Multidominance in light of conflicting formalizations of minimalism and syntactic economy

Chomsky introduced merge in the early 1990s as the elementary syntactic operation and defined it as binary set formation of the form $M(a, b) = \{a, b\}$. Multidominance, introduced in the minimalist sense later on (Citko 2005), has been argued to be at odds with Chomskian conception of merge. For example, Collins and Groat (CG, Collins and Groat 2018) write about a multidominant tree: "One issue that comes up right away is that [the structure] is a graph theoretic object. In minimalism, Merge forms sets $\{X, Y\}$" (2). This paper aims to argue against such claims, and argue instead that: **(Claim 1)** There is no meaningful distinction between merge-based sets and graphs or trees (= directed graphs), particularly in the context of labeled merge-based sets. **(Claim 2)** Given that the definition of syntactic "economy" is severely lacking, there are several conceptions of economy on which multidominant derivations *are more economic* than conservative merge-based ones. These points make multidomimance more minimalist than Chomsky's IM/EM.

(1). Multiple papers argue against some "graph-theoretic structures" and in favor of some "set-theoretic ones". However, this is usually done without any whatsoever discussion of which graph theory or which set theory is being considered. For example, CG (3) write that graphs are necessarily "relations (sets of ordered pairs)" and hence are more complicated than merge-produced sets. This is not necessarily the case, and the two are not at all the same. The following objections can be posed to these and similar line of arguments:

(1) A set, as a structure, is necessarily a relation. Straightforwardly, $S = \{a, b\} = \in(a, S) \wedge \in(b, S)$. Note that S is *not* a label in this case. It's also true that Chomsky is incorrect when he claims binary set formation is the simplest operation (over and over: The MP, On Phases, &c.). Negating is the simplest, and conjoining is the simplest two-place. Merge requires relations, which is more complex than either negation or truth-functional conjunction alone.

(2) While it's true that conventionally, $\langle a, b\rangle = Rab$; ordered pairs are not the same as relations/predicates and they are not part of set theory without an additional axiom to specify them. Never did anyone define ordered pairs in syntactic literature[1]. Notably, since node relations are *relations*, they do not inherently require ordered pairs. With a convenient axiomatization of graph theory, graphs can be stated as sets of the form $R(a, b) \wedge R(c, d)$, making them as simple as Merge-produced structures.

(3) Labeled merge-produced sets take the form $\{a, \{a, b\}\}$. In this context, upon adoption of Kuratowski's[2] ordered pair definition, we obtain $\langle a, a\rangle$. Therefore, if labeling is adopted, there's no difference in complexity between graphs-as-ordered-pairs and Merge-sets.

As such, the claim that minimalist/conservative Merge is formally preferable to multidominance does not hold up to any scrutiny.

(2). Economy (local or global) has been stated in the form of multiple conditions on possible structures (like inclusiveness, NTC, &c.), but no rigorous (or any) formal definitions have been given. Once Chomsky's objection concerns accessibility: in $\{a, \{b, c\}\}$, c should not be accessible to form an object $\{c, \{a, \{b, c\}\}\}$. Such arbitrary constraints usually reference putative notions of "economy" and "third-factor principles". We look at the questions from a purely computational standpoint. Either way, in a structure like $\{a, \{c, \{b, a\}\}\}$, there are two a-objects (whether they're occurrences in the sense of Collins and Stabler (2016) or objects proper = lexical items). The lexical items come from a workspace, a set with a numeration of lexical items requisite for the derivation. So, the question appears to be as follows: which

[1] Options would have included $\{\{\{a\}, \{\}\}, \{\{b\}\}\}, \{\{a, 1\}, \{b, 2\}\}, \{a, \{a, b\}\}, \{\{a\}, \{a, b\}\}$, &c.

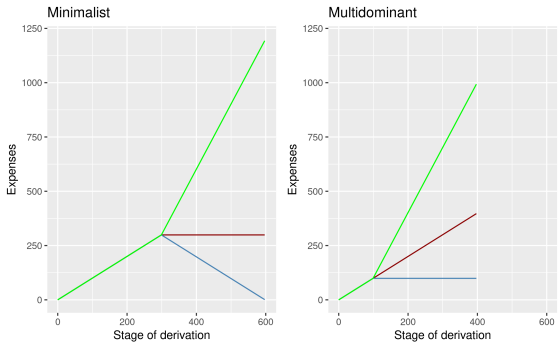
[2] Which is usually adopted implicitly and so without justification, together with consequences like $\langle a, a\rangle = \{\{a\}\}$.

derivation is more costly – (a) the one that accesses the workspace to obtain another occurrence of a lexical item, or (b) the one that makes the previous stages of derivation accessible.[3] To answer the question, a distinction needs to be made between memory resources, processing resources, and accessibility resources. Storing items in workspace requires memory costs; putting/taking items in/from workspace and using them in derivation requires processing costs; being able to access items in previous steps of the derivation requires accessibility costs. We can, however, model these and compare the competing views on costs.

To do so, we'll use python functions – not given immediately in the abstract per space considerations – which both take a list of **l**exical **i**tems and the **num**ber of items which move (as *John* in *John has been seen* t_{John}.) We take a number of assumptions for granted, though:
(4) accessibility is modelled as # of syntactic objects accessible *at any point in derivation*;
(5) costs for pulling an object from lexicon to workspace, from workspace for derivation, and to implement an instance of merge (i.e., $M(a, b) = \{a, b\}$), are the same;
(6) in a multidominant approach, there are no occurrences; all items in the workspace are accessible at any point in derivation.

Below, we present the results of a simulation which calculates procedural, memory, and accessibility expenses. Items accessible at any given point are in blue, memory expense is in red, and processing is in green.

For the minimalist approach, items are accessible in the workspace until they are used in the derivation, after which only one item – the formed sentence – is accessible. Memory expenses is the same after all the items are selected for the workspace.



We should note that there little if any discussion in the syntactic literature of the different types of expenses involved in the process of syntactic derivation, and Chomsky's eclectic appeal to accessibility does not seem to a defensible argument against multidominance so long as no argument for why accessibility is more important than processing or memory is given.

The crucial result that we report on is that once we move to the point where the number of instances of movement (= additional occurrences in the workspace) is greater than the number of syntactic objects, merge is no longer effective. This is clearly seen from the figure in the abstract (100 elements in lexicon, 200 instances of movement in the derivation; other graphs are not included per the space limits). The number of steps in derivation is vastly larger, because it takes longer to pull things from the lexicon. While there might possible ways of remedying that, none are available in the existing formalizations of minimalist syntax. Importantly, conservative merge loses in accessibility as well, because of the sheer number of items in the WS. Yet – while existence of such movement-rich sentences is of question, the computational point stands.

Conclusion. It seems that a lot of debates surrounding economy and conflicts between minimalism and multidominance are in fact nested either in uninformed usage of formal tools or in confusions surrounding the status of "economy". The hope is that this poster will inspire further discussion in the area.

Citko. 2002. On the nature of Merge. Linguistic Inquiry, 36(4):475-496.
Collins and Groat. 2018. Copies and Repetitions.
Collins and Stabler. 2016. A Formalization of Minimalist Syntax. Syntax, 19(1):43-78

[3] Note that this is not necessarily equivalent to tracking the derivation.