(((m-1)*3 > d+a+n+7) || ((a-1)*2 > d+m+n+7) || ((m+1)*2 > d+a+n+5) || (a+m+2 > n+d+4) || ((n-1 > d+m+a+7) && (((a+m+2)*2>d+n+6) || ((m+1)*3>d+a+n+7)||((m-1)*4>d+a+n+9) ) ) ) &&(((d+1)*2> n+a+m+5) &&((m+n+a-3)*(u-2)>d+1)&&((a+n-2)*(t-1)>d+m+2)&&((n-1)*(s-1)>d+m+a+3) ) && ! ( ((n-1)*(s-1)> a+m+2)&&((n+a-2)*(u-2)>m+1) &&( ((a+m)*(u-1)>d+n+2) || ((m-1)*u>d+a+n+3) || ((a-1)*u>d+m+n+3) || ( ((n-1)*(u-1)>d+m+a+3) && (((a+m+2)*(u-1)>d+n+2)||((a+m)*u>d+n+4)||((m-1)*(u+1)>d+a+n+5)))) ) ) &&! ((a+m > d+n+4)&&((n-1)*s>d+m+a+3)&&((n+a+m-1)*(u-2)>d+1)&&((n+a-2)*(u-2)>m+1) ) &&! ( ((n+a-2)*(u-2)>m+1) && ( ((m-1)*(u-1)>d+a+n+1) || ( ((n-1)*(s+u-2)>d+a+m+1)

```


REFERENCES

- Abels, Klaus. 2003. *Successive Cyclicity, Anti-locality, and Adposition Stranding*. Ph.D. dissertation, University of Connecticut.
- Abels, Klaus. 2011. Hierarchy-order relations in the Germanic verb cluster and in the noun phrase. *Groninger Arbeiten zur Germanistischen Linguistik* 53.2: 1-28.
- Abels, Klaus and Ad Neeleman. 2009. Universal 20 without the LCA. In J.M. Brucart, A. Gavarro, and J. Sola, eds., *Merging features: Computation, interpretation, and acquisition*. New York: Oxford University Press, 60-79.
- Abney, Steve. 1987. *The English Noun Phrase in its Sentential Aspect*. Ph.D. dissertation, MIT.
- Aboh, Enoch. 2004. *The Morphosyntax of Complement-Head Sequences*. New York: Oxford University Press.
- Ackema, Peter & Ad Neeleman. 2004. *Beyond Morphology*. New York: Oxford University Press.
- Alexiadou, Artemis & Elena Anagnostopoulou. 1998. Parameterizing Agr: word order, verb movement and EPP-checking. *Natural Language and Linguistic Theory* 16.3: 491-539.
- Anderson, Stephen R. 1992. *A-morphous Morphology*. Cambridge: Cambridge University Press.
- Antinucci, Francesco, and R. Miller. 1976. How children talk about what happened. *Journal of Child Language* 3: 167-189.
- Aoun, Joseph and Dominique Sportiche. 1983. On the Formal Theory of Government. *Linguistic Review* 2: 211-36.
- Bahloul, M., and W. Harbert. 1993. Agreement asymmetries in Arabic. In *Proceedings of the Eleventh West Coast Conference on Formal Linguistics*. Stanford, Calif.: CSLI Publications, 15-31.
- Baker, Mark. 1988. *Incorporation*. Chicago: Chicago University Press.
- Baker, Mark. 1985. The Mirror Principle and morphosyntactic explanation. *Linguistic Inquiry* 16: 373-416.
- Ball, Phillip. 1999. *The Self-Made Tapestry: Pattern Formation in Nature*. New York:

Oxford University Press.

Barker, Chris and Geoffrey Pullum. 1990. A theory of command relations. *Linguistics and Philosophy* 13: 1-34.

Barss, Andrew, ed. 2003. *Anaphora: A Reference Guide*. Malden, MA: Blackwell.

Bejan, Adrian. 2000. *Shape and Structure from Engineering to Nature*. Cambridge: Cambridge University Press.

Bejar, Susana and Milan Rezac. 2009. Cyclic Agree. *Linguistic Inquiry* 40(1): 35-73.

Belletti, Adriana, ed. 2004. *Structures and Beyond: The Cartography of Syntactic Structures*. New York: Oxford University Press.

Benmamoun, Elabbas. 1992. *Functional and Inflectional Morphology: Problems of Projection, Representation and Derivation*. Ph.D. dissertation, University of Southern California, Los Angeles.

den Besten, Hans. 1983. On the interaction of root transformations and lexical deletive rules. In W. Abraham, ed., *On the Formal Syntax of Westgermania*. Philadelphia: John Benjamins, 47-131.

den Besten, Hans, and Gert Webelhuth. 1987. Remnant topicalization and the constituent structure of VP in the Germanic SOV languages. *GLOW Newsletter* 18:15-16.

Biberauer, Tina, Anders Holmberg and Ian Roberts. 2007. Disharmonic word-order systems and the Final-over-Final-Constraint (FOFC), *Proceedings of the Incontro di Grammatica Generativa XXXIII* edited by A. Bisetto & F. Barbieri, 86-105.

Binder, Philippe M. 2008. Frustration in complexity. *Science* 320, 322-323.

Bobaljik, Jonathan. 2008. Where's Phi? Agreement as a post-syntactic operation. In Daniel Harbour, David Adger, and Susana Béjar, eds., *Phi Theory: Phi-Features across Modules and Interfaces*. New York: Oxford University Press, 295-328.

Boeckx, Cedric. 2003. *Islands and chains*. Philadelphia: John Benjamins.

Boeckx, Cedric and Massimo Piatelli-Palmarini. 2005. Language as a natural object, linguistics as a natural science. *The Linguistic Review* 22(2-4): 447-466.

Boeckx, Cedric. 2008. *Bare Syntax*. New York: Oxford University Press.

- Boeckx, Cedric. 2010. Linguistic minimalism. In Bernd Heine and Heiko Narrog, eds., *The Oxford Handbook of Linguistic Analysis*. New York: Oxford University Press, 485-505.
- Boeckx, Cedric. 2011. Some reflections on Darwin's Problem in the context of Cartesian biolinguistics. In Anna Maria Di Sciullo and Cedric Boeckx, eds., *The Biolinguistic Enterprise: New Perspectives on the Evolution and Nature of the Human Language Faculty*. New York: Oxford University Press, 42-64.
- Borer, Hagit. 1984. *Parametric Syntax: Case Studies in Semitic and Romance Languages*. Dordrecht: Foris.
- Bošković, Željko. 1997. *The Syntax of Nonfinite Complementation: An Economy Approach*. Cambridge, MA: MIT Press.
- Bošković, Željko. 2001. *On the Nature of the Syntax-Phonology Interface: Cliticization and Related Phenomena*. Amsterdam: Elsevier Science.
- Bowers, John. 1993. The syntax of predication. *Linguistic Inquiry* 24(4): 591-656.
- Brandi, Luciana and Patrizia Cordin. 1989. Two Italian dialects and the Null Subject Parameter. In Osvaldo Jaeggli and Kenneth Safir, eds., *The Null Subject Parameter*. Dordrecht: Kluwer, 111-142.
- Bresnan, Joan. 1971. Syntactic stress and syntactic transformations. *Language* 47(2): 257-281.
- Bresnan, Joan. 1972. *The Theory of Complementation in English Syntax*. Ph.D. dissertation, MIT.
- Cardinaletti, Anna. 2004. Towards a cartography of subject positions. In Luigi Rizzi, ed., *The Structure of CP and IP. The Cartography of Syntactic Structures, Vol.2*. New York: Oxford University Press, 115-165.
- Cardinaletti, Anna and Michal Starke. 1999. The typology of structural deficiency: A case study of the three classes of pronouns. In Henk van Riemsdijk, ed., *Clitics in the Languages of Europe*. Berlin: Mouton de Gruyter, 145-233.
- Carnie, Andrew. 2010. *Constituent Structure*. 2nd ed. New York: Oxford University Press.
- Carnie, Andrew and David P. Medeiros. 2005. Tree maximization and the Extended Projection Principle. *Coyote Working Papers in Linguistics* 14: 51-55.

- Cecchetto, Carlo and Catarina Donati. 2010. On labeling: Principle C and head movement. *Syntax* 13(3): 241-278.
- Chametzky, Robert. 1996. *A Theory of Phrase Markers and the Extended Base*. Albany: SUNY Press.
- Chametzky, Robert. 2000. *Phrase Structure: From GB to Minimalism*. Malden, MA: Blackwell.
- Cherniak, Christopher, Zekeria Mokhtarzada, Raul Rodriguez-Esteban, and Kelly Changizi. 2004. Global optimization of cerebral cortex layout. *Proceedings of the National Academy of Sciences* 101(4): 1081-1086.
- Chomsky, Noam. 1957. *Syntactic Structures*. Berlin: Mouton de Gruyter.
- Chomsky, Noam. 1970. Remarks on nominalization. In R. Jacobs and P. Rosenbaum, eds., *Readings in English Transformational Grammar*. Waltham: Ginn, 184-221.
- Chomsky, Noam. 1973. Conditions on transformations. In Stephen R. Anderson and Paul Kiparsky, eds., *A Festschrift for Morris Halle*. New York: Holt, Rinehart & Winston, 232-86.
- Chomsky, Noam. 1981. *Lectures on Government and Binding*. Dordrecht: Foris.
- Chomsky, Noam. 1982. *Some Concepts and Consequences of the Theory of Government and Binding*. Cambridge, MA: MIT Press.
- Chomsky, Noam. 1986. *Barriers*. Cambridge, MA: MIT Press.
- Chomsky, Noam. 1995a. Bare phrase structure. In Gert Webelhuth, ed., *The Principles and Parameters Approach to Syntactic Theory: A Synopsis*. Malden, MA: Blackwell, 385-439.
- Chomsky, Noam. 1995b. *The Minimalist Program*. Cambridge, MA: MIT Press.
- Chomsky, Noam. 2000. Minimalist inquiries: The framework. In Roger Martin, David Michael, and Juan Uriagereka, eds., *Step by Step: Essays on Minimalist Syntax in Honor of Howard Lasnik*. Cambridge, MA: MIT Press, 89-155.
- Chomsky, Noam. 2001. Derivation by phase. In Michael Kenstowicz, ed., *Ken Hale: A Life in Language*. Cambridge, MA: MIT Press, 1-52.
- Chomsky, Noam. 2005. Three factors in language design. *Linguistic Inquiry* 36(1): 1-22.

- Chomsky, Noam. 2007. Approaching UG from below. In Uli Sauerland and Hans-Martin Gärtner, eds., *Interfaces + Recursion = Language? Chomsky's Minimalism and the View from Syntax-Semantics*. Berlin: Mouton de Gruyter, 1-30.
- Chomsky, Noam. 2008. On phases. In Robert Freidin, Carlos Otero, and Maria Luisa Zubizarreta, eds., *Foundational Issues in Linguistic Theory*. Cambridge, MA: MIT Press, 133-166.
- Chomsky, Noam. 2009. Opening remarks. In Massimo Piatelli-Palmarini, Juan Uriagereka, and Pello Salaburu, eds., *Of Minds and Language: A dialogue in the Basque Country with Noam Chomsky*. New York: Oxford University Press, 13-43.
- Chomsky, Noam and Morris Halle. 1968. *The Sound Pattern of English*. New York: Harper and Row.
- Chomsky, Noam and George Miller. 1963. Introduction to the formal analysis of natural languages. In R. Duncan Luce, Robert R. Bush, and Eugene Galanter, eds., *Handbook of Mathematical Psychology*, Vol. 2. New York: Wiley, 269-321.
- Cinque, Guglielmo. 1993. A null theory of phrase and compound stress. *Linguistic Inquiry* 24: 239-297.
- Cinque, Guglielmo. 1994. On the evidence for partial N movement in the Romance DP. In Guglielmo Cinque, Jan Koster, Jean-Yves Pollock, Luigi Rizzi, and Raffaella Zanuttini, eds., *Paths Towards Universal Grammar: Essays in Honor of Richard Kayne*. Washington, D.C.: Georgetown University Press, 85-110.
- Cinque, Guglielmo. 1996. The antisymmetric programme: Theoretical and typological implications. *Journal of Linguistics* 32: 447-464.
- Cinque, Guglielmo. 1999. *Adverbs and Functional Heads: A Crosslinguistic Perspective*. New York: Oxford University Press.
- Cinque, Guglielmo. 2000. On Greenberg's universal 20 and the Semitic DP. In Laura Brugè, ed., *University of Venice Working Papers in Linguistics* 10(2):45-61.
- Cinque, Guglielmo. 2003. The prenominal origin of relative clauses. Paper presented at the Workshop on Antisymmetry and Remnant Movement, New York University.
- Cinque, Guglielmo, ed. 2002. *Functional Structure in DP and IP: The Cartography of Syntactic Structures*, Vol. 1. New York: Oxford University Press.
- Cinque, Guglielmo. 2004. Issues in adverbial syntax. *Lingua* 114: 683-710.

- Cinque, Guglielmo. 2005. Deriving Greenberg's universal 20 and its exceptions. *Linguistic Inquiry* 36(3): 315 – 332.
- Cinque, Guglielmo. 2006. Are all languages 'numeral classifier languages'? *Rivista di Grammatica Generativa* 31:119-122.
- Cinque, Guglielmo and Luigi Rizzi. 2010. The cartography of syntactic structures. In Bernd Heine and Heiko Narrog, eds., *The Oxford Handbook of Linguistic Analysis*. New York: Oxford University Press, 51-66.
- Croft, William. 2001. *Typology and Universals*. 2nd ed. Cambridge: Cambridge University Press.
- Culbertson, Jennifer, Paul Smolensky, and Géraldine Legendre. 2012. Learning biases predict a word order universal. *Cognition* 122: 306-329.
- Déchaine, Rose-Marie and Martina Wiltschko. 2002. Decomposing pronouns. *Linguistic Inquiry* 33: 409-442.
- Dediu, Dan and D. Robert Ladd. 2007. Linguistic Tone is related to the population frequency of the adaptive haplogroups of two brain size genes, ASPM and Microcephalin. *Proceedings of the National Academy of Sciences* 104(26): 10944-10949.
- den Dikken, Marcel. 1995. *Particles: On the Syntax of Verb-Particle, Triadic, and Causative Constructions*. New York: Oxford University Press.
- Diesing, Molly. 1992. *Indefinites*. Cambridge, MA: MIT Press.
- Douady, S. and Y. Couder. 1992. Phyllotaxis as a physical self-organized growth process. *Physical Review Letters* 68: 2098-2101.
- Dryer, Matt. 2009. The branching direction theory of word order universals revisited. In Sergio Scalise, Elisabetha Magni, and Antonietta Bisetto, eds., *Universals of Language Today*. London: Springer, 185-207.
- Emonds, Joseph. 1970. *Root and Structure-Preserving Transformations*. Ph.D. dissertation, MIT.
- Epstein, Samuel, Erich Groat, Ruriko Kawashima, and Hisatsugu Kitahara. 1998. *A Derivational Approach to Syntactic Relations*. New York: Oxford University Press.

- Fassi Fehri, Abdelkader. 1993. *Issues in the Structure of Arabic Clauses and Words*. Dordrecht: Kluwer.
- Fox, Danny and David Pesetsky. 2005. Cyclic linearization of syntactic structure. *Theoretical Linguistics* 31: 1-45.
- Frank, Robert and Fero Kuminiak. 2000. Primitive asymmetric c-command derives X-bar theory. In M. Hirotani, A. Coetzee, N. Hall, and J.-Y. Kim, eds., *Proceedings of NELS* 30: 203-17.
- Frank, Robert and K. Vijay-Shankar. 2001. Primitive c-command. *Syntax* 4: 164-204.
- Freidin, Robert and Jean-Roger Vergnaud. 2001. Exquisite connections: Some remarks on the evolution of linguistic theory. *Lingua* 111, 637-666.
- Fukui, Naoki. 1995. *Theory of Projection in Syntax*. Stanford, CA: CSLI Publications.
- Furuya, Kaori. 2008. DP hypothesis for Japanese “bare” noun phrases. *University of Pennsylvania Working Papers in Linguistics* 14: 149-162.
- Goldberger, Ary L., Bruce J. West, Timothy Dresselhaus and Vinod Bhargava. 1985. Bronchial asymmetry and Fibonacci scaling. *Experientia* 41.
- Greenberg, Joseph. 1963. Some universals of grammar with particular reference to the order of meaningful elements. In Joseph Greenberg, ed., *Universals of Language*. Cambridge, Mass.: MIT Press, 73-113.
- Grimshaw, Jane. 1991. Extended projections. Ms., Brandeis University.
- Grimshaw, Jane. 2000. Locality and extended projection. In Peter Coopmans, Martin Everaert, and Jane Grimshaw, eds., *Lexical Specification and Insertion*. Philadelphia: John Benjamins, 115-135.
- Grohmann, Kleanthes K. 2000. *Prolific peripheries: A radical view from the left*. PhD dissertation, University of Maryland, College Park.
- Guimarães, Max. 2000. In defense of vacuous projections in bare phrase structure. *University of Maryland Working Papers in Linguistics* 9: 90-115.
- Hale, Kenneth. 1983. Warlpiri and the grammar of non-configurational languages. *Natural Language and Linguistic Theory* 1: 5-47.
- Hale, Kenneth and Samuel J. Keyser. 1993. On argument structure and the lexical expression of syntactic relations. In Kenneth Hale and Samuel J. Keyser, eds., *The*

- View from Building 20: Essays in Linguistics in Honor of Sylvain Bromberger.* Cambridge, MA: MIT Press, 111-176.
- Halle, Morris and Jean-Roger Vergnaud. 1987. *An Essay on Stress*. Cambridge, MA: MIT Press.
- Halle, Morris and Alec Marantz. 1993. Distributed morphology and the pieces of inflection. In Kenneth Hale and Samuel J. Keyser, eds., *The View from Building 20: Essays in Linguistics in Honor of Sylvain Bromberger*, 53-110. Cambridge, MA: MIT Press.
- Halle, Morris, and Alec Marantz. 1994. Some key features of Distributed Morphology. In *MIT Working Papers in Linguistics 21*, Andrew Carnie, Heidi Harley, and Tony Bures, eds., 275-288.
- Harley, Heidi, & Elizabeth Ritter. 2002. Structuring the bundle: A universal morphosyntactic feature geometry. In Horst J. Simon and Heike Wiese, eds., *Pronouns: Grammar and Reference*. Philadelphia: John Benjamins, 23- 62.
- Hauser, Mark, Noam Chomsky and Tecumseh Fitch. 2002. The faculty of language: What is it, who has it, and how did it evolve? *Science* 298: 159-79.
- Hawkins, John. 1983. *Word Order Universals*. New York: Academic Press.
- Heine, Bernd and Tania Kuteva. 2002. *World Lexicon of Grammaticalization*. Cambridge: Cambridge University Press.
- Hinzen, Wolfram. 2011. Emergence of a systemic semantics through minimal and underspecified codes. In Anna Maria Di Sciullo and Cedric Boeckx, eds., *The Biolinguistic Enterprise: New Perspectives on the Evolution and Nature of the Human Language Faculty*. New York: Oxford University Press, 417-439.
- Hiraiwa, Ken. 2002. Movement and derivation: Eliminating the PBC. In *Penn Linguistics Colloquium 26*. Philadelphia, PA.
- Holmberg, Anders. 1986. *Word Order and Syntactic Features in the Scandinavian Languages and English*. Ph.D. dissertation, University of Stockholm.
- Hornstein, Norbert, Jairo Nunes, and Kleanthes Grohmann. 2005. *Understanding Minimalism*. Cambridge: Cambridge University Press.
- Hornstein, Norbert. 2009. *A Theory of Syntax: Minimal Operations and Universal Grammar*. Cambridge: Cambridge University Press.

- Idsardi, William. 2008. Combinatorics for metrical feet. *Biolinguistics* 2: 233-236.
- Idsardi, William and Juan Uriagereka. 2009. Metrical combinatorics and the real half of the Fibonacci Sequence. *Biolinguistics* 3(4): 404-406.
- Jackendoff, Ray. 1977. *X-bar Syntax: A Study of Phrase Structure*. Cambridge, MA: MIT Press.
- Jayaseelan, K. A. 2008. Bare phrase structure and specifier-less syntax. *Biolinguistics* 2: 87-106.
- Jean, Roger V. 1994. *Phyllotaxis: A Systematic Study in Plant Morphogenesis*. Cambridge: Cambridge University Press.
- Johnson, Kyle. 1991. Object positions. *Natural Language and Linguistic Theory* 9: 577-636.
- Julien, Marit. 2002. *Syntactic Heads and Word Formation*. New York: Oxford University Press.
- Kahnemuyipour, Arsalan and Diane Massam. 2006. Patterns of phrasal movement: the Niuean DP. In Hans-Martin Gartner, Paul Law and Joachim Sabel, eds., *Clause Structure and Adjuncts in Austronesian Languages*. Berlin: Mouton de Gruyter, 125-150.
- Kayne, Richard. 1984. *Connectedness and Binary Branching*. Dordrecht: Foris.
- Kayne, Richard. 1994. *The Antisymmetry of Syntax*. Cambridge, MA: MIT Press.
- Kayne, Richard. 2000. *Parameters and Universals*. New York: Oxford University Press.
- Kayne, Richard. 2002. Pronouns and their antecedents. In Samuel Epstein and T. Daniel Seely, eds., *Derivation and Explanation in the Minimalist Program*. Malden, MA: Blackwell, 133-166.
- Kayne, Richard. 2003. Silent years, silent hours. In *Grammar in Focus. Festschrift for Christer Platzack*, Vol. 2. Lund: Wallin and Dalholm, 209-226.
- Kayne, Richard. 2005. Some Notes on Comparative Syntax with Special Reference to English and French. In Richard Kayne and Guglielmo Cinque, eds., *Oxford Handbook of Comparative Syntax*. New York: Oxford University Press, 3-69.
- Kayne, Richard. 2008. Antisymmetry and the Lexicon. *Linguistic Variation Yearbook* 8(1): 1-31.

- Koizumi, Masatoshi. 1993. Object Agreement Phrases and the split VP hypothesis. *MIT Working Papers in Linguistics* 18, 99-148.
- Koopman, Hilda. 1996. The spec-head configuration. *Syntax at Sunset: UCLA Working Papers in Syntax and Semantics*. ed. E. Garret and F. Lee, 37-64.
- Koopman, Hilda and Dominique Sportiche. 1991. The position of subjects. *Lingua* 85: 211-258.
- Koopman, Hilda and Anna Szabolcsi. 2000. *Verbal Complexes*. Cambridge, MA: MIT Press.
- Koster Jan. 1975. Dutch as an SOV language. *Linguistic Analysis* 1, 111-136.
- Koster, Jan. 2007. Structure-preservingness, internal Merge, and the strict locality of triads. In Simin Karimi, Vida Samiiian, & Wendy K. Wilkins, eds., *Phrasal and Clausal Architecture: Syntactic Derivation and Interpretation*. Philadelphia: John Benjamins: 188-205.
- Kratzer, Angelika. 1996. Severing the external argument from its verb. In Johan Rooryk and Laurie Zaring, eds, *Phrase Structure and the Lexicon*. Dordrecht: Kluwer, 109-37.
- Kratzer, Angelika and Elisabeth Selkirk. 2007. Phase theory and prosodic spellout: The case of verbs. *The Linguistic Review* 24: 93-105.
- Langacker, Ronald. 1966. On pronominalization and the chain of command. In W. Reibel and S. Schane, eds., *Modern Studies in English*. Englewood Cliffs, NJ; Prentice Hall, 160-86.
- Langendoen, D. Terence. 2003. Merge. In Andrew Carnie, Heidi Harley, and MaryAnn Willie, eds., *Formal Approaches to Function in Grammar*. Philadelphia: John Benjamins, 307-18.
- Larson, Richard. 1988. On the double object construction. *Linguistic Inquiry* 19(3): 335-91.
- Larson, Richard. 1990. Double objects revisited: Reply to Jackendoff. *Linguistic Inquiry* 21: 589-632.
- Lasnik, Howard. 1976. Remarks on coreference. *Linguistic Analysis* 2: 1-22.
- Lasnik, Howard. 2001a. A note on the EPP. *Linguistic Inquiry* 32(2): 356-361.

- Lasnik, Howard. 2001b. Subjects, objects, and the EPP. In William Davies and Stanley Dubinsky, eds., *Objects and Other Subjects*. Dordrecht: Kluwer, 103-121.
- Lasnik, Howard and Mamoru Saito. 1991. On the subject of infinitives. In *Papers from the 27th Regional Meeting of the Chicago Linguistic Society Part One: The General Session*, Lise M. Dobrin, Lynn Nichols, and Rosa M. Rodriguez, eds. Chicago: Chicago Linguistic Society, 324-343.
- Lasnik, Howard and Mamoru Saito. 1992. *Move Alpha*. Cambridge, MA: MIT Press.
- Levitov, Leonid S. 1991. Phyllotaxis of flux lattices in layered superconductors. *Physical Review Letters* 66(2): 224-227.
- Lieberman, Mark. 1975. *The Intonational System of English*. Ph.D. dissertation, MIT.
- Livio, Mario. 2002. *The Golden Ratio*. New York: Broadway books.
- Lu, Bingfu. 1998. *Left-right Asymmetries of Word Order Variation: A Functional Explanation*. Ph.D. dissertation, University of Southern California, Los Angeles.
- Mahajan, Anoop. 2000. Eliminating head movement. *GLOW Newsletter* 44: 44-45.
- Mandelbrot, Benoit. 1967. How long is the coast of Great Britain? Statistical self-similarity and the fractional dimension. *Science* 156: 636-638.
- Marácz, László. 1989. *Asymmetries in Hungarian*. Ph.D. dissertation, Rijksuniversiteit Groningen.
- Marantz, Alec. 1997. No escape from syntax: Don't try morphological analysis in the privacy of your own lexicon. *University of Pennsylvania Working Papers in Linguistics* 4: 201-225.
- May, Robert. 1985. *Logical Form: Its Structure and Derivation*. Cambridge, MA: MIT Press.
- McCloskey, James and Kenneth L. Hale. 1984. On the syntax of person-number inflection in modern Irish. *Natural Language and Linguistic Theory* 1(4): 487-533.
- McCloskey, James. 2002. Resumption, successive cyclicity, and the locality of operations. In Samuel Epstein and T. Daniel Seely, eds., *Derivation and Explanation in the Minimalist Program*. Malden, MA: Blackwell, 184-226.

- Medeiros, David P. 2008. Optimal growth in phrase structure. *Biolinguistics* 2: 152-95.
- Medeiros, David P. 2011. X-bar structure and the golden string. Paper presented at Complex Systems in Linguistics workshop, University of Southampton, UK.
- Miller, George and Noam Chomsky. 1963. Finitary models of language users. In R. Duncan Luce, Robert R. Bush, and Eugene Galanter, eds., *Handbook of Mathematical Psychology*, Vol. 2. New York: Wiley, 419-492.
- Miyagawa, Shigeru. 2010. *Why Agree? Why Move? Unifying Agreement-based and Discourse-configurational Languages*. Cambridge, MA: MIT Press.
- Moro, Andrea. 2000. *Dynamic Antisymmetry*. Cambridge, MA: MIT Press.
- Müller, Gereon. 1998. *Incomplete Category Fronting: A Derivational Approach to Remnant Movement in German*. Dordrecht: Kluwer.
- Muysken, Pieter. 1982. Parametrizing the notion "head". *Journal of Linguistic Research* 2: 57-75.
- Narita, Hiroki. 2010. Phase cycles in service of projection-free syntax. Ms., Harvard University.
- Nunes, Jairo. 1995. *The Copy Theory of Movement and Linearization of Chains in the Minimalist Program*. Ph.D. dissertation, University of Maryland, College Park.
- Oltra-Massuet, Maria and Karlos Arregi. 2005. Stress-by-structure in Spanish. *Linguistic Inquiry* 36(1):43-84.
- Ouhalla, Jamal. 1991. *Functional Categories and Parametric Variation*. London: Routledge.
- Partee, Barbara, Alice ter Meulen, and Robert Wall. 1993. *Mathematical Methods in Linguistics*. Dordrecht: Kluwer.
- Pearson, Matt. 2000. Two types of VO Languages. In Peter Svenonius, ed., *The Derivation of VO and OV*. Philadelphia: John Benjamins, 327-363.
- Pearson, Matt. 2007. Predicate fronting and constituent order in Malagasy. Ms., Reed College.
- Pereltsvaig, Asya. 2007a. *Copular Sentences in Russian: A Theory of Intra-Clausal Relations*. Dordrecht: Springer.

- Pereltsvaig, Asya. 2007b. On the universality of DP: A view from Russian. *Studia Linguistica* 61: 59-94.
- Piatelli-Palmarini, Massimo and Juan Uriagereka. 2008. Still a bridge too far? Biolinguistic questions for grounding language on brains. *Physics of Life Reviews* 5: 207-224.
- Pollock, Jean-Yves. 1989. Verb movement, Universal Grammar, and the structure of IP. *Linguistic Inquiry* 20: 365-424.
- Postal, Paul M. 1972. On some rules that are not successive cyclic. *Linguistic Inquiry* 3: 211-222.
- Postal, Paul M. 1974. *On Raising: One Rule of English Grammar and its Theoretical Implications*. Cambridge, MA: MIT Press.
- Prince, Alan and Paul Smolensky. 1993. *Optimality Theory: Constraint Interaction in Generative Grammar*. RuCCs Technical Report 2, Rutgers University Center for Cognitive Science.
- Progovac, Ljiljana. 1998. Determiner Phrase in a language without determiners. *Journal of Linguistics* 34: 165-179.
- Rackowski, Andrea. 1998. Malagasy adverbs. In Ileana Paul, ed., *The Structure of Malagasy Vol. II*, UCLA Occasional Papers in Linguistics 20.
- Rackowski, Andrea & Lisa Travis. 2000. V-initial languages: X or XP movement and adverbial placement. In Andrew Carnie and Eithne Guilfoyle, eds., *The Syntax of Verb Initial Languages*. Oxford Studies in Comparative Syntax. New York: Oxford University Press, 117-142.
- Reinhart, Tanya. 1976. *The Syntactic Domain of Anaphora*. Ph.D. dissertation, MIT.
- Richardson, John F. and Robert Chametzky. 1985. A string-based redefinition of c-command, *Proceedings of the Northeastern Linguistic Society* 15: 332-361.
- Rizzi, Luigi. 1990. *Relativized Minimality*. Cambridge, MA: MIT Press.
- Rizzi, Luigi. 1997. The fine structure of the left periphery. In Lilliane Haegeman, ed., *Elements of Grammar: Handbook of Generative Syntax*. Dordrecht: Kluwer, 281-337.
- Rizzi, Luigi, ed. 2004. *The Structure of CP and IP: The Cartography of Syntactic Structures*, Vol. 2. New York: Oxford University Press.

- Roopun, Anita, Mark Kramer, Lucy Carracedo, Marcus Kaiser, Ceri Davies, Roger Traub, Nancy Koppell, and Miles Whittington. 2008. Temporal interactions between cortical rhythms. *Frontiers in Neuroscience* 2(2): 145-154.
- Rosenbaum, Peter S. 1967. *The Grammar of English Predicate Complement Constructions*. Cambridge, MA: MIT Press.
- Ross, John R. 1967. *Constraints on Variables in Syntax*. Ph.D. dissertation, MIT.
- Scott, Gary-John. 2002. Stacked adjectival modification and the structure of nominal phrases. In Guglielmo Cinque, ed., *Functional Structure in DP and IP: The Cartography of Syntactic Structures. Vol. 1*. New York: Oxford University Press, 91-120.
- Schlenker, Phillippe. 1999. *Propositional Attitudes and Indexicality: A Cross-Categorical Approach*. Ph.D. dissertation, MIT.
- Schlyter, Suzanne. 1990. The acquisition of tense and aspect. In: J. Meisel, ed., *Two First Languages: Early Grammatical Development in Bilingual Children*. Dordrecht: Foris, 87-121.
- Schroeder, Manfred. 1997. *Number Theory in Science and Communication, with Applications in Cryptography, Physics, Digital Information, and Self-similarity*. 3rd edition. New York: Springer-Verlag.
- Shlonsky, Ur. 2004. The form of Semitic nominals. *Lingua* 114: 1465-1526.
- Shlonsky, Ur. 2010. The cartographic enterprise in syntax. *Language and Linguistics Compass*, Wiley Online.
- Snow, Mary and Robert Snow. 1962. A theory of the regulation of phyllotaxis based on *Lupinus albus*. *Philosophical Transactions of the Royal Society of London B* 244(717): 483-514.
- Speas, Margaret. 1990. *Phrase Structure in Natural Language*. Dordrecht: Kluwer.
- Starke, Michal. 1995. On the format for small clauses. In Anna Cardinaletti and Maria Guasti, eds., *Syntax and Semantics 28: Small Clauses*. New York: Academic Press, 255-292.
- Starke, Michal. 2004. On the inexistence of specifiers and the nature of heads. In Adriana Belletti, ed., *The Cartography of Syntactic Structures Vol. 3: Structures and Beyond*. New York: Oxford University Press, 252-268.

- Steddy, Sam & Vieri Samek-Lodovici. 2011. On the ungrammaticality of remnant movement in the derivation Greenberg's universal 20. *Linguistic Inquiry* 42(3): 445-469.
- Stowell, Timothy. 1981. *Origins of Phrase Structure*. Ph.D. dissertation, MIT.
- Stuurman, Frits. 1985. *Phrase Structure Theory in Generative Grammar*. Dordrecht: Foris.
- Svenonius, Peter. 1994. *Dependent Nexus: Subordinate Predication Structures in English and the Scandinavian Languages*. Ph.D. dissertation, University of California, Santa Cruz.
- Svenonius, Peter. 2007. 1... 3-2. In Gillian Ramchand and Charles Reiss, eds., *Oxford Handbook of Linguistic Interfaces*. New York: Oxford University Press, 239-288.
- Svenonius, Peter. 2008. The position of adjectives and other phrasal modifiers in the decomposition of DP. In Louise McNally and Chris Kennedy, eds., *Adjectives and Adverbs: Syntax, Semantics, and Discourse*. New York: Oxford University Press.
- Thompson, D'Arcy Wentworth. 1917/1992. *On Growth and Form* [abridged edn., prepared by John Tyler Bonner]. Cambridge: Cambridge University Press.
- Thráinsson, Höskuldur. 2001. Object shift and scrambling. In Mark Baltin and Chris Collins, eds., *The Handbook of Contemporary Syntactic Theory*. Malden, MA: Blackwell, 148-202.
- Travis, Lisa. 1984. *Parameters and Effects of Word Order Variation*. Ph.D. dissertation, MIT.
- Tsang, Kwok Y. 1986. Dimensionality of strange attractors determined analytically. *Physical Review Letters* 57: 12, 1390-1393.
- Turing, Alan. 1952. The chemical basis of morphogenesis. *Transactions of the Royal Society of London B* 237: 37-72.
- Uriagereka, Juan and Norbert Hornstein. 2002. Reprojections. In Samuel Epstein and T. Daniel Seely, eds., *Derivation and Explanation in the Minimalist Program*. Malden, MA: Blackwell, 106-132.
- Uriagereka, Juan. 1998. *Rhyme and Reason: An Introduction to Minimalist Syntax*. Cambridge, MA: MIT Press.

Uriagereka, Juan. 1999. Multiple spellout. In Samuel Epstein and Norbert Hornstein, eds., *Working Minimalism*. Cambridge, MA: MIT Press, 251-82.

Uriagereka, Juan. 2002. *Derivations: Exploring the dynamics of syntax*. London: Routledge.

Wasow, Thomas A. 1972. *Anaphoric Relations in English*. Ph.D. dissertation, MIT.

Weist, Richard, 1986. Tense and aspect. In P. Fletcher and M. Garman, eds., *Language Acquisition. Studies in First Language Development*. Cambridge: Cambridge University Press, 356-374.

Williams, Edwin and Anna-Maria Di Sciullo. 1987. *On the Definition of Word*. Cambridge, MA: MIT Press.

Yngve Victor. 1960. A model and an hypothesis for language structure. *Proceedings of the American Philosophical Society* 104: 444-466.

Zwart, C. Jan-Wouter. 1994. Dutch is head-initial. *The Linguistic Review* 11: 377-406.