

## **On directionality of phrase structure building**

Cristiano Chesi (NETS, IUSS Pavia)

### **Abstract**

Minimalism, in grammatical theorizing (Chomsky 1995) led to simpler linguistic devices and a better focalization of the core properties of the structure-building engine: a lexicon and a free (recursive) phrase formation operation, dubbed Merge, are the basic components that serve in building syntactic structures. Here I suggest that looking at the elementary restrictions that apply to free Merge (i.e. selection and licensing of functional features), we could conclude that a re-orientation of the syntactic derivation (from bottom-up/right-left to top-down/left-right) is necessary to make the theory simpler, especially for long-distance (filler-gap) dependencies, and empirically more adequate: if our structure building operations would roughly assemble lexical items in the order they were pronounced (Phillips 1996, Chesi 2004-2012 a.o.), on-line performance data could better fit the grammatical model, without resorting to external “performance factors”. The phase-based, top-down (and, as consequence, left-right) Minimalist Grammar here discussed goes in this direction, ultimately showing how Strong Islands (Huang 1982) and intervention effects (Gordon et al. 2001) could be better explained in structural terms assuming this unconventional derivational direction.

## 1. Introduction

In this paper I provide formal and empirical arguments in favor of a derivational view of the grammar in which structure building takes place incrementally top-down (Chesi 2004, 2007) and, as a consequence, from left to right (Phillips 1996, 2003).

Following the spirit of the Minimalist Program (Chomsky 1995, 2008), I will show that the bottom-up orientation ('bottom-to-top' in the sense of Phillips 1996) of phrase structure building is not a "conceptual necessity" (§2). In fact, both descriptive adequacy and processing transparency can be gained by moving away from the idea that phrases are created by the recursive application of operations like bottom-up Merge (§2.1). In a nutshell, because of the empirical restrictions that apply to Merge, I propose reversing the tree-structure building procedure: If phrases are expanded (in a way that is reminiscent to traditional phrase structure grammar derivations, §3.1) rather than being created by bottom-up Merge, Chomsky's notion of "phase" can be interpreted as the minimal domain in which a given set of syntactic features must be sequentially computed (either lexicalized or further expanded). One of the main assumptions here is that tree growth is driven by local lexical selection. When a verb selects an argument, for instance, it will trigger prediction of the features associated with that required argument (e.g. all necessary features that make such an argument a full DP, like a functional feature D, for determiner, and a lexical feature N, for nominal). This will introduce into the syntactic tree the relevant features (e.g.  $[_{VP} \dots \text{buy} [_{DP} D N]]$ , i.e. a full DP endowed with "D" and "N") that must then be sequentially lexicalized or further expanded (e.g. first  $[_{DP} [_D \text{the}] [_N ]]$ , then  $[_{DP} [_D \text{the}] [_N \text{apple}]]$ ). The feature insertion procedure will be computationally costless in the case of the final selection triggered by a certain lexical head, which completes the satisfaction of selectional

requirements of that head. However, an expansion can be computationally more demanding if the features to be inserted are not the last expected ones. For example, in “Adam gave a book [that had been bought by his wife [who visited the bookshop yesterday]] to Barbara” the feature expansions extending the direct object with a relative clause will occur while the matrix verb is still incomplete, because another selection requirement (i.e. the indirect object) remains to be expanded (§3.3). This is a source of computational complexity, roughly expressible in terms of the number of mandatory features that are yet to be computed.

I will also argue that Movement (creating a long-distance ‘filler-gap’ dependency, Fodor 1978, §2.2) is triggered by what I will call ‘unexpected’ features that are introduced into the tree (§3.2). Such features enter the derivation when they are part of a feature bundle (a complex lexical item or a phrase) which contains one or more other features in addition to the ones that are required (selected) at that location. In English, for instance, a *wh*-element like “who” (as opposed to “why” and “when”, see §3.2 for discussion) that is inserted in the left periphery of a question, to lexicalize an interrogative feature (*wh*), also has argument features (it qualifies as a determined, D, nominal, N, phrase, i.e. a DP), which are not licensed in that peripheral *wh*-position (since no predicate selects them there). This prevents the unselected feature bundle from being interpreted properly in this position, and forces the derivation to keep these unexpected features ‘on hold’ until a point at which a proper selector is found for it (e.g. a verb that requires a DP as its argument). For a Top-Down, Left-Right implementation, I propose using a simple Memory Buffer, reminiscent of the HOLD register proposed by Wanner and Maratsos (1978) (but see §3.2 for relevant differences), where the unexpected feature bundles are stored until a relevant

selected (thematic) position is found. By regulating the access and transmission mechanisms of the memory buffer in terms of phases, we succeed in capturing islandhood (Chesi 2004) and parasitic gap constructions (Bianchi & Chesi 2006) (§4.2). It is important to stress that the grammatical formalism discussed here does not provide a processing account of these phenomena — I am not proposing that “the grammar is the parser” (Phillips, 1996). Rather, I show how a formal grammatical model that embodies top-down directionality is empirically more adequate in unifying a set of facts that are otherwise only mysteriously related. From the linguistic perspective, then, I will suggest that the proposed grammatical model (§3) is explanatorily more adequate than the standard Minimalist model (§2). Specifically, parasitic gaps are not exceptional constructions but follow from the assumption of top-down movement. From the processing perspective (§4), I will argue that this phase-driven, memory-based, filler-gap association mechanism affords a better characterization of observed processing difficulties due to feature confusions (Gordon et al. 2001, McElree et al. 2003, a.o.) which stem from locality constraints (Relativized Minimality, Rizzi 1990), as for instance in the comprehension of subject-gap vs. object-gap relative clauses (Tavakolian 1981, Warren & Gibson 2002, 2005, a.o.), without resorting to the assumption of specific performance constraints.

## **2. Virtual conceptual necessities**

A common assumption within the Minimalist framework (Chomsky 1995) is that the grammar is an effective bridge between sounds and meanings. In this respect, any syntactic structure-building procedure should interface with both, and comply with constraints that ensure the creation of structures which are legible objects for these

“external” systems. In this sense, the Sensory-Motor (S-M) and Conceptual-Intentional (C-I) interfaces are conceptual necessities (Chomsky 1995:169). Unfortunately, apart from a few constraints that these external modules should comply with, very little has been said about what requirements they should meet. The only two generally accepted assumptions related to the interfaces are that:

1. only features that are “interpretable” should reach the Conceptual-Intentional interface (Chomsky 1995:277). This amounts to saying that the interface system should discard any abstract feature used for creating syntactic objects which has no semantic value (e.g. case features).
2. syntactic objects that are shipped to the S-M systems should be linearizable (Fox & Pesetsky 2005).

Another conceptual necessity is that (some of) the structure building operations should include a recursive component, since they need to combine a finite set of lexical items in potentially infinite sentences. In the next sections I will discuss how these necessities are encoded in the structure building procedure that applies locally (§2.1) and non-locally (§2.2).

### **2.1. Creating phrase structures through recursive Merge**

Since Chomsky’s 1995 seminal work, much effort has been expended on correctly characterizing the minimal requirements of the simplest possible recursive structure building procedure. ‘Merge’ (Chomsky 1995:243, 2013:42) implements such a recursive mechanism simply by means of an unconstrained, binary, set formation

operation that combines two adjacent elements like the lexical items “the” and “book” in (4)<sup>1</sup>:

$$(4) \text{ Merge}(\textit{the}, \textit{book}) = \{ \textit{the}, \textit{book} \}$$

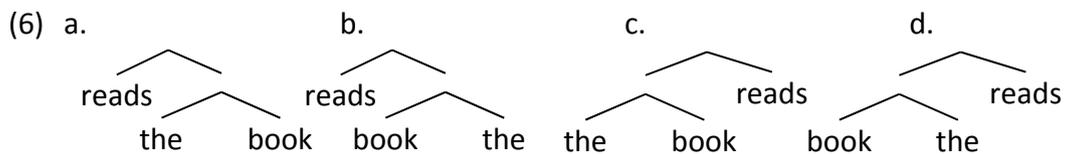
Recursion results from the fact that Merge does not only take lexical items as arguments, but also the results of previous Merge operations:

$$(5) \text{ a. Merge}(\textit{reads}, \{ \textit{the}, \textit{book} \}) = \{ \textit{reads} \{ \textit{the}, \textit{book} \} \}$$

$$\text{ b. Merge}(\textit{John}, \{ \textit{reads} \{ \textit{the}, \textit{book} \} \}) = \{ \textit{John} \{ \textit{reads} \{ \textit{the}, \textit{book} \} \} \}$$

Assuming that the subset relations resulting from Merge express dominance relations,

(5.a) can be associated with any of the tree-structures shown in (6).



This is because minimalist Merge does not entail any linear order among its arguments. Thus Merge is more minimal than a traditional rewriting rule (such as DP → D N) which imposes both structure and linear order, since linearization is not part of Merge but it is an external requirement imposed by the S-M interface and evaluated after the hierarchical syntactic structure has been created (Fox & Pesetsky 2005).

One implicitly accepted departure from this minimal conception of Merge is to limit the freedom of Merge by means of selection<sup>2</sup>:

<sup>1</sup> In the examples below, Merge is a function taking two arguments (inside round brackets); curly brackets indicate unordered sets (i.e. the result of the Merge operation). Squared brackets will be used for ordered constituents [D N] (labeled, when necessary [<sub>DP</sub> D N]) and ordered sets of features within a lexical item L: [<sub>F1 F2 F3 ...</sub> L].

<sup>2</sup> The recent discussion on ‘labeling’ seems to me to go exactly in this direction (Cecchetto & Donati 2010): the ‘label’ (i.e. the interpretable result of Merge, accessible to other Merge operations) is the ‘probe’, i.e. the merged item that selects the other.

(7) Selection restriction on Merge

- i.  $\alpha$  merges with  $\beta$  successfully iff either  $\alpha$  selects  $\beta$  or  $\beta$  selects  $\alpha$ ;
- ii. the item that selects, also projects (i.e. if  $\alpha$  selects  $\beta$ , then  $[\alpha \ \beta]$ )

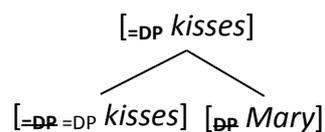
Stabler (1997), for instance, uses this restriction to limit Chomsky's (1995) Merge.

Similarly, Collins (2002) proposes that Merge itself is triggered by saturation considerations (i.e. the unsaturated verb selects, then merges with, its complement).

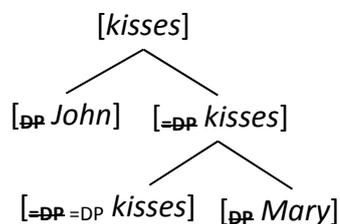
Following Stabler's formalism, I will prefix selecting features with the "=" sign<sup>3</sup>. As a result of the Merge operation, the selecting head will be promoted, and the selection feature that is now satisfied will be deleted; the constituent so created will consist of the two merging items, dominated by the selecting one, pruned of the features that triggered the successful Merge operation:

- (8) Result of two Merge operations (active features triggering Merge are struck through in these diagrams):

- i.  $\text{merge}([\text{=DP} \text{ kisses}], [\text{DP} \text{ Mary}]) = [\text{=DP} \text{ kisses}]$



- ii.  $\text{Merge}([\text{=DP} \text{ kisses}], [\text{DP} \text{ John}]) = \text{kisses}$



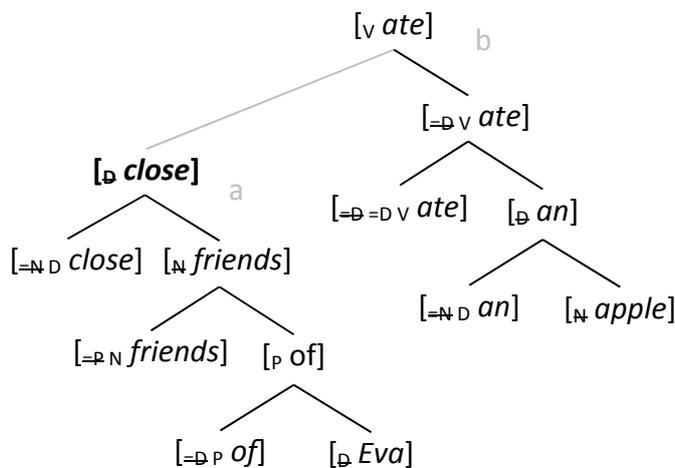
---

<sup>3</sup> Note that the "=" sign does not denote equivalence when employed in this way; "=X" means that X is selected.

According to Stabler, the right branching nature of natural languages (Kayne 1994) can be captured by stipulating that ‘complex’ constituents (i.e. items that result from previous Merge operations) always lie on the right branch of the tree<sup>4</sup>. Any left-branching constituent must be assembled before being merged into the main tree. This is the case for ‘complex’ subjects like “close friends of Eva” in the sentence “[close friends of Eva] eat an apple”:

- (9) Lexicon: {[<sub>=P</sub> N *friends*], [<sub>N</sub> *apple*], [<sub>D</sub> *Eva*], [<sub>=N</sub> D *close*], [<sub>=N</sub> D *an*], [<sub>=D</sub> P *of*], [<sub>=D</sub> =D V *ate*]}

- a. Derivation of the complex subject:
- i. Merge ([<sub>=D</sub> P *of*], [<sub>D</sub> *Eva*]) = [<sub>P</sub> *of*]
  - ii. Merge ([<sub>=P</sub> N *friends*], [<sub>P</sub> *of*]) = [<sub>N</sub> *friends*]
  - iii. Merge ([<sub>=N</sub> D *close*], [<sub>N</sub> *friends*]) = [<sub>D</sub> ***close***]
- b. Derivation of the main tree:
- iv. Merge ([<sub>=N</sub> D *an*], [<sub>N</sub> *apple*]) = [<sub>D</sub> *an*]
  - v. Merge ([<sub>=D</sub> =D V *ate*], [<sub>D</sub> *an*]) = [<sub>=D</sub> V *ate*]
  - vi. Merge ([<sub>D</sub> ***the***], [<sub>=D</sub> V *ate*]) = [<sub>V</sub> *ate*]



<sup>4</sup> As I will discuss in §2.2, all other possible word orders should result from Movement (i.e. displacement of certain items to obtain a configuration able to remove features that are uninterpretable at the C-I interface, as discussed in §2).

It is proposed by Nunes & Uriagereka (2000) that a left-branch constituent, once composed, behaves like a single lexical item, explaining why it can appear in the left-hand position and also why it is an “island” (Ross 1967) for movement, as shown in (10).

(10) \**Who* did [close friends of \_] eat an apple?

Nothing can be extracted from the frozen subtree (“close friends of *who*”) when it is merged in the subject position, because its internal constituents can no longer be accessed (see §4.2).

Though formally satisfactory in many ways, it should already be clear that the tree-building procedure driven by Merge is not attractive from a processing perspective.

Merge builds phrase structure from bottom to top, which in right-branching constructions entails building from right to left. This strongly contrasts with processing operations such as sentence parsing and production, which proceed sequentially, word after word (in this sense “from left to right”), in the opposite sequence from what Merge theory predicts. A formal theory more compatible with processing models will be proposed in §3.

## 2.2. Non local dependencies

Natural languages exhibit relations among items within a tree structure which are not local, such as the relation between the verb “buy” and its object “what” below:

(11) “What<sub>i</sub> will John buy \_<sub>i</sub>?”

The related elements need to be connected using devices other than Merge, since, as discussed in §2.1, Merge only combines elements that are strictly adjacent. Minimalist scholars assume that such specific non-local relations are established via ‘Movement’:

“what” is first merged next to “buy” as direct object, then, once the complementizer phrase is merged higher up in the structure, “what” is copied and re-merged next to the complementizer, then deleted from its base position. This operation is often dubbed ‘internal Merge’ to highlight the fact that a subcomponent of Movement is exactly a Merge operation (which however does not pick up a new item from the lexicon, but takes an item already used in the phrase structure built so far). The original minimalist explanation for Movement is that it creates the correct local configuration between the relevant complementizer position and the moved wh-constituent that, in such configuration, can get its ‘uninterpretable’ wh- feature removed by the ‘interpretable’ wh- feature of the complementizer (Chomsky 1995:280).

Despite different possible formalizations of such operation, it was clear since early studies (Ross 1967) that Movement should be restricted somehow to correctly exclude instances of displacement that are not allowed in any natural language, e.g. island (e.g. (10), §4.2) and locality (§4.3) constraints. There are two possible approaches to such restrictions:

- i. representational approach – after the phrase structure is completed it is evaluated and if it violates any constraint(s) the derivation is discarded;
- ii. derivational approach – during an incremental derivation, any illegal step that violates a constraint is avoided or else leads the derivation to crash.

The derivational perspective is more appealing from a computational point of view, since it is more economical (no generation of impossible structures) and more targeted (fewer derivational alternatives are considered). Within the Minimalist program, the derivational approach is pursued: using the ‘probe-goal’ idea to rephrase Movement

Chomsky (2001) suggests that once the left-peripheral position is created, some feature(s) there, e.g. *+wh*, function as a probe, triggering an active search within the tree, down to the first (hierarchically speaking) accessible target, a goal with the feature *-wh* that will be moved higher, next to the probe, subject to whatever constraints are applicable. I will limit discussion here to certain long distance dependencies (A'-dependencies such as *wh*-movement), which hold between elements in two major positions. The structurally higher one, which I will call the 'Criterial' position, following Rizzi (2006), that involves discourse-related features like topic and focus. And the lower position, which is a thematic position (an argument selected by a predicate)<sup>5</sup>. This second one is the position where the item was first merged in the derivation and it is often referred to as the 'trace' (in Minimalist theory, an unpronounced copy) of the higher element.

Contrary to standard processing accounts of filler-gap dependencies (Fodor 1978, De Vincenzi 1991), the probe-goal mechanism predicts that the filler (probe) and the gap (goal) are introduced in the derivation in the inverse order with respect to the order in which they are parsed and pronounced: first the goal is merged in the position that will later become the gap position, then, some steps later, the probe (the criterial item that will trigger movement) will be merged higher up in the structure, and it will require exploring the tree built so far, searching for the goal that will be moved up next to the probe.

Without losing important generalizations captured by the bottom-up derivational approach, I want to bring syntactic derivations into closer harmony with processing

---

<sup>5</sup> Such relation between the criterial position and the thematic one is not present in all *wh*-dependencies. As discussed in Rizzi (1999) and Ko (2005), "why", e.g., is a *wh*-operator item generated in a left-peripheral position and not involved in any movement operation of the kind discussed here.

concerns. In particular, in §4, I will consider how effectively the grammatical constraints on long-distance dependencies interact with the processing routines (Gordon et al. 2001, 2004, Warren & Gibson 2005). Driving this research with a Minimalist habit (nothing but what is strictly necessary should be considered as part of the grammar) I will try to show that a grammatical theory that succeeds in formulating a filler-gap dependency in a processing-friendly sequential order is descriptively, but also explanatorily, more adequate.

### **3. A processing-friendly, empirically adequate, minimalist grammar**

Within a general minimalist framework, some proposals have been made for reformulating the structure building operations in a way that would make the grammar more processing friendly (Weinberg 2001) but also more descriptively adequate (e.g. explaining conflicts among constituency tests, Phillips 1996, 2003). Here I want to follow this thread, showing that while there is no need to conclude that “the parser is the grammar” (in the sense of Phillips 1996:255), nevertheless a top-down oriented derivation, which creates phrase structure from left to right, is compatible with the minimalist mode of inquiry, and is more processing friendly than the bottom-up approach that is currently standard in minimalist analyses.

In the remainder of this paper, I will continue to use a simplified version of Stabler’s formalism to express features associated with lexical items as illustrated in (12).

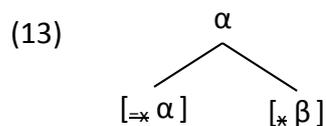
(12)      [<sub>+T</sub> =<sub>DP</sub> v *ate*]

An ordered set of features is associated in (12) to the word “*ate*” (past tense of “eat”, “*ate*” will be used as place-holder for phonetic/semantic features). Features prefixed

by “+” are functional features (e.g. +T expresses a tense specification); features prefixed with “=” express selection requirements that must be satisfied locally (e.g. “=DP” forces Merge with a “DP” argument, as in (8)); features without a prefix are lexical categorial features (e.g. “V” indicates that the item carrying this feature is a verbal element).

### 3.1 Expecting features and lexical items

As discussed in §2.1, the major restriction on Merge is lexical selection:  $\alpha$  will successfully merge with  $\beta$  if and only if  $\alpha$  and  $\beta$  are minimally of the form  $[\text{=}_X \dots \alpha]$  and  $[\text{}_X \dots \beta]$ , where  $X$  is the most prominent<sup>6</sup> set of features both of  $\alpha$  and  $\beta$  (Stabler 1997, Collins 2002 a.o.). The  $X$  element satisfies the selectional requirement  $=X$ , as shown with strikethrough in (13).



With a bottom-up perspective,  $\alpha$  and  $\beta$  cannot be ordered without making some extra assumption as discussed in §2.1. On the other hand, from a top-down perspective,  $\alpha$  logically “precedes”  $\beta$ , since  $\alpha$  selects  $\beta$  and the item that selects also projects (i.e. the selector  $[\text{=}_x \alpha]$  dominates the selectee  $[\text{}_x \beta]$  as illustrated in (13).

In this regard, the top-down perspective is more restrictive than the bottom-up one, since only the former predicts a possible order for such  $\alpha$  and  $\beta$  items, given their select and base features.

Assuming that phrase structure building is an incremental procedure, we should conclude that  $\alpha$  must be introduced in the derivation first, then  $\beta$  can be licensed by

---

<sup>6</sup> Remember that features associated to a lexical item are ordered; being prominent means being the leftmost feature in the lexical feature list.

selection. I suggest interpreting selection by  $\alpha$  as an *expectation* for  $\beta$ . That is, once the select features of  $\alpha$  are computed, a minimal set of features complying with those selection needs are expected to be integrated right after  $\alpha$ . This idea forces a radical departure from standard minimalist assumptions: since select features in the lexicon are active, in the sense they trigger a continuation that must be satisfied locally before continuing the computation, Merge reduces to local selection satisfaction. Thus linear order is mainly determined by features triggering new lexical insertions and there is no need to assume that this is part of an external system (the Sensory-Motor one)<sup>7</sup>.

To see how the derivation unrolls following this intuition, let us use first the lexical items in (8) and let us apply to them this top-down procedure<sup>8</sup>:

(14) Top-down derivation for items  $[_{=DP=DP} \textit{kisses}]$ ,  $[_{DP} \textit{John}]$  and  $[_{DP} \textit{Mary}]$ :

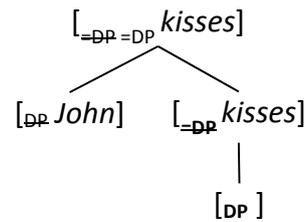
- |   |   |
|---|---|
| i. merge/insert <i>kisses</i> in the derivation   | $[_{=DP=DP} \textit{kisses}]$                                 |
| ii. read the first select feature (=DP) and project the expectation for a compatible item         | $[_{=DP=DP} \textit{kisses}]$<br> <br>$[_{DP}]$               |
| iii. merge/insert a compatible item satisfying the projected expectation, $[_{DP} \textit{John}]$ | $[_{=DP=DP} \textit{kisses}]$<br> <br>$[_{DP} \textit{John}]$ |

---

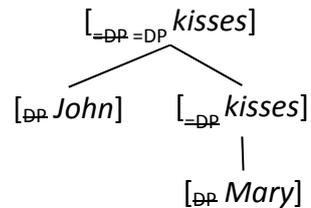
<sup>7</sup> The null hypothesis (“being pronounced as soon as merged”) is obviously too strong. Morpho-phonetic constraints should be taken into consideration: for instance the same constraints that are considered to be responsible for (parameterized) V-to-I movement (Pollock 1989) should justify the inflected verb position. Another source of cross-linguistic variation (resulting in a difference of linear order) can be related to different lexical selection: as suggested by Choi and Yoon (2006) in languages like Japanese, case-marked DPs select (i.e. creates expectations for) specific verb types, leading to head-final order (the last source of variation in linear order is obviously movement, §3.2).

<sup>8</sup> The diagram below shows the derivation history. Only relevant features are considered (=DP and DP). Notice that multiple “copies” of the verbal head “kisses” are present in the diagram, but only one of them will be pronounced. While the selection mechanism is assumed to be universal, the option to spell-out a specific copy could be parameterized cross-linguistically.

iv. as long as *kisses* has other select features (=DP), keep it prominent (i.e. to the right edge of the structure<sup>9</sup>), read the first available select feature and project the expectation for a compatible item (i.e. repeat ii.)



v. merge/insert a compatible item satisfying the projected expectation,  $[_{DP} \textit{Mary}]$  (i.e. repeat iii.)



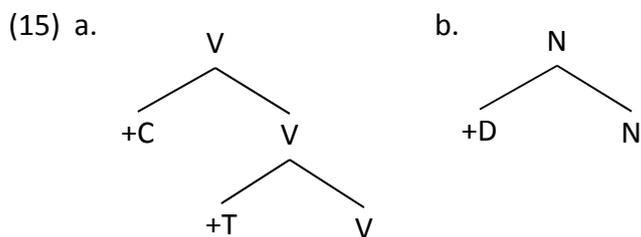
The result of the complete satisfaction of the selection requirements will produce a cascade of argument-dedicated projections<sup>10</sup> (cf. the VP-shells of Larson 1988); that is, a collection of layers each dominated by a lexical head with one relevant select feature responsible for the related expectation. As for standard bottom-up Merge, the top-down Merge/Insertion operation can be thought of as binary, since it applies a new lexical item to the result of previous merges, and creates sets (i.e. hierarchies), more precisely it expands feature subsets by refining expectations feature by feature, according to the chosen lexical items. The sequential growth (“left-right”), postulated by Phillips 1996 (‘Right Merge’), is a consequence of the logical constraints of the selection mechanism: we Merge what is compatible with our expectations, and expectations come from what we have already merged.

<sup>9</sup> This is not a stipulation, but an empirical point: expansions “to the right” will be lexicalized (and eventually pronounced) later than what has already been lexicalized (and pronounced).

<sup>10</sup> It would be interesting to derive the argument selection order from independent principles, instead of postulating it. Exploring this option is however beyond the scope of this article.

### 3.2 Moving unexpected features

A derivation simply driven by local selection, as discussed so far, would derive only right branching structures, of the kind sketched in (14). To account for different word orders (see also footnote 7) we need an implementation of a ‘displacement’ operation of the kind discussed in §2.2 and a way to integrate non-selected constituents (like true adjuncts and relative clauses, for instance). To do so, following Grimshaw (1991), I will call the collection of functional heads which canonically dominate a lexical category (like V and N) ‘extended projections’: in this sense, from now on, I will refer to CPs and DPs as extended projections headed by one verb and one noun respectively. Here I assume that functional features are neither properly selected nor proper selectors of lexical heads but simply licensed by the extended projection of the relevant lexical category, as in (15).a,b (cf. Pollard & Sag 1994):



In formal terms, such ‘extended projections’ are ordered lists of functional features associated to the lexical head. I also assume that feature clusters like the ones characterizing declarative finite sentences (i.e. CPs of the form [<sub>V</sub> +C +T V]) as in (15).a), or DPs, i.e. [<sub>N</sub> +D N] as in (15).b) are part of our grammatical competence and can be used to expand an expectation<sup>11</sup>. To see how this is integrated in the derivation in (14), let us explore English interrogative structures<sup>12</sup>: I propose that the derivation starts by

---

<sup>11</sup> Within the Tree Adjoining Grammar framework (Joshi 1985) similar, unlexicalized, feature structures are dubbed ‘initial trees’.

<sup>12</sup> Here I will only discuss argumental wh-items like *who*, *what* and *which...*. Adverbial wh- phrases (like “why”, “how” etc.) introduce an operator that takes scope from the peripheral position where such

projecting an expectation compatible with a question, that is, for instance, a verbal extended projection ( $[_V \dots V]$ ) with an overt specification of a +wh functional feature, a tensed specification (+T)<sup>13</sup> and, eventually, a topic specification distinct from the +wh one, which I will call +S since related to subject (criterial) position:<sup>14</sup>  $[_V \text{ +wh +T + S V}]$ . Since features are computed in sequence, wh- items like “what” must be generated at the beginning of the sentence and linked to the relevant thematic position of “buy” by movement.

(16)        What<sub>i</sub> will John buy  $_{-i}$ ?

According to the sketched derivation in (14), the subject that is pronounced before the selecting verb to satisfy the +S subject-related, cross-linguistically parameterized functional feature (e.g. topic or focus in SVO languages), presupposes a movement operation as well. In conclusion, all the Criterial positions are computed before the thematic ones (terminology introduced in §2.2). This is compatible with what is

---

elements are first merge (see footnote 5); crucially they differ from “thematic” wh-items since they do not introduce any unexpected feature, hence they will not be moved in the memory buffer. Similarly, wh-in situ languages like Chinese and Japanese (Cheng 2003) introduce an operator (either empty or morphologically realized) in the periphery of the phrase that must be questioned. This operator binds a variable introduced with the wh-in-situ item. Such dependency is subject to standard C-command restrictions that, from a top-down perspective, are expressible looking at nesting (Cheng 2003 for a summary of non-movement based proposal and for different typologies of wh-in-situ).

<sup>13</sup> We can safely assume that semantic or formal constraints are associated to the features of the lexical item: e.g. a wh- complementizer can be endowed with the semantic necessity, which must be satisfied locally, to be followed by a tensed specification compatible with the extended projection of V. In this sense the order of functional items could be semantically driven (Cinque 1999, Ramchand & Svenonius 2013).

<sup>14</sup> This surely requires cross-linguistic parameterization: for instance the subject criterial position, signaled by +S feature, in English, which is assumed to be usually a focalized position (Huck & Na 1990), is absent in (17), because of the focalized wh- item; as result of the absence of +S, the wh item is also interpreted as topic of the predication (and it will be interpreted as the subject of the predication in the relevant selected position). Since the preverbal subject position in other languages (like Italian) is not focalized but topicalized (Calabrese 1982), this position will be always lexicalized in these languages (though a null subject can be used).

assumed in the processing literature, at least for filler-first, filler-gap constructions such as these (Fodor 1978)<sup>15</sup>.

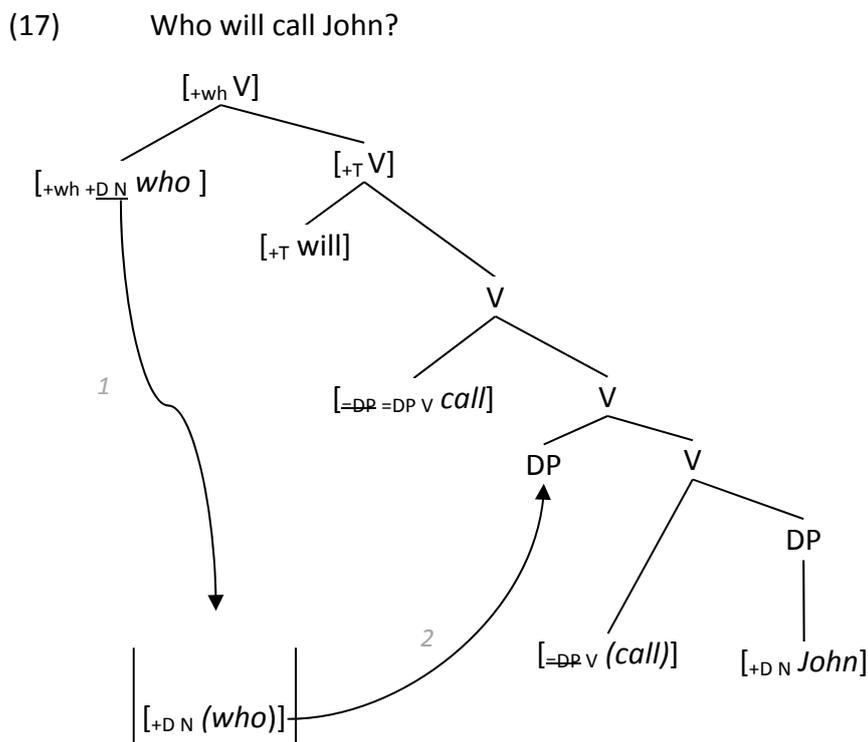
The formalization of this idea hinges on the fact that merging an item into the sentence to lexicalize a particular feature can result in bringing into the computation other features that are “unexpected”, where “unexpected” has a precise technical meaning: those features are not selected at the position in question. Elements like “who” [<sub>+wh +D N</sub> *who*] or “what” [<sub>+wh +D N</sub> *what*] (i.e. extended nominal projections endowed with +wh and +D functional features) can be chosen from the lexicon because of their [+wh] feature<sup>16</sup>, but they will also introduce into the structure a cluster of extra features [+D N] that are unselected in the criterial position, which licenses only the [+wh] feature. I propose that these unselected (in this sense “unexpected”) features force a copy of the item to be put “on hold” in memory while the computation keeps introducing further lexical items as result of the lexicalization procedure of the remaining expected features. This is what creates the “filler” status of the item. During the phrase structure building, when (and only when) a lexical item is added to the tree which bears a selection compatible with the unexpected features of the item on hold, the filler in memory will be retrieved and re-merged into the derivation at the position at which they are properly selected. Optimally, only unused features are put on hold, namely the unexpected ones. In this sense the element ‘on hold’ is not an exact copy of the one first merged (pace Chomsky 1995:202). Even

---

<sup>15</sup> Several kinds of gap-first constructions will not be discussed here.

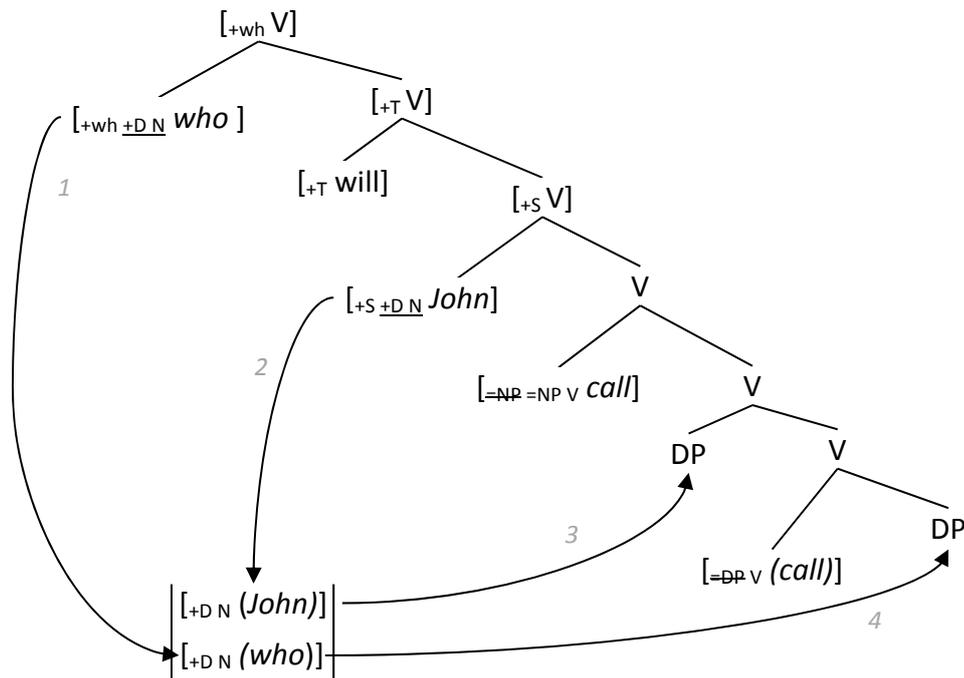
<sup>16</sup> Also an unlexicalized feature cluster (cf. ‘initial trees’, Joshi 1985, footnote 11) like [+wh +D N] can be used to expand the +wh feature expected. This can result in lexicalizations of this kind: [<sub>+wh +D</sub> *which*] [<sub>N</sub> *boy*]. Three constraints apply in this case: first, such expansion will introduce extra feature ([+D N]) that force movement as for the insertion of single wh- words like “what” or “who”; second, such expansion must be properly lexicalized; third, expansions are not recursive: \*[+wh[+wh[+wh[...] +D N] +D N] +D N] (contrary to ‘auxiliary trees’, Joshi 1985, cf. Chesi 2013).

though empirical evidence might suggest, *prima facie*, that this strongest hypothesis should be somehow weakened, pursuing it has some interesting consequences in terms of economy of computation: this explains why in such filler-gap construction only the highest “copy” of the item is usually pronounced since the phonetic features associated to such items are already used in the first (criterial) merge position<sup>17</sup>. As result of a successful re-merge, the item used will be deleted from the memory buffer. A simple implementation of this “Move” operation requires a Last In First Out (LIFO, or stack) memory buffer (as in Push-Down automata, Hopcroft et al. 2001), i.e. the last inserted item in the buffer will be the first one to be retrieved and remerged. The LIFO structure is enough to account for the difference in the phrase structures (17) and (18):



<sup>17</sup> This is again an option that could be parameterized to explain ‘partial copies’ spell-out (Felser 2004). As for *wh*-in-situ languages, the structure of the derivation does not change, only lexical items do, as discussed in footnote 12.

(18) Who will John call?<sup>18</sup>



In both examples, the derivation proceeds similarly:

- i. a top-down expectation for an interrogative phrase is generated, i.e. a tensed (+T) extended V projection with +wh feature expressed and, possibly, +S in case the wh- item is not also the topic of the matrix predication;
- ii. the first feature [+wh] is computed and a compatible wh-item is retrieved and merged, in this case ([+wh +D\_N who]);
- iii. because of the unexpected features [+D N], a copy of “who” equipped with the unselected features [+D N (who)] is moved into the memory buffer<sup>19</sup>. In the same

<sup>18</sup> As discussed before, +S is a criterial feature related to the surface subject position: in SV languages, this licenses the insertion of the subject in the preverbal position. This feature can be interpreted as a case marker (which is evident, in English, only on first and third person pronouns).

<sup>19</sup> As suggested before, only items that are not used or fully interpreted in a given position are moved in the memory buffer. “(who)” indicates that the item remerged in the argument position will be silent (since phonetic features are already used in the criterial position where it is first introduced). I will not discuss here semantic features, but it is fair to assume (Saurleand 2004) that a crucial part of the lexical content of a DP must be interpreted in the selected argument position and not in the criterial position.

- way, later in (18), [<sub>+S +D N</sub> John] is first merged in the [+S]-related position, and then moved into the buffer because of its unselected [+D N] features);
- iv. other features are lexicalized (first [+T], then the [+S] subject case-related position, as discussed);
  - v. then the verbal head “call” is merged and two expectations are projected (both DPs, i.e. expectations compatible with the [<sub>N +D N</sub>] feature set);

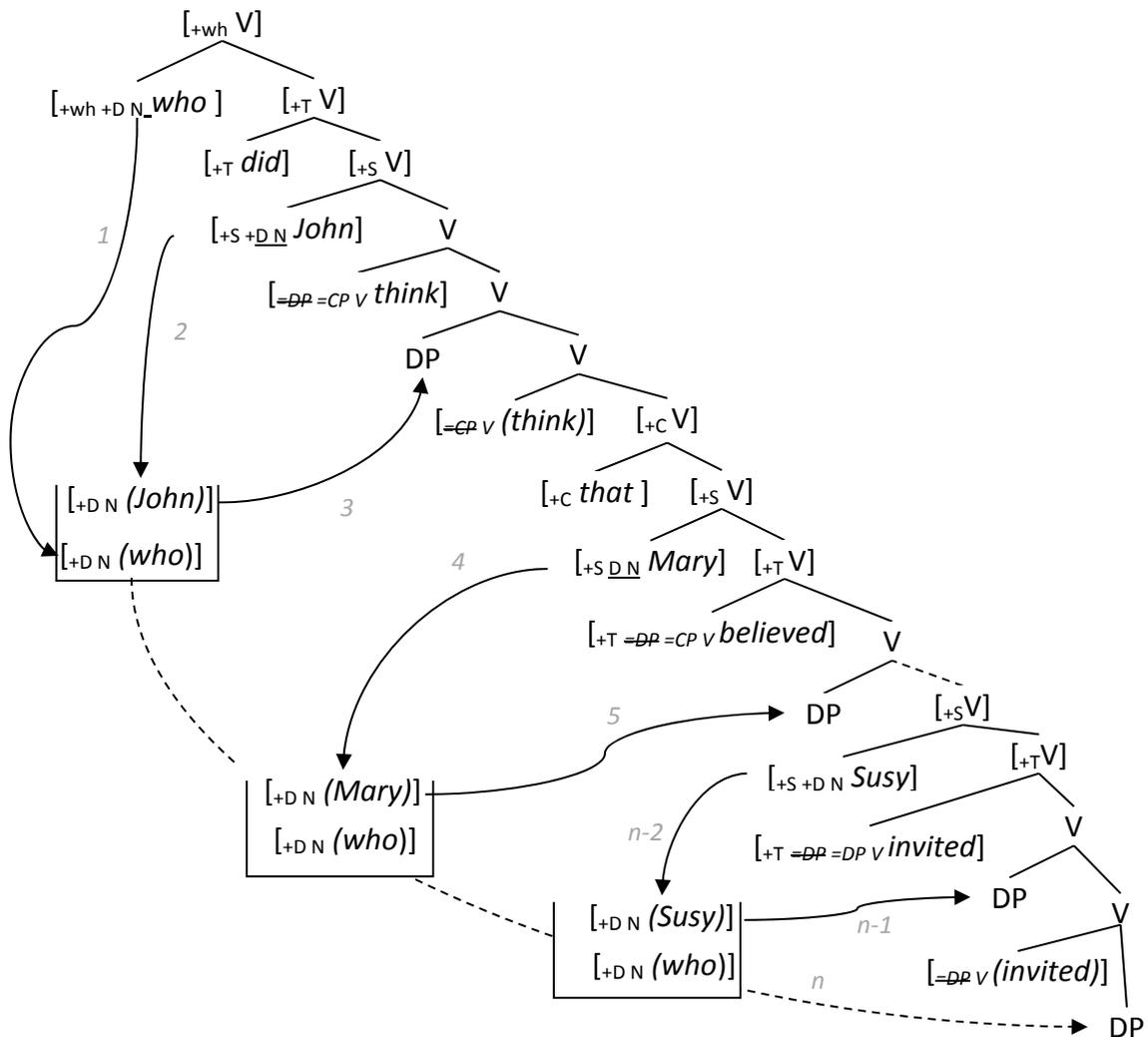
At any point in the parse where there are selection requirements to be fulfilled, a compatible element in the memory buffer prevents the derivation from pulling an extra item from the lexicon to satisfy it<sup>20</sup>. In this way the satisfaction of a pending dependency always preempts the introduction of new items from the lexicon (cf. Active Filler Strategy, Frazier and Clifton, 1989:95 discussed in §4.1). Using the memory buffer could be enough to account for the observed difference in processing costs of subject vs. object extraction: in the latter case, two DPs are stored in the buffer, while only the subject is stored in (18) (cr. §4.3).

This grammatical mechanism will also readily allow for long-distance object wh-dependencies as exemplified in (19) without great effort.

---

<sup>20</sup> An anonymous reviewer noticed that this might conflict with resumption, i.e. “last resort” strategies that replace an unlicensed gap with a resumptive pronoun rescuing an otherwise ill formed sentence. Even though this topic is out of the scope of this article, it is interesting to notice that, on the line of Shlonsky (1992), we can assume that “gaps” (or “traces” in standard minimalist terms) and “resumptive pronouns” are in a complementary distribution and that the gap vs. pronoun alternation depends on the edge-properties of the nested constituent. For instance, in Palestinian Arabic, the relative clause complementizer comes in two morphologically distinct varieties (corresponding to the “aN” vs “aL” split in Irish discussed by McCloskey 1990): one complementizer is responsible for instantiating an operator that should bind the resumptive pronoun that acts as a variable at logical form; the other, in the current framework, would request for a copy of the accessible content of the memory buffer and this copy will be used to fill a trace at arbitrary distance within such relative clauses. Notice that resumption is not readily explainable in a bottom-up approach, where the trace/pronoun alternation must be resolved well before the insertion of the licensing complementizer; since the choice between the gap and the pronoun is dependent from the choice of the complementizer, such problem dissolves in a system, like the one I propose here, where the complementizer is selected before the rest of the relative clause (see Boeckx 2003 for a minimalist attempt to account for resumption, which however does not solve the problem of the logical order of the operations).

(19) Who<sub>i</sub> did John think that Mary believed that... Susie invited \_<sub>i</sub>?



Such usage of the memory buffer to satisfy long-distance filler-gap dependencies would work perfectly well as in (19). However, there are some linguistic facts which indicate that the grammatical derivation cannot be that simple.

### 3.3 Processing chunks: phases

From an algorithmic point of view, within the standard Minimalist program, probe-goal dependencies pose searching space problems. Empirically, it is clear that a probe does not retrieve a goal that is unnecessarily distant, e.g. deriving ungrammatical sentences such as “\*John<sub>i</sub> is likely that it was told \_<sub>i</sub> that Mary left” (Collins 1997:6). In formal

terms, exploring an unbounded structure might require a computationally demanding procedure, in which case limiting the searching space becomes a theoretical necessity (Chomsky 1998:99-104). These empirical and formal considerations led to splitting the derivation into computational chunks, technically dubbed 'phases'. In the standard minimalist view, a phase is a complete derivational stage during which a syntactic object is built using a finite set of lexical items pre-selected from the lexicon (a Lexical Array, Chomsky 1998:100,106). These lexical items will be available in active memory only at this stage of processing, so the phase is a closed derivation with respect to (external) Merge operations<sup>21</sup>. It ends once the Lexical Array is exhausted and all formal requirements are satisfied. More precisely, a phase consists of one head plus its domain (i.e. the syntactic object assembled up to the head merging point, in a strictly bottom-up fashion) and its 'edge', that is, in classical X-bar terms, the specifier of the phase head. Both the thematically complete verbal head (dubbed 'little v', Chomsky 1995:315) and the complementizer (C) are generally considered to be phase heads (Chomsky 2008). Once the phase head is merged and its formal requirements in terms of thematic roles ('little v') or discourse-related operators (C) are satisfied, the phase is delivered to the external systems (the Sensory-Motor and Conceptual-Intentional systems, §2). Apart from the edge, whose content will still be available at the next phase, all the rest of the phase will be inaccessible to further syntactic operations. This means that only material that has been moved to the vP or CP edge will be licensed to move further, higher up in the structure. What stays in the phase domain (under the

---

<sup>21</sup> For external vs. internal Merge see §2.2.



introduce an unjustified ambiguity in the feature set associated with the moving elements<sup>24</sup>.

Unless we make extra assumptions<sup>25</sup>, this mechanism, by itself, does not capture the contrasts revealed in extraction of wh-phrases internal to subject (islands) (21).a vs. object (non-island) (21).b phrases:

(21) a. \*[Which famous singer]<sub>i</sub> did [close friends of   <sub>i</sub>] admire you?

b. [Which famous singer]<sub>i</sub> did you admire [close friends of   <sub>i</sub>]?

In conclusion, the phase idea helps in rephrasing long-distance dependencies in terms of cyclic operation, but neither explains the look-ahead status of each step nor general empirical constraints such as islandhood<sup>26</sup>.

Here I propose that the complexity-reduction benefit of phase theory is preserved in a top-down derivational perspective, where problems concerning cyclic movement and islandhood can also be readily solved without additional stipulations. The complexity-reduction is possible because once the phase head has projected its last selection expectation, such a phase can be safely closed and ‘forgotten’ (apart from ‘unexpected’ items that are in the memory buffer, §3.2); this entails that no other syntactic operations will ever target previous phases, as predicted by the standard bottom-up model (but exactly in the inverse sequence). On the other hand, cyclicity (without need of any look-ahead device) and islandhood readily follow from the

---

<sup>24</sup> For a critics of the (apparent) solution for cyclicity in terms of ‘edge features’ (Chomsky 2008) refer to Chesi 2007:58-60.

<sup>25</sup> For instance, within a ‘multiple spell-out’ framework (Uriagereka 1999) we need to introduce the notion of ‘independent derivational workspaces’ and accept the idea of multiple ‘numerations’ (Nunes & Uriagereka 2000, see discussion in §2.1)

<sup>26</sup> There are other theories postulating different movement triggers that do not need features (notably Moro’s 2000 ‘dynamic antisymmetry’). Even though the feature-related problem dissolves under these assumptions, as far as I know, neither a discussion on the nature of phases nor accounts of islands phenomena, like the one presented in (21), have been proposed yet within these frameworks.

definition of phase and the order in which they are computed: I assume that a phase is the smallest computational domain in which a set of expectations must be satisfied.

This minimally corresponds to a lexical head (N, V, or A) its selection requirements and its extended projections (as discussed in §3.2).

Building a sentence top-down “by phase” needs a sequence of ordered steps: first a phase expectation is projected, that is a list of features associated to a certain lexical (N, V, or A) head; each functional feature of this extended projection is computed (either lexicalized or first further expanded before being lexicalized<sup>27</sup>) up to the phase head. Once the phase head has been inserted, its (ordered) select features are computed and one by one, they will generate the expectations for other phases to the right of the selecting head (as discussed in §3.1).

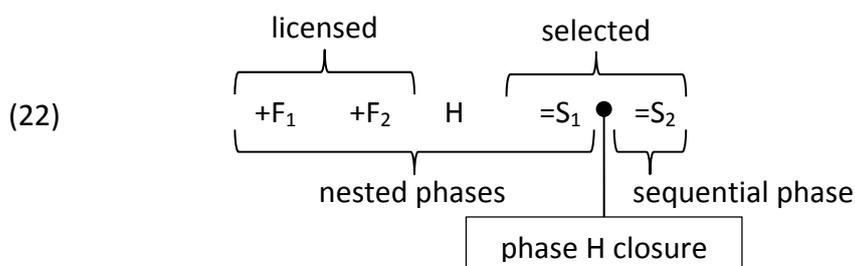
Computing features in sequence, following a depth-first procedure (that is, every nested phase must be completed, before returning to the superordinate one) has an important consequence: given the sequence of features expected in a simple declarative phase P, such that  $P = [{}_V +S +T V]$ , when the subject-related +S feature in P is computed (for instance, expanded as  $P_s = [{}_N +S +D N]$ , as discussed in §3.1), this will happen while other features (i.e. +T and V) are still on the expansion agenda of P.

Suppose that  $P_s$  (the subject of P) is expanded with a N headed phase such that  $N = [{}_{=PP} {}_N \text{ friends}]$ ; this head will bring into the derivation a select feature for creating a

---

<sup>27</sup> Unselected adjuncts and (restrictive) relative clauses are introduced in the derivation as result of functional feature expansions (e.g. ‘+R’ for restrictive relative clause; ‘+MOD’ for modal adjuncts like ‘with absolute certainty’). Their relative position should be universally determined (see footnote 13, Cinque 1999), but their final order can be affected by shifting operations triggered by scope necessities or preferences for computational complexity minimization (‘NP shift’-like operations, in head initial languages; ‘scrambling’ in head-final languages, Chesi 2012:183).

complex PP<sup>28</sup> (as in the case of “(close) friends of John”). The select features related to that head will be ‘nested’ since they will create other phase expectations while the superordinate phase is not yet closed (other features need to be expanded and the matrix phase head selection requirements projected). However nesting is not the only possible case: when the last select feature of a given phase is expanded (e.g. [<sub>=NP =NP =PP send</sub>]), that will complete the requirements triggered by that phase. In this case, the last selected argument qualifies as ‘sequential’ phase, rather than ‘nested’ (Bianchi and Chesi 2006, 2012) since no more feature can be added to the selecting phase (or computed) when the last selected phase starts being lexicalized. Hence once a phase head has had its last expected selection projected, it will be closed and forgotten to other syntactic operations (see Chesi 2012:183 for a discussion on exceptions to this). In (22) a summary of the terminology used (H is the phase head; +F<sub>n</sub> are functional features, i.e. the extended projection of H, while =S<sub>n</sub> are phases selected by H; features are sequentially computed, from left to right):



Because of phase expansion while the superordinate phase is not yet closed (i.e. phase ‘nesting’), I propose the following restriction on the memory buffers usage<sup>29</sup>:

<sup>28</sup> Also PPs are considered extended nominal projections: [<sub>N +P +D N</sub>]. +P is integrated in the phrase structure as a functional case marker. As discussed in footnote 13, this does not prevent any semantic (predicative) function from being associated to an item bearing the functional +P feature.

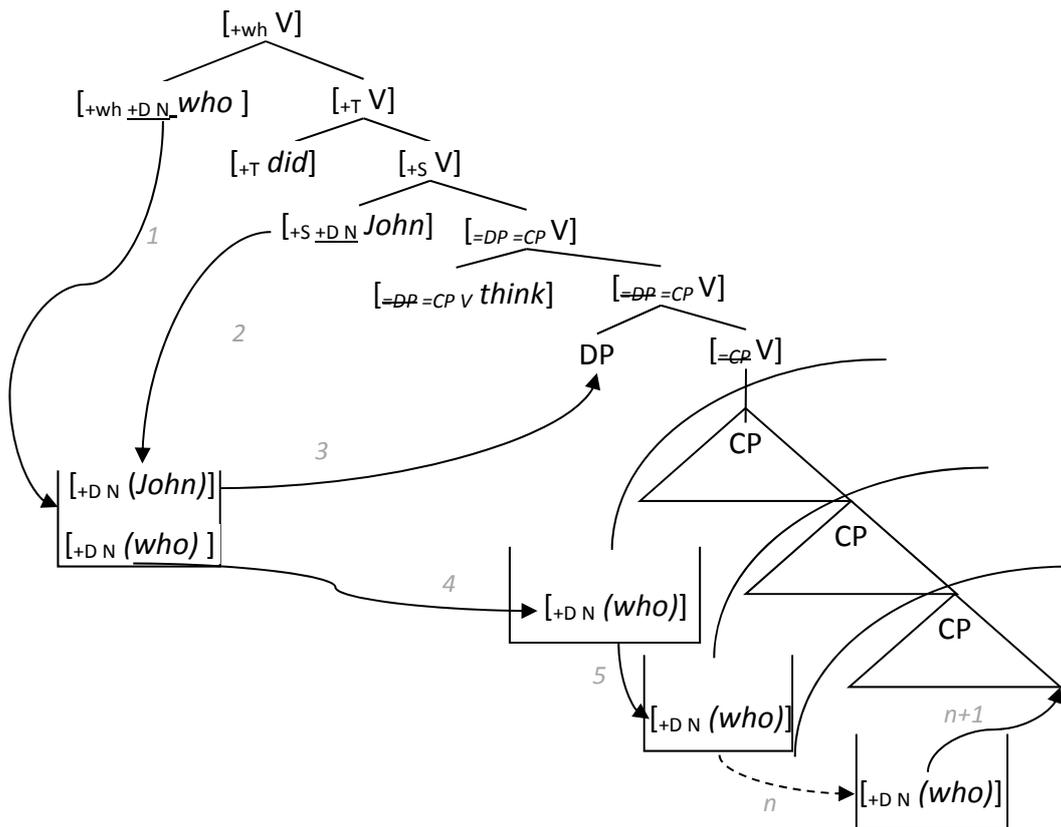
<sup>29</sup> What is taken here as stipulation could be derived from computational complexity considerations: i.e. ‘true recursion’ (nesting) vs ‘tail recursion’ (sequential) (Abelson & Sussman 1996, Bianchi & Chesi 2006, Chesi 2012:166-170).

(23) Content regulation of memory buffer

- a. every phase has its own memory buffer;
- b. the content of the memory buffer must be discharged, through appropriate selection, either within the current phase or in a sequential phase.
- c. the memory buffer must be empty at the end of the derivation (success condition)

The effect of (23).b is to re-cast a long-distance filler-gap dependency as in (19) as a series of successive cyclic movements, as illustrated in (24).

(24) Who<sub>i</sub> did John think that Mary believes that... Susie invited <sub>i</sub>?



Notice that unselected features are not 'consumed' up to the selection point, explaining cyclicity without look-ahead features triggering any intermediate step.

Moreover, because of (23).b, every nested phase qualifies as an island since the

content of the memory buffer cannot be discharged there (Chesi 2004, Bianchi & Chesi 2006, 2012). This captures the major cases of strong (configurational) islands (subject island, adjunct island, complex NP island, none of which involve selected phases), recasting Huang's (1982) insight (extraction domains are properly selected domains) in derivational terms.

#### **4. Processing advantages of a Top-down derivation**

Phillips (1996, 2003) motivated the shift to left-right derivations on the basis of empirical phenomena (a better understanding of conflicts between some constituency tests) and so did Bianchi & Chesi by discussing parasitic gaps and islandhood constructions (§3), Quantifier Raising (Bianchi & Chesi 2010), Pronominal Binding (Bianchi 2009) and 'rightward movements' like Extraposition and Heavy NP-shift (Chesi 2013). In this chapter I want to discuss two other empirical facts that should be of some interest from a processing point of view. First, the radically different perspective on island phenomena presented in §3.3 will be shown (§4.2) to be more in line with parsing expectations and experimental linguistic judgments (Phillips 2012). Second, the predictions, in terms of top-down memory-driven derivation, concerning intervention effects in Relative Clauses, will subsume promising formal accounts (Friedmann et al. 2009, Belletti & Rizzi 2012) and show a better fit with the processing evidence (Gordon et al. 2001, 2004) (§4.3). Before addressing these empirical facts, I describe (§4.1) how major parsing constraints can also be related to the grammatical derivation proposed here.

#### 4.1 Explaining (some) parsing biases

Despite some debate (Gibson & Pearlmutter 2000, Lewis 2000) it is generally agreed that in case of ambiguity not all of the possible grammatical analyses of a sentence are pursued by the human parser. The existence of garden path sentences (Bever 1970) is a consequence of this. The derivations I proposed here are largely deterministic; that is, given the first top-down expectation and the step by step choice of lexical items complying with the expected features, the grammar generates sentences with no ambiguity about when a particular operation such as Move or phase projection will apply. Therefore, garden-path examples like (25).b do not have a purely “grammatical” explanation; the grammar is not biased toward generating (25).a (hence unable to integrate ‘fell’ at the end of the sentence) rather than (25).b.

(25) a. [<sub>V</sub> [<sub>+D</sub> the <sub>N</sub> horse] [<sub>+T</sub> <sub>V</sub> raced] [past the barn]] \*fell

b. [<sub>V</sub> [<sub>+D</sub> the <sub>N</sub> <sub>+R</sub> horse [<sub>RC</sub> ([<sub>+C</sub> that] [<sub>+T</sub> was])] [<sub>V</sub> raced] [past the barn]] ] fell ]

The difference between these structures is attributable to a different lexicalization of ‘horse’: in (25).a. [<sub>N</sub> horse] is lexicalized, while (25).b. [<sub>N</sub> <sub>+R</sub> horse] is generated with a +R (restrictive relative) feature that needs to be expanded by a restrictive relative clause modifying the noun (see footnote 27). That relative clause is a nested (since unselected) verbal phase.

In some naïve sense, [<sub>N</sub> horse] is simpler than [<sub>N</sub> <sub>+R</sub> horse] since it has fewer features, but this does not entail that the grammar should favor lexical items with a minimized feature set. In order to guarantee full expressiveness, structure building operations should not tend to minimize the richness of the feature structures associated to lexical items. On the other hand, keeping a phase open has a computational cost in terms of memory load: nested phases introduce extra features to be kept in the active

workspace while other features in the superordinate phase must be processed. So it could be expected that a parsing algorithm that directly uses the structure building operations provided by the grammar would guess sequential expansion rather than nesting (hence  $[_V +S [+D N] +T =CP V] [v...]$  rather than  $[_V +S [+D N +R [v...]] +T V]$ ). This could recast simple economy metrics like the Minimal Attachment Principle (Frazier 1978, Frazier & Fodor 1978) in terms of memory load, since a reduced relative clause on the subject, being a nested phase, would prevent the matrix phase from being closed.

Moreover, the Active Filler Hypothesis (AFH, Clifton & Frazier 1989) finds here a coherent grammatical support. Wherever a potential gap is found (i.e. a selected thematic position), the filler (i.e. the item with unexpected features being held in the memory buffer) should be discharged at that point. This explain the following bias<sup>30</sup>:

- (26) a. Which boys is...  
b. ... Mary taking to the circus

Many English speakers consider the beginning of the sentence (25).a to be ungrammatical, despite the fact that there are compatible grammatical continuations such as (25).b. This is expected under the AFH, and it is also plausible to think that the parser, following the top-down derivation here discussed, would prefer this expectation  $[_V +wh +T V]$  where 'is' lexicalize both +T and V than the alternative  $[_V +wh +T +S V]$ , simply because the number of steps in the first case will be minimized, more features lexicalized at once and the filler discharged in a closer gap position.

---

<sup>30</sup> I wish to thank Janet Fodor who pointed out this example to me.

## 4.2 Rethinking Island effects

Dating back to Ross (1967), the standard understanding of island phenomena is that a violation occurs when an extracted item moves out of a constituent of a certain type, illustrated in (27).a-c in comparison with non-extracted counterparts (27).a'-c'.

- (27) a. \***Who** did [close friends of \_ ] eat an apple?  
a'. Did [close friends of Eva] eat an apple?
- b. \***What** did you shelve a book [without reading \_ ]?  
b'. Did you shelve a book [without reading the title ]?
- c. \***Who** did you shelve a book [that Adam gave to \_ ]?  
c'. Did you shelve a book [that Adam gave to Eva]?

Subjects (27).a, adjuncts (27).b and relative clauses (27).c usually qualify as islands. All of these classes of islands have been seriously challenged as sharp and unified phenomena<sup>31</sup>. Nevertheless, there are some clear violations, as in the examples (27).a-c above, and all the familiar bottom-up analyses (Chomsky 1973, 1981, 1986, Huang 1982, Lasnik & Saito 1992, Nunes & Uriagereka 2000, a.o.) attribute the ungrammaticality to the impossibility of establishing the dependency between the gap within the island and the *wh*-filler in its criterial position.

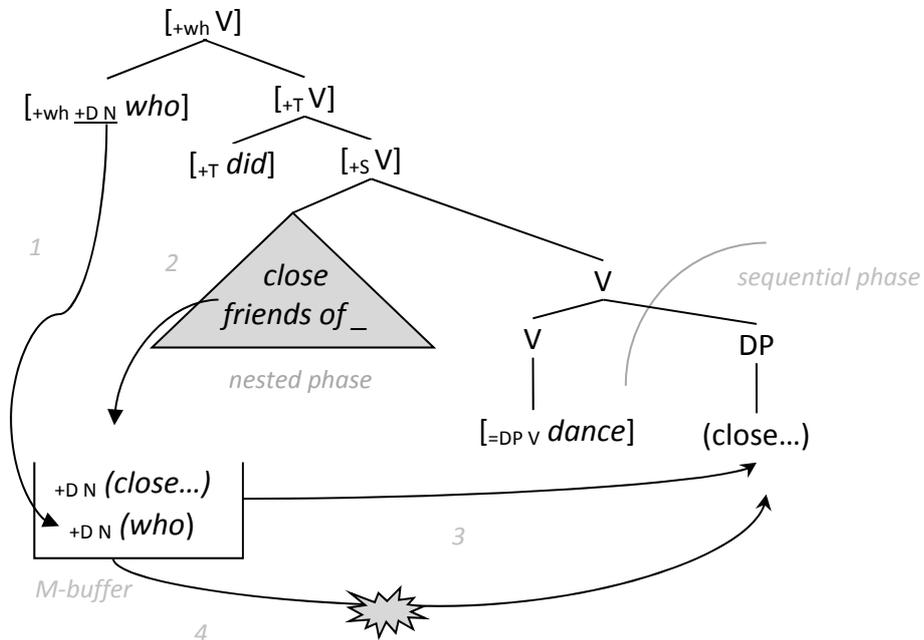
I think that focusing on constraints on “extraction” out of certain domains has been misleading, and that looking at these constructions from a top-down perspective provides us with a more explanatory alternative. As discussed in §3.2, in a top-down derivation, the *wh*-filler is processed first; then the restrictions on memory buffer usage stated in (23) above predict that sentences like (27).a are ungrammatical not

---

<sup>31</sup> See, for instance, for Subject Islands, Bianchi & Chesi 2012; for Adjunct Islands, Levine & Sag 2003 and Stepanov 2007; and Phillips 2012 for a formal and processing oriented comparison and comprehensive review of these phenomena.

because of the gap within the subject constituent (as predicted by standard approaches) but because of a missing, compatible, gap in the main verb phrase

(27) a. \*Who did [close friends of \_] dance?



The derivation depicted before shows that the wh item (who) is moved into the memory buffer (because of [+D N] unselected features and so does the subject “close friends of” for the same reason). The parasitic gap within the subject island can be interpreted (parasitically) using a ‘copy’ of the available content of the memory buffer of the matrix phase already available at that stage (Bianchi & Chesi 2006). This will not violate the memory buffer condition in (23).b, since the item in the buffer is not discharged at this point. That item will remain in the buffer and this will lead to the ungrammaticality of the sentence because of (23).c (i.e. the buffer must be empty at the end of the derivation).

This interpretation of island phenomena relates them to parasitic gap constructions (Engdahl 1983, Kayne 1983, Culicover & Postal 2001), illustrated in (28).

- (28) a. **Who** did [close friends of \_ ] like \_ ?  
b. **What** did you shelve \_ [without reading \_ ]?

The sentences (27).a-b constitute minimal pairs with (28).a-b. From the top-down perspective, the extra gap in the matrix clause in (28) ameliorates the “island effect” because it allows the memory buffer to be discharged in the usual legitimate fashion (i.e. properly selected position) within the phase in which the filler was introduced. As discussed in Phillips (2006), the improvement due to the extra gap in the matrix sentence has been evaluated experimentally and is substantial.

It should be clear at this point that this is not a “reductionist” solution in which the perception of ungrammaticality is explained in terms of restrictions or complications attributable to processing, pragmatics, or information structure (see Phillips 2012, and Sprouse et al. 2012 in response to Hofmeister & Sag 2010). The top-down proposal explains why island effects are caused by the missing integration of an unselected constituent within the phase in which it was introduced first, relating the ‘impenetrability’ of islands to the computational properties of nested constituents as defined in §3.3. This solution is not readily available from a bottom-up perspective. As far as I know, Nunes & Uriagereka (2000) are the only ones who offered a coherent explanation of parasitic gaps within a standard minimalist framework, but they need to postulate other devices, such as ‘numerations’ and ‘sideward movement’ (as well as ‘multiple spell-out’ and ‘independent derivational workspaces’), in order to guarantee that the two wh- items inserted independently in two different gap positions are in fact the same item. This follows straightforwardly from the Top-Down perspective by the uniqueness of the filler. Another merit of the top-down perspective is that it predicts some observed on-line parsing phenomena. An increase in processing

difficulty at the left edge of the island (McKinnon & Osterhout 1996, McElree & Griffith 1998, Phillips 2006) corresponds to the fact that a nested phase starts to be expanded there. Also predicted is the observed (milder) cost of gap-filling within the island while the filler is in the memory buffer. (But see Phillips 2006 for some unsettled cases discussed in the literature.)

### 4.3. Intervention effects and processing asymmetries

Another advantage of a top-down grammatical formalization that uses memory buffers to deal with filler-gap dependencies is the transparency of the proposed computation with respect to a processing model that predicts memory interference effects due to similarity of the elements entering a non-local relation (Gordon et al. 2001-04, Van Dyke & McElree 2006 a.o.). The well studied case of non-local dependencies in headed restrictive relative clauses (henceforth RCs) clarifies this point. It is well known that both in understanding and production of Subject vs. Object RCs adults (and children) show an asymmetric performance, with Subject RCs (S-RCs) (29).a generally easier to process than Object RCs (O-RCs) (29).b<sup>32</sup>:

(29) a. **The banker** [that \_ praised **the barber** ] climbed the mountain

b. **The banker** [that **the barber** praised \_ ] climbed the mountain

In (29) (Gordon et al. 2001), “the banker” (the head of the RC) is related to a gap within the RC that corresponds either to the Subject position of the RC (S-RC), (29).a, or to its Object position (O-RC), (29).b. The fact that O-RCs are generally harder to process than S-RC has been shown systematically using self-paced reading experiments (since King & Just 1991 a.o.), probe-task paradigms (Warner & Marastos 1978 a.o.), brain activity

---

<sup>32</sup> A substantial literature is present on the subject; see Gordon et al. 2004 for discussion.

analysis (Just et al. 1996 a.o.) and eye-tracking techniques (Traxler et al. 2002 a.o.). O-RC difficulty, however, can be mitigated by varying the type of RC subject (Gordon et al. 2001<sup>33</sup>):

(29) b'. **The banker** that **Ben** praised \_ climbed the mountain

b''. **The banker** that **you** praised \_ climbed the mountain

When pronouns are processed in the subject position of the relative clause, as in (29).b'', self-paced reading experiments show that the critical verbs (“praised” and “climbed”) are read faster than when proper names are present, (29).b'; the same critical regions are read slower when definite descriptions occupy the subject position of the relative clause, as in (29).b.

Memory-load accounts<sup>34</sup> (Gibson 1998, Warren & Gibson, 2002, 2005 a.o.) explain these contrasts considering an “integration cost” (Gibson 1998) associated to new discourse referents that are introduced in the structure: since pronouns do not introduce any new discourse referent and names are considered referentially lighter than full DPs (Warren & Gibson 2005), memory-load accounts predict faster reading time at the first critical verb when the subject is a pronoun and slightly longer reading time when it is a proper name. A problem emerges when they predict faster reading time for the Name-Name condition ('it was **Patricia** who **Dan** avoided at the party'),

---

<sup>33</sup> Since “Ben”/“you” items are shorter than “the barber”, the discussed effects could be biased by the relative NP length (thanks to Janet Fodor for pointing out this to me). However, the same contrast is reported in other experiments (e.g. Warren & Gibson 2002) where longer names are used, so I assume here that the contrast is not related to NPs length.

<sup>34</sup> I will not discuss here neither role-determinant accounts (MacWhinney & Pleh 1988) nor any frequency-based accounts (Kidd et al. 2007). For a discussion of role-determinant accounts vs. memory-load accounts refers to Belletti & Rizzi (2013). As for frequency-based accounts, the aim of this paper is eventually to explain why a certain processing asymmetry is found; if this asymmetry correlates with the distribution of specific feature patterns across corpora (e.g. subjects are usually animate while objects are inanimate), pointing out that low frequencies are the source of complexity is simply unexplanatory.

since this is empirically wrong: no significant difference is revealed in reading time of the critical verb between the DP-DP (>360ms) and the N-N conditions (340ms).<sup>35</sup>

The intervention-based accounts (Gordon et al. 2001, Friedmann et al. 2009, Belletti & Rizzi 2013 a.o.) can explain the symmetry revealed in the DP-DP and N-N matching conditions in terms of similarity of the intervening features: Friedmann et al. 2009 building on Rizzi (1990) locality constraint, assume that whenever features are shared between a filler, X, (e.g. “the banker” in (29).b) and a structural intervener, Z, (e.g. “the barber” in (29).b), the relation between X and the related selected gap, Y, gets disrupted in a way that is proportional to the kind (and number) of features involved. Assuming that ‘lexical restriction’<sup>36</sup>, rather than referentiality, is computed and that features expressing such lexical restriction in definite descriptions, proper names and pronoun are distinct (N for common nouns, N<sub>prop</sub> for proper names, and null N for pronouns), the intervention-based accounts predict exactly that the matching conditions in which common nouns and proper names are present are comparable, while pronouns are easier, since N is null. This idea can be precisely formalized (see below), but first notice that a crucial assumption of this account is that only features triggering movement should cause intervention (Friedmann et al. 2009:83). In this respect, the lexical restriction should not play a significant role, since this “feature” is buried in the NP within the DP and does not clearly trigger any movement in the standard bottom-up approach. This problem dissolves from the top-down perspective, since in a left-right derivation exactly the “lexical restriction” triggers the filler-gap

---

<sup>35</sup> The results of the experiment testing any condition permutation (‘it was DP/N/pro who the DP/N/pro **avoided** at the party’; where, e.g., DP = ‘the lawyer’, ‘the businessman’; N = ‘Patricia’, ‘Dan’; pro = ‘you’, ‘we’) are discussed in Warren & Gibson 2005. Reading times are approximated from their work.

<sup>36</sup> However it will be correctly expressed in semantic terms, this corresponds to what distinguishes a definite description from a pronoun (Friedmann et al. 2009:72).

dependency. Such lexical restriction is in fact represented by the unselected features that must be stored in the memory buffer. From this point of view let me propose a simple complexity metrics that takes into account the retrieval cost associated to memory buffer access, depending on the number of items stored ( $m$ ), the number of new features to be retrieved from memory ( $nF$ ), mitigated by the number of distinct cued features ( $dF$ ); this will generate the Feature-based integration cost ( $C_{FIC}$ )<sup>37</sup> below:

$$(30) C_{FIC} = \prod \frac{(1+nF)^m}{(1+dF)}$$

For example, considering DP = {+D, +number, N}, N = {+D, +number, N<sub>prop</sub>} (with N vs. N<sub>prop</sub> distinctiveness counting half, since expression of a subcategorization, of the distinctiveness related to +number or +person mismatches) and pro = {+D, +case, +person, +number}, for the DP-DP matching condition (“it was the lawyer that the businessmen avoided...”), the  $C_{FIC}$  at “avoided” will be equals to 64 (average reading time on the critical verb:  $\approx 360\text{ms}$ <sup>38</sup>), that is  $16 \cdot 4$  (16 for retrieving “the businessmen”, since  $nF=3$ ,  $m=2$  because two DPs are in memory at this time, and  $dF=0$  because no feature is cued by the verb distinguishing one DP from the other; 4 for retrieving “the lawyer”, since  $nF=3$ ,  $m=1$  and  $dF=0$ ). The same  $C_{FIC} = 64$  (average reading time on the critical verb:  $\approx 350\text{ms}$ ) applies at the N-N condition (“it was Dan that Patricia avoided...”) since  $nF$  will always be 3 for both Ns and no distinct feature will be cued at retrieval, hence  $dF=0$  ( $m=2$  for retrieving “Patricia”, since also “Dan” is stored in memory; and  $m=1$  when “Dan” is retrieved since “Dan” will be the only item in memory at that time).

---

<sup>37</sup> More complex (and probably accurate) equations can be built on cue-based retrieval strategies as discussed in Van Dyke (2007). Here, the intent of this formula is simply to highlight that the three factors used are sufficient to account for the minimal variations in the paradigm under discussion.

<sup>38</sup> All reading times are approximated from Warren and Gibson 2005:360.

On the other hand, we expect a  $C_{FIC} = 15.6$  for the pro-pro condition<sup>39</sup> (average reading time on the critical verb:  $\approx 290$ ms): for the first pronoun,  $nF=4$ ,  $m=2$  and  $dF=3$  because number, person and case mismatch cued by the verb, while  $nF=4$ ,  $m=1$  and  $dF=1$  (because of the cued case) for the second pronoun.

Also for DP-N and DP-pro conditions, the  $C_{FIC}$  makes the correct predictions:

(31) it was the lawyer that Patricia **avoided**... (DP – N condition)

N retrieval at **avoided**:  $nF=3$ ,  $m=2$ ,  $nF=0.5$  ( $N_{pro}$  vs. N counts as half)

DP retrieval at **avoided**:  $nF=3$ ,  $m=1$ ,  $dF=0$

$C_{FIC} = 42,6$  (i.e.  $10,7 \cdot 4$ ) (medium reading time of **avoided**:  $\approx 320$ ms)

(32) it was the lawyer that I **avoided**... (DP – pro condition)

pro retrieval at **avoided**:  $nF=4$ ,  $m=2$  and  $dF=3$  (person, number and case)

DP retrieval at **avoided**:  $nF=3$ ,  $m=1$  and  $dF=0$

$C_{FIC} = 25$  (i.e.  $6,25 \cdot 4$ ) (medium reading time of **avoided**:  $\approx 300$ ms)

Let me summarize, for convenience, medium reading times reported in Warren & Gibson (2005) at the critical verb and  $C_{FIC}$  values:

(33) Condition:	DP-DP	N-N	DP-N	DP-pro	pro-pro
Reading time (ms):	$\approx 360$	$\approx 350$	$\approx 320$	$\approx 300$	$\approx 290$
$C_{FIC}$ :	64	64	42,6	25	15.6

Despite the fact that names and definite descriptions requires a deeper featural scrutiny that I cannot address here<sup>40</sup> the  $C_{FIC}$  plainly subsumes and integrate Friedmann et al. 2009 account. Moreover, it can easily accommodate animacy features

<sup>39</sup> Notice that all pro-pro items used in Warren & Gibson 2005 present a mismatch in number, person and case. This facilitates processing (and generates lower  $C_{FIC}$ ).

<sup>40</sup> The semantic literature on this is rather rich, see Elbourne 2005 for a review. See also Belletti & Rizzi 2013 for N vs.  $N_{proper}$  distinction.

(whenever cued by the verb, Mak et al. 2002) and precisely characterize the filler-gap dependency triggers under the cue-based memory retrieval approach (Van Dyke & McElree 2006) expecting confusions (higher  $C_{FIC}$ ) when the cued characteristic features are non-unique during retrieval. This also explain the cross-linguistic variation revealed in Hebrew vs. Italian with respect to gender vs. number (Belletti et al. 2012): in Hebrew the verb agrees with the subject using gender not number, while the opposite happens in Italian (subject-verb agreement uses number, not gender). As expected, gender mismatch in Hebrew facilitates O-RCs processing, because the verb uses these cues to retrieve the relevant argument from memory (hence, in  $C_{FIC}$  terms, gender cued mismatches increase  $dF$  in Hebrew), while number mismatches helps in Italian for the same reason (hence, in  $C_{FIC}$  terms, number cued mismatches increase  $dF$  in Italian).

## 5. Concluding Remarks

I have presented here a derivational grammatical framework which falls within the general minimalist philosophy but which diverges from previous versions by generating phrase structure incrementally top-down, contrary to what the standard bottom-up Merge-based approach predicts. The proposed derivational shift stems from the observation that there are linguistic advantages to be gained from seeing syntactic trees as created systematically by lexical selection of lower phases by higher ones. This emphasis on selection, combined with the idea of phase as the minimal computational domain including the extended projection of lexical items, leads to the important distinction between ‘sequential’ (last selected argument) versus ‘nested’ domains (selected arguments that are not the last one or unselected adjuncts related to the expansion of functional features within the extended projection). Once this change has

been made on strictly linguistic grounds, we find that it brings formal syntactic derivations and on-line processing operations into a much closer alliance. Most obviously, sentence structures are generated from left to right and from top to bottom, as is presumably the case in both sentence production and sentence parsing. A specific instance discussed here concerns application of the island constraints on long-distance dependencies and their relation to the phenomenon of parasitic gaps, which is naturally understood in terms of a grammatical apparatus that includes a memory buffer which conveys information about elements in higher criterial ('filler') positions down to lower positions where thematic properties are interpreted .

Another comfortable alliance cited here between the proposed grammatical model and the facts of online processing concerns intervention effects, i.e., the disruption of a filler-gap dependency related to the presence of a similar item in the same memory-buffer, finds here a coherent interpretation predicting fine grained linguistic distinctions as the one expected by 'locality' theory (Rizzi 1990, Friedmann et al. 2009) suggesting a relevant correlation between grammatical theories and cue-based memory retrieval mechanisms (Van Dyke & McElree 2006).

It has been often suggested in literature that parsing related complexity metrics might influence grammaticality judgment or that the grammar reduces to parser (notably Dynamic Syntax, Kempson et al. 2001, Phillips 1996 for a review of the literature on this), but the perspective given here is radically different since this is a generation procedure that simply provides an effective and constrained way to generate sentences starting from simple expectations to be satisfied incrementally by merging lexical items that creates other expectations themselves. This seems to subsume correctly some parsing strategies, like the Active Filler Hypothesis, but it needs extra

machinery to become “a parser” (for instance neither backtracking is considered, nor parallel processing, which might be part of a realistic parser).

The virtue of the Minimalist research program in linguistics has been to drive the inquiry toward a simpler computational machinery: basic structure building operations, which are limited within a phase domain, and a lexicon. Pressing forward on the basis of computational considerations we have arrived here at an unconventional formal account of syntactic derivations which provides insights into empirical phenomena that have often been relegated too quickly to the status of merely ‘performance issues’.

## References

- Abelson, H., & Sussman, J. (1996). *Structure and Interpretation of Computer Programs*, MIT Press, Cambridge, MA, pp. 261–264.
- Belletti, A. & Rizzi, L. (2013). Intervention in grammar and processing. In I. Caponigro & C. Cecchetto (eds) *From Grammar to Meaning: The Spontaneous Logicality of Language*. pp. 293-311. Cambridge, UK: Cambridge University Press.
- Belletti, A., Friedmann, N., Brunato, D., Rizzi, L. (2012). Does gender make a difference? Comparing the effect of gender on children’s comprehension of relative clauses in Hebrew and Italian. *Lingua*. 122(10):1053–1069
- Bever, T. G. (1970). The cognitive basis for linguistic structures. In J. R. Hayes (Ed.), *Cognition and the development of language* (pp. 279–362). New York, NY: Wiley.
- Bianchi, V. (2009). A note on backward anaphora. *Rivista di Grammatica Generativa*. 34, 3-34

- Bianchi, V., & Chesi, C. (2006). Phases, left branch islands, and computational islands. *University of Pennsylvania Working Papers in Linguistics*. 12(1), 15–28.
- Bianchi, V., & Chesi, C. (2010). Reversing the perspective on Quantifier Raising. *Rivista di Grammatica Generativa*. 35, 3-38
- Bianchi, V., & Chesi, C. (2012). Subject islands and the Subject Criterion. in *Enjoy Linguistics! Papers offered to Luigi Rizzi on the occasion of his 60th birthday*. (Eds. Bianchi, V., & Chesi) pp. 25-53. Siena: CISCL Press.
- Boeckx, C. (2003). *Islands and chains: Resumption as stranding*. John Benjamins.
- Calabrese, A. (1982). Alcune ipotesi sulla struttura informazionale della frase in italiano e sul suo rapporto con la struttura fonologica. *Rivista di Grammatica Generativa*. 7: 3-79.
- Cecchetto, C. & Donati, C. (2010). On Labeling: Principle C and Head Movement. *Syntax*. 13:3, 241–278
- Cheng, L. L.-S. (2003). Wh-in-situ, Part II. *Glott International*. 7: 129-137
- Chesi, C. (2004). Phases and cartography in linguistic computation: Toward a cognitively motivated computational model of linguistic competence. PhD thesis, Università di Siena.
- Chesi, C. (2007). An introduction to phase-based minimalist grammars: Why move is top-down from left-to-right. *Studies in Linguistics*. 1, 49–90.
- Chesi, C. (2012). *Competence and Computation: toward a processing friendly minimalist Grammar*. Padova: Unipress
- Chesi, C. (2013). Do the right thing. *Studies in Linguistics*. 6, 131-164.
- Choi, Y. & Yoon, J. (2006). Argument Cluster Coordination and Constituency Test (Non)-Conflicts, *paper presented at NELS 37*, University of Illinois at Urbana-Champaign.

- Chomsky, N. (1957). *Syntactic Structures*. Berlin: Mouton de Gruyter.
- Chomsky, N. (1973). Conditions on transformations. In S. Anderson & P. Kiparsky (eds.), *A festschrift for Morris Halle*, pp. 232-286. New York: Holt, Rinehart, & Winston.
- Chomsky, N. (1981). *Lectures on government and binding*. Dordrecht: Foris.
- Chomsky, N. (1986). *Barriers*. Cambridge, MA: MIT Press.
- Chomsky, N. (1995). *The minimalist program*. Cambridge, MA: The MIT Press.
- Chomsky, N. (1998). *Minimalist inquiries: The framework* (No. 15). MIT Working Papers in Linguistics, MIT, Department of Linguistics.
- Chomsky, N. (2001). Derivation by phase. In M. Kenstowicz. (Ed.) *Ken Hale : A life in language*. (pp. 1-52.) Cambridge, Mass.: MIT Press.
- Chomsky, N. (2008). On phases. In R. Freidin, C. P. Otero, & M.-L. Zubizarreta (Eds.), *Foundational issues in linguistic theory: Essays in honor of Jean-Roger Vergnaud*. Cambridge, MA: The MIT Press.
- Chomsky, N. (2013). Problems of projection. *Lingua*. 130, 33--49
- Cinque, G. (1999). *Adverbs and Functional Heads: A Cross-linguistic Perspective*, Oxford: Oxford University Press.
- Clifton, C. & Frazier, L. (1989). Comprehending sentences with long-distance dependencies. In: Carlson, G.N., Tanenhaus, M.K. (Eds.), *Linguistic Structure in Language Processing*. Kluwer, Dordrecht.
- Collins, C. (1997). *Local Economy*. Cambridge, MA: MIT Press.
- Collins, C. (2002) Eliminating Labels, in *Derivation and Explanation in the Minimalist Program* (eds S. D. Epstein and T. D. Seely), Blackwell Publishing, MA: Malden.
- Culicover, P. W. & Postal, P. M. (2001). *Parasitic Gaps*. MIT Press, MA: Cambridge.

- De Vincenzi, M. (1991). *Syntactic parsing strategies in Italian: The minimal chain principle*. Springer. Berlin.
- Elbourne, P. (2005). *Situations and Individuals*. Cambridge, MA: MIT Press.
- Engdahl, E. (1983). Parasitic Gaps. *Linguistics and Philosophy*. 6:5-34.
- Felser, C. (2004). Wh-Copying, Phases and Successive Cyclicity. *Lingua*. 114(5):543–574
- Fodor, J.D., (1978). Parsing strategies and constraints on transformations. *Linguistic Inquiry*. 9, 427–473.
- Fox, D., & Pesetsky, D. (2005). Cyclic Linearization of syntactic structure. In *Object shift*, ed. By Katalin E. Kiss, special issue, *Theoretical Linguistics* 31(1–2):1–46.
- Frazier, L. & Clifton, C., Jr. (1989). Successive cyclicity in the grammar and the parser. *Language and Cognitive Processes*, 4(2), 93-126.
- Frazier, L. & Flores d'Arcais, G.B. (1989) Filler-driven parsing: A study of gap filling in Dutch. *Journal of Memory and Language*. 28:331-344.
- Frazier, L. & Fodor, J.D. (1978). The sausage machine: A new two-stage parsing model. *Cognition* 6, 291-325.
- Frazier, L. (1978). *On comprehending sentences: Syntactic parsing strategies*. PhD Thesis, University of Connecticut.
- Friedmann, N., Belletti, A., Rizzi, L. (2009). Relativized relatives: types of intervention in the acquisition of A-bar dependencies. *Lingua*. 119:67–88.
- Gibson, E. (1998). Linguistic Complexity: Locality of Syntactic Dependencies. *Cognition* 68, 1-76.
- Gibson, E., & Pearlmutter, N. J. (2000). Distinguishing serial and parallel parsing. *Journal of Psycholinguistic Research*, 29(2):231–240.

- Gordon, P., Hendrick, R., Johnson, M. (2004). Effects of noun phrase type on sentence complexity. *Journal of Memory and Language*. 51:97-114.
- Gordon, P.C., Hendrick, R., Johnson, M. (2001). Memory interference during language processing. *Journal of Experimental Psychology: Learning, Memory and Cognition*. 27:1411-1423.
- Grillo, N. (2008). *Generalized Minimality*. Utrecht Institute of Linguistics, OTS.
- Grimshaw, J. (1991). Extended projection. In P. Coopmans, M. Everaert & J. Grimshaw (Eds.) *Lexical specification and insertion* (pp. 115-134). The Hague: Holland Academic Graphics.
- Hofmeister, P., & Sag, I. A. (2010). Cognitive constraints and island effects. *Language*
- Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2001). *Introduction to Automata Theory, Languages, and Computation*, 2nd. Addison-Wesley
- Huang, C. T. J. (1982). *Logical Relations in Chinese and the Theory of Grammar*. PhD thesis. MIT.
- Huck, G. & Na, Y. (1990). Extraposition and Focus. *Language* 66: 51-77
- Joshi, A. K. (1985). *Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions?* (pp. 206-250). University of Pennsylvania, Moore School of Electrical Engineering, Department of Computer and Information Science.
- Kayne, R. (1983). *Connectedness and Binary Branching*. Foris.
- Kayne, R. (1994). *The antisymmetry of syntax*. Cambridge, MA: MIT Press.
- Kempson, R., Meyer-Viol & W. & Gabbay, D. (2001). *Dynamic Syntax: The Flow of Language Understanding*. Blackwell

- Kidd, E., Brandt, S., Lieven, E. & Tomasello, M. (2007). Object relatives made easy: A crosslinguistics comparison of the constraints influencing young children's processing of relative clauses. *Language and Cognitive Processes*. 22: 8860-897.
- Kluender, R. & Kutas, M. (1993). Bridging the Gap: Evidence from ERPs on the Processing of Unbounded Dependencies. *Journal of Cognitive Neuroscience*. 5:2, pp. 196-214.
- Larson, R. (1988). On the double object construction. *Linguistic Inquiry*. 19, 335-391.
- Lasnik, H. & Saito, M. (1992). Move alpha: Conditions on its application and output. Cambridge, MA: MIT Press.
- Levine, R., I. A. Sag. (2003). Some empirical issues in the grammar of extraction. In S. Müller (ed.), *Proceedings of the HPSG03 Conference*, Michigan State University, East Lansing. CSLI Publications. Life in Language (pp. 1-52). Cambridge: MIT Press.
- Lewis, R. L. (2000). Falsifying serial and parallel parsing models: empirical conundrums and an overlooked paradigm. *Journal of Psycholinguistic Research*, 29(2), 241-248.
- MacWhinney, B., & Pleh, C. (1988). The processing of restrictive relative clauses in Hungarian. *Cognition*, 29(2), 95-141.
- Mak, W. M., Vonk, W., Schriefers, H. (2002). The Influence of Animacy on Relative Clause Processing. *Journal of Memory and Language*, 47( 1) 50-68.
- McCloskey J. (2002). Resumption, successive cyclicity, and the locality of operations. *Derivation and Explanation in the Minimalist Program*, 184-226.
- McCloskey, J. (1990). Resumptive Pronouns, A'-Binding and Levels of Representation in Irish, in *The Syntax and Semantics of Modern Celtic Languages*. San Diego, CA: Academic Press.

- McElree, B. & Griffith, T. (1998). Structural and lexical constraints on filling gaps during sentence processing: A time-course analysis. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 24, 432-460.
- McElree, B., Foraker, S., & Dyer, L. (2003). Memory structures that subserve sentence comprehension. *Journal of Memory and Language*, 48(1), 67-91.
- McKinnon, R. & Osterhout, L. (1996). Event-related potentials and sentence processing: Evidence for the status of constraints on movement phenomena. *Language and Cognitive Processes*, 11, 495-523.
- Moro, A. (2000). *Dynamic antisymmetry*. Cambridge, MA: MIT Press.
- Nunes, J. & Uriagereka, J. (2000). Cyclicity and extraction domains. *Syntax*, 3, 20-43.
- Phillips, C. (1996). Order and structure. PhD thesis, Massachusetts Institute of Technology.
- Phillips, C. (2003). Linear order and constituency. *Linguistic Inquiry*, 34(1), 37-90.
- Phillips, C. (2012). On the nature of island constraints. I: Language processing and reductionist accounts. In J. Sprouse & N. Hornstein (eds.), *Experimental syntax and island effects*. Cambridge University Press.
- Pollard, C., & I. Sag (1994). *Head-driven phrase structure grammar*. Stanford: CSLI.
- Pollock, J. 1989. Verb movement, Universal Grammar, and the structure of IP. *Linguistic Inquiry*. 20:365-424.
- Ramchand, G. & P. Svenonius (2013) Deriving the Functional Hierarchy. *GLOW Newsletter* 70, 148-149
- Rizzi, L. (1990). *Relativized Minimality*. Cambridge MA: MIT Press.
- Rizzi, L. (2006). On the form of chains: Criterial positions and ECP effects. In L. Cheng & N. Corver (Eds) *Wh-movement: Moving on*, pp. 97-134. Cambridge, MA: MIT Press.

- Ross, J. R. (1967). *Constraints on variables in syntax*. PhD Thesis. MIT.
- Sauerland, U. (2004). The interpretation of traces. *Natural Language Semantics* 12, 63-127.
- Shlonsky, U. (1992). Resumptive Pronouns as a Last Resort. *Linguistic Inquiry*. 23(3):443-468.
- Sprouse, J., Wagers, M., & Phillips, C. (2012). A test of the relation between working memory capacity and syntactic island effects. *Language*.
- Stabler, E. (1997). Derivational minimalism. in Retoré (Ed.) *Logical Aspects of Computational Linguistics*. Springer.
- Stepanov, A. 2007. The end of CED? Minimalism and extraction domains. *Syntax* 10, 80-126.
- Tavakolian, S.L. (1981). The conjoined-clause analysis of relative clauses. In S.L. Tavakolian (Ed.), *Language Acquisition and Linguistic Theory* (pp. 167–187) Cambridge, MA: MIT Press.
- Traxler, M. J., R. K. Morris, & R. E. Seely (2002). Processing subject and object relative clauses: Evidence from eye movements. *Journal of Memory & Language*, 47(1), 69–90.
- Uriagereka, J. (1999). Multiple Spell-Out. In S. D. Epstein & N. Hornstein (eds) *Working Minimalism*. Cambridge, MA: MIT Press, 251-282.
- Van Dyke, J. A., & McElree, B. (2006). Retrieval interference in sentence comprehension. *Journal of Memory and Language*, 55(2), 157-166.
- Wanner E. & Maratsos, M. (1978). An ATN approach to comprehension. In Morris Halle, Joan Bresnan, and George A. Miller, editors, *Linguistic Theory and Psychological Reality*, chapter 3, pp. 119–161. Cambridge, MA: MIT Press.

Warren, T. & Gibson, E. (2005). Effects of NP type in reading cleft sentences in English.

*Language and Cognitive Processes*. 20: 751-767

Warren, T., Gibson, E. (2002). The influence of referential processing on sentence

complexity. *Cognition*. 85, 79–112.

Weinberg, A. (2001). A Minimalist Theory of Human Sentence Processing. in Epstein

and N. Hornstein (eds), *Working Minimalism*. Cambridge, MA: MIT Press.