The Asymmetry of Merge

Anna Maria Di Sciullo & Daniela Isac

This paper addresses the following question: What kind of properties must the structure-building operation Merge have such that, given a Numeration, the grammar will build the 'right' structure and avoid generating ill-formed configurations? The answer we will propose is that Merge should be seen as an asymmetric operation in the sense of relating two items whose sets of morpho-syntactic features are in a proper inclusion relation. In addition, we propose a partition of features into two stacks: categorial features and operator features. This distinction is independently motivated as it feeds into the definition of External Merge and Internal Merge (Chomsky's 2001). The proper inclusion condition will be assumed to hold for both of these operations, but the set of features under consideration for the evaluation of the proper inclusion relation differs: strictly categorial features for External Merge, and the whole set of features of lexical items for Internal Merge.

Keywords: asymmetrical relations, External Merge, feature valuing, Internal Merge, proper sub-set relation

1. Introduction

This paper is concerned with the properties of Merge, the operation that builds syntactic structures in the Minimalist program (Chomsky 1993, 1995, and related works). Our starting point is the observation that the existing definitions of Merge are relatively unconstrained, in the sense that many unattested structures may be built from a given Numeration.[1] Descriptively, what seems to matter in teasing apart the grammatical vs. ungrammatical structures that can be built from a given Numeration is *the particular order* in which the items are merged. For instance, if Merge first applies to the verb and the object, and the subject is merged later on in the derivation, a grammatical structure emerges. If on the other hand, Merge applies to the subject and little *v* before it incorporates the

This work is supported in part by funding from the Social Sciences and Humanities Research Council of Canada to the Asymmetry Project, grant number SSHRC 214-97-0016, as well as by a grant to the Dynamic Interfaces Project from the Fonds de recherche sur la société et la culture, grant number FQRSC 103690.

1 This is also the case with Chomsky's (2001) definition of merge ramifying in External Merge (the merger of two syntactic objects that have not been previously merged) and Internal Merge (i.e. Move).

ISSN 1450–3417
Biolinguistics 2.4: 260–290, 2008
http://www.biolinguistics.eu

object into the *v*P, the resulting structure is ungrammatical. However, the *order* of application of Merge does not follow in any way from the current characterizations of Merge. In this paper we propose that such ordering constraints on the application of Merge follow from a view on *Merge as an asymmetric operation*. Our specific proposal is that Merge can apply only if the sets of features of the two merged elements are in a *proper inclusion relation*. Support for this view is offered not only by the fact that under this assumption we can block unwanted derivations, but also by a number of welcome consequences which follow from our view on Merge. One such consequence is that specifiers emerge as different from both complements and adjuncts. This is a desirable result, which converges with work by Nunes & Uriagereka (2000) and others. Another consequence of our view is that adjuncts also come out as having a special status with regard to the way they are integrated in the structure. Essentially, adjuncts are added to the structure that is being built by an operation that is not subject to the proper inclusion condition, and thus cannot be equated with Merge. This makes adjuncts different from both complements and specifiers. On the other hand, our proposal is also able to capture the fact that at some level of analysis adjuncts and specifiers pattern alike, and differ from complements. These asymmetries between complements/non-complements, specifiers/non-specifiers, adjuncts/non-adjuncts are thus related to a more basic asymmetry which is in-built as a property of the structure-building operation itself.

This paper unfolds as follows. In section 2, we describe the properties of Merge, as described by Chomsky (1995), and we discuss the predictions with respect to the order of application of Merge. In section 3, we discuss some possible solutions to these problems and show why these solutions fail to produce the desired effect. In section 4, we present our proposal, and in section 5, we lay out the consequences and the conclusions.

2. Merge

In the minimalist program (Chomsky 1995), syntactic structure is built bottom-up by the operation Merge, which has two crucial properties: (i) Merge is a *binary* operation, which combines two elements into one constituent, and (ii) Merge is *recursive*, where the output of Merge may subsequently be submitted to Merge with other elements yielding a further constituent. The items that are subjected to Merge are drawn from a list called the Numeration. A Numeration is defined as a set of pairs (*LI*, i), where LI is a lexical item and *i* is an index indicating the number of times that *LI* is selected. Every time a lexical item is selected from the numeration in order to enter the derivation, its index is reduced by one. The derivation terminates when all indices are reduced to zero. The first application of Merge selects two items from the numeration and all the other (subsequent) instances of Merge iteratively select items from the numeration, one by one, until the numeration is exhausted and a complex object is formed that contains all of the items that started out as individual elements. The iterative application of Merge is thus responsible for building up the structure from bottom to top; the initial input to the first application of Merge consists of terminal items, and the final output of the last application of Merge is a hierarchical structure.

Whenever Merge applies to two items α, β, a new syntactic object is formed whose label is determined according to the following rule:

(1) *Merge* (Chomsky 1995)

Target two syntactic objects α and β, form a new object Γ {α,β}, the label LB of Γ(LB(Γ)) = LB(α) or LB(β).

Chomsky (1995: 243) distinguishes between two types of syntactic objects which can be subject to Merge: (a) lexical items and (b) objects of the type K = {γ, {α, β}}, where α, β are objects and γ is the label of K.

Initially, all objects in the numeration are of type (a), i.e. lexical items. Once Merge applies, it creates an object of type (b). The next stage of the derivation will thus include objects of type (a) and one object of type (b). Once an object of type (b) is created, Merge may assign a new element from the Numeration to this object (the workspace) and increase its complexity. Crucially, when an element from the numeration is merged to an already formed syntactic object, it can only be merged at the root of the structure existing in the workspace.

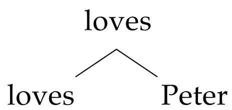
The implementation of Merge under this view presupposes the existence of a workspace: the space where the derivation unfolds and which will eventually contain the output of the recursive application of Merge.

To illustrate how the procedure works, consider the following Numeration:

(2) N = {(Mary,1), (*v*, 1), (loves, 1), (Peter, 1)}

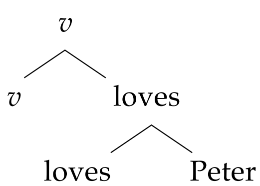
If *loves* and *Peter* are selected from this list and submitted to the operation Merge, the result will be a binary structure whose label is determined by the projection rule in (1) above.

(3)

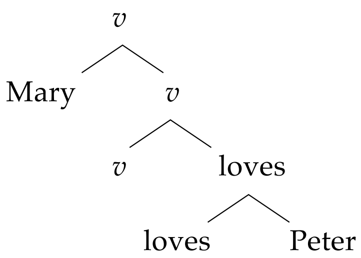


Next, one could select *v* from the Numeration and apply Merge again. The label of the resulting binary structure would again be determined according to rule (1).

(4)



The last application of Merge will bring together *Mary* and the complex structure labeled as *v* in (4), and will produce the structure in (5).

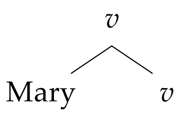
(5)


Since the Numeration now contains only items whose indices have been reduced to zero, the derivation will now stop.

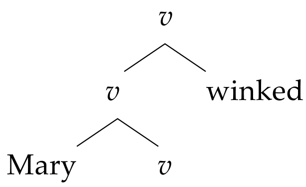
The Merging procedure described above may indeed generate a well-formed structure, as in (5). However, this procedure may also build many unattested structures from the same set of terminals. To illustrate, let us start with the numeration in (6).[2]

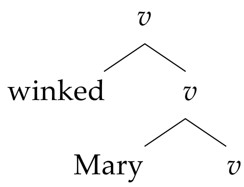
(6) N = {Mary, *v*, winked}

If *Mary* and *v* are selected from the Numeration and if Merge applies to them, the result will be a binary structure labeled *v*, as in (7).

(7)


Next, Merge will assign the only remaining element in the Numeration, namely *winked*, and will build one of the structures in (8), which is ill-formed. (8a) yields the right word order but does not capture the predication relation between the subject and the verb, and (8b) fails both to capture the predication relation and to yield the right word order.

(8) a.


b.


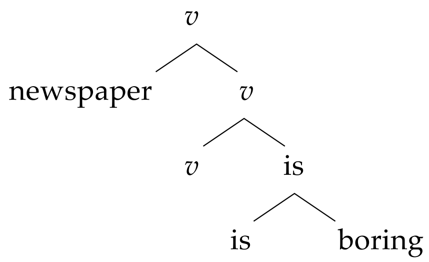
Similar considerations apply when deriving a transitive sentence. The 'right' derivation should first Merge the object with the verb and only then Merge the subject, but there is no property of Merge as defined above that would secure this order of application of Merge. Moreover, if the Numeration we start with is something like (9a), and the first instances of Merge produce a complex object like {*v*, {*v*, {is, {is, boring}}}}, there is no way to guarantee that what is Merged as a sister to this object is a phrase itself, and not just a head, as in (9b).

---

[2] In the rest of the paper, the numerations we will use for illustration purposes will ignore the numerical indices associated with the lexical items.

(9) a. N = {this, newspaper, *v*, is, boring}

b.



What seems to be needed is that the subject XP be built separately, through a successsion of Merge operations, and only then merged with {*v*, {*v*, {is, {is, boring}}}}. However, there is nothing in the description of the operation Merge above that would guarantee this.

This latter problem raises issues related not only to the order of application of Merge, but also to the notion of workspace. If at a certain stage in the derivation, there are two syntactic objects that were previously formed, as in (10), there is nothing in this procedure that flags {is, {is, boring}} as the workspace, rather than {this, {this, newspaper}}.

(10)

The choice is clearly important, since taking {is, {is, boring}} to be the workspace would allow a successful derivation (by selecting little *v* from the numeration and merging it to this syntactic object), but taking {this, {this, newspaper}} to be the workspace would lead to a crash, since the resulting structure will contain uninterpretable features. Such dead ends are undesirable from the point of view of economy. What seems to be needed is (i) a way to guarantee that objects are merged before subjects and that (ii) subject XPs are built separately, through a succession of Merge operations, and only then Merged to the workspace.

3. Possible Solutions

3.1. Renumeration

Johnson (2002) in an unpublished manuscript proposes that Merge is constrained by three factors: (i) the condition in (11), (ii) the Projection Rules, as stated in (12), and (iii) language-specific well-formedness conditions on the constituency of particular phrases.

(11) If an X^0 merges with a YP, then YP must be its argument.

(Johnson 2002: (12))

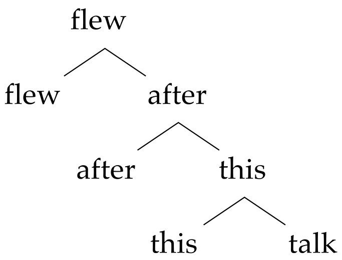
(12) *The Projection Rules* (Johnson 2002)

In $[_{\gamma}\ \{\alpha, \beta\}]$,

a. if just one of α and β is a phrase, then make γ a projection of the non-phrase;
b. if both α and β are phrases, then make γ a projection of the phrase that dominates the host.

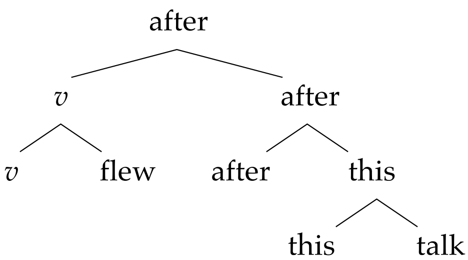
These three factors will ensure in Johnson's view that certain phrases re-numerate after they are built in the workspace, and thus, that the right order of Merge follows. To illustrate, Johnson discusses the derivation of the constituent *flew after this talk*, and he considers two alternatives that can be chosen at the point where *after this talk* has been built. One possibility is for the derivation to proceed by selecting and merging the verb, as in (13) below. If this happens, a violation of the condition in (11) occurs.

(13) *Merge: (Johnson 2002: (19d))



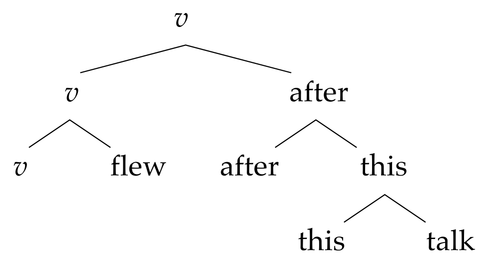
If, on the other hand, little *v* and the verb *flew* are selected and merged first, then renumerated, and then later merged with the PP *after this talk*, the derivation crashes because the application of the projection rules conflict with the well-formedness conditions on PPs in English (namely PPs cannot begin with a VP in English). In (14) below, it is the adjunct PP that projects according to the projection rules in (12), and this results in an ill-formed PP..

(14) *Merge: (Johnson 2002: (21g))



Instead of these two derivations, the successful one would first build *after this talk*, renumerate it, and then later merge it onto the independently constructed *v*P. Notice that the tree in (15) is identical to (14) except that the projected node is different in each case. This difference follows from the projection rules: In (14) above, the relevant projection rule is (12b) and it is the PP that is the host. In contrast, in (15), again the relevant projection rule is (12b), but this time the host is the little *v*.

(15) *Merge: (Johnson 2002: (22g))



These constraints manage to account for Johnson's problem, namely why subjects and adjuncts are grouped together in disallowing extraction from them: Adjunct phrases and subject phrases, but not other phrases, will be forced to go through a stage in which they are renumerated. However, the question we are trying to answer in this paper is a different one: Given a numeration, is there any way to determine the order in which the items will be merged in order to form a complex syntactic object? This question does not receive an obvious answer in Johnson's terms. More specifically, once the derivation gets to the point where *after the talk* is built, two choices are in principle possible, according to Johnson: Either the derivation continues by further selecting items from the numeration and successively merging them to the already built syntactic object, or the derivation continues by first renumerating *after the talk* and then selecting other elements from the numeration. The choice depends, in Johnson's system, on two factors: (i) whether or not a violation of condition (11) occurs and (ii) whether or not well-formedness conditions of particular constituents are violated.

The first condition crucially makes reference to the argument status of a phrase: The verbal head can be merged to the PP only if the PP is interpreted as the verb's argument. The problem is that this decision must be taken at the point where the workspace contains the PP only and there is no intrinsic feature of the PP itself which determines whether the PP is an argument or not, and hence whether the PP is going to be renumerated or not. In order to decide this, the features of the verb need to be considered. Johnson's system does not exclude this possibility and is in fact compatible with it. However, his analysis does not explicitly spell out this direction. The reason is obviously related to the fact that his focus is different from ours; his aim is to provide an explanation of why adjuncts and subjects fall into a natural class, as evidenced by the fact that they are both islands for extraction.

The second condition, i.e. the observance of well-formedness conditions of particular lexical items, again depends on evaluating the features of the relevant head, in this case the preposition. Given the Projection rules in (12), not renumerating the PP would result in building an object whose label would be a P, since the PP would be the host for the next application of Merge. This in turn would violate the well-formedness condition on English PPs, namely the constraint that PPs cannot begin with a VP in English. The problem with extending this as a solution to our concern is that it is not clear where these well-formedness conditions come from. One could claim that the well-formedness conditions on English PPs are encoded as selectional restrictions on the head of the PP. If so, this indicates that such features must be taken into account.

3.2. *Selectional Features*

Another possible solution to the ordering question is to capitalize on the *c-selectional features* of lexical items. This notion goes back to Chomsky's (1965) 'strict subcategorization rules' which were meant to analyze a lexical category in terms of its local distributional context, or 'frames' where it can be inserted. In Chomsky's view, strict subcategorization rules are part of the set of Phrase Structure rules, but at the same time, they are also features that characterize some lexical subcategories, and as such, they are part of various items' lexical entries.

It is thus possible to determine whether an object in the Numeration is an argument or an adjunct by looking at the selectional features of the other items in the numeration. If an item has a selectional feature that selects another object in the Numeration, then the selected object is an argument. Hence, condition (11) above will apply to it. If, on the other hand, an item in the Numeration is not selected by any of the other items in the list, then it is an adjunct, and condition (11) will apply to it.

While this is clearly a possible solution to our problem, it comes with a price: Selectional features will have to be assigned a special status as compared to all the other features. The feature system will need to flag selectional features in a way that will guarantee that they will be involved in the operation Merge, to the exclusion of other features.

To discuss only a few examples of such views, in the *Aspects*-model, subcategorization features are contextual features that need to match corresponding Phrase Structure rules. In other approaches, such as Stabler's (1998), selectional features are again assumed to be different from other features. This difference is signaled by a special notation in Stabler's system: A selectional D feature on an item, for example, is encoded as '=d', and this is different from the way in which other features are encoded. Likewise, in Müller's (2007) view, subcategorization features are singular in that they are always at the top of an assumed hierarchy of features of lexical items. Moreover, if one lexical item has more than one subcategorization feature, the respective subcategorization features are ordered with respect to one another and are 'discharged' one after another, depending on their relative position in the hierarchy of features.

What these views share is that regardless of how the special status of selectional features is formalized, they are treated as different from other features. While the order of application of Merge can indeed be derived from the selectional features of the items in the Numeration, the 'cost' of this proposal consists in the burdening of the theory with a special type of features, which are different from the other types of features computed in the syntax. Ideally, one would need a system in which these features play a role but are not given a special status as compared to other features. There are several proposals in the literature to this effect. For example, in the approaches advocated by Svenonius (1994), Holmberg (2000), Julien (2000), Matushansky (2002), among others, c-selectional features are uninterpretable categorial features that must be checked against the categorial feature of the selected object. Insofar as selectional features are treated as members of a larger set of features (strong features, or uninterpretable features), these views are less stipulative, and therefore more desirable than theories that consider selectional features as 'special' in any way. However, in these views it is not obvious how the order of Merge can be derived from selectional features.

One approach that seems to have both advantages, i.e. assume that selectional features are similar to other features and at the same time offer a solution to the ordering of Merge problem, is the one proposed in Adger (2003). In this approach, selectional features are uninterpretable and strong on a par with other (non-selectional) features. The notion of 'strength' of a feature is adopted from Chomsky (1993) and it essentially imposes a locality constraint on the feature that bears it: A strong uninterpretable feature must be checked under sisterhood.

In such a system, the ordering of Merge would be related to the locality requirement on the checking of strong uninterpretable features. Given that a verb like 'bring' for instance, has a strong uninterpretable [*u*N] feature, this feature will have to be checked under sisterhood, so Merge with an item bearing a matching [N] feature is the only possible choice.

However, in Adger's system, the ordering problem that we raise in this paper is still unresolved, we think. One point where the ordering problem becomes apparent is the case of items having more than one selectional feature, that is, more than one uninterpretable strong feature. One such case would be the verb 'show'. As discussed in Adger (2003), this verb has two selectional features, encoded as uninterpretable strong features: [*u*P] and [*u*N]. The fact that one of these features will be checked before the other one does not follow from the nature of uninterpretable strong features, since they are both of the same nature.

We thus conclude that views that would capitalize on c-selectional features in order to derive the ordering of Merge would not ultimately offer a solution to this problem.

4. Our Proposal

Our proposal is that the ordering of Merge follows from assuming that Merge is an asymmetric operation in the sense of (16).[3]

(16) *Asymmetry of Merge*

Merge is an operation that applies to a pair of elements in the Numeration whose sets of features are in a proper inclusion relation

Before we go on to illustrate how a derivation would proceed under this assumption, several remarks are in order.

First, we will assume that the numeration contains subarrays, i.e. sub-Numerations that define phases. The concept of Lexical Subarray was introduced by Chomsky (2000: 106ff.), who claims that Lexical Subarrays can be selected straightforwardly from the initial lexical array.

---

3 See also Di Sciullo & Isac (in press) for the role of set inclusion in movement chains, and Di Sciullo (2005) for the role of set inclusion in morphological merger.

Second, when viewed as a procedure, Merge is an operation that involves sub-components, including select and search. Merge first selects an element from the Numeration, and then searches for an item whose features are in a proper inclusion relation with the initially selected one. Once search is complete, the two items are joined as sisters. The asymmetry itself is built into the sub-operation 'search'. Given that 'search' is part of Merge, we end up with a picture in which the proper inclusion relation is not a precondition on the application of Merge, but an organic part of Merge itself.

The third observation concerns the feature sets of lexical items that are relevant for the condition in (16). Only morpho-syntactic features that are active in the syntactic computation are taken into account. In particular, we exclude phonological features, and semantic features. Moreover, in line with recent studies (Hale & Marantz 1993, Di Sciullo 2005, Bobaljik 2007), we also exclude Case features and phi-features, since we assume that such features are not computed in the syntax, but in a different space. Apart from these exclusions, we are making the following assumptions regarding the morpho-syntactic features of lexical items. To begin with, we assume that each lexical item has two types of features: interpretable and possibly uninterpretable ones. This is a fairly common view in the Minimalist literature. Moreover, we will assume that c-selectional features are not derivable from the semantic (theta) properties of lexical items and thus that they should be listed in the lexicon.[4] In line with other authors (see the discussion above, section 3.2), we will not assign any special status to selectional features: We will assume that they are just uninterpretable features, on a par with other uninterpretable features that lexical items might have. For example, the fact that little *v* selects a VP will be captured by positing an uninterpretable V feature, [*u*V], on little *v*. However, according to standard assumptions, little *v* also has another uninterpretable feature — [*u*T], meant to capture the morphosyntactic relationship between Tense and little *v*. This latter feature is clearly not a selectional feature, but in our system it is treated the same, i.e. simply as an uninterpretable feature.

Fourth, we will assume that both lexical and functional items have selectional features. This differs from other proposals (Adger 2003, among others), that tie in selectional features to theta marking properties of lexical items, and under which functional items do not have selectional features, given that they do not enter into theta marking. Thus we do not assume the existence of an independent hierarchy of functional projections and the hierarchical relation between little *v* and VP, or between T and little *v*, is treated the same as the relation between a V and its object, i.e. as a reflex of the proper set inclusion relation in our view.

Last, and perhaps most importantly, apart from the interpretable/uninterpretable distinction, we will assume an additional division between categorial features and operator features. The operator features that we will assume are features like [wh], [Topic] or [Focus] involved in A-bar movement and

---

[4] This assumption converges with views like Grimshaw (1991), but see Pesetsky (1982) for an acquisition based argument that the primitives of theta theory are epistemologically prior to the primitives of c-selection.

responsible for creating configurations that will be assigned an operator-variable interpretation at LF.[5] However, our proposal is that these features are involved not only in movement to an A-bar position, but in all movement. In our view, the distinction between categorial features and operator features underlines Chomsky's (2001: 7f.) distinction between External Merge and Internal Merge (i.e. Move): Categorial features are relevant for External Merge, whereas operator features are for Internal Merge. This distinction thus feeds into the definition of these two operations, which are independently assumed to exist. Characterizing External Merge and Internal Merge in terms of the types of features that get operated on makes it possible to characterize both the differences and the similarities between these two operations. The proper inclusion condition will be assumed to hold for both of these operations, but the set of features under consideration for the evaluation of the proper inclusion relation differs: In the case of External Merge the features under consideration are exclusively categorial features, whereas in the case of Internal Merge (i.e. Move) the whole set of features of the lexical items involved are considered. We can thus restate the proper inclusion condition in (16) in terms of this distinction, as follows:

(17) a. *Asymmetry of External Merge*

External Merge is an operation that applies to a pair of elements in the Numeration whose categorial features are in a proper inclusion relation.

b. *Asymmetry of Internal Merge*

Internal Merge is an operation that applies to a pair of elements in the workspace whose (total set of) features are in a proper inclusion relation.

What this means is that the relevant set of features that have to be evaluated for proper inclusion is different for the two operations. As it will be illustrated below, the point in the derivation where the shift from considering strictly the set of categorial features to considering the total set of features (including the operator ones) occurs is predictable. Internal Merge will be viewed as a last resort operation that applies only if external Merge cannot apply.[6]

A similar shift is discussed in Müller (2007), who proposes, together with Chomsky (2000, 2001, 2005), that some features are not obligatorily present on certain heads and that they can be added in the course of the derivation.[7] The relevant heads that can benefit from such 'expansions' of their features are edge heads in Müller's approach. Similarly, in our view, the expansion from the set of categorial features to the total set of features (including operator features) also occurs on certain heads only, namely those heads that have operator features to

---

5 See Rizzi (1997) on operators involved in topic configurations.

6 This is again not a new point, and it has been coined in the literature as the Merge over Move preference.

7 Chomsky (2000, 2001, 2005) proposes that this can happen *after* the phase is complete, but in Müller's (2007) view, the timing is different, i.e. a feature can be added on a phase head *before* the phase is complete.

begin with. In line with recent proposals, it is only edge heads that actually have operator features (see Butler 2004 and references cited therein) so in fact in our view too this expansion is only possible once the derivation reaches the point at which a phase head is merged. The 'last resort' flavor of this extension is not however limited to the availability of heads having operator features, but goes beyond that. In our view, this extension is warranted only if the numeration is exhausted and External Merge cannot apply. Moreover, as discussed by Müller, features can be added on edge heads only if this has an effect on the outcome of the derivation. What "having an effect on the outcome" means for Müller is being instrumental in implementing intermediary movement steps required by the Phase Impenetrability Condition. Similarly, what "having an effect on the outcome" means for us is triggering movement. The extension from the set of categorial features to the total set of features is licensed when the effect is to trigger movement. Alternatively, in Heck & Müller (2000, 2003), Fischer (2004), and Müller (2004), 'having an effect on the outcome' means 'balancing a phase'. In other words, under these approaches, features can be added on edge heads only if this is needed for balancing a phase. A phase counts as balanced if for every movement inducing feature on items in the numeration, there is an available matching feature. This idea is also paralleled in our proposal in that once the extension to all features is available for the edge head, it also becomes available to other heads in the c-command domain of the edge head. These latter heads will provide the matching feature for the operator feature in the edge head.

One question that arises is related to how 'costly' this distinction between categorial features and operator features is. In particular, even though we do not assign any special status to selectional features in our theory, we do introduce a distinction between categorial features and other types of features, and this seems to be just as 'costly' as a theory that grants singular properties to selectional features. It is obviously true that a system that would manage to predict the order of Merge without relying on this distinction would be simpler and more elegant. However, we think that this distinction is independently motivated by virtue of the fact that it underlines the distinction between the two structure-building operations: External Merge and Internal Merge. To the extent that these are two different operations, with distinct properties, and to the extent that their differences can be reduced to a difference in the type of features they compute over, the distinction between the relevant types of features is thus not that stipulative. Moreover, even if by some measures we traded a stipulation of selectional features with one on categorial features, our theory, unlike a theory based on selectional features, can account for the order of Merge in cases in which one item has more than one selectional features.

Below, we illustrate the feature specification that we assume for the most common lexical items that could enter a derivation:

(18)

nouns:	[N]	
Indefinite D:	[Num]	
	[*u*N]	
definite D:	[D]	
	[*u*Num]	
wh-D:	[D]	
	[*u*Num]	
	[wh]	
unergative V:	[V]	
transitive V:	[V]	
	[*u*D]	
unaccusative V:	[V]	
	[*u*D]	
v	[*v*]	
	[*u*V]	
	[*u*D]	
	[*u*Tense]	
unaccusative *v*:	[*v*]	
	[*u*V]	
	[*u*Tense]	
tense	[Tense]	
	[*uv*]	
	[*u*D] [EPP]	
	[*u*ClauseType:]	(operator feature)
C_1	[D]	
	[ClauseType]	(operator feature)
	[*u*Tense]	
C_2	[D]	
	[ClauseType]	(operator feature)
	[*u*Tense]	
	[wh]	(operator feature)

4.1. Sample Derivation

The purpose of this section is to provide an illustration of how asymmetric Merge can lead to the 'right' order of Merge. Taking as a starting point the lexical entries provided above, let us see how an analysis based on asymmetric Merge predicts the 'right' order of Merge. Let's assume that the Numeration consists of the following elements grouped into subarrays as indicated:

(19) N = {C, T, {D, Num, N, *v*, V, D, Num, N}}

The successive steps in building the structure are all in compliance with asymmetric Merge, as defined in (17). The derivational steps, given a Numeration like

(19), are given below:

Step 1. Select an item from Numeration that has interpretable features only
$\Rightarrow$ Select **N** {[N]}

Step 2. Select an item from Numeration that properly includes N
$\Rightarrow$ Select Num {[Num], [*u*N]}

Step 3. External-Merge N with Num.

(20) **NumP**

Num **N**
[Num] [N]
[*u*N]

Given the Earliness Principle (Pesetsky & Torrego 2001), the uninterpretable feature of Num will get checked and erased as soon as possible[8].

(21) **NumP**

Num **N**
[Num] [N]
[~~*u*N~~]

The newly created object is projected according to the rule in (1), i.e. out of the two items that merge, one will project its features. The newly created object thus has identical features with the item that projects. In our particular case, the projecting element is Num. The next step of the derivation will be to search for a new item in the Numeration whose features stand in a proper inclusion relation with the set of features of the object in the workspace.

Step 4. Select an item that properly includes Num
$\Rightarrow$ Select D {[D], [*u*Num]}

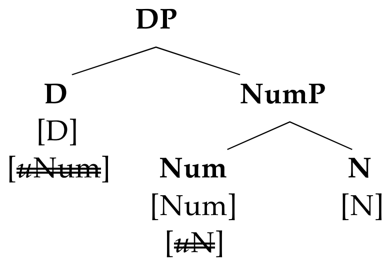
Notice that even though the set of features of D does not properly include the set of features of Num, as given in the Numeration, once Num is part of the derivation, its uninterpretable (selectional) feature gets checked and deleted, and thus its set of features comes to be properly included by the set of features of D.

Step 5. External-Merge D to the workspace and check uninterpretable features, as enforced by the Earliness Principle.

---

8 Notice that checking and deletion of uninterpretable features is subsequent to actual Merge. Even though after checking and deleting uninterpretable features, the proper inclusion relation does not hold, what is important is that it holds at the point where Merge applies, i.e. before checking and deletion of uninterpretable features of the newly merged item.

(22)



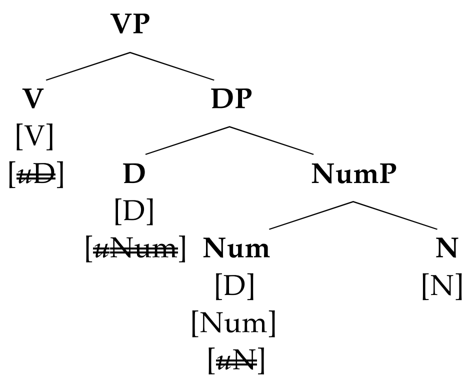
The workspace now contains a DP. If the Numeration contains more than one Noun, these 5 steps are taken for each Noun and all DPs are built in a parallel way. The result is that the workspace will potentially contain several DPs. The derivation will continue by Merging new elements selected from the Numeration to one of these DPs and the other DPs will be put on hold. What this means is that these DPs will be in the search domain for each of the next step of the derivation. The selection of the new element to be Merged to the workspace will again be made in compliance with the proper inclusion requirement.

Step 6. Select an item from the Numeration whose set of features properly includes the set of features of the DP in the workspace. Notice that at this point there are several choices, since not only the set of features of V properly includes the set of features of the DP, but also the set of features of little *v*, or T. We propose that the element that is selected must be the one whose features are the smallest superset of the set of features of the object in the workspace. In our particular case, the item that will be selected is V. This is a set theoretic version of 'locality'.

⇒ Select V {[V], [*u*D]}

Step 7. External-Merge V and DP and check uninterpretable features on V.

(23)

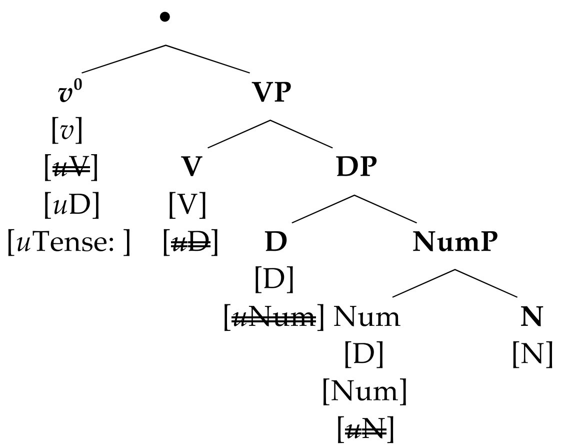


Step 8. Select an item from the Numeration whose set of features properly includes the set of features of the object in the workspace, i.e. VP.

⇒ Select little *v* {[*v*], [*u*V], [*u*D], [*u*Tense]}

Step 9. External-Merge little *v* to the VP which was in the workspace and check uninterpretable features on little *v*.

(24)

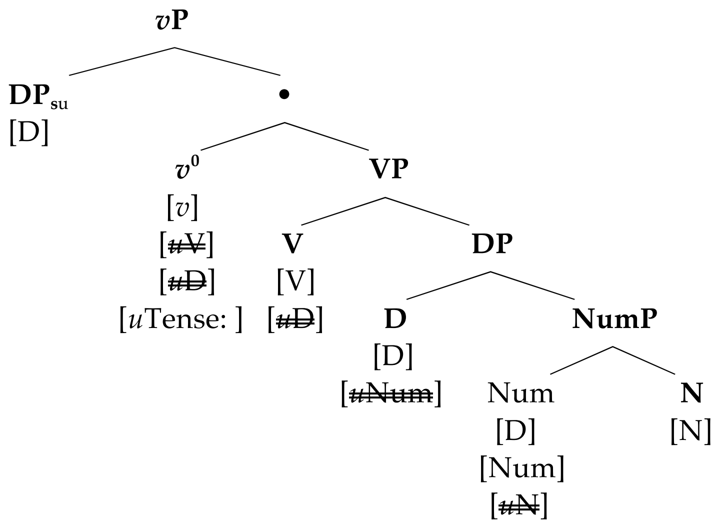


Step 10. The next step in the derivation should be to select an item from the Numeration whose set of features properly includes the set of features of the object in the workspace. There is one element in the numeration that fulfills this condition, i.e. Tense. However, this item is not part of the same subarray as all the items that have been already selected. The only syntactic object belonging to the same subarray is the DP that was put on hold in the workspace.[9] Even though this DP does not properly include the set of features of the syntactic object in (24), the proper inclusion condition can be met in the reverse direction, since the set of features of the object in (24) does properly include the set of features of the DP on hold.

⇒ Select DP {[D]} from the workspace/numeration

Step 11. External-Merge DP to the workspace and check uninterpretable features.

(25)



9 Remember that several DPs can be built in a parallel way with the first five steps above, and that the derivation continues by Merging new items to one of these DPs, while the others are set aside for future derivational steps. Alternatively, the algorithm could have the operation ‘renumerate’.

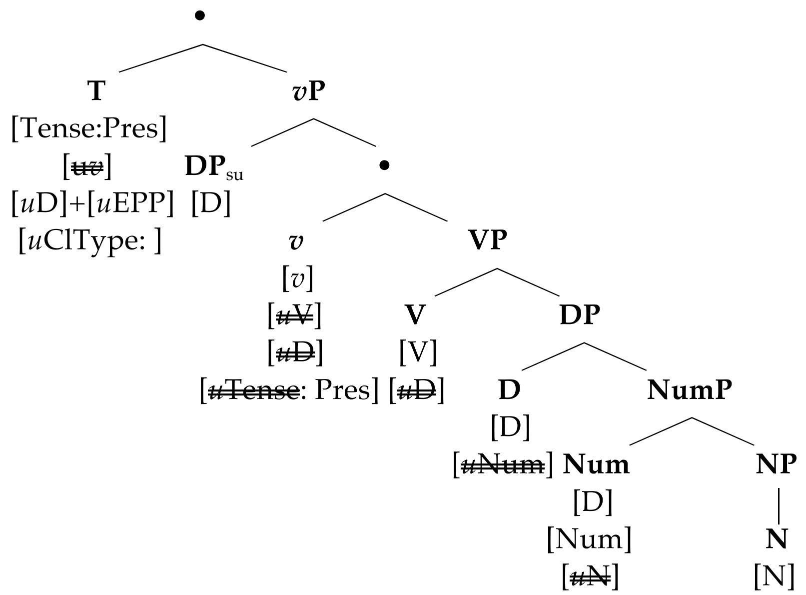
Notice that little *v* has three uninterpretable features: [*u*V], [*u*D], and [*u*Tense:]. Two of these, namely [*u*V] and [*u*D], correspond to selectional features. However, the ordering of checking of these two features is not a problem in our system, since it is a reflex of how the derivation grows and not a function of the properties of the selecting head.

Step 12. Select an item from the Numeration whose set of features properly includes the set of features of little *v*. The item that meets this condition is T, with the following set of features: {[Tense:Pres], [*uv*], [*u*D] + [*u*EPP], [*u*ClType:]}. However, not all of these features are considered when the proper inclusion relation is evaluated, since the Clause type feature is an operator feature. In our proposal, it is only categorial features that are considered for External Merge, to the exclusion of operator features. Notice that in this particular case, the proper inclusion relation holds regardless of whether the whole set of features of T is considered, or only the subpart of categorial features. So, restricting the set of features to categorial features for External Merge seems to be irrelevant in this case. However, there are other instances, such as the merger of a DP bearing a *wh*-feature, in which this restriction will be shown to play a role.

⇒ Select T {[Tense:Pres], [*uv*], [*u*D] +[*u*EPP], [*u*ClType:]}

Step 13. External-Merge T to the workspace and check uninterpretable features.[10]

(26)



Step 14. The next step, according to our proposal, should be the selection of an item whose set of features properly includes the set of features of the object in

10 Even though only the categorial features are considered for the purposes of External Merge, when External Merge applies, it applies to the lexical item, and so not only the categorial features are External-Merged, but all of the features that are part of the feature matrix of the respective item. The obvious analogy is pied-piping in the case of movement.

(26). However, at this point, there is only one item left in the numeration, namely the Complementizer, and clearly, C's set of features does not properly include the set of features of the object in (26). Moreover, there is no DP on hold anymore: The numeration only contained enough 'material' for the construction of two DPs, and both DPs have already been merged at this point.

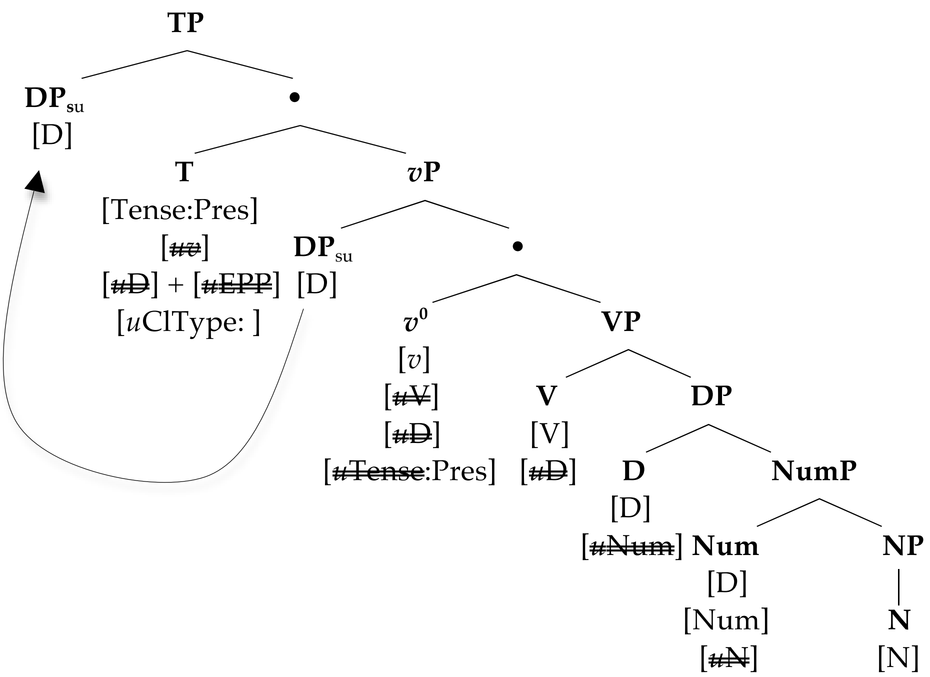
Our proposal is that this is exactly the kind of situation when internal Merge becomes an option and the search domain for the next Merge is therefore switched from the Numeration to the list of terminal items in the already constructed object. In other words, rather than looking in the numeration in order to select the next item to be merged, the derivation can look within the tree in (26) for the selection of the next item to be merged.

The selected item will have to satisfy the same proper inclusion requirement, i.e. its set of features must be in a proper inclusion relation with the set of features of the highest node, i.e. T. Moreover, since External Merge is not possible at this point, and Internal Merge is the only option, in our view the sets of features that are evaluated will have to include operator features as well, as discussed above. So far, the only item in the numeration that had operator features was T itself. Since no element has a set of features that properly includes the set of features on T, the search will select items whose set of features is properly included by the set of features of T. The closest DP will thus be selected.

⇒ Select D(P) {[D]} in [Spec,*v*P]

Step 15. Internally Merge DP to the workspace and check uninterpretable features.

(27)



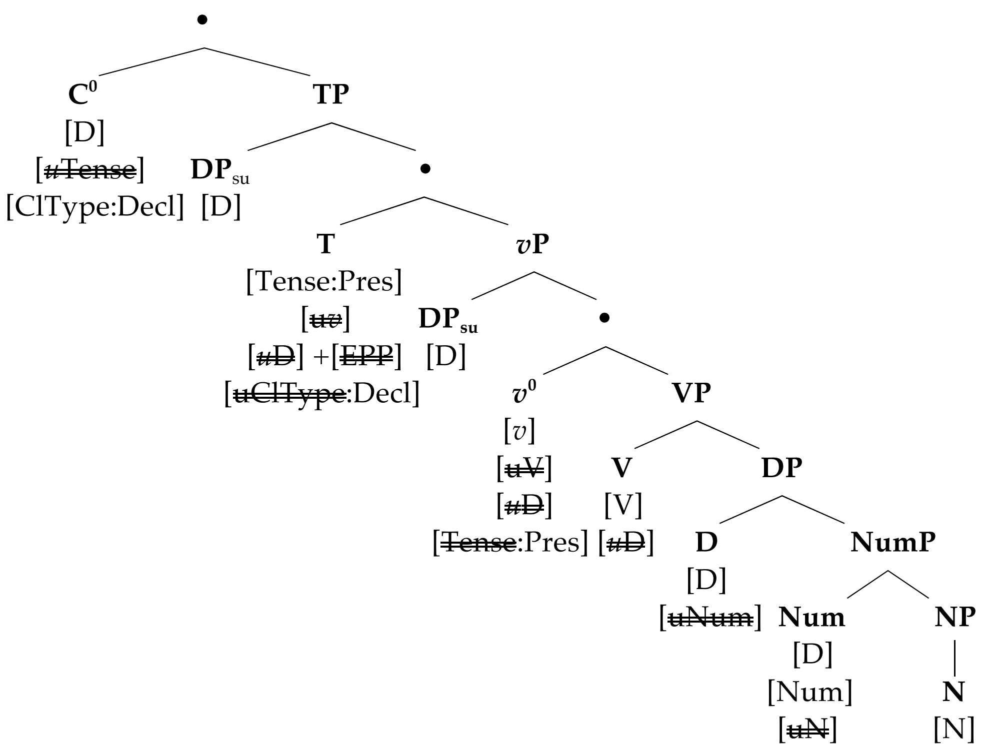
Step 16. The derivation continues by searching for items in the numeration whose set of features properly includes the set of features of T. Notice that the features of T that were taken into account in the previous step, i.e. Internal

Merge, included the operator feature on T, namely its Clause type feature. This operator feature will however be ignored in this next step, given that the search domain is the numeration and thus that the derivation is searching for an item to be External-Merged. The only item left in the numeration is the complementizer. Its total set of features includes {[D], [ClType], [*u*Tense]}. However, for purposes of External Merge, only the categorial features are considered, and thus the Clause type feature on the complementizer is ignored.

⇒ Select C {[C], [*u*ClType:], [*u*Tense]}

Step 17. External-Merge C to the workspace and check uninterpretable features.

(28)



Once the C is External-Merged, T's [*u*ClType] feature will check against the matching interpretable [ClType] feature of the complementizer. The numeration is now exhausted, and all the uninterpretable features are checked. The derivation is now completed.

Let us now see how the derivation of a *wh*-interrogative would proceed. In this case, the Numeration will obviously contain different feature specifications for the complementizer and the *wh*-DP. The complementizer will have to include an uninterpretable [*u*wh] feature, which is an operator feature, and the *wh*-DP will be headed by a D with a [wh] feature, again an operator feature. Moreover, we propose that the feature specification of T will also be different in a *wh*-interrogative, in the sense that it will also include a *wh*-feature. Apart from C, T and the displaced *wh*-item, all the other items in the numeration will have the same feature specification as assumed above. The assumption that C includes a *wh*-feature in *wh*-interrogatives, or that the displaced constituent has one are not new. The assumption that the T also has a *wh*-feature in *wh*-interrogatives can be supported by the overt manifestation of this feature in Bantu languages like Kinanda, as well as in other languages such as Berber, Celtic, Fiorentino and Trentino, Kikuyu, Palauan, Somali, Turkish, Ojibwe (see Schneider–Zioga 2002 and references cited therein). These languages show *wh*-agreement on the verb when the *wh*-DP is the object, and a preverbal *wh*- agreement affix when the moved *wh*-DP is the subject. What is crucial for us is that the overt *wh*-morphology *on* the verb is different depending on whether the *wh*-constituent is the object or the subject. This difference can be accounted for in our proposal in a natural way. As will become apparent below, we take *wh*-agreement with a subject to be a reflex of checking the [wh] feature on T against the *wh*- feature on the subject in a Spec-head configuration, whereas *wh*-agreement with the object is assumed to be the morphological manifestation of the checking relation between the same two features, but in a different syntactic configuration. More specifically, the object does not move and T's [wh] feature enters agreement (checking) with the [wh] feature of the object in situ.

In what follows we will not go through all the steps of a derivation of a *wh*-interrogative, but will focus only on the steps that differ from the derivation presented above. To begin with, we expect a first difference in the step that merges a *wh*-object and the verb. However, given that the *wh*-feature is an operator feature, and given that the Merge between the verb and the *wh*-object would be an instance of External Merge, the potential operator features of the verb or of the object are ignored. In other words, the features that are evaluated for proper set inclusion in this case are exactly the same as in the case of a non-*wh*-object, namely the categorial features, to the exclusion of operator *wh*-features. If the *wh*-item is not the object, but the subject, one expects a difference in the step that merges the *wh*-subject with a syntactic object like (24). Again, given that the *wh*-feature on the subject is an operator feature, and given that the merging of the subject in the specifier position of the little *v* is an instance of External Merge, the features under consideration will be the categorial ones. Thus, the proper inclusion relation between the little *v* and the subject DP will be unaffected by the presence or absence of a *wh*-feature on the subject.

The next steps in the derivation that are likely to be affected by the presence of *wh*-features are the steps involving T and C, since we assume that both T and C have *wh*-features. When T is merged to *v*P, the proper inclusion relation still holds, since T is supposed to be the superset and since the *wh*-feature is a feature posited on T. Moreover, since T is merged to *v*P as an instance of External Merge, the *wh*-feature on T will be irrelevant, as it is an operator feature. Now, in the next step of the derivation described above, the subject DP moves to [Spec,TP]. Since Move is Internal Merge, operator features become visible. If the *wh*-constituent is the object, T will attract the closest DP, namely the subject DP. The proper inclusion relation between the set of features of T and the set of features of the subject will clearly hold (the only difference from the equivalent step described above will be the presence of a *wh*-feature on T, hence an extra feature on what is supposed to be the superset anyway). If the *wh*-constituent is the subject, the proper inclusion relation will not change as compared to the equivalent step in the derivation above, since both T and the attracted subject will have an additional *wh*-feature.

Let us now examine what happens when the derivation gets to the level of the *wh*-C. When a *wh*-C is Merged, the *wh*-feature is ignored, since this is an instance of External Merge. However, in contrast to the derivation described above, the derivation of a *wh*-interrogative will not end with the merging of the C. This is because the *wh*-feature on C is uninterpretable and moreover associated with an uninterpretable EPP feature. What is similar to the derivation above is the fact that at this point the numeration is exhausted in both cases. So the only way for the derivation to continue is by Internal Merge. The closest constituent with a matching *wh*-feature will be attracted — it could be either the subject or the object, depending on which of the two bears a *wh*-feature. Whichever is attracted, the set of features of C, which should now be assumed to include not only categorial features, but also the *wh*-operator features, properly includes the set of features of the *wh*-DP.

Before concluding, we would like to discuss an apparent problematic consequence of our proposal that was brought to our attention by one of the reviewers. More specifically, our system seems to have the undesirable consequence that skipping projections in the functional domain is impossible, since we need a selectional relationship between each Merge in the functional sequence. Even if selection is treated as an uninterpretable feature, the problem still seems to remain, since the occurrence of aspectual projections like PerfP or ProgrP will need to be encoded as uninterpretable features on the head that selects them, and their occurrence, on the other hand, is optional. The T head, for instance, will have to bear an uninterpretable [*u*Perf] feature, to *encode* the fact that PerfP is a complement of T, and the Perf head will bear a [*u*Progr] feature, in order to capture the fact that the ProgP is a complement of Perf. Given that these projections are optional, we will end up either with multiple T's, Perf's, and Prog's (each with a different uninterpretable feature encoding selection), or else with a number of optional selectional features on each of these heads. It therefore looks like our system needs to stipulate that a *v*P is obligatory in the structure, as is TP, while the remainder of the projections are optional. One way to deal with this problem would be to stipulate a hierarchy of functional projections that would take care of the ordering relation between these projections. However, once we do this, our proposal that the ordering of Merge can be seen as a reflex of the proper set inclusion relation will lose substantiation.

While the problem of optional projections is obviously hard, we think that it is not impossible to offer a solution in terms of the proper set inclusion relation. What follows is a tentative solution to this problem, one that builds on a suggestion we found in Matushansky (2002) and Adger (2003). Based on a motivation that is independent from selection or from the need to provide an ordering relation between these projections,[11] Adger suggests that little *v*, Prog, Perf, etc. bear a more general [*u*Infl] feature, rather than more specific features like [*u*Tense] or [*u*Perf]. Similarly, Matushansky (2002), who builds on Julien

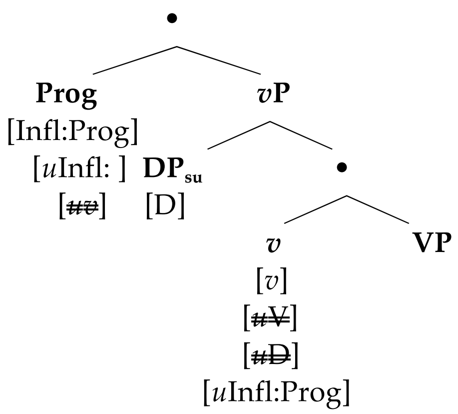
---

[11] Adger (2003) in fact assumes a hierarchy of functional projections and proposes a system in which the relation between little *v* and VP or between T and *v*P, for instance, is conceptually distinct from the selection relation that holds between a V and its object.

(2000), proposes that the lexical head determines the c-feature of all the functional heads in the extended projection of that lexical head. This proposal provides a means of formally expressing the notion of extended projection (Grimshaw 1991).

Building on this suggestion, we suggest the following feature specification for the relevant functional heads. For little *v*, we suggest that instead of the [*u*Tense] feature that we assumed above, a [*u*Infl:] feature could be considered. This [*u*Infl] feature could be valuated by any of the possible functional heads that could optionally appear on top of *v*P. If a Progressive head is present, its features will be {[Infl:Prog], [*u*Infl:], [*uv*]}. This set of features properly includes the set of features of little *v* (i.e. {[*v*], [*u*Infl:]}). Once the Prog head is merged, the [*u*Infl] feature on little *v* will be valuated as Progressive, and the uninterpretable [*uv*] feature on Prog will be checked.

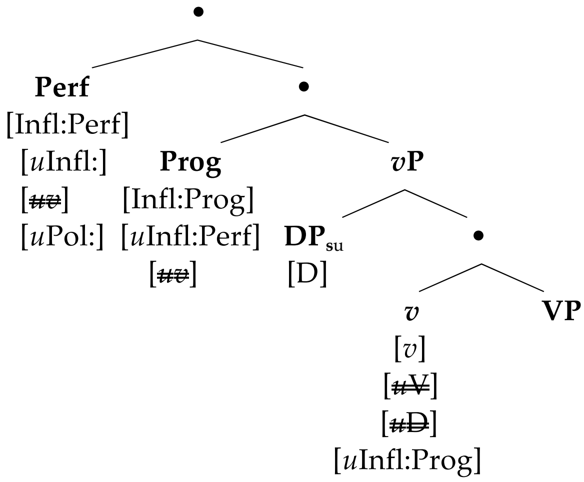
(29)



Now, if a Perfect aspectual head is also present in the numeration, its features can be assumed to include {[Infl:Perf], [*u*Infl:], [*u*Pol:]}. This set of features properly includes the features of the object in (29) above. Crucially, the [*uv*] feature on the Progressive head has been checked and erased by the time the Perf head is merged. Notice also that positing a [*uv*] feature on both the Prog and the Perf head does not commit us to having a *v*P as an obligatory complement to any of these heads. Having the *v*P as a complement is possible for these heads, but not obligatory. This is because selectional features are treated as uninterpretable features in our system, and uninterpretable features are under no restriction to be checked locally: They could be checked locally, under sisterhood, but they could also be checked at a distance, under c-command.

Once the Perf head is merged, the [*u*Infl:] feature on the Prog head is valued as Perf, and the [*uv*] feature on Perf is checked.

(30)

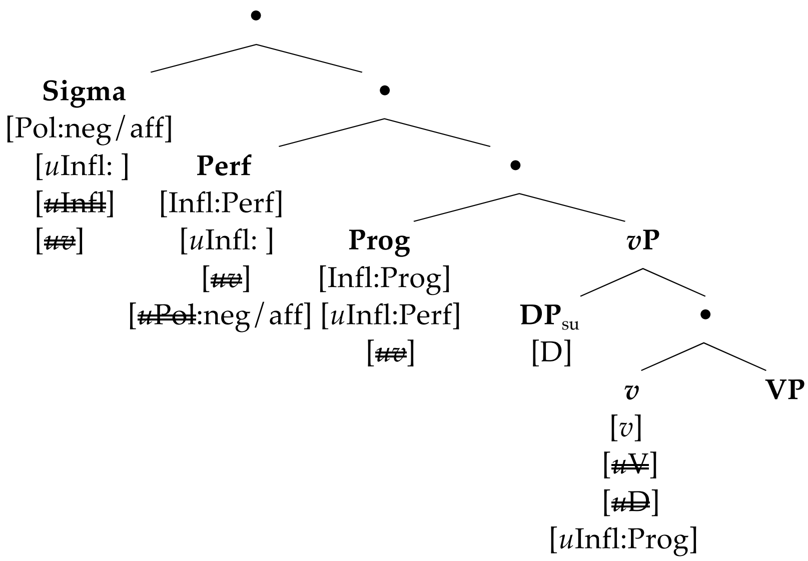


It is easy to see how the Prog head could be missing in this configuration without disturbing the proper set inclusion relation between the Perf head and the little *v*.

Apart from these two aspectual heads, and possibly a Voice head, which we have not considered here, but which would work along the same lines, there is another projection that for simplification purposes we have not taken into account in the sample derivation presented above, but which we nevertheless take to be an obligatory projection. This projection (that we will call a SigmaP, following Laka 1990 and others) hosts a polarity feature that can bear either the Negative or the Affirmative value. The total set of features of the Sigma head that we are assuming is: {[Pol:neg/aff], [*u*Infl:], [*u*Infl], [*uv*]}. This set properly includes the set of features of the object in (30) above, that is, {[Infl:Perf], [*u*Infl:], [*u*Pol:]}.

When the Sigma head is merged, there are two features on the Sigma head that get checked: its uninterpretable [*u*Infl] feature that captures the selectional property of Sigma, as well as its [*uv*] feature. Likewise, when the Sigma head is merged, the [Pol:] feature on the Perf head is also checked.

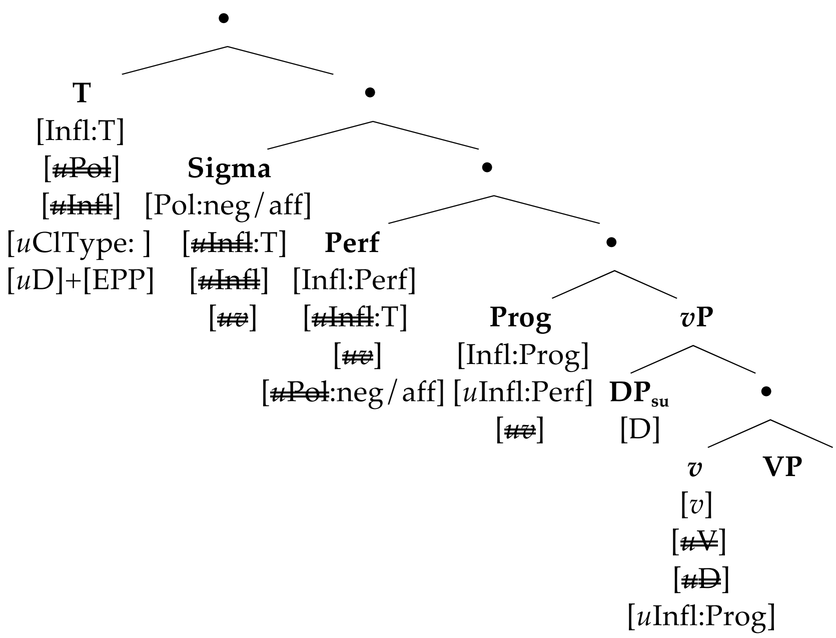
(31)



If the PerfP is not projected, the proper inclusion relation between the features of Sigma and the features of Prog still obtains, and if both Perf and Prog are absent, the proper inclusion relation between the features of Sigma and the features of little *v* is likewise preserved.

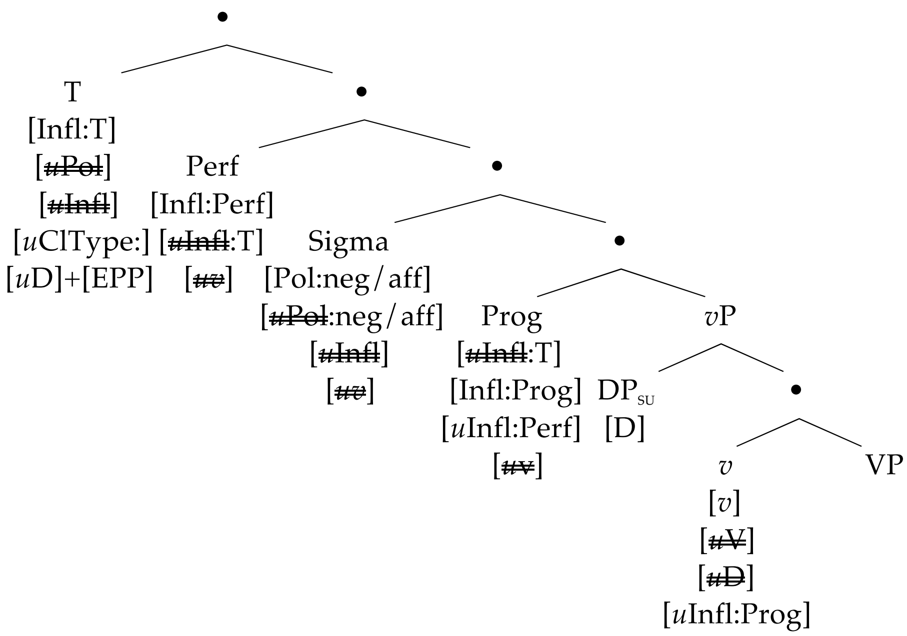
Finally, the T head in the numeration will have the following features: {[Infl:Tense], [*u*Pol], [*uv*], ~~[*uv*]~~, [*u*Infl], [*u*ClauseType:], [*u*D], [EPP]}. T's set of features properly includes the set of features of the object in (31) above, even under the restrictive assumption that only categorial features should be taken into account when evaluating the proper inclusion condition. When T is merged, its ~~[*uv*]~~ [*u*Infl] feature is checked against the interpretable [*v*] feature of Perf, and its [*u*Pol] feature is also checked against the interpretable [Pol] feature of the Sigma head. Moreover, the [*u*Infl:] feature of the Perf head and of the Sigma head can now be valuated as Tense.

(32)



Notice that at the point in the derivation where ProgP has been built, and a new item has to be selected from the numeration, the choice between Perf and Sigma could be switched. The proper set inclusion condition would still be met, but PerfP would be higher than SigmaP.

(33)



If Sigma is merged first, its [*uInfl*] feature is checked against the [Infl:Prog] feature of the Prog head, and its [*uv*] feature is checked against the [*v*] feature of the little *v* head. The remaining set of features — {[Pol:neg/aff], [*u*Infl:]} — is properly included in the set of features of the Perf head which is merged next. Once Perf is merged, its [*u*Infl] feature is checked against the [Infl:Prog] feature of the Prog head, its [*uv*] feature is checked against the [*v*] feature of the little *v* head, and its [*u*Pol:] feature is checked and valued by the [Pol:neg/aff] feature of he Sigma head. The remaining features of the Perf head — {[Infl:Perf], [*u*Infl:], [~~uPol~~:neg/aff]} — is properly included in the set of features of the T head, which is merged next.

This is in fact a desirable result, in spite of the fact that it seems to introduce optionality in our theory. A Sigma phrase carrying polarity features can indeed occur on either side of the Perf head.

(34) a. He claims to have not understood the instructions.
b. He claims to not have understood the instructions.

This is predicted to be possible under a theory such as Butler's (2004) who argues that phases should be defined as any domain that has a predicative core, a layer of functional structure and a quantificational layer (including modality and polarity phrases). In this view, not only *v*P and CP are phases, but also aspectual phrases, such as PerfP and ProgP. Given that the periphery layer can be reiterated on top of each of these phases, we end up with a picture in which the Sigma Phrase could show up both higher and lower than the Perf head.[12] This is

[12] Sigma Phrase can also be projected on top of TP, according to Butler, as part of the CP layer on top of TP. The higher Sigma head will have different features than the Sigma in (32) and (33). In particular, its set of features will include {[*u*Pol:], ([EPP]), [*u*Infl], [*u*ClType:]} and will thus properly include the set of features of T. Crucially, we follow Zanuttini (1994) and

captured in (32) and (33) above by the fact that the Perf head is assumed to have a [*u*Pol:] feature which can be checked and valued either by a lower Sigma, or by a higher one.

As already mentioned, the discussion above is tentative. Questions arise for example as to why can the [*u*Infl] feature on Perf or Sigma for instance not be valued from below, i.e. from the Prog head or from the Perf head respectively, and why do they have to be valued from a c-commanding head. We leave this for future research.

On the other hand, the exact feature content of each item in the numeration plays a crucial role in our account. It could be claimed that even though we have eliminated stipulations related to selectional features, as well as to the hierarchy of functional projections, we have moved the stipulations into the lexicon, in the sense that to a certain extent we have stipulated the feature content of the items in the numeration. At least some of the features we have assumed are features that have been used elsewhere in the literature, and are independently motivated. For the rest, ideally, it should also be possible to show that they are motivated independently from our proposal on Merge, something we have not done in this paper. Rather, we have treated the feature specification of lexical items as a hypothesis and then explored the consequences of this hypothesis. To the extent that the order of Merge can be derived from this hypothesis, this can be taken as motivating the latter.

5. Conclusion

In this paper we proposed a procedure for deciding the order in which items in the Numeration must be Merged. The criterion that we proposed is the proper inclusion relation: The set of features of the merged item must stand in a proper inclusion relation with the set of features of the object derived in the workspace. In addition, we assumed that the proper inclusion relation is 'local' in the sense that the element that is selected must be the one whose features make up the smallest superset of the set of features of the object in the workspace.

Our proposal does not assign any special status to selectional features. The latter are assumed to be uninterpretable features, on a par with other uninterpretable features, which cannot be seen as selectional. In addition, the order of Merge in the view presented in this paper is linked not to selectional features but to a relation between sets of features of lexical items, namely a proper inclusion relation.

We have proposed that the sets of features that are evaluated for the proper inclusion condition is different in the case of External Merge vs. Internal Merge. For External Merge, we have assumed that the relevant set of features is reduced to categorial features, whereas for Internal Merge, we have proposed that there is

---

Culicover (1996) in assuming that the concept of sentential negation is syntactically distributed across two functional projections: one under T and one higher than T. For language like Italian, where the negative marker occurs higher than T, the negative marker is assumed to move from the lower Sigma position to the higher one.

a last resort shift from the set of categorial features to the total set of features of lexical items, including both categorial features and operator features. Characterizing External and Internal Merge in terms of relevant sets of features involved allowed us to capture both the similarities and the differences between the two structure-building operations.

The content we provide to the notion of asymmetric Merge is different from what other authors have proposed. In particular, it is different from a notion of asymmetry related to which of the two elements that enter Merge will project, as expressed by Chomsky (1995: 246), or Langendoen (2003: 310).

> The operation Merge (α, β) is asymmetric, projecting either α or β, the head of the object that projects becoming the label of the complex formed.
> (Chomsky 1995: 246)

Our notion of asymmetric Merge is also different from the derivational view advocated in Jaspers (1998), Zwart (2006), and Johnson (2002). Under this latter view, Merge is asymmetric in the sense that it does not simply join two elements A and B, but it actually transfers one element at a time from the numeration to a workspace (derivation). Element A is thus merged to a workspace B, instead of A and B merging together. The result of Merge can then be described as an ordered pair <A,B>, where B is the current stage of the derivation and A the newly added element. In Zwart's terms, merge turns the current derivation into a dependent. Dependency is a semantic relation which must be syntactically realized. The core dependency relations are (a) head-complement: The complement is the dependent of the head, and (b) subject-predicate: The predicate is the dependent of the subject. The latter relation might be surprising, but the argument against considering the subject as being the dependent, rather than the predicate, is that the subject is not directly related to the verb. Arguments are related to the verb, but a subject can be any type of argument and even a non-argument. Therefore, the asymmetric relation of dependency is not between head and dependent but between dependent and non-dependent.

Our proposal on Merge as involving a proper inclusion relation between the two elements that undergo Merge does not exclude this view, but is different from it. Our claim is that the asymmetry of Merge is reflected in the morpho-syntactic properties of the members of <A,B>, in the sense that the set of morpho-syntactic features of A properly includes the set of morpho-syntactic features of B. This view is thus compatible with a derivational approach of the theory of grammar, in that we are not assuming that derivations are driven by some internal global syntactic architecture, but we assume instead that derivations proceed on a strictly local basis, caring only about the syntactic relations between members of sister pairs.

Crucially, the 'derivational' view on Merge cannot account for the ordering problem: given a Numeration, how can we predict the order in which the elements of the Numeration will be Merged to the workspace?

Under the assumptions that we are making in this paper about the nature of selectional features and about the feature specification of lexical items in the numeration, the ordering of Merge operations follows. In addition, our proposal also has the desired effect of building subject DPs separately and of merging them as phrases, rather than as lexical items. This result is obtained by our assumption that all DPs are built in a parallel way and simultaneously in the workspace. In our view, after these DPs are built, the derivation continues by selecting one of them as the host for the next Merge and leaving the others on hold. Moreover, Specifiers emerge as having a special status not only in the sense that they need to be built separately, but also from the point of view of the directionality of the proper inclusion relation. In the step-by-step derivation described above the proper inclusion condition always holds between the workspace and the merged item, but the directionality of proper inclusion is not consistent. Given that the object in the workspace is extended by the Merge of a new element, the expectation is that the direction of proper inclusion should be from the Merged element to the object in the workspace. In other words, the set of features of the merged element should properly include the set of features of the object in the workspace. However, even though in most of the steps shown in the derivation above this is the directionality of proper inclusion, sometimes this directionality is reversed and the set of features of the object in the workspace properly includes the set of features of the merged item. This change of direction of the proper inclusion relation is apparent only when items are merged in a 'specifier' position.

An additional result of our analysis is that adjuncts are integrated into the derivation by an operation that has different properties than Merge. More specifically, the operation that puts together an adjunct and the XP that is its adjunction site is not subject to the proper inclusion condition. Clearly, there is no proper inclusion relation between the set of features of, say, a PP adjunct, and the set of features of *v*P.

Lastly, the question arises as to why there should be a proper subset relation built in the definition of Merge. One possible answer to this question could be that this is required by conceptual necessity, in order to ensure legibility of the interfaces by the external systems. Interestingly, there is a morphism (structure preserving mapping) between asymmetric Merge and the semantic operation of functional application. In functional application, generally, the function must be of a higher type than its argument. In asymmetric Merge, generally, the grammatical features of the selector must properly include the ones of the selectee. We see this morphism as a conceptual motivation as to why the proper subset should hold for the merger of predicates with arguments. The morphism ensures the mapping of the expressions derived by the core operations of the grammar and the ones derived by the operations of the conceptual-intentional system.

References

Abney, Steven. 1987. The English noun phrase in its sentential aspect.

Cambridge, MA: MIT dissertation.
Adger, David. 2003. *Core Syntax*. Oxford: Oxford University Press.
Adger, David & Gillian Ramchand. 2003. Merge and move: Wh-dependencies revisited. In Lisa L.-S. Cheng & Norbert Corver (eds.), *Wh-Movement Moving On* (Current Studies in Linguistics 42). Cambridge, MA: MIT Press.
Bennis, Hans, Norbert Corver & Marcel den Dikken. 1998. Predication in nominal phrases. *The Journal of Comparative Germanic Linguistics* 1, 85–117.
Bobaljik, Jonathan David. 2007. Where's phi? Agreement as a post-syntactic operation. In David Adger, Susana Béjar & Daniel Harbour (eds.), *Phi-Theory: Phi-Features Across Interfaces and Modules* (Oxford Studies in Theoretical Linguistics 16). Oxford: Oxford University Press.
Butler, Jonny. 2004. Phase structure, phrase structure, and quantification. York: University of York dissertation.
Carstens, Vicki. 1991. The morphology and syntax of determiner phrases in Kiswahili. Los Angeles, CA: University of California dissertation.
Carstens, Vicki. 2000. Concord in minimalist theory. *Linguistic Inquiry* 31, 319–355.
Chomsky, Noam. 1965. *Aspects of the Theory of Syntax*. Cambridge MA: MIT Press.
Chomsky, Noam. 1993. A minimalist program for linguistic theory. In Ken Hale & Samuel Jay Keyser (eds.), *The View from Building 20: Essays in Linguistics in Honor of Sylvain Bromberger* (Current Studies in Linguistics 24), 1–52. Cambridge, MA: MIT Press.
Chomsky, Noam. 1995. *The Minimalist Program* (Current Studies in Linguistics 28). Cambridge, MA: MIT Press.
Chomsky, Noam. 2000. Minimalist inquiries: The framework. In Roger Martin, David Michaels & Juan Uriagereka (eds.), *Step by Step: Essays on Minimalist Syntax in Honor of Howard Lasnik*, 89-155. Cambridge, MA: MIT Press.
Chomsky, Noam. 2001 Derivation by phase. In Michael Kenstowicz (ed.), *Ken Hale: A Life in Language* (Current Studies in Linguistics 36), 1-52. Cambridge, MA: MIT Press.
Chomsky, Noam. 2005. Three factors in language design. *Linguistic Inquiry* 36, 1–22.
Culicover, Peter. 1991. Polarity, inversion, and focus in English. *ESCOL* 8, 46–68.
Di Sciullo, Anna Maria. 2005. *Asymmetry in Morphology* (Linguistic Inquiry Monographs 46). Cambridge, MA: MIT Press.
Di Sciullo, Anna Maria & Daniela Isac. In press. Movement chains at the interfaces. *Canadian Journal of Linguistics — Special Issue on Interfaces*.
Fischer, Silke. 2004. *Towards an Optimal Theory of Reflexivization*. Tübingen: Eberhard-Karls-Universität Tübingen dissertation.
Grimshaw, Jane. 1991. Extended projection. Ms., Brandeis University.
Hale, Morris & Alec Marantz. 1993. Distributed morphology and the pieces of inflection. In Ken Hale & Samuel Jay Keyser (eds.), *The View from Building 20: Essays in Linguistics in Honor of Sylvain Bromberger* (Current Studies in Linguistics 24), 111–176. Cambridge, MA: MIT Press.
Hauser, Marc D., Noam Chomsky & W. Tecumseh Fitch. 2002. The faculty of language: What is it, who has it, and how did it evolve? *Science* 298, 1569–1579.

Heck, Fabian & Gereon Müller. 2000. Successive cyclicity, long-distance superiority, and local optimization. *WCCFL* 19, 218–231.
Heck, Fabian & Gereon Müller. 2003. Derivational optimization of Wh-movement. *Linguistic Analysis* 33, 97–148.
Holmberg, Anders. 2000. OV order in Finnish. In Peter Svenonius (ed.), *The Derivation of VO and OV* (Linguistik Aktuell / Linguistics Today 31), 123–152. Amsterdam: John Benjamins.
Jaspers, Dany. 1998. Categories and recursion. *Interface* 12, 81–112.
Johnson, Kyle. 2002. Towards an etiology of adjunct islands. Ms., University of Massachusetts, Amherst.
Julien, Marit. 2000. Syntactic heads and word formation: A study of verbal inflection. Tromsø: University of Tromsø dissertation.
Kayne, Richard S. 1993. Towards a modular theory of auxiliary selection. *Studia Linguistica* 47, 3–31.
Kennedy, Christopher & Jason Merchant. 2000. Attributive comparative deletion. *Natural Language and Linguistic Theory* 20, 553–621.
Laka, Itziar. 1990. Negation in syntax: On the nature of functional categories and projections. Cambridge, MA: MIT dissertation.
Langendoen, Terence. 2003. Merge. In Andrew Carnie, Heidi Harley & Mary Ann Willie (eds.), *Formal Approaches to Function in Grammar: In Honor of Eloise Jelinek* (Linguistik Aktuell / Linguistics Today 62), 307–318. Amsterdam: John Benjamins.
Matushansky, Ora. 2002. Movement of degree/degree of movement. Cambridge, MA: MIT dissertation.
Matushansky, Ora. 2006. Head movement in linguistic theory. *Linguistic Inquiry* 37, 69–109.
Müller, Gereon. 2004. Phrase impenetrability and wh-intervention. In Arthur Stepanov, Gisbert Fanselow & Ralf Vogel (eds.), *Minimality Effects in Syntax*, 289–325. Berlin: Mouton de Gruyter.
Müller, Gereon. 2007. On deriving CED effects from PIC. Paper presented at the *30th Annual Colloquium of Generative Linguistics in the Old World (GLOW 30)*, Tromsø, Norway. [12–14 April, 2007]
Nunes, Jairo & Juan Uriagereka. 2000. Cyclicity and extraction domains. *Syntax* 3, 20–43.
Pesetsky, David. 1982. Paths and categories. Cambridge, MA: MIT dissertation.
Pesetsky, David. 1987. Wh-in-situ: Movement and unselective binding. In Eric Reuland & Alice ter Meulen (eds.), *The Representation of (In)definiteness* (Current Studies in Linguistics 14), 98–129. Cambridge, MA: MIT Press.
Pesetsky, David & Esther Torrego. 2001. T–to–C: Causes and consequences. In Michael Kenstowicz (ed.), *Ken Hale: A Life in Language* (Current Studies in Linguistics 36), 355–426. Cambridge, MA: MIT Press.
Picallo, Carmen. 1991. Nominals and nominalizations in Catalan. *Probus* 3, 279–316.
Ritter, Elizabeth. 1991. Two functional categories in noun phrases: Evidence from Modern Hebrew. In Susan D. Rothstein (ed.) *Perspectives on Phrase Structure: Heads and Licensing* (Syntax and Semantics 25), 37–62. San Diego, CA: Academic Press.

Rizzi, Luigi. 1997. The fine structure of the left periphery. In Liliane Haegeman (ed.), *Elements of Grammar: Handbook of Generative Syntax* (Kluwer International Handbooks of Linguistics 1), 281-337. Dordrecht: Kluwer.

Schneider–Zioga, Patricia. 2002. The case of anti-agreement. *MIT Working Papers in Linguistics, Proceedings of Austronesian Formal Linguistics Association (AFLA 8).* Cambridge, MA: MIT Press.

Siloni, Tal. 1997. *Noun Phrases and Nominalizations: The Syntax of DPs* (Studies in Natural Language and Linguistic Theory 40). Dordrecht: Kluwer.

Stabler, Edward. 1998. Acquiring languages without movement. *Syntax* 1, 72–97.

Svenonius, Peter. 1994. C-selection as feature checking. *Studia Linguistica* 48, 133–155.

Szabolcsi, Anna. 1989. Noun phrases and clauses: Is DP analogous to IP or CP? In John Payne (ed.), *Proceedings of the Colloquium on Noun Phrase Structure*. Manchester: University of Manchester, Department of Linguistics.

Zanuttini, Raffaella 1994. Re-examining negative clauses. In Guglielmo Cinque, Jan Koster, Jean-Yves Pollock, Luigi Rizzi & Raffaella Zanuttini (eds.), *Paths Towards Universal Grammar: Studies in Honor of Richard S. Kayne,* 427–451. Washington, DC: Georgetown University Press.

Zwart, C. Jan-Wouter. 2006. Local agreement. In Cedric Boeckx (ed.), *Agreement Systems* (Linguistik Aktuell / Linguistics Today 92), 317–339. Amsterdam: John Benjamins.

Anna Maria Di Sciullo
University of Quebec at Montreal
Department of Linguistics

P. O. Box 8888, Downtown Station
Montreal, Quebec H3C 3P8
Canada
di_sciullo.anne-marie@uqam.ca

Daniela Isac
Concordia University
Department of Classics, Modern Languages and Linguistics
H-663 1455, de Maisonneuve West
Montreal, Quebec H3G 1M8
Canada
disac@alcor.concordia.ca