Projection as an Epiphenomenon of Optimal Packing

Ways of Structure Building, Nov. 14 2008

David P Medeiros, University of Arizona
medeiros@email.arizona.edu

1.0 Introduction

My agenda: 'biolinguistic' explanation of the X-bar schema & the notion of 'projection'.

In the Minimalist Program: "The goal with respect to X-bar theory is, we might say, to have no cake and to eat it, too." (Chametzky 2000: 149)

My claim: "function follows form".
In Minimalism, the explanatory buck often stops with interface requirements and lexical features – such that (syntactic) form follows function (in non-syntactic components).
I claim that "blind" optimization of branching structure is to blame.
Such optimization is best achieved by X-bar-like geometry…
Which is endocentric/"projective" as a happy accident.
We will end up with a sort of "physics suffices" view of syntax doing what it does for naturalistic/Galilean reasons, with "legibility" as a spandrel -- a consequence, rather than a cause, of syntactic facts.

Projection is not so much *necessary* (i.e., 'required' by what syntax is 'for') as *inevitable* (falling out 'for free' from mathematical principles of optimal packing).

(1)The enemy destroyed the city.
(2)the enemy's destruction of the city

The properties of complex expressions are most directly determined by a single most salient lexical item (a 'head') in their interior.

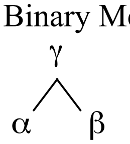
Obviously, we should not be happy with any theory of language that does not end up capturing this fundamental descriptive fact in some way.

But that doesn't mean our theory should *start* with that, by stipulating it as a primitive.
…What if we could derive projection from more elementary properties?

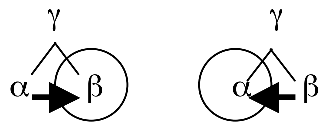
2.0 C-command and dominance in binary-branching structure

Assuming without comment that syntax builds strictly binary branching structure:

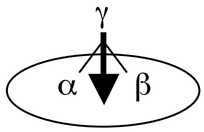
(3) Binary Merge:



(4) C-command: α c-commands b = # nodes in β, β c-commands a = # nodes in α.



(5) Dominance (containment): γ dominates $a + b$ nodes. (NB: taken as irreflexive)



(6) By induction:
In a binary-branching tree, **# of c-command relations = # of dominance relations.**

Now, for the **Big Observation**:

(7) Merging the *same* objects, via binary Merge, can lead to *different* totals of c-command/dominance relations.

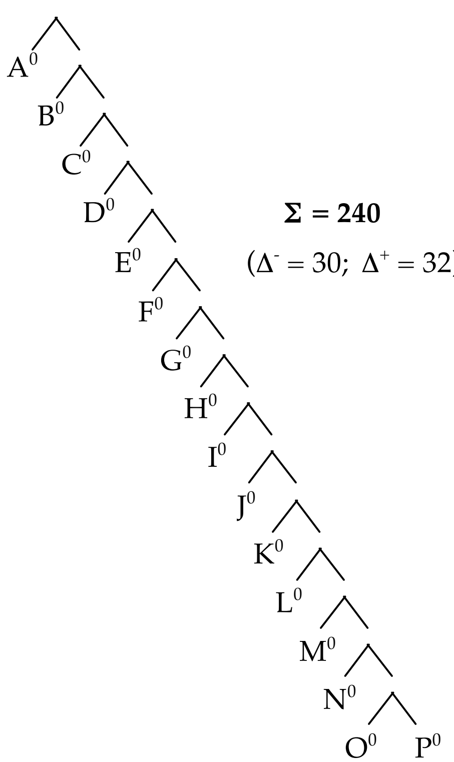
Put another way:

(7′) ***Not all binary-branching structure is created equal!***

Compare two extremes for building binary-branching structure via Merge.

Strategy 1: "*Stack them up.*" Begin by Merging two terminal atoms. Continue Merging one terminal at a time to the result until all have been combined into a single object.

(8)

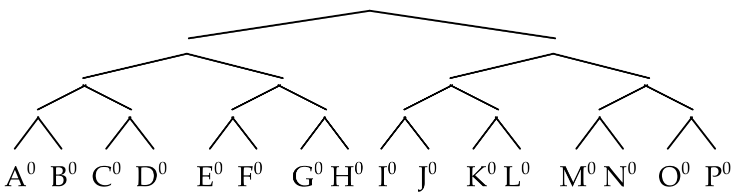


Σ = 240

($\Delta^- = 30$; $\Delta^+ = 32$)

Strategy 2: "*Pack them in.*" Merge so that the final structure has all terminals at a uniform depth (or at two adjacent levels of depth).

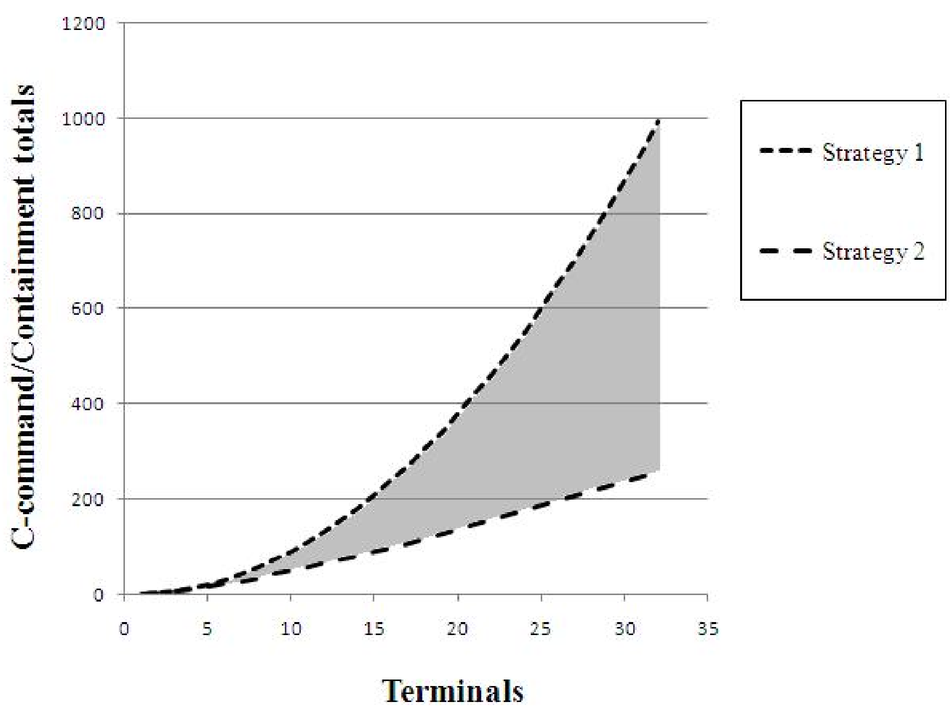
(9)



Σ = 98
($\Delta^- = 8$; $\Delta^+ = 10$)

Strategy 1 is as simple an algorithm as possible to get the job done.
- *worst possible* choice for minimizing c-command and containment relations.

Strategy 2: absolute minimization of c-command and containment relations.
-But not a straightforward algorithm to carry out.
Representational condition on the final form. Actually deriving such a form?
-Full 'vertical' and 'horizontal' informational flow in the derivation.
Vertical: match internal structure of objects Merged, all the way down.
Horizontal: pan-derivational synchronization. E.g., to add one more terminal to (9), better to do so at bottom (early asymmetry) than at the top. (+10 vs +32)



3.0 Why does this matter?

"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." (Langendoen 2003: 307; emphasis added)

C-command and dominance are THE relations for syntax; all syntactic relations seem to be "carried" by these structural relations…

C-command:
Probe-goal (Chomsky 2000)
Linearization (Kayne 1994)
Determination of scope (May 1985)
Binding theory (Reinhart 1976), and so on,

Dominance relations:
(Potential) 'is-a' relations
Number of steps of cyclic stress (re-)computation an embedded element may be subject to (Chomsky and Halle 1968, Hayes 1995).

Interleaving of Merge and search (cf. Epstein et al 1998).
That is: syntax proceeds by iterating the following cycle:
Merge α and β,
Compute new c-command/dominance relations,
Repeat.

Minimal search: Chomsky argues for complement as domain for c-command because it is smallest possible 'readily defined' space; spec-head ruled out. Thus, minimal search *for individual instances of search.*

Making trees flatter/more parallel: minimize sum over spaces-searched…
minimal search, *in aggregate*.

→ Minimizing *total number* inevitably goes hand-in-hand with minimizing *length of individual relations*.
So, we can sensibly talk about this structural factor in terms of *locality*, as a generic drive to "minimize links".

→ Surprisingly, if length of relations is extrinsically capped (e.g., no c-command relations of length greater than 3 nodes is ever computed), minimization (in both senses – individual and total) *still* favors more-balanced structure.

4.0 Mini-max optimization:

To minimize both:
(i) c-command/containment totals,
and
(ii) algorithmic complexity (keep to relatively 'local' information in derivation)

The ‘best’ format is:

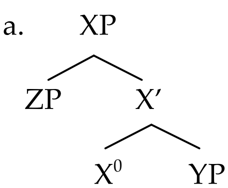
(10) **[α [β…[γ [X^0 δ]]…]]**
Where α, β, … are themselves structured according to (10),
and X^0 is a terminal. (see Medeiros 2008 for details)

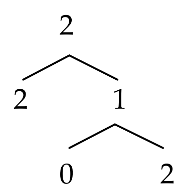
For example,
(11) [X^0 α]
(12) [α [X^0 β]]
(13) [α [β [X^0 γ]]]

…but not:
(14) [X^0 [α β]]

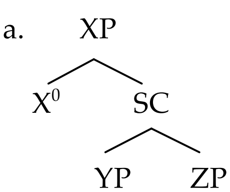
The idea: we should be willing to be surprised by this.

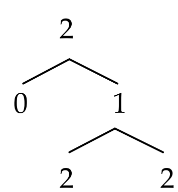
Phrasal formats: patterns for combining terminals into structures with indefinite recursion. There are many ways of doing this, very few of which fit (10).

(15) a.


b.


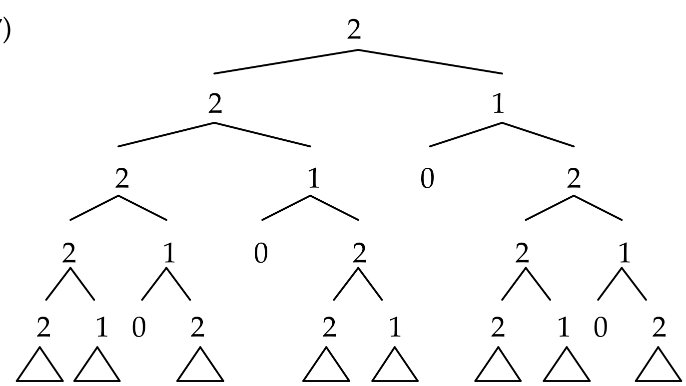
X-bar schema (Chomsky 1970, Jackendoff 1977)
(basically, (12) from above)

(16) a.


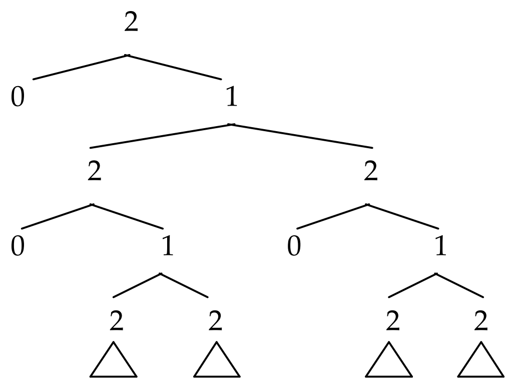
b.


Moro (2000)-style analysis of copula + small clause
(basically, (14) from above)

Iterate these patterns: each XP (2) in (b) portion made the same way as full template, all the way down….

(17)


(18)

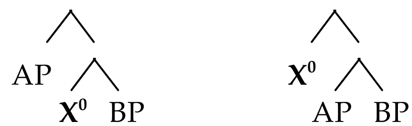


(15)/(17) is a "denser packing" than (16)/(18). As a consequence:

(19) X^0, AP, BP
- X^0 a bare lexical item
- AP and BP are internally complex.

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

(21) [X^0 [AP BP]]



Let *a* be the number of nodes in AP, and let *b* be the number of nodes in BP:
$a, b > 2$

(20) : $a + 2b + 3$ new c-command and containment relations.
(21) : $2a + 2b + 2$ new c-command and containment relations, which is strictly greater.

Look at (10) again (the optimal format):

(10) **[α [β…[γ [X^0 δ]]…]]**
Where α, β, … are themselves structured according to (8),
and X^0 is a terminal.

The elements entering into the specification of (10) are purely structural (geometric).

But note: insofar as syntax keeps to this optimal pattern, it is natural to pick out X^0 as the unique "head" of (10), and interpret (10) as a "projection" containing other "projections" (α, β, …), each with its own unique terminal/"head".

That is, (10) "looks" like a projection, but no *explicit device* associates non-terminal nodes with their "heads".
→ **Endocentricity** (projection) is an *emergent property* of optimized structure.

Here, (interpretive) function follows (syntactic) form:
"as if syntax carved the path interpretation must blindly follow." (Uriagereka 2002: 64)
(see also Hinzen 2006)

References

Chametzky, Robert. 2000. *Phrase Structure: From GB to Minimalism*. Oxford: Blackwell.

Chomsky, Noam. 1970. Remarks on nominalization. In Roderick Jacobs & Peter Rosenbaum (eds.) *Readings in English Transformational Grammar*, 184-221. Waltham, MA: Ginn and Co.

Chomsky, Noam. 1995a. Bare phrase structure. In Gert Webelhuth (ed.) *Government and Binding Theory and the Minimalist Program*, 385-439. Oxford: Blackwell.

Chomsky, Noam. 1995b. *The Minimalist Program*. Cambridge, MA: MIT Press.

Chomsky, Noam & Morris Halle. 1968. *The Sound Pattern of English*. New York: Harper and Row.

Collins, Chris. 2002. Eliminating labels. In Samuel Epstein & Daniel Seely (eds.) *Derivation and Explanation in the Minimalist Program*, 42-64. Oxford: Blackwell.

Epstein, Samuel D., Erich Groat, Ruriko Kawashima & Hisatsugu Kitahara. 1998. *A Derivational Approach to Syntactic Relations*. Oxford: Oxford University Press.

Hayes, Bruce. 1995. *Metrical Stress Theory*. Chicago: University of Chicago Press.

Hinzen, Wolfram. 2006. *Mind Design and Minimal Syntax*. Oxford: Oxford University Press.

Hornstein, Norbert & Jairo Nunes. 2008. Adjunction, Labeling, And Bare Phrase Structure. *Biolinguistics* 2: 57-86.

Jackendoff, Ray. 1977. *X'-Syntax: A Study of Phrase Structure*. Cambridge, MA: MIT Press.

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

Langendoen, D Terence. 2003. Merge. In Carnie, Andrew, Heidi Harley, and MaryAnn Willie (eds.) *Formal Approaches to Function in Grammar: In Honor of Eloise Jelinek*, 307-318. Amsterdam: John Benjamins.

May, Robert. 1985. *Logical Form*. Cambridge, MA: MIT Press.

Medeiros, David. 2008. Optimal Growth in Phrase Structure. *Biolinguistics* 2:2 156-195.

Moro, Andrea. 2000. *Dynamic Antisymmetry*. (Linguistic Inquiry monographs 38). Cambridge, MA: MIT Press.

Pietroski, Paul. 2004. *Events and Semantic Architecture*. Oxford: Oxford University Press.

Reinhart, Tanya. 1976. The Syntactic Domain for Anaphora. MIT dissertation.

Speas, Margaret. 1990. *Phrase Structure in Natural Language*. Dordrecht: Kluwer.

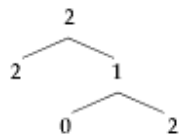
Starke, Michael. 2001. Move dissolves into Merge. University of Geneva dissertation.

Uriagereka, Juan. 2002. *Derivations: Exploring the Dynamics of Syntax*. London: Routledge.

Appendix 1: Comparisons of phrasal formats

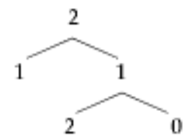
a. 2 → 2 1 ('X-bar')
1 → 0 2

b.



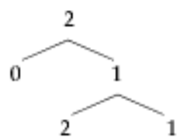
a. 2 → 1 1 ('D-bar')
1 → 2 0

b.



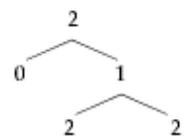
a. 2 → 1 0 ('high-headed X-bar')
1 → 2 1

b.



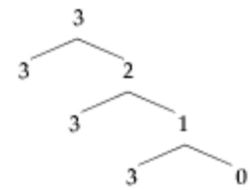
a. 2 → 1 0 ('high-headed D-bar')
1 → 2 2

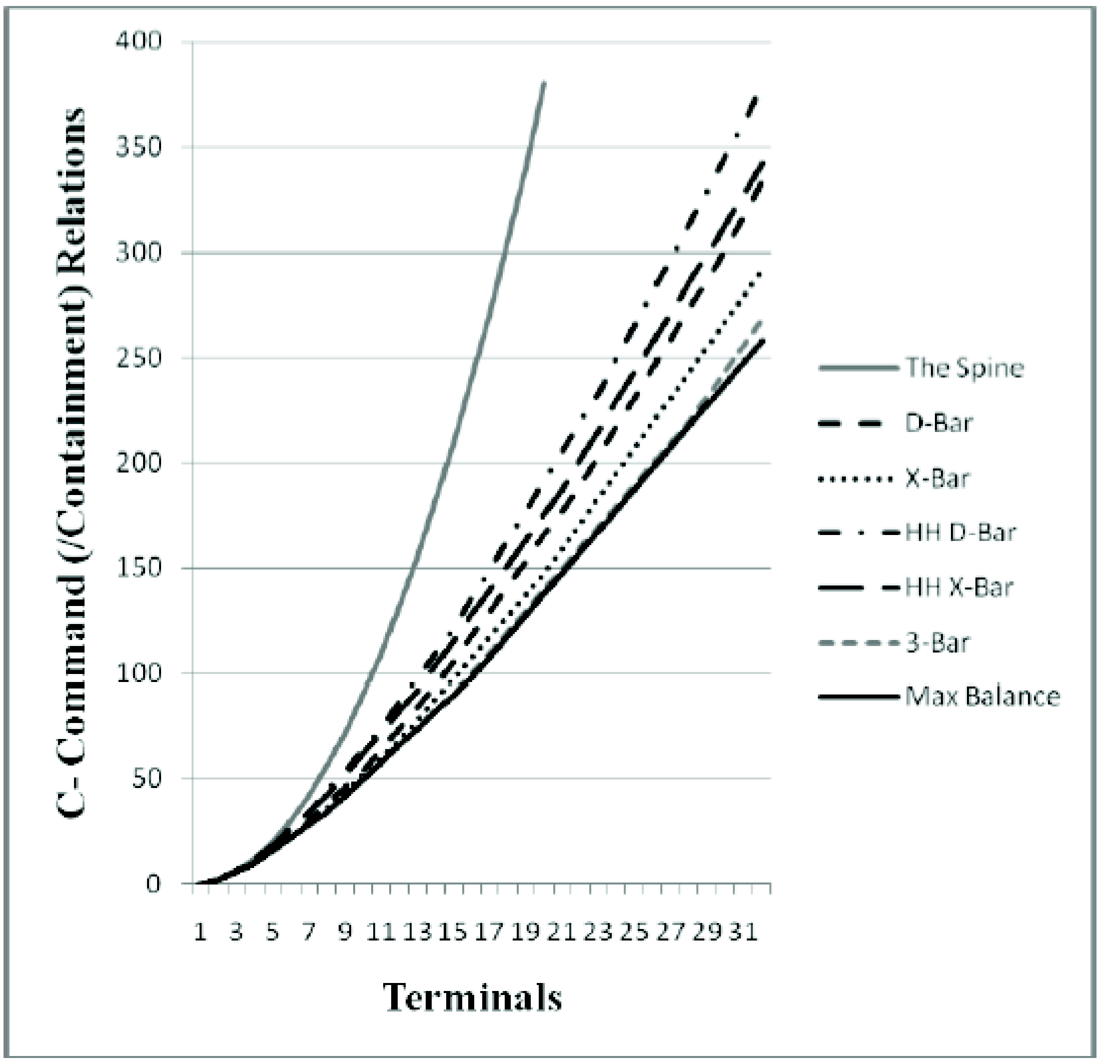
b.



a. 3 → 3 2 ('3-bar')
2 → 3 1
1 → 3 0

b.





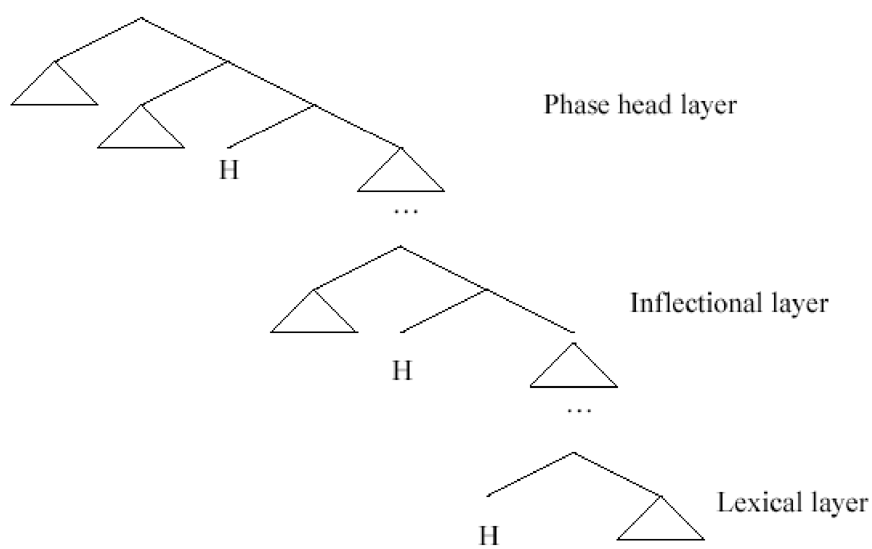
Appendix 2: Mode transitions

The argument above assumes that the syntactic system will tend toward a consistent pattern providing an optimal balance between (i) the cost of accumulating c-command and containment relations, and (ii) the cost of the horizon of information flow in the derivation (a larger, costlier window is required for more intricate phrasal patterning).

But the informational-horizon cost is fixed for a given size of window, while the cost of c-command and containment relations grows for larger trees.

As a result, we might reasonably expect that the optimal balance will shift as trees grow larger. That is, at the earliest stages of the derivation, the small number of c-command and containment relations may only warrant a minimal amount of information flow; as the tree grows, it becomes worthwhile to allow greater amounts of information flow, as this will open up less costly (in terms of c-command and containment relations) options for phrasal organization.

Thus, a basic prediction is that phrasal organization will exhibit transitions to successively more elaborated 'projective' forms as the derivation unfolds: head-complement structures at the most deeply embedded levels, then classical X-bar forms with a single specifier, with multiple specifiers only in the highest portions of the tree.



This seems like a reasonable approximation to the facts of natural language, under current analyses. Phases seem to be organized into three tiers: a lexical layer plausibly involving only head-complement structures, an inflectional layer allowing the introduction of arguments in specifier positions, and a top phasal layer possibly permitting multiple specifiers (e.g., multiple wh-phrases in SpecCP; subjects, shifted objects, and successive-cyclically moving wh-phrases in Spec*v*P).