Using the *Survive* Principle for Deriving Coordinate (A)symmetries*

John R. te Velde

This analysis examines the symmetries of coordinate structures, specifically how they can be generated in a minimalist, crash-proof grammar. I show that a phase-based model with selection of lexical items (LIs) before merge must have a matching operation across conjuncts, but this operation is prohibited by this model's own constraints. An alternative is presented that uses the *Survive* principle by which LIs are selected as needed for the merge operations of a coordinate structure. This selection process is guided and assisted by algorithms that map certain features from a leading conjunct to the next conjunct undergoing concatenation. With selection on demand and the mapping of features, coordinate symmetries can be generated that otherwise require global operations spanning all the conjuncts such as across-the-board movement. Additionally, the asymmetries that occur in coordinate structures are accounted for as consequences of additional mergers that do not require coordinate matching across conjuncts. Issues related to the limits of working memory can also be addressed.

1. Introduction

The development of generative syntactic theories within the minimalist framework has shown considerable progress in the last decade. Yet, some fundamental properties of coordinate structures remain a challenge. The most central property of coordination, symmetry, and its inverse, asymmetry, will be investigated here from the perspective of a minimalist, crash-proof grammar presented in Stroik (2007) and Putnam & Stroik (2008) called *Survive*-Minimalism. The derivational model they propose is outlined in section 4. In section 2 we review some important background to minimalist theory and coordinate structures, and in section 3 we consider a phase-based model as a point of reference for *Survive*-minimalism. In section 4 we derive some coordinate structures using this model, with special focus on how it handles symmetry and asymmetry. In section 5 we sum up the outcomes of section 4 and list areas that need further research.

2. Some background

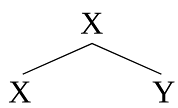
2.1 Assumptions about Merge, structure building and coordinate (a)symmetry

In the Minimalist Program of Chomsky (1995, 2000, 2005) the syntactic structure of a construction does not exist when lexical items (LIs) are selected from the lexicon. Rather, it is

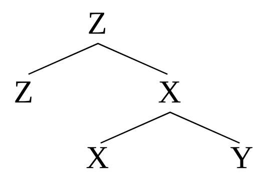
* Without the enthusiastic and inspired work of Mike Putnam and Tom Stroik, especially Mike's Crash-Proof Grammar conference, this paper would not have gotten off the ground. I am indebted to the conference participants, especially John Hale, Daniel Steely, Tom Stroik and Jan-Wouter Zwart for their comments, but most of all, again, to Mike for his many helpful suggestions. The remaining errors and problems stem from me only.

created when LIs merge in parallel, beginning at “the bottom”; further merge operations build upward from this initial structure:

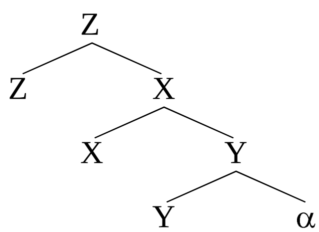
(1) a. after one merge operation



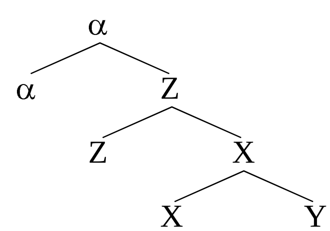
b. after a second merge operation[1]



c. conjuncts Z and Y sharing α



d. the same sharing relation, reverse ordering



This type of structure building and the syntactic relations that result from it directly predict certain properties of coordinate structures, for instance that two conjoined XPs, such as Z and Y in (1c,d), are not symmetric with respect to an element external to the coordinate structure, such as α in (1c,d), with X representing the coordinating conjunction (cf. work by Munn 1987, Johannessen 1998, among others). Support for the assumption that relations between conjuncts is asymmetric like those in (1c,d) comes from the binding relations in (2), or the subject-verb agreement in (3):[2]

(2) a. Sam$_i$ and his$_i$ dog went for a walk in the park (cf. sharing relation in (1c))
 b. *His$_i$ dog and Sam$_i$ went for a walk in the park (anaphor binding requires asymmetry)

(3) a. There is a man and his dog in the kitchen (cf. sharing relation in (1d))
 b. *There are a man and his dog in the kitchen

Other properties of coordinate structures, most notably their symmetries, are not predicted by this kind of structure building, if we assume that these symmetries are related to the structure in which they are found (cf. Camacho 2000 for a structure-based account of symmetric tense features):

[1] The merge operation that assembles the structures in (1) has another property of asymmetry in that either one or the other of the categories can project when merge occurs. Here I have chosen the left one for projection.

[2] This observation goes back at least as far as early work of Munn (1987) and forms the core thesis of Johannessen’s (1998) study of coordination. Alternate phrase structures can also produce the same results w.r.t. the asymmetry.

(4) a. Sam and his dog walked.PAST in the park and visited.PAST their friends
a' *Sam and his dog walked.PAST in the park and visit.PRES their friends
b. The park and the street both offer good walking areas
b' #The park and the bench both offer good walking areas
c. *Der Park und *(die) Bank bieten beide gute Sitzplätze* (German)
the.MASC park and the.FEM bench offer both good sitting-places
c' *Der Park und (der) Rasen bieten beide gute Sitzplätze* (*Rasen*=turf, grass)
d. The boys and (the) girls like to play in the park too
d' The man and (the) woman enjoy sitting in the park

As (4a,a') indicate, tense features must be symmetric; in (4b,b') the semantic features of the DPs sharing a verb+complement must be symmetric. The comparison between (4c) and (4c') indicates that the sharing of a determiner in German requires a gender symmetry. In (4d,d') the determiner may or may not be omitted, depending on the intended reading; thus, the interpretation depends on the presence or absence of sharing, which in turn depends on the presence of symmetric features.

Adding to the complexity of coordinate symmetry vs. asymmetry, there are constructions in V2 languages in which an asymmetric agreement relation is tolerated when the verb has fronted ahead of the two conjoined DPs that it agrees with, as in (5a); this asymmetric agreement is ungrammatical, however, in the SVO configuration in (5a'). Asymmetric agreement is required with Comp-agrement in (5b), while symmetric agreement is required when verbal agreement occurs in (5b'), both from van Koppen (2003):

(5) a. *So verrauschte Scherz und Kuss*[3]
so dissipated.3SG joke and kiss
'In this way joking and kissing dissipated'

a' *Scherz und Kuss verrauschten* / **verrauschte so*
joke and kiss dissipated.3PL / dissipated.3SG so

b. *Ik tink dat-st [do en Marie] dit wykein yn Rome west ha* (Frisian)
I think that.2SG [you.2SG and M].PL this weekend in Rome been have.PL
'I think that you and Marie were in Rome this weekend'

b' *Ha /*Ha-st [do en Marie] dit wykein yn Rome west?* (Frisian)
Have.PL /Have.2SG [you.2SG and M].PL this weekend in Rome been
'Were you and Marie in Rome this weekend?'

[3] J. W. Goethe, from "An den Mond."

As the data indicate, both symmetric and asymmetric agreement relations are possible in coordinate structures, and the configurations in which each occurs is not 100% consistent, such that the one always occur in one type of configuration and the other in the inverse configuration. These findings undermine the hypothesis that symmetry and asymmetry are always structurally determined.

2.2 Further data indicating coordinate symmetry

In addition to the lack of a reliable correlation between structure and (a)symmetry, a number of symmetries in coordinate structures are clearly not related to structure, and thus not to the derivational principles and operations of the narrow syntax per se. These include the symmetries resulting from the matching or pairing of lexical features in corresponding conjuncts, as in (4b,d,d'). Further examples are given in (6):

(6) a. #The woman and the shopping bag went to the mall
 a' The woman and her shopping bag always go to the mall together
 b. #The car sped down the street and leaped over the railing
 b' The boy sped down the street and leaped over the railing

In (6a) the conjunction is ill-formed because the two subjects do not both have a semantic feature, possibly [+volitional], required of the subject of 'went'. In (6b) the two verbs cannot be conjoined and share the subject 'the car' because this subject is not capable of leaping.

The features in (6) that create semantic symmetry are part of the feature matrix of the respective LIs when selected from the lexicon and are thus not created or valued in the narrow syntax. Any derivational grammar, minimalist or otherwise, must account for the matching of the lexical – especially the semantic – features required for well-formed conjunction, as often determined by whether they can share a lexical item, that is, co-exist in parallel agreement relations between two or more conjuncts and a single element. This parallel co-existence in matching coordinate structures requires what will be referred to here as coordinate feature matching.[4]

In the next section we turn to some minimalist accounts of coordinate symmetry that have Phase Theory as a basis. We will use these as a reference point for illustrating the advantages that the *Survive* model outlined in section 4 offers for deriving coordinate (a)symmetries.

[4] The symmetry requirements of coordinate relations – whether syntactic or semantic, but especially the syntactic symmetries – suggest that the coordinate structure itself is also symmetric. This point has been discussed extensively in the literature, with studies generally leaning one way or the other, i.e. either arguing that coordinate symmetries are all semantic and advocating syntactic asymmetry (the leading example Johannessen 1998), or attempting to accommodate them in a syntactic account that explains the symmetries in terms of syntactic relations or operations (Camacho 1997, 2000; Johnson 2002, among others).

3. Phase-based accounts of coordinate symmetry

3.1 Some assumptions of Phase Theory

We consider first an assumption about the selection of LIs from the lexicon in the phase models. In the previous section we noted that lexical features come with the LI when it is selected, regardless of the model used. Presumably the lexical features of LIs selected for conjuncts must be scanned at selection in order to determine whether they have the features needed for coordinate symmetry, as required by an agreement relation they enter into with a shared element, as illustrated in (1) – (6).

The exact procedure by which LIs are selected can make a significant difference as to how they are matched for coordinate symmetry. Chomsky (2000: 100-101) makes the following assumption about selection in the context of considering ways to address computational complexity and optimal design:

> Is it also possible to reduce access to Lex [as with [F], features], the second component of the domain of L? The obvious proposal is that derivations make a one-time selection of a *lexical array* LA from Lex, then map LA to expressions, dispensing with further access to Lex. That simplifies computation far more than the preceding steps. If the derivation accesses the lexicon at every point, it must carry along this huge beast, rather like cars that constantly have to replenish their fuel supply. Derivations that map LA to expressions require lexical access only once, thus reducing operative complexity in a way that might well matter for optimal design.

If access to the lexicon is restricted in the way Chomsky proposes, then we must assume for coordinate structures that this selection process involves a matching mechanism so that the LIs of paired conjuncts that enter into an agreement relation with a shared element have the required symmetries, as in (6b') 'the boy [sped… and leaped…]'.[5] A certain complexity, or at the very least a division of labor, ensues with this one-time selection: it can only guarantee matching *lexical* features; the symmetry of the verb tense in (6b') must be guaranteed in the narrow syntax. Thus, Chomsky's strategy for constraining the grammar results in a two-layered coordinate feature matching procedure. In section 4 we will consider whether this is the best strategy.

Let us turn now to an assumption about phases themselves. When a phase is complete, it is transferred to the interfaces, thus relieving the narrow syntax of the burden of holding it in working memory until the rest of the derivation is complete. This "multiple spell-out" approach to handling derivations, as advocated particularly by Uriagereka (1999) and adopted by Chomsky (2005), raises a number of questions, particularly with regard to coordinate structures. Boeckx (2007) addresses several of them; he points out that the computational system must be

[5] Stroik (2008) discusses at length – apart from any considerations regarding coordinate symmetry – the implications of "blind" selection from the lexicon. He argues that it results in an extremely high number of merge possibilities that are neither manageable by C_{HL}, nor desirable in a minimalist grammar.

able to retrieve previously spelled-out material to provide a complete, coherent surface string and thus can't ignore or delete spelled-out elements. Boeckx (2007: 418) gives Norbert Hornstein credit for pointing out that interfaces appear to examine the internal content of full representations for specific processes. For example, the semantic component needs to see multiply spelled-out chunks for pronominal binding, and "PF quite possibly needs full clauses to determine intonational patterns, such as falling intonation ending up with a declarative (statement) or rising intonation yielding an interrogative …" (Ibid.).

Accounts using phases and multiple spell-out also leave unanswered how certain coordinate structures can be derived. In a phase-based approach coordinate TPs need to be derived and spelled out individually and sequentially, unlike e.g. conjoined DPs or PPs which, since they are not phases, lend themselves to parallel derivation and possibly to matching as well (but see discussion in §3.2 of Citko's (2005) proposal for conjoined TPs sharing a *wh*-element). Structures made up of CP conjuncts have an additional phase; they first require the spell out of the *v*P phase, creating a TP, and then the spell-out of the CP phase. At this point LIs for the second CP conjunct are extracted as subarrays, first the elements for the *v*P phase (which is then spelled out) and then those for the CP phase. After spell-out, however, the first *v*P phase of the first CP conjunct is no longer available for matching with the second *v*P phase. Matching is the basis for valuing the syntactic features so that the required coordinate symmetries are generated. Evidence that conjoined TPs must meet certain syntactic symmetry requirements comes from constructions like those in (7):

(7) a. First Sam walked.PAST the dog, then he cooked.PAST dinner
a' *First Sam walked.PAST the dog, then he cooks.PRES dinner
b. Sam prepared$_i$ the main dish, and Sue e_i the dessert
b' Sam prepared the main dish, and Sue **e* (=baked) the pie
c. Their guests generally prefer e_i, and Sam and Sue also enjoy [wine with the meal]$_i$
c' #Their guests prefer e_i, but Sam and Sue never buy [wine with the meal]$_i$

The symmetry requirement in (7a) can be narrowed down to the tense inflection and the ϕ-features, both of which must be valued in the narrow syntax. In (7b) the symmetry involves not only tense and ϕ-features, but also the lexical features of the two finite verbs. In (7c) the match arguably involves more than one LI, i.e. the two finite verbs 'prefer' and 'enjoy' as well as the adverb 'also', and the shared elements in brackets must also be part of matching. As the ill-formed (7c') makes quite clear, matching must be global, i.e. essentially all of the LIs of the first conjunct must be matched with those in the second.

Earlier we noted that the matching of lexical features must precede the narrow syntax in a phase-based model, if selection of the LIs proceeds as Chomsky (op. cit.) proposes. This type of selection eliminates the possibility of matching lexical and syntactic elements globally. Syntactic features must be matched separately in the narrow syntax, and for this matching, the relevant features of the conjuncts must be present in the narrow syntax. But if the conjuncts are TPs or

CPs, then multiple spell-out has eliminated the first TP or CP conjunct from narrow syntax before the next one is derived and thus makes it impossible to match the required syntactic features before transfer to the interfaces, at which point unfulfilled feature matching for coordinate symmetry causes a crash.[6]

Conjuncts unaffected in this way by multiple spell-out, for instance DP conjuncts such as those in (4), also present a problem for a phase-based grammar without either the adoption of certain assumptions about phrase structure and syntactic categories (Camacho 2000) or the operation Copy (Frazier & Clifton 2001). In the next section we turn to a proposal for an alternative phrase structure resulting from an assumption about merge as an example of a strategy within a phase-based model to address a type of coordinate symmetry.

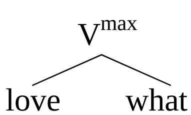
3.2 Citko (2005): alternative way to merge SOs

Addressing a construction type that has inspired a long history of syntactic analysis, Citko (2005) proposes "Parallel Merge" to generate the relations that exist in across-the-board (ATB) *wh*-questions of the sort in (8), which require coordinate sharing – and therefore coordinate symmetry – not unlike the constructions considered earlier:

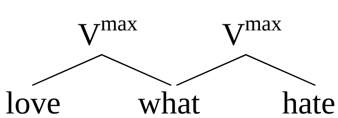
(8) What did Sam love *t* and Sally hate *t* ?

Citko's Merge and Parallel Merge applied in sequence results in the multiple-dominance and sharing relation in (9b) so that *what* is dominated by two nodes and shared by two verbs:

(9) a. Merge *love* and *what*; project *love*:



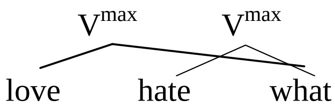
b. Parallel Merge *hate* and *what*; project *hate*



The linearization of the elements in (9b) requires crossing branches:[7]

[6] Matching in the narrow syntax in a derivational grammar may not be possible without major modification of assumptions on phrase structure and representation. In essence, the grammar would need to accommodate 3D or parallel representations along the lines of proposals by Grootveld (1994), Moltmann (1992), Muadz (1991) and Wesche (1995), or it would need to assume a form of Copy, such as proposed by Clifton & Frazier (2001) and Frazier (2008). The question remains: How derivational and crash-proof are such grammars?

[7] Citko points out that the relations resulting from Parallel Merge do not cause problems for the linearization requirements of syntactic derivation, as formulated by Kayne (1994) in his Linear Correspondence Axiom (LCA), because the LCA, following Chomsky (1995), does not apply until the derivation reaches the SM interface, i.e. its

(10) 

The merit of Citko's proposal is that Parallel Merge requires no new operations or principles, it simply combines the properties of Internal and External Merge (IM and EM) proposed by Chomsky (2005). It is like EM in that it involves two distinct rooted objects; it is like IM in that it combines the two by taking a subpart of one of them.

Thus, Citko's proposal can account for ATB *wh*-questions, if we assume that "taking a subpart" and Parallel Merge in general are operations that conform to assumptions about Merge and Copy.[8] Even if we assume this is the case, Citko's proposal would still require an additional linearization rule for constructions such as those seen earlier with conjoined DPs that do not have IM available to them for lining up the LIs at the PF interface. These constructions do not require any overt displacement like *wh*-movement to linearize the LIs; therefore, the only option for linearizing them after Parallel Merge has generated the sharing relations would be to propose some kind of NP-movement operation that accomplishes this. However, the generative literature provides no arguments or evidence that a realignment operation is necessary in the derivation of, for instance, the constructions in (11):

(11) a. The woman and her purse went to the mall
b. The car ran the light and flipped over the railing

In other words, what is transferred to the PF interface follows directly from the Merge operations that construct (11a,b); that is, neither the subject-verb relation in (11a) nor the verb-object relation in (11b) requires an overt realignment that is comparable to the *wh*-movement required in (8). Yet this is what would be required if Merge and Parallel Merge constructed (12a,b); the result would be:[9]

---

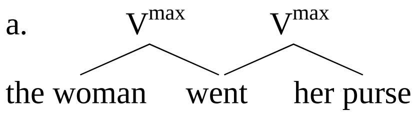
affects remain invisible. As long as Internal Merge (IM) repositions the shared object *what* before the SM interface, the derivation will not crash.

A point left unaddressed by Citko is how Parallel Merge is unified with transfer to the interfaces. Presumably (8) requires more than one phase. Thus, Parallel Merge must somehow "cancel" transfer until the merge operation is complete. This appears to be an ad hoc mechanism.

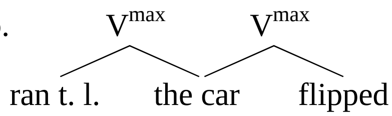
[8] Chomsky (2007) states (n. 10): "Citko argues that parallel Merge is 'predicted' as IM is, but that is not quite accurate. It requires new operations and conditions on what counts as a copy, hence additional properties of UG."

[9] An objection to this argument might be that in the narrow syntax there is no linear relation between the SOs. We can counter this objection, however, following this line of reasoning: Merge holds to the head parameter of a particular language. If Merge were not sensitive to this parameter, i.e. were completely non-linear, then overt movement (IM) would be required every time orderings like *went – her purse* and *ran the light – the car* were transferred to the PF interface. Presumably such orderings are just as likely as their reverse in a completely non-linear narrow syntax that is insensitive to the head parameter of the language for which it is being used.

(12) a.



b.



These structures would require further linearization before they could meet the PF interface; yet, unlike the linearization of the structure (9b) in (10), there would be no syntactic motivation for it.

Beyond these limitations of Citko's proposal, there is the more general one that it has no operation or mechanism that matches the features of conjoined SOs; thus, it is not able to assure the symmetry of syntactic inflections as required, for instance, with the finite verbs in (11b), or the symmetry of lexical features in (4) and other constructions we have seen. Thus, limitations and problems remain with even this "enhanced" phase-based approach.

We turn in the next section to the *Survive* model of Stroik (2007) and Putnam & Stroik (2008) to investigate whether the problems of the Citko proposal and other phase-based models can be avoided.

4. The *Survive* principle and its application to coordinate (a)symmetry

4.1 The Survive *model and its advantages*

A single principle guides the concatenation of SOs in the *Survive* model without resort to IM or phases, which Stroik and Putnam argue do not make a minimalist grammar more strictly derivational or crash-proof. For these and a number of other reasons, they employ a version of Merge that may target a given LI more than once, depending on its feature matrix. For instance, in (11), *who* is targeted by Merge three times (i.e. is remerged twice) because it has three features, each of which must be checked by a different head. Each time a SO has a remaining unchecked feature, it "survives" and remains active in the numeration and remerges for checking (from Putnam 2007: 14 - 15):

(13) Who cares?

a. Merge {who, cares} → who cares
b. Survive {who}
c. Merge {T, {**who**, cares}} → T **who** cares
d. Remerge {who, {T, {**who**, cares}}} → who T **who** cares
e. Survive {who}
f. Merge {C, {**who**, {T, {**who**, cares}}}} → C **who** T **who** cares
g. Remerge {who, {C, {**who**, {T, {**who**, cares}}}}} → who C **who** T **who** cares

In (13) *who* first merges with *cares* (in a) for checking its θ-feature. It survives and remerges (in d) to check its ϕ-features with T and survives again. Finally, it remerges a second time (in g) to check its Q-feature with C. Note that other merge operations intervene between the remergers of *who* to make available the heads that check the remaining features of *who*. In this way a syntactic structure is built from the bottom up, as are structures using Phase Theory. Note, however, that the entire numeration remains active until all features have been checked, i.e. nothing is transferred to the interfaces until the entire derivation is completed in the narrow syntax.

What advantages does this type of merge operation offer for the derivation of the coordinate structures we have seen? We recall that a major shortcoming of phase-based approaches for deriving coordinate structures was the lack of a clear derivational or checking mechanism for coordinate symmetries, either at selection (for lexical features) or in the narrow syntax (syntactic features); in fact, the one-time selection of LIs for merge and the multiple spell-out of merged structures create major obstacles to the matching necessary for coordinate symmetries, as we have seen, to the detriment, in my estimation, of fundamental minimalist objectives. The *Survive* model's method of selection and feature checking provide a possible solution to these problems. For considering this method, some additional aspects of a typical derivation in the *Survive* model that aren't apparent in (11) must be outlined.

In contrast to the form of selection described by Chomsky (2000: 100) cited above, the *Survive*-model uses an "on-demand" or dynamic selection process, in contrast to one-time selection, that does not require extraction of LIs from a lexical array or subarray when they are merged. Rather, the LIs to be merged are selected directly from the lexicon. Furthermore, when merged, mapping algorithms are established for pronoun-antecedent relations and other long-distance relations that cannot be captured in a local feature-checking relation. Such mapping algorithms, along with dynamic selection, offer a means to first generate and then check coordinate symmetries. Consider the derivation of the German construction *Frank ruinierte den Laptop, aber er behielt das Zubehör* 'Frank ruined the laptop, but he kept the accessories' in (14):

(14) Deriving *Frank ruinierte den Laptop, aber er behielt das Zubehör* in the *Survive* model)

(each → is a selection; s.m.=semantic matrix; WB=WorkBench[10], C=conjunct; c/c=compare/contrast)

L E X I C O N

a. Merge {Frank, ruinierte} → Frank ruinierte - - - - - - [$_{F}$ g, n], [$_{r}$ s.m.] - - - > - - - - - - -

a' Merge in WB {den, Laptop} → den Laptop[11]

b. Survive {Frank} b' Survive in WB {den}

c. Merge {Fr. ruinierte {den Laptop}} → Fr. ruinierte den Laptop - - [$_{DP}$ θ_{th}][12] - >

c' Merge in WB {D {**den** Laptop}}→ D den Laptop

d. Merge {*v*, {Fr. ruinierte den Laptop}}→ Fr. ruinierte den Laptop -[$_{DP}$ ACC][13]-

e. Merge {T, {Frank ruinierte den Laptop, }} → T Frank ruinierte den L

f. Remerge {**Fr**, **ruin.**[14] {T Frank ruinierte den Laptop}} - - - [ϕ, T] - > - -

→ Frank ruinierte den Laptop

(C-2 begins; merge proceeds as in C-1, with the aid of algorithmic input.)

g. Merge {er, behielt} → er behielt - - - - - - [$_{er}$ g, n], [$_{beh}$ (c/c Vs)]- - - - - < - - - - - - - -

g' Merge in WB {das Zubehör}} → das Zubehör

h. Survive {er} h' Survive in WB {das}

i. Merge {er behielt {das Zubehör}} → er behielt das Zubehör - - - - - < - - - - - - -

i' Merge in WB {D {**das** Zubehör}}→ D das Zubehör

j. Merge {*v*, {er behielt das Zubehör}}→ er behielt das Zubehör - - - < - - - - -

k. Merge {T, {er behielt das Zubehör}} → T er behielt das Zubehör

l. Remerge {**er**, **behielt** {T er behielt das Zub.}} → Er behielt das Zub. <-

m. Merge {**aber**, {er behielt das Zubehör}}- - - - - - - - (c/c C-1 & C-2[15]) - -

→ aber er behielt das Zubehör

n. Merge (conjoin): {Frank, ruinierte, den Laptop, {aber, er, behielt, das Zubehör}}

→ Frank ruinierte den Laptop aber er behielt das Zubehör

[10] "WorkBench" is where DPs are assembled in the *Survive* model. Putnam (2007: 99) explains the need for the WB this way: "...if a *Derivation* D is restricted to the computation of an event (expressed in the V-*v* projections) and its temporal and speech event coordinates (expressed in the T-C projections), then D must be built from two different sources of syntactic objects – the *Numeration* N and a workspace (WS) [= WB] not contained in D."

[11] The parsing formalisms differ somewhat from those in Putnam (2007). Boldface is used to single out an LI that (re)merges; thereafter it occurs in normal typeface; category labels are not repeated once they have been introduced into the numeration. In like manner, the "copies" of LIs remerged are not repeated after a remerger.

[12] I leave open if and when the V2 structures have a clause-final [V].

[13] In a more detailed outline, *Frank ruinierte den Laptop* would remerge in the *v*P after [*v*] has merged; the same with *er behielt das Zubehör* in C-2 (assuming the *v*P is required in the *Survive* framework).

[14] [T] triggers the remerger of both *Frank* and *ruinierte* in C-1 and *er behielt* in C-2 because it has features for both agreement and tense. In a more detailed outlined, each LI would remerge separately.

[15] Comparison and contrast are presumably required for the selection of *aber* 'but' as the coordinating conjunction.

The mapping algorithms indicated on the right with broken horizontal and solid vertical lines accomplish two things: i) they guide selection from the lexicon by mapping the features of LIs in the numeration with the features of potential LIs for the second conjunct, and ii) they map a particular feature from one conjunct that has completed derivation to another that is being assembled. Thus, algorithms assure certain symmetries and eliminate the randomness of selection that would otherwise generate numerations that crash at the interfaces.[16]

In the example construction, first the number and gender features of *Frank* and the semantic feature matrix of *ruinierte* 'ruined' must be matched with the same features of corresponding LIs selected for the second conjunct. Later, in step l the ϕ-features of *Frank* and the tense features of *ruinierte* are mapped to their equivalents in C-2. The features of the LIs *er behielt* 'he kept' must match those of their counterparts in the first conjunct for grammaticality and semantic well-formedness; they represent the minimum required by coordinate symmetry.[17]

Likewise, the Case features of the DPs *den Laptop* and *das Zubehör* 'the accessories' must match for similar coordinate symmetry requirements. If one or the other DP had a Case inflection other than [ACC], the construction would be ungrammatical even without conjunction, if the verbs remained the same. Selecting another verb for the second conjunct that requires a dative object, for instance, results in non-matching Case features that render the construction ill-formed:[18]

(15) #*Frank ruinierte den Laptop, aber er dankte dem Besitzer*
F ruined the laptop, but he thanked the owner

Other verbs that require a dative object, such as *ähneln* 'resemble', *begegnen* 'encounter', *dienen* 'serve', *drohen* 'threaten', *fehlen* 'lack', etc. produce even worse results in this construction (some completely ungrammatical). Accordingly, both the verbs and their objects must be matched, as the mapping algorithms in (14) indicate, and both lexical and syntactic features must match. We note that in the *Survive* model both types of features are matched via a single mapping algorithm.[19] Thus, the division of labor seen earlier in the phase-based model does not

[16] Putnam and Parafita (this volume) outline how the *Survive* principle can be applied to the selection and merger of LIs in a grammar that involves code switching. Their proposal addresses simplex constructions and thus presents a picture of derivational operations that precede the assembly of coordinate constructions, explored here.

[17] We note that in the *Survive* model there is no distinction between [+interpretable] and [-interpretable] features as to where what features are checked. This distinction is problematic for a number of reasons. In German, for instance, Case features cannot be [-interpretable] and thus eliminated from the numeration by checking because they must be realized as inflections in PF.

[18] A great deal of contextualization might rescue this construction. Hence, we cannot assume that all conjunctions of DP_{ACC} and DP_{DAT} will be ill-formed or ungrammatical, i.e. inter-clausal or pragmatic factors/features can rescue, i.e. render adequately symmetric, some such conjunctions.

[19] An anonymous reviewer pointed out correctly that tense mismatches between verbs of conjoined clauses are generally more severe than ACC-DAT mismatches between conjoined DPs like those above. The same is not true, however, with NOM-ACC mismatches, which result in total ungrammaticality:

(i) Der Vater ging spazieren, aber sein / *seinen Sohn spielte Billard
the father went walking buts his-NOM / his-ACC son played billiards

exist in this model; this simplification of the grammar model has the advantage for coordinate feature matching of allowing a single matching operation for both feature types. Furthermore, the mapping algorithms, because they extend from one clausal conjunct to another, make global matching feasible.[20] The lack of phases and multiple spell-out in the *Survive* model means that the problems associated with them are avoided.

In the next section we turn to advantages of the *Survive* model related to computational efficiency.

4.2 Working memory and C_{HL} in the derivation of coordinate structures

As we have just seen, the *Survive* model makes no distinction between lexical and syntactic features when it comes to their matching for coordinate symmetry; both can be mapped via an algorithm during concatenation. In a phase-based model, the two types of features must be handled in their respective areas of the grammar; the lexical features must match at selection (since the narrow syntax cannot read them), and the syntactic features in the narrow syntax. This added complexity of a phase-based model comes in addition to the one already noted, namely that because all LIs are selected prior to computation in the narrow syntax, the selection of LIs for conjuncts must proceed in parallel planes or in 3D mode for matching to occur (cf. Moltmann 1992, Muadz 1991, Wesche 1995 for proposals along these lines). By contrast, in the *Survive* model with on-demand selection, an LI that constitutes a second (or third, etc.) conjunct is matched with the previous conjunct via a mapping algorithm when it enters the numeration. In a phase-based model, matching must occur again in the narrow syntax for the syntactic features of paired conjuncts to achieve the required symmetry.

This second matching procedure raises an important question: Where are the SOs of the first conjunct located when the second conjunct is derived? If both conjuncts are in the same phase, such as conjoined DP subjects, then nothing has been transferred to the interfaces when the second conjunct is assembled.[21] If, however, the conjuncts are located in two different phases, as is the case in (14), then matching between the interfaces and the narrow syntax – where the second conjunct, located in the next phase, is being assembled – must proceed. This kind of matching would require the derivation to look back from the interfaces into the narrow syntax, and thus by definition from one phase into another. Look-back, however, is prohibited in phase-based models, as formulated in Chomsky's (2000: 108) Phase Impenetrability Condition. This problem obviously does not occur in the *Survive* model since it does not employ phases.

There is nevertheless a problem for the *Survive* model that relates to computational efficiency in the derivation of coordinate structures:

---

[20] Whether an object is suitable for a verb selected must also be checked, but this can be accomplished when the verb and its object merge (in a local relation) and thus does not require an algorithm unless it involves a coordinate relation in which the second conjunct, for purposes of cooordinate symmetry, must has an equivalent verb-object relation.

[21] Even if the two conjuncts are located in the same phase, the matching of syntactic features is not a trivial matter. Instead of matching, a copy operation might get the job done (cf. Frazier & Clifton 2001, te Velde 2005b).

(16) *Memory load problem*: How can multiple matrix-clause conjuncts be merged and matched, if all conjuncts remain active in the numeration (and are thus retained, i.e not transferred to the interfaces) until derivation (and matching) is completed?[22]

This problem is the counterpart to the look-back problem of a phase-based model addressed earlier in that it is created by not transferring the derivation to the interfaces. We consider below how it is manifested in the German construction in (17):

(17) Multiple conjoined TPs with a shared *wh*-element (copies in bold)

Welches Buch hat Frank gern gelesen, Peter nicht verstanden, Lars nicht gekauft ...?
which book has F gladly read, P not understood, L not bought
'Which book did Frank like reading, Peter not understand, Lars not buy…?'
[$_{CP}$ Welches.Q Buch.ACC [hat.3SG [$_{TP}$ Frank.NOM **hat** [**welches Buch** gern gelesen
[$_{TP}$ Peter.NOM **hat** [**welches Buch** nicht verstanden
[$_{TP}$ Lars.NOM **hat** [**welches Buch** nicht gekauft]]]]]]]...

Some preliminary observations: The phrase structure in (17) implies a reduced ATB derivation by which *welches Buch* 'which book' is fronted from each VP to the Spec,CP position. It implies a *reduced* ATB derivation in that this element does not make "intermediate stops" in the Spec,CP position of its own clause and then progress upward to the Spec,CP of the next clause, etc. until it reaches the highest one available. This derivation is not tenable in current minimalist syntax for at least two reasons: 1) It violates Phase Theory, which doesn't allow movement from one phasal unit to another without use of the edge (Spec,CP in this case); 2) It requires head movement for raising *hat* to the C° position; for principled reasons, head movement is generally assumed to be non-existent (cf. Chomsky 2001).

Parallel Merge in Citko's (2005) proposal would, will some minor modification, be able to derive this construction. However, this proposal, as we have seen, cannot derive some of the other constructions under consideration here; for this reason we will leave it aside here.

Deriving this construction using the *Survive* principle requires a certain assumption about bottom-up derivation, as it applies to coordinate structures of this sort. My line of argument is that a matrix clause (whether a TP or a CP) defines the limit of a derivation. For the conjunction of matrix clauses this means that the "bottom" of the derivation is the bottom of the first matrix

[22] This problem is not restricted to coordinate structures. Any sentence can theoretically be expanded by *n*-number of mergers of embedded clauses. The memory load problem with the derivation of embedded clauses can be dealt with uncontroversially by transferring clauses (or portions of them, phases) to the interfaces as they are merged, since they, unlike conjuncts, do not have to be retained in the numeration for matching purposes (though potentially other syntactic relations may prevent transfer). Related problems arise from the "bottom-up" approach assumed in much minimalist work: Either look-ahead or the management of working memory are required, see below.

clause in a string of conjoined matrix clauses. No condition or rule needs to be formulated for this assumption, as stated in (18), to apply in the *Survive* model:

(18) A matrix clause defines the limits of a single derivation; the conjunction of *n*-number of matrix clauses consists of *n*-number of such derivations that thereafter become the syntactic object of 'conjunction', a "complex" derivational operation that merges the output of a simplex derivation with the output of a preceding symmetric simplex derivation.[23]

For the derivation of (17), this means that first the matrix clause *Welches Buch hat Frank gern gelesen* 'Which book did Frank like reading' will be derived first (cf. (13) for an example of how a *wh*-question is derived using the *Survive* principle). The entire sequence of derivations for (17) looks like this:

(19) Sketch of the derivations required for (17) using the *Survive* principle

a. Derive *Welches Buch hat Frank gern gelesen* a la (13) with the addition of WB merger for the DP *welches Buch*

b. Derive ***Welches Buch hat*** *Peter nicht verstanden* with **copies** of *welches Buch* and *hat* mapped to the CP domain via an algorithm

c. Conjoin the merged matrix clauses[24]

d. Repeat b for the next conjunct ***welches Buch hat*** *Lars nicht gekauft*

e. Conjoin with the previous clauses

f. Elide at the left edge, eliminating the redundant CP domains[25]

The copies of *welches Buch* 'which book' and *hat* 'has' mapped to the CP domain of the second and third conjuncts contain all the syntactic and semantic features of the originals. Lacking are

---

[23] This assumption addresses the dilemma faced by the bottom-up derivation of coordinate structures: If merge begins at the bottom of constructions like those in (i), then look-ahead is required into the previous clause:

(i) a. Sam bought the beer, but he forgot the milk

b. The car sped down the street, and then it flipped over the railing

If derivation proceeds as proposed in (18), then look-ahead is coordinated with the logic and temporal sequence of the construction. See also the discussion of *Link!* in §4.3 and 5.

[24] Since this construction is a hypothetical, "limitless" conjunction of CPs, no coordinating conjunction is indicated and need not be merged as long as the appropriate prosodic features are in place.

[25] For an analysis of left-edge deletion in the conjunction of CPs and a proposal using a phase-based approach, see te Velde (2005a,b). This proposal would be compatible with the *Survive* model on the assumption that the mapping algorithms established at (re)merge for coordinate matching purposes are also suitable for the "deletion" (marking for non-realization in the PF component) of redundant elements. Further work is needed.

the (morpho-)phonetic features, which have been "deleted" by "coordinate ellipsis," an economy-based operation that makes non-realization of phonetic features possible when the deletion site is c-commanded by [&] and recovery is possible in LF when the coordinate symmetry requirements have been met. A more detailed discussion would take us too far afield, and is already available in Chapter Four of te Velde (2005b). Relevant here is that remerge is not required for this coordinate ellipsis operation itself. Further work is needed to determine whether this operation is fully compatible with derivation by (re)merge as conceived in *Survive*-minimalism.[26]

In order for conjunction in steps c and e to proceed, it is necessary to have all of the clauses merged up to those points active in the numeration, if we assume that the conjunction of matrix clauses is a syntactic merge operation. If it were not, it remains a puzzle how a coordinating conjunction could be merged, which is required before the last of a string of conjuncts in German, English and many other languages. But assuming that conjunction is a syntactic merge operation brings us back to the question posed in (16): How can *n*-number of conjuncts (of *any* size) be managed in working memory? The dilemma of managing a potentially infinite number of conjuncts while maintaining conjunction as a syntactic merge operation forces us to look beyond the syntactic mechanisms of the *Survive* model presented so far. This is the objective of the next subsection.

4.3 Link!

The primitive function *Link!* in the *Survive* model is described as follows:

(20) *Link!* (Putnam 2006: 6)

If XP is a syntactic object (SO) bearing the interpretable feature α, and a syntactic head Y˚ enters the derivation bearing the interpretable feature α, XP will remerge into the structural position immediately dominating Y˚ [SpecYP] at which point the matching features will amalgamate and be interpreted at the interfaces.

The central idea with *Link!* is to have features and concatenated SOs be checked for interpretability as soon as appropriate configurations arise in the narrow syntax. One of these "appropriate configurations" in (19) is the first CP derived. Upon completion of its derivation, *Link!* determines on the basis of the Q-feature and the finite verb that it can be interpreted as a matrix *wh*-question. *Link!* thus brings the derivation closer to conjunction in that its output is identifiable as a SO that can be conjoined with another of its kind that is concatenated immediately thereafter.

---

[26] It is important to note that the derivation of each conjunct, with all the required mergers for θ-role and Case assignment and subject-verb agreement, are completed in each conjunct, with the relevant features mapped to the next conjunct as in (14) for purposes of coordinate symmetry.

Link! does not by itself solve the working memory dilemma, however. For that *Link!* must extend beyond the interpretive component, call it LF for convenience, to a new area of memory, call it short-term memory. I will assume here that short-term memory is the area of human memory where meaningful (interpretable) utterances can be stored, in contrast to working memory where linguistic computations are performed. If C_{HL} indeed has access to short term memory via *Link!*, then working memory is in fact relieved of a burden with this linkage. Further research will need to verify whether this transfer to short-term memory actually occurs. If so, a potential answer to the question in (16) is available.[27]

Relieving working memory of the burden of holding a matrix clause until another is derived is of no help to the conjunction of the matrix clauses, however, unless the first matrix clause, now in short-term memory, can be recalled by the narrow syntax for conjunction purposes. *Link!* as outlined above does not provide for this kind of recall. For this, a system is needed in which transfer goes both directions between the narrow syntax and the interfaces, i.e. they are symmetric in this regard. Boeckx (2007) discusses a system in which Spell-Out is eliminated and the interfaces have access to the narrow syntax and are not passive recipients of information. He points to a symmetric relation between the narrow syntax and the interfaces that is broken – made asymmetric – or "hidden by time, i.e. disrupted by derivational processes" (Ibid., p. 422). This is precisely the situation with the derivation of conjoined matrix clauses: The derivation extends over the entire coordinate structure, but it is broken by the time it takes to assemble each clause, with each read by the interfaces at the appropriate point. What is needed to make this system ideal for the derivation of coordinate structures is a recall ability; that is, the narrow syntax is able to take advantage of the fundamental symmetry in the narrow syntax-LF system and recall features as needed for coordinate feature matching. This operation is part of one derivation that consists of subparts, matrix clauses that stand in a coordinate relation to each other, that is broken or hidden by time, i.e. by the derivational processes involved with each one, but emerging in the end as one construction by conjunction. Several non-trivial assumptions about the principles of derivation – most notably look-back – must be investigated here.

4.4 Coordinate Asymmetry in the Survive *model*

So far we have considered only how the symmetries of coordinate structures can be handled in the *Survive* model. As Johannessen (1998) has made clear, asymmetries occur quite commonly in coordinate structures and cannot be ignored.[28] Some simple examples of asymmetric subject-

---

[27] The transfer of a phase to LF (the Conceptual-Intentional Interface in Chomsky's terminology) also relieves working memory of a burden. The difference between the phase-based model and the *Survive* model is that the problem with executing the matching required for coordinate symmetry – accomplished via mapping algorithms in the *Survive* model – has not been eliminated in the phase-based model with the alleviation of the working memory burden.

[28] I differ with Johannessen on the premise that asymmetry is the defining property of coordinate structures, i.e. that a theory of coordination should revolve around asymmetry. It is precisely the symmetry of coordinate structures that distinguish them from subordinate structures. This distinction does not force us, however, to use 3D or parallel

verb agreement that resemble many others because of their C/VSO configuration are (b from van Koppen 2003):

(21) Examples of asymmetric coordination in C/VSO configurations

a. *So verrauschte Scherz und Kuss*[29] (German)
so dissipated.3SG joke and kiss
'In this way joking and kissing dissipated.'

b. *Ik tink dat-st [do en Marie] dit wykein yn Rome west ha* (Frisian)
I think that.2SG [you.2SG and M].PL this weekend in Rome been have.PL
'I think that you and Marie were in Rom this weekend.'

Van Koppen analyzes the asymmetry in (20b) as a result of a probe-goal relation between the complementizer – which has inflection for agreement – and the first subject. Using the *Survive* model, we can assume a very similar relation, but one that does not induce Internal Merge, as does van Koppen's. Rather, the merging of the complementizer *dat* 'that' in Frisian and the remerging of the first subject *do* 'you' results in a checking relation; *do* 'you' survives for remerge and checking with *dat* 'that' because both have ϕ-features. The second, conjoined subject *Marie* does not require a checking relation with *dat* 'that' because it has no such features. In the event there is no Comp-agreement because a verb sits in C° as in (22a), a symmetric agreement relation is required, and as (22b) indicates, the SVO configuration also requires symmetric agreement:

(22) a. Ha / *Ha-st [do en Marie] dit wykein yn Rome west? (Frisian)
Have.PL/Have.2SG [you.2SG and M].PL this weekend in Rome been
'Were you and Marie in Rome this weekend?'

b. [Do en Marie] ha / *ha-st dit wykein yn Rome west (Frisian)
[you.2SG and M].PL have.PL /have.2SG this weekend in Rome been
'You and Marie were in Rome this weekend.'

Other West Germanic data indicate that the VSO configuration doesn't guarantee either symmetry or asymmetry; rather, the agreement relation and features chosen for checking determine this, i.e. there is an optionality involved:

---

structures that become highly problematic for syntactic theory. My approach here and in earlier work (te Velde 2005b) is to capture coordinate symmetries within an essentially asymmetric phrase structure.

[29] J. W. Goethe, from "An den Mond." The *–te* ending on *verrauschte* also marks [PAST], ignored here for simplicity's sake.

(23) a. In den Wald ging / gingen [der Jäger und sein Hund] (German)
into the wood went.SG/went.PL [the hunter.SG and his dog].PL

b. In de kamer kom / komen [ik en Jan] vaak (Dutch)
into the room come.SG/ come.PL [I and J].PL often

We must keep in mind that the SVO equivalents of (23) require symmetric agreement:

(24) a. [Der Jäger und sein Hund] gingen /*ging in den Wald
[the hunter.SG and his dog].PL went.PL / went.SG into the wood

b. [Ik en Jan] komen / *kom vaak in de kamer
[I and J].PL come.PL / come.SG often into the room

In American dialects SVO does not guarantee symmetric agreement, however, for reasons that can be debated (see van Gelderen 1997 for an account); in general, non-prescriptive, colloquial speech in English favors asymmetric subject-verb agreement in coordinate structures. It appears that because English inflectional agreement morphology is either impoverished or non-existent, asymmetry results, given the lack of inflections to make a syntactic symmetry clearly apparent. In other words, what features are available and what speech registers or prescriptive rules come into play determine the symmetry or asymmetry of the agreement.[30]

Other West Germanic data that need to be considered include verb-object and preposition-object agreement that have inflections sufficiently rich to indicate symmetry or asymmetry. The data are too extensive to be investigated here (see te Velde 2005b for discussion), but one generalization can be made: When a feature-checking head precedes conjoined DPs, asymmetry is sometimes tolerated, at times even required, as in Frisian constructions with Comp-agreement like (21b). The reverse configuration shows less asymmetry; in languages like German with relatively rich inflectional systems, asymmetry in the [DP & DP]–head configuration is ungrammatical.

This limited amount of data is insufficient for coming to final conclusions, but a picture begins to emerge: rich inflectional agreements systems favor symmetry in the configuration in which the checking head surfaces *after* the conjoined DPs. This correlation suggests that for a merge system like the one used in the *Survive* model, the remerging of a checking head in a position preceding the conjoined elements with which it agrees can result in the breakdown of symmetry. This breakdown is predicted by the fact that the remerger of a head in this position results from an *additional* checking requirement. For instance, in (21b) with Comp-agreement in which the asymmetric, Comp–first conjunct agreement relation is required, this additional checking requirement – the remerger of the coordinate structure [*do en Marie*] 'you and Marie'

[30] Emonds (1986) provides an interesting study on the role of sociolinguistic factors in the choice of (a)symmetric agreement in coordinate structures.

in the CP domain after feature checking with *ha* 'have' in the VP[31] – is induced because both Comp and the first conjunct *do* 'you' have the same feature in Frisian.

What exactly the make-up of the feature for Comp-agreement is must be determined on the basis of data from Germanic dialects that require it. In (17) the remerger of the finite verb in C˚is required for checking the Q-feature present on the interrogative in Spec,CP; the same checking occurs for the same reasons occurs in (22a), with the difference that in the latter no lexical element, just this feature, occupies Spec,CP. The feature that induces the remerger of the finite verb in C˚in constructions like (23a,b) is obviously different, for here the remerger is optional. Hence we must assume that in the *Survive* model there are features that induce stylistic changes as well as the required syntactic operations. Given the flexibility of this model – we saw earlier that it makes no distinction between interpretable and uninterpretable features – "stylistic features" can presumably be selected and merged with SOs in the same way required syntactic features. However, this is another area for further research.

5. Evaluation and further research

Several other aspects of this analysis leave open areas for further research. One of the most complex involves the precise make-up of the mapping algorithms required for the matching of coordinate features. Much has been left up to them, but further work is required to determine whether mapping algorithms can i) accomplish the kind of matching required for coordinate symmetry, and ii) handle both lexical and syntactic feature checking, both without resort to mechanisms that render the grammar non-derivational. This work relates to the division of labor in a derivational grammar and what has been proposed here for the *Survive* model: Is the "shared" labor – in the sense discussed earlier, i.e. that both syntactic and semantic features are handled in the narrow syntax by mapping algorithms – both welcome and feasible? Related to the nature of mapping algorithms is the question of whether they are also suitable for ellipsis/deletion, as assumed for the derivation outlined in (19).

Another area for further research is the investigation of whether *Link!* enables the transfer of derivational units via LF to short-term memory, thus relieving the narrow syntax of a memory burden (essentially two questions). The extension of *Link!* proposed here – a symmetric relation between the narrow syntax and LF enabling the recall of features as needed for coordinate feature matching – also requires more investigation.

Finally, the statement in (18) on 'conjunction' as a complex derivational operation that merges the output of simplex derivations must be made more concrete and precise. The prediction here is that this derivation concatenates matrix clauses as SOs in much the same manner as other derivations concatenate LIs. Nevertheless, the syntactic structure required and the status of the coordinating conjunction within it – especially what feature it has that must match with the conjunct it merges with – must be investigated further.

[31] Presumably tense feature checking occurs in the TP domain as well, as it does in (22a) before *ha* raises to check the Q-feature in the CP domain.

References

Boeckx, C. 2007. Eliminating Spell-Out. *Linguistic Analysis* 33, 3-4: 414 – 425.

Camacho, J. 1997. The Syntax of NP Coordination. Ph.D. diss., Univ. of Southern California.

Camacho, J. 2000. On the Structure of Conjunction. In *Ellipsis in Conjunction*, K. Schwabe & N. Zhang (eds.), 23–50. Tübingen: Niemeyer.

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

Chomsky, N. 2000. Minimalist Inquiries: the framework. In *Step by Step*, Roger Martin, D. Michaels & J. Uriagereka (eds.), 89-155. Cambridge, MA: MIT Press.

Chomsky, N. 2001. Derivation by Phase. In *Ken Hale: A life in language*, M. Kenstowicz (ed.), 1 – 52. Cambridge, MA: MIT Press.

Chomsky, N. 2005. On Phases. Ms. [to appear in *Foundational Issues in Linguistic Theory*, R. Freidin, C. P. Otero and M.-L. Zubizaretta (eds.). Cambridge, MA: MIT Press].

Chomsky, N. 2007. Approaching UG from Below. In *Interfaces + Recursion = Language? Chomsky's Minimalism and the View from Syntax-Semantics*, Uli Sauerland and Hans-Martin Gärtner (eds.), 1–29. Berlin: Mouton de Gruyter.

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

Emonds, J. 1986. Grammatically Deviant Prestige Constructions. In *A festschrift for Sol Saporta*, M. Braeme, H. Contreras & F. Newmeyer (eds.), 93-129. Seattle: Noit Amrofer.

Frazier, L. and Clifton, C., Jr. 2001. Parsing Coordinates and Ellipsis: Copy α. *Syntax* 4,1: 1–19.

Frazier, L. 2008. Processing Ellipsis: A Processing Solution to the Undergeneration Problem? In *Proceedings of WCCFL* 26, C. B. Chang & H. J. Haynie (eds.), 21-33. Somerville, MA: Cascadilla Proceedings Project.

van Gelderen, E. 1997. *Verbal Agreement and the Grammar behind its* Breakdown. Tübingen: Niemeyer.

Grootveld, M. 1994. *Parsing Coordination Generatively*. Den Haag: Holland Institute of Generative Linguistics.

Johannessen, J. B. 1998. *Coordination*. New York: Oxford University Press.

Johnson, K. 2002. Restoring Exotic Coordinations to Normalcy. *Linguistic Inquiry* 33: 97-156.

Kayne, R. 1994. *The Antisymmetry of Syntax*, [Linguistic Inquiry Monograph 25]. Cambridge, MA: MIT Press.

van Koppen, M. 2003. First Conjunct Agreement. Frisian, Tegelen Dutch and Hellendoorn Dutch. Paper presented at the Comparative Germanic Syntax Workshop 18, University of Durham, Sept. 18–20, 2003.

van Koppen, M. 2005. *One Probe – Two Goals: Aspects of agreement in Dutch dialects*. Utrecht: LOT.

van Koppen, M. 2006. One Probe – Multiple Goals. Paper presented at GLOW 2006.

Moltmann, F. 1992. Coordination and Comparatives. Ph.D. diss., MIT, Cambridge, MA.

Muadz, H. 1991. Coordinate Structures: A planar representation. Ph.D. diss., Univ. of Arizona.

Munn, A. Coordinate Structure and X-bar Theory. *McGill Working Papers in Linguistics* 4: 121-140.

Putnam, M. 2006. Scrambling in West Germanic as XP-adjunction: A critical analysis of prolific domains. Ph.D. dissertation, University of Kansas-Lawrence.

Putnam, M. 2007. *Scrambling and the Survive Principle.* Amsterdam: John Benjamins.

Putnam, M. and T. Stroik. 2008. Syntactic relations in Survive-minimalism. Ms.

Stroik, T. 2007. *Locality in Minimalist Syntax*. Cambridge, MA: The MIT Press.
Stroik, T. 2008. The Numeration in *Survive*-Minimalism. Ms.
te Velde, J. R. 2005a. Unifying Prosody and Syntax for Right- and Left-Edge Coordinate Ellipsis. *Lingua* 115: 483-502.
te Velde, J. R. 2005b. *Deriving Coordinate Symmetries. A phase-based approach integrating Select, Merge, Copy and Match*. Amsterdam: John Benjamins.
Uriagereka, J. 1999. Multiple Spell-Out. In *Working Minimalism,* S. D. Epstein and N. Hornstein (eds.), 251-282. Cambridge: MIT Press.
Wesche, B. 1995. *Symmetric Coordination. An Alternative Theory of Phrase Structure*, [*Linguistische Arbeiten* 332]. Tübingen: Max Niemeyer Verlag.