BASIC OPERATIONS

Winfried Lechner

1. INTRODUCTION

A grammar of a language L is a structure that generates all and only the well-formed strings in L and assigns these representation a semantic interpretation. Ignoring for the moment the semantic component, such a grammar consists of a set of basic expressions (the lexicon) and a set of operations on these operations that recursively assemble the complex expressions of L. Some of these operations are in turn complex, in that they are built from two or more sub-operations that are also independently operative in the grammar. Movement is a case in point, as it involves dislocation, copying, PF-deletion and possibly further manipulations of the lower copy. By *basic operations*, I will refer to operations that are not composite in this sense, and cannot be further reduced to less complex operations.

In this vignette, I will examine some properties of basic operations, laying special emphasis on the question which consequences a particular choice of basic operations has on the architecture of the grammar. More specifically, it will be argued that certain observations about the correspondence between surface representations and interpretation are incompatible with grammars that assume incremental top-down, left-to-right structure building. From this, it follows first, that the set of basic operations should not include incremental mechanisms for creating syntactic representations, and, second, that the relation between the grammar and parsing needs to be mediated by bridge laws in the sense of Marr (1982).

Section 2 gives an overview of some basic operations standardly used in current Minimalist theories. In sections 3 to 5, I discuss arguments against top-down, left-to-right structure building. Section 6 concludes with some brief remarks on the consequences of these findings for the relation between the grammar and the parser.

2. A SHORT OVERVIEW OF BASIC OPERATIONS

The present section provides a short survey and rough classification of some of the most widely used basic operations. The main objective is not to attain completeness, but to outline the backdrop against which the discussion in sections 3 to 5 is to unfold.

Basic operations fall into discrete classes. In the Minimalist framework, binary *Merge* is the basic mechanism for assembling syntactic representations. Merge comes in various flavors, among them Binary Merge, Pair Merge and Parallel Merge (Chomsky 1995; Citko 2005; i.a.), which mainly differ with respect to the nexus between the two merged terms (head-complement relation vs. modification; Hunter 2015) and the way the compound category is assigned its label (projection vs. adjunction).

Most theories within the Minimalist research program (for exceptions see below) are based on the premise that structure is built *monotonically*. Thus, each application of the basic operation *Merge* (Chomsky 1995) results in a tree that expands previously existing structure, either by standard, “first” merge or movement, which can in turn be interpreted as an instance of “Re-Merge” (Chomsky 2001). Non-monotonic manifestations of Merge are excluded by principles such as the Strict Cycle Condition, which prohibits reaching into already established structure, or explicit filters such as the Extension Condition (Chomsky 1995), which demands all applications of Merge to target the root node. That said, it has also become practice to admit a limited amount of non-monotonicity in the shape of so-called *Late Merge* (LM) or *countercyclic merge*. I will turn to a more detailed discussion of LM and its ramifications in section 4.

Next, there are rules re-organizing syntactic structure in order to make it readable at the interface to semantics and phonology. Prominently among them are mechanisms which render movement configurations interpretable for the semantic component. One widely adopted algorithm for modeling dislocation consists in separating the movement index from its host category at LF, adding a λ-prefix and re-attaching the result to the sister node of the host, as shown in the transition from (1)a to (1)b (von Stechow 1993; Heim and Kratzer 1998; Büring 2005):

(1) a. [$_{CP}$ Who$_1$ [t_1 is here]]?
b. [$_{CP}$ Who [$_{C'}$ λ_1 [t_1 is here]]]?

In course of the semantic interpretation procedure, the object-language symbol 'λ_1' is subsequently translated as a lambda operator that binds a variable in position of the trace, resulting in λ-abstraction (see section 5).

Reconstruction effects indicate that (1) is incomplete in that movement traces also contain descriptive content. In (2)a, Condition A either would have to apply prior to movement, an analytical option incompatible with the assumption that interpretation systematically follows all syntactic operations. The conflict is famously resolved by moving from the representation in (2)a to the one in (2)b, which provides the anaphor with a c-commanding local antecedent (Chomsky 1995):

(2) a. [Which book about ***himself$_1$***]$_2$ did ***he$_1$*** like t_2?
b. [Which book about ***himself$_1$***]$_2$ did ***he$_1$*** like [which book about ***himself$_1$***]$_2$?

Rendering the lower movement copy of (2)b interpretable is delegated to a designated sub-group of operations in this class, jointly referred to as *Trace Conversion* (Fox 2000; Sauerland 1998).

In Minimalist orthodoxy, movement is seen as the result of the binary relation *Agree* between features on the constituent to be dislocated and the goal of movement. The distribution of features in the syntactic representations as well as constraints on licit feature combinations is regulated by a third class of basic operations. Depending on theoretical proclivities, mechanisms that fall into this category include insertion rules that combine features with their hosts; upward, downward or bidirectional Agree (Chomsky 2000; Bjorkman and Zeijlstra 2019, i.a.); feature valuation and feature deletion.

Three remarks are in order here. First, the use of selectional features and hierarchically organized feature structures makes it possible to reduce Merge to feature matching, with a concomitant reduction in the number of basic operations. This strategy has for example been pursued by the formal interpretation of Minimalism known as *Minimalist Grammars* (Stabler 1997, 2010).

Second, while the basic operations above, properly defined, characterize grammars for substantial parts of the well-formed expressions, they do not include designated rules for eliding structure (ellipsis) or the inverse operations of doubling and resumption. This is so because these phenomena are also amenable to feature based analyses (Merchant 1997) and independent mechanisms such as movement. Thus, separate basic operation like deletion are not needed.

Finally, some theories assume that lexical atoms, usually referred to as roots, are combined with category labels and features in the lexicon. Clearly, these lexical rules represent yet another, fourth class of basic operations.

3. Two conceptions of Merge

On standard minimalist assumptions, Merge operates bottom-up and - given the strong empirical tendency for syntactic structure to be right-branching and left-associative (Kayne 1994; Haider 2013) - by assembling terms from right to left. Over the last two decades, a diverging, alternative view has emerged, though, according to which structure is built incrementally from left to right and top-down (Phillips 1996, 2003; Richards 2004; Chesi 2015, 2016; i.a.). Support for this alternative conception mainly comes from two sets of observations. First, a large body of psycholinguistic evidence has accumulated over the last decades indicating that “both language comprehension and language production seem to be incremental in surprising respects” (Stabler 2014: 259). Second, top-down structure building has been claimed to afford broader empirical coverage than the conservative bottom-up versions of Merge in the analysis of a number of phenomena, among them movement and ellipsis (Philips 2003; Richards 2004). To maximize synergies between arguments from competence grammar and psycholinguistic evidence, it has moreover been suggested to remove the separation between the grammar and the parser, resulting in a radical departure from the tradition of asserting the autonomy of grammar from performance and the slogan “The grammar is the parser” (Phillips 2003; Phillips and Lewis 2013).

Adopting left-to-right structure building clearly effects the decision of what is to be included in the set of basic operations. In what follows, I will briefly present such a system that afforded important insights and generated interesting new questions, followed by two case studies indicating that incremental left-to-right theories at the moment still fail to provide answers to problems that do not arise for a conservative conception of Merge. As a corollary, these findings suggest that abandoning the distinction between the grammar and the parser is, at least at the time of writing, premature.[1]

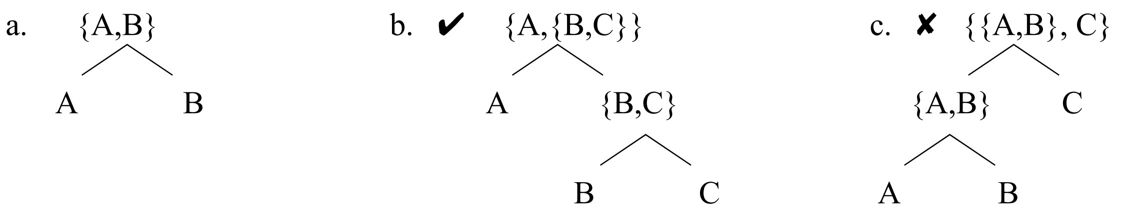
Phillips (1996) proposes that syntactic structure is incrementally built by the complex operation *Incremental Merger*, informally defined by the two rules in (3). Incremental Merger assembles nodes from left to right and attaches new terms low, and not high, as in orthodox theories of Merge.

(3) ***Incremental Merger***
   a. Merge Right: New terms are introduced at the right edge of a structure.
   b. Branch Right: Merge as low as possible.

The algorithm is seen at work in (4). Once the unit A and B has been formed in (4)a, adding a further term C creates the representation (4)b, in which the constituency of the previously assembled group AB is broken up. C forms now a unit with B to the exclusion of A. Thus, the grammar does, at least in the absence of intervening factors, not generate left-branching trees such as (4)c:

[1] Further empirical arguments against left-to-right structure building can be found in Lechner (2002, 2003). For a reply see Phillips and Lewis (2013).

(4)



Crucially, while the two nodes A and B form a constituent in (4)a, they no longer do so in the final representation (4)b. This is a general property of Incremental Merger, which will become relevant in the discussion to follow: left-to-right structure building re-organizes previously established constituency.

There are at least three arguments against incremental structure building of the type above, two of which I will attend to in turn: Late Merge, variable binding and phrase structure paradoxes. The first two arguments are, as far as I know, new. The third one is the topic of Lechner (2002, 2003).

4. Late Merge and incremental structure building

The first obstacle for the IM-analysis is posed by its incompatibility with the standard analysis of binding theoretic opacity effects that arise in certain movement constellations. More specifically, Ā-movement creates opaque contexts in which the application of interpretive principles, among them Condition A and Condition C of Binding Theory is not visible in the output (Freidin 1986; Johnson 1987; Lebeaux 1990; van Riemsdijk and Williams 1981; i.a.). To exemplify, the anaphor in (2)a, repeated below as (5)a, finds a c-commanding antecedent only after reconstruction. Reconstruction is also instrumental to account for the disjoint reference effect in (5)b.

(5) a. [Which book about ***himself***$_1$]$_2$ did ***he***$_1$ like best t_2?
b. *[Which picture of ***John***$_1$]$_2$ did ***he***$_1$ like best t_2?

Both cases are successfully handled by the Copy Theory of movement which posits additional occurrences of the reflexive and the name, respectively, in the lower trace position (details suppressed). Binding Theory evaluates then these lower copies:

(6) a. [Which book about himself$_1$]$_2$ did ***he***$_1$ like best [which book about ***himself***$_1$]?
b. *[Which picture of John$_1$]$_2$ did ***he***$_1$ like best [which picture of ***John***$_1$]?

If the name is embedded within an adjunct, though, instead of an argument-like *of*-PP ((6)b), the disjoint reference effect disappears (Lebeaux 1990):

(7) [Which picture <u>near</u> ***John***$_1$]$_2$ did ***he***$_1$ like best t_2?

Such instances of Principle C obviation require the introduction of additional machinery. The standard minimalist account, due to Lebeaux (1988), operates on the assumption that insertion of adjuncts can be delayed by a (new) basic operation known as *Late Merge* (LM; see also McCawley 1968; Sauerland 1998; Fox 2000; Takahashi and Hulsey 2009; Stanton 2016; Abe 2018; i.a.). On this analysis, parts of which are made explicit in (8), only a fragment of the fronted NP moves, and the adjunct PP is merged countercyclically once the host has reached a position outside the c-command domain of the pronoun:

(8) a. Move host: [Which picture] did he_1 like best [which picture]?
b. Late Merge of adjunct: [Which picture [near ***John$_1$***]]$_2$ did ***he$_1$*** like best t_2?

By restricting LM to adjuncts, a corresponding countercyclic derivation is blocked for (6)b. LM derivations have, with different motivation, also been proposed for the integration of comparative complements in structures such as *Mary is smarter than John is* (Bhatt and Pancheva 2004), restrictions on preposition stranding (Stanton 2016) and argument ellipsis (Abe 2018).

What is of particular relevance for present purposes is the observation that the LM mechanism is, as far as I can see, incompatible with a system that adopts an incremental structure building operation such as Incremental Merge. Consider a left-to-right derivation of (7). On the IM-analysis, symbols are parsed in their linear order, starting from the left edge. Accordingly, *which picture* is assembled first ((9)a), and the result is then combined with the PP *near John* ((9)b).[2] Once the rest of the tree has been merged ((9)c), a copy must be inserted into the object position of *like*. Moreover, given that copying is subject to syntactic parallelism, the conclusion is inescapable that the lower copy must match its antecedent, yielding something akin to (9)d. But clearly, this representation wrongly leads one to expect a disjoint reference effect. Note that merging only the host NP *which picture* into the thematic position is not an option, because such a move would violate parallelism.

(9) a. Which picture
b. Which picture near John
c. Which picture near John did he like
d. [Which picture near John] did he like [which picture near John]

A possible response consists in assuming that an even smaller constituent, the head noun *picture*, is inserted into the postverbal position:

(10) [Which [$_{NP}$ [$_{NP}$ picture] [$_{PP}$ near John]]] did he like [$_{NP}$ picture]

This derivation abides now by parallelism. Furthermore, the lower copy does no longer contain the offending name, also a welcome result. But a new complication arises at this point, since the antecedent in (10) does not c-command the lower copy in violation of the c-command requirement on movement. Once again, this problem might not have to be fatal for the IM approach given that proponents of orthodox LM also need to appeal to an exemption from a fundamental structural constraint, viz. the monotonicity of Merge. By definition, LM does not operate on the root node, but submerges adjuncts into the specifier of the root. Thus, LM violates the general principle dictating that newly merged items c-command all other constituents of the tree. One might therefore posit that selected violations of the c-command requirement on movement in a left-to-right grammar correspond to failure to abide by the Extension Condition in standard systems. Both theories need to be granted some degree of liberty in the treatment of anti-reconstruction opacity.

As it turns out, though, even this proviso, shady to begin with, fails to provide a full analysis of (7). To begin with, as the more complete rendering of the derivation in (11) reveals, limiting

---

[2] I am ignoring here the question why the resulting constituency is *[[which picture][near John]]* instead of *[which [picture [near [John]]]];* see Philips (1996) for discussion.

the copy mechanism to the head noun has the detrimental effect of rendering the structure ineligible for semantic interpretation by commonly sanctioned rules:

(11) [Which $[_{NP}$ $[_{NP}$ picture] $[_{PP}$ near John]]]$_2$ did he like $[_{NP}$ picture]

What is missing in (11) is the index 2 on the lower copy, which is needed in order to introduce into the logical form a variable to be bound by the antecedent. But this index is part of the whole DP *which picture near John*, and not just the head noun *picture*. Thus, abstraction applies vacuously and the sentence cannot be mapped to the desired meaning.

Second, it can be shown that there are constellations in which the lower copy contains more than the head noun. In (12), it is possible to construe the pronoun *his* as a variable bound by *every artist* while the lower subject *she* is simultaneously coreferential with *Mary*:

(12) Which of his$_2$ paintings on Mary$_1$'s wall did every artist$_2$ think that she$_1$ would like?

In order to derive this reading, the lower copy has to embed *his painting* without also including the PP *on Mary's wall,* as sketched in (13):

(13) Which of $[_{DP}$ his$_2$ $[_{NP}$ $[_{NP}$ paintings] on ***Mary$_1$***'s wall]] did ***every artist$_2$*** think that ***she$_1$*** would like which of ***his$_2$*** paining?

However, if constituency is calculated from left to right, there is no way for the representation to both fulfill the requirement above on the shape of the lower copy and to satisfy movement parallelism. On the assumption that the lower occurrence matches the full antecedent *his painting on Mary's wall*, the lower copy has to include *on Mary's wall* and the absence of a disjoint reference effect remains unaccounted for. If, alternatively, the lower copy consists of the bare head noun *painting* only, the variable remains unbound. Consequently, (13) is not a legitimate representation for (12).

More generally, it is hard to conceive of any analytical strategy that would both fall within the bounds laid down by the Incremental Merger theory and have the capacity to account for anti-reconstruction effects. Left-to-right structure building appears to be inherently incompatible with Late Merge. Note in the end that the problem is not limited to Principle C obviation, but also affects Late Merge accounts of other constructions (Bhatt and Pancheva 2004; Takahashi and Hulsey 2009; Stanton 2016; Abe 2018).

5. Variable Binding

A second, qualitatively different challenge for left-to-right approaches towards structure building comes from variable binding. On an influential strand of analysis (Heim and Kratzer 1998), variable binding is modeled by the rule of Predicate Abstraction (14), which substitutes the index of the binder - more precisely, the object language symbol λ prefixed to the index; see (1) - with a metalanguage lambda operator, resulting in lambda abstraction over the coindexed variable.[3]

(14) ***Predicate Abstraction (PA)***
For any binder index $n \in \mathbb{N}$ and assignment function g: $[\![\alpha\ \lambda_n]\!]^g = \lambda x.[\![\alpha]\!]^{g[n \rightarrow x]}$

In the sample derivation (15), the variable in the trace position is bound by the metalanguage

[3]For a fully compositional treatment of binding and quantification see Bennet (1979).

lambda operator subsequent to the application of Predicate Abstraction in (15)c:

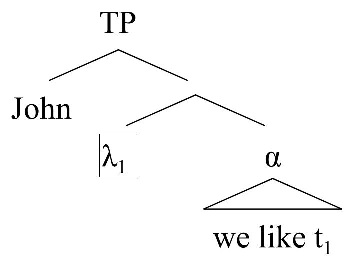
(15)
a. John_1, $[_{TP}$ we like $t_1]$
b. LF: John λ_1 $[_{TP}$ we like $t_1]$
c. $[\![\lambda_1 [_{TP} \text{ we like } t_1]]\!]^g = \lambda x.[\![[_{TP} \text{ we like } t_1]]\!]^{g[1\to x]} =$ (PA)
d. $= \lambda x.[\![\text{like}]\!]^{g[1\to x]} ([\![t_1]\!]^{g[1\to x]}) [\![\text{we}]\!]^{g[1\to x]} =$
e. $= \lambda x.[\![\text{like}]\!] (g^{[1\to x]}(1)) (\text{we}) =$
f. $= \lambda x.\text{we like } x$

In order to facilitate binding, the structure containing the variable must be assembled *before* the abstraction step. That is, Predicate Abstraction is triggered by contexts that contain both the variable and the binder index.

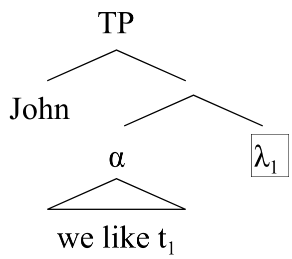
Arguably, the guiding idea underlying left-to-right structure building, epitomized by the slogan “The grammar is the parser”, should also extend to interpretation, such that meaning is, just like syntactic structure, computed incrementally. Assuming this to be the case, variable binding constitutes an obvious problem, because the translation procedure will target the binder index first, and encounter the variable bound by the metalanguage translation of this index only later. Clearly, this strategy will not succeed in assigning (15) the desired interpretation as a derived predicate.[4]

There is a simple fix for this problem. Instead of merging the λ-binder to the left ((16)a), abstraction can in an incremental system be modeled by attaching the λ-binder to the right, as in the *late abstraction* representation (16)b:

(16) a. *Standard analysis*



b. *Late abstraction*



The semantic derivation then proceeds just as in regular bottom-up systems in that the semantic value of the category (reflexively) containing the trace is computed first, followed by late abstraction.[5]

(17) $[\![[_{TP} \text{ we like } t_1]\ \lambda_1]\!]^g = \lambda x.[\![[_{TP} \text{ we like } t_1]]\!]^{g[1\to x]}$ (PA)
$= \lambda x.\text{we like } x$

Even though promising at first sight, it can be show that shifting the attachment site of the λ-binder has arguably unwelcome consequences, two of which will be attended to in the remainder

[4]In Shan and Barker (2006), binding is analyzed in a system that evaluates expressions incrementally. This move comes at the cost of abandoning the notion of c-command, though.

[5]Late abstraction only affords a way for restoring the correct order between the λ-binder and the trace. An additional assumption is needed in order to ensure that the semantic computation of (16)b starts with the subject *we* instead of the moved category *John*. One (admittedly stipulative) option might be to store *John* in a stack from where it can be retrieved at a later point.

of this chapter.

Observe to begin with that irrespective of whether the set-up in (16)a or (16)b is chosen, the trace is formally related to its λ-binder only, while the relation between the antecedent and the trace is indirect and only mediated by function application. Furthermore, in the standard system (16)a, the antecedent is adjacent to the λ-binder. It follows that the distance between the antecedent and the trace co-varies with the path length between the λ-binder and the trace. By contrast, in the late abstraction representation (16)b, the λ-binder is parsed into the tree-final position. (Recall that this is so because the λ-binder needs to be computed last). But unlike in the orthodox model (16)a, the distance between the antecedent and the trace is *inversely* correlated with the distance between the λ-binder and the trace: the closer the trace is to the antecedent, the further away it is from the λ-binder, and *vice versa*.

The key observation is now that this particular property of the late abstraction system conflicts with a solidly established psycholinguistic insight according to which longer movement paths, measured from the *overt* antecedent to the trace, incur higher processing costs. A classic instance of this generalization is manifest in the experimental finding that subject relative clauses (SRC) are easier to process than object relatives (ORC; Graf et al 2017; Yatsushiro 2017; i.a.):

(18) *Standard analysis: SRC < ORC* (from Graf et al. 2017)
a. The reporter [who $\boldsymbol{\lambda_1}$ $\boldsymbol{t_1}$ attacked the senator] admitted the error. (SRC)
b. The reporter [who $\boldsymbol{\lambda_1}$ the senator attacked $\boldsymbol{t_1}$] admitted the error. (ORC)

Consider at this point the derivation on the late abstraction regime, relevant parts of which are made explicit in (19):

(19) *Late abstraction: ORC < SRC*
a. The reporter [who $\boldsymbol{t_1}$ attacked the senator] $\boxed{\lambda_1}$ admitted the error. (SRC)
b. The reporter [who the senator attacked $\boldsymbol{t_1}$] $\boxed{\lambda_1}$ admitted the error. (ORC)

Unlike on the standard view, the linear distance between the trace and its binder is now greater in subject relatives than it is in the object variant. The path lengths are effectively reversed. Thus, the left-to-right system fails to model a fundamental design feature of the parser - cognitive plausibility - which motivated the account in the first place.

Moreover, right-attachment of the binder also has consequences for configurations of multiple movement, in that it leads to a reversal of the relation between linear order and path structure. On standard assumptions, order preserving movement as in (20)a creates crossing paths, while the nested configuration (20)b is the signature of movements that invert the original order of the constituents. If, by contrast, the binder index is right adjoined, the order reversing constellation is parsed into a crossing dependency ((20)c) and nested paths correlate with order preservation ((20)d).

(20)
a. [α $\boxed{\lambda_1}$... [β $\underline{\underline{\lambda_2}}$ ... [... $\boxed{t_1}$ $\underline{\underline{t_2}}$ ...]]]] (crossing and order preserving)
b. [α $\boxed{\lambda_1}$... [β $\underline{\underline{\lambda_2}}$ ... [... $\underline{\underline{t_2}}$ $\boxed{t_1}$ ...]]]] (nested and order reversing)
c. [α ... [β ... [... $\underline{\underline{t_2}}$ $\boxed{t_1}$ ...]] $\underline{\underline{\lambda_2}}$] $\boxed{\lambda_1}$] (crossing and order reversing)
d. [α ... [β ... [... $\boxed{t_1}$ $\underline{\underline{t_2}}$ ...]] $\underline{\underline{\lambda_2}}$] $\boxed{\lambda_1}$] (nested and order preserving)

Thus, one should find crossing paths where orthodox analyses posit nested dependencies and *vice versa*. To exemplify, the well-formedness of (21)a, which is usually taken to be a reflex of nested movement paths (Chomsky 1977; Pesetsky 1982; i.a.), would have to be explained by recourse to representation (22)a, which implicates crossing paths on the right-adjunction account. Conversely, the illicit constellation (21)b, commonly thought to be the excluded because the paths are crossing, would on such an alternative conception have to be linked to the nested relations in (22)b.

(21) *Standard analysis*

a. Which violins $\boxed{\lambda_1}$ are these sonatas easy $\underline{\underline{\lambda_2}}$ to play $\underline{t_2}$ on $\boxed{t_1}$ (nested)

b. *Which sonatas $\boxed{\lambda_1}$ are these violins easy $\underline{\underline{\lambda_2}}$ to play $\boxed{t_1}$ on $\underline{t_2}$ (crossing)

(22) *Late abstraction*

a. Which violins [are these sonatas [$_{AP}$ easy to play $\underline{t_2}$ on $\boxed{t_1}$ $\underline{\underline{\lambda_2}}$] $\boxed{\lambda_1}$] (crossing)

b. *Which sonatas [are these violins [$_{AP}$ easy to play $\boxed{t_1}$ on $\underline{t_2}$ $\underline{\underline{\lambda_2}}$] $\boxed{\lambda_1}$] (nested)

It might of course be possible to re-define the laws regulating multiple movements such that they reflect the changes described above. But the generalizations derived from differences in path structures are arguably grounded in independent, deeper principles such as syntactic locality conditions or the cycle, and it is far from obvious how these explanations are to be replicated in the alternative late abstraction system.

In sum, it seems fair to conclude that at the moment, theories that build structure and compute meaning incrementally still owe an account of variable binding, at least if the analysis is to be compatible with the principles gained from the study of other components of the grammar such as parsing efficiency and the path containment condition.

6. Summary

In this vignette, I set out to define two arguments against the view that syntactic and semantic representations are assembled incrementally, from left-to-right. On the one side, it was argued that Incremental Merger and the mechanism of Late Merge are mutually exclusive. The inherent incompatibility of these two operations was exemplified by anti-reconstruction effects in section 4 above, but the result are more general and extend to the analysis of a range of unrelated phenomena (Bhatt and Pancheva 2004; Stanton 2016; Abe 2018). In the absence of a standardly accepted alternative to Late Merge (but see Graf 2014; Sportiche 2019), this suggests that the set of basic operations does not include a left-to-right version of merge.

On the other side, the observation that variable binding cannot be accommodated in a model that computes meanings incrementally was seen to pose a substantial challenge to theories in which not only natural language syntax, but also the semantic component is organized in a left-to-right manner. This was of relevance to the discussion of left-to-right computations of linguistic representations because the motto “The grammar is the parser”, which underlies the concept of incremental merger, is most naturally understood as a general statement about the architecture of the grammar, generating the expectation that incrementality is not confined to the formal, syntactic part of the grammar.

If these findings can be further substantiated, they also have broader, ontological ramifications for the relation between competence and performance. More precisely, the conclusion appears at least at the moment ineluctable that the grammar should be isolated from the principles characterizing the way language is processed. Thus, these two levels of analysis still need to be connected by bridge laws, which link the cognitive mechanisms of linguistic competence to their actual physical manifestations (Nagel 1961; del Pinal & Nathan 2013; Nathan & del Pinal 2016; Phillips and Lewis 2013, i.a.).

Bibliography

Abe, Jun. 2018. Late Merge and Phases for Anti-C-Command Requirements. *Syntax* **21**.2: 91-111.

Bennett, Michael. 1979. *Questions in Montague Grammar.* Indiana University Linguistics Club.

Bhatt, Rajesh, and Pancheva, Roumyana. 2004. Late Merger of Degree Clauses. *Linguistic Inquiry* **35**. 1: 1-45.

Bjorkman, Brownyn and Hedde Zeijlstra. 2019. Checking Up on (Phi-)Agree. *Linguistic Inquiry* **50**.3: 527-569.

Chesi, Christopher. 2015. On directionality of phrase structure building. *Journal of Psycholinguistic Research* **44**.1: 1-50.

Chesi, Christopher (ed.). 2016. Studies in linguistics. Special Issue on "Directionality of Phrase Structure Building". Siena: CISCL.

Chomsky, Noam, 1977. On Wh-Movement. In: P. Culicover, T. Wasow and A. Akmajian (eds.), *Formal Syntax*. New York, New York: Academic Press, pp. 71-132.

Chomsky, Noam. 1995. *The Minimalist Program*: Current Studies in Linguistics. Cambridge, Massachusetts: MIT Press.

Chomsky, Noam, 2000. Minimalist Inquiries: The framework. In: Roger Martin, David Michaels and Juan Uriagereka (eds.), *Step by Step: Essays on minimalist syntax in honor of Howard Lasnik.* Cambride, Mass: MIT Press, pp. 89-155.

Chomsky, Noam. 2001. Beyond Explanatory Adequacy. In *MIT Occasional Papers in Linguistics 20.* Cambridge: MITWIPL.

Citko, Barbara. 2005. On the Nature of Merge: External Merge, Internal Merge, and Parallel Merge. *Linguistic Inquiry* **36**.4: 475–496.

del Pinal, Guillermo and Marco J. Nathan. 2013. There and Up Again: On the Uses and Misuses of Neuroimaging for Psychology. *Cognitive Neuropsychology,* **30**.4: 233-252.

Fox, Danny. 1999. Reconstruction, Variable Binding and the Interpretation of Chains. *Linguistic Inquiry* **30**.1: 157-196.

Freidin, Robert. 1986. Fundamental issues in the theory of binding. In: Barbara Lust (ed.), *Studies in the Acquisition of Anaphora,* Volume I, 151–188. Dordrecht: Reidel.

Graf, Thomas. 2014. Late Merge as lowering movement in Minimalist grammars. In LACL 2014, ed. by Nicholas Asher and Sergei Soloviev, 107-121. Heidelberg: Springer

Graf, Thomas, James Monette, and Chong Zhang. 2017. Relative clauses as a benchmark for Minimalist parsing. *Journal of Language Modelling* **5**.1: 57-106.

Haider, Hubert. 2013. *Symmetry breaking in syntax.* Cambridge: Cambridge University Press.

Hunter, Tim. 2015. Deconstructing Merge and Move to Make Room for Adjunction. *Syntax* **18**.3: 266-319

Johnson, Kyle. 1987. Against the Notion SUBJECT. *Linguistic Inquiry* **18**.2: 354–361.

Kayne, Richard. 1994. *The antisymmetry of syntax.* Cambridge, Massachusetts: MIT-Press

Lebeaux, David. 1988. Language Acquisition and the Form of the Grammar. Doctoral Dissertation, University of Massachusetts, Amherst.

Lebeaux, David. 1990. Relative clauses, licensing, and the nature of the derivation. In: Juli Carter, Rose-Marie Dechaine, Bill Philip, and Tim Sherer (eds.) *Proceedings of North East Linguistic Society 20,* 318–332. University of Massachusetts, Amherst: GLSA.

Lechner, Winfried. 2002. Deriving Phrase Structure Paradoxes by Conditions on Merge. In *Proceedings of NELS 32*, Amherst, Mass: GLSA, pp. 287-306.

Lechner, Winfried. 2003. Phrase Structure Paradoxes, Movement and Ellipsis. In Kersting Schwabe and Susann Winkler (eds.), *The Interfaces*. Berlin and New York: Mouton de Gruyter.

Lewis, Shevaun and Phillips, Colin. 2015. Aligning grammatical theories and language processing models. *Journal of Psycholinguistic Research* **44**.1: 27-46.

McCawley, James D. 1968. Lexical Insertion in a Transformational Grammar without Deep Structure. In *Proceedins of the 4th Chicago Linguistics Society.*

Marr, David. 1982. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. New York: W.H.Freeman & Co Ltd.

Nagel, Ernst. 1961. *The Structure of Science.* New York: Harcourt, Brace and World.

Nathan, Marco J, and Guillermo Del Pinal. 2016. Mapping the Mind: Bridge Laws and the Psycho-Neural Interface. *Synthese* **193**.2: 637–657.

Pesetsky, David. 1982. Paths and Categories. Doctoral Dissertation, Massachusetts Institute of Technology.
Phillips, Colin. 1996. *Order and Structure.* Doctoral Dissertation, MIT.
Phillips, Colin. 2003. Linear Order and Constituency. *Linguistic Inquiry* **34**.1: 37-90.
Phillips, Colin and Shevau Lewis. 2013. Derivational Order in Syntax: Evidence and Architectural Consequences. *STIL – Studies in Linguistics* **6**: 11-47.
Richards, Norvin. 2004. Against bans on lowering. *Linguistic Inquiry* **35**.3: 453-463.
van Riemsdijk, Henk and Edwin Williams. 1986. Introduction to the Theory of Grammar. Cambridge, Massachusetts: MIT Press.
Sauerland, Uli. 1998. The Meaning of Chains. Doctoral Dissertation, MIT.
Shan, Chung-Chieh and Chris Barker. 2006. Explaining Crossover and Superiority as Left-to-right Evaluation. *Linguistics & Philosophy* **29**.1: 91-134.
Sportiche, Dominique. 2019. Somber Prospects for Late Merge. *Linguistic Inquiry* **50**.2: 416-424.
Stabler, Edward P. 1997. Derivational minimalism. In *Logical aspects of computational linguistics*, ed. Christian Retoré, volume 1328 of *Lecture Notes in Computer Science*, 68–95. Berlin: Springer.
Stabler, Edward P. 2010. After Government and Binding Theory. In *Handbook of Logic and Language (2nd ed.)*, ed. Johan F.A.K. van Benthem and Alice ter Meulen. Elsevier. pp. 395–414.
Stabler, Edward. 2013. Two Models of Minimalist, Incremental Syntactic Analysis. *Topics in Cognitive Science* **5**: 611-633.
Stanton, Juliet. 2016. Wholesale late merger in Ā-movement: Evidence from preposition stranding. *Linguistic Inquiry* **47**.1: 89-126
Takahashi, Shoichi, and Sarah Hulsey. 2009. Wholesale Late Merger: Beyond the A/Ā-Distinction. *Linguistic Inquiry* **40**.3: 387-426.
Yatsushiro, Kazuko. 2017. Relative clause acquisition in German: the effect of case and agreement. Talk presented at CIALT 1, Rethymon, Crete, October 6-8, 2017.