View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Edinburgh Research Explorer

THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Effects of short-term storage in processing rightward movement

Citation for published version:
Ackema, P & Neeleman, A 2002, 'Effects of short-term storage in processing rightward movement'. in SG Nooteboom, F Weerman & FNK Wijnen (eds), Storage and computation in the language faculty. Studies in Theoretical Psycholinguistics, vol. 30, Kluwer , pp. 219-256.

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Preprint (usually an early version)

Published In:
Storage and computation in the language faculty

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 20. Feb. 2015

Chapter #

Effects of Short-Term Storage in Processing Rightward Movement

PETER ACKEMA and AD NEELEMAN
Utrecht University and University College London

Key words: rightward movement, processing, short-term storage, (a)symmetric syntax, Verb Raising, P-incorporation, Heavy NP Shift

Abstract: There is a striking asymmetry between leftward and rightward movement in syntax: whereas leftward movement can in principle be unbounded, rightward movement is subject to very strict locality conditions. There are two possible approaches to explaining this asymmetry. One can either assume that some syntactic principle disfavours rightward movement, or that some mechanism having to do with sentence processing is responsible. In this chapter we will argue that a processing approach to limitations on rightward movement is more fruitful. In particular, we will argue that the human parser cannot process certain instances of rightward movement because the introduction of an antecedent-trace relation leads to a conflict with information about the parse which is already stored in short-term memory before this relation can be established. Similar problems do not occur in cases of leftward movement.

1. INTRODUCTION

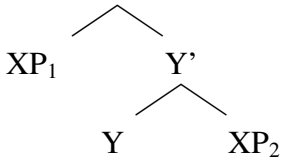
At least since Ross 1967 it is known that there is a striking asymmetry between leftward and rightward movement in syntax.[1] Whereas

[1] Natural language allows 'displacement': elements may show up in position they do not usually occupy. For example, in *who did she see*, *who* is interpreted as the direct object of *see*. The analysis of this example involves an unpronounced 'trace'.that occupies the

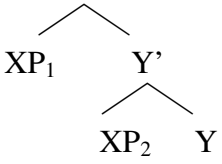
leftward movement can in principle be unbounded, rightward movement is subject to very strict locality conditions. There are two possible approaches to explaining this asymmetry. One can either assume that some syntactic principle disfavours rightward movement, or that some mechanism having to do with sentence processing is responsible.

A syntactic approach to the problem has been advanced by Kayne (1994). Kayne proposes a principle which has the effect (for reasons that need not concern us here) that syntactic trees go 'downward' from left to right. If a head Y has two dependents XP_1 and XP_2, the right-branching structure in (1a) is the only structure allowed; (1b-d) are ruled out.

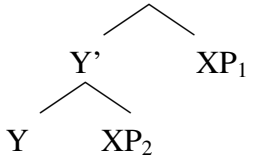
(1) a.



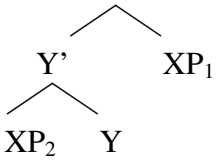
b. *



c. *



d. *



Since there is overwhelming evidence that an element cannot be moved to a position that is lower in the tree than the position it originates in (see for instance Van Riemsdijk & Williams 1986:202), the so-called antisymmetric theory illustrated in (1) implies that rightward movement cannot exist, since in this theory 'rightward' implies 'downward in the tree'.

Kayne's theory of antisymmetry faces difficulties in two areas. First, there appears to be pervasive evidence that trees as in (1b-d) exist. (We will discuss this evidence in some detail below). Second, there are in fact instances of rightward movement. Under a Kaynean

---

object position of *see*. The relation between the fronted category *who* and the trace is called 'movement'.

approach, these are unexpected and hence require an alternative account (see Büring & Hartmann 1997 for discussion).

In this chapter we will argue that a processing approach to limitations on rightward movement is more fruitful than a syntactic approach. In sentence processing a syntactic representation is constructed on the basis of an essentially linear (left-to-right) input. As pointed out by Just & Carpenter (1992), Gibson (1998), and Kaan & Stowe (this volume), this process involves both storage and computation. The parser must compute a syntactic representation of the incoming sentence. Since it cannot do so for the whole sentence at once, short-term storage of partially analysed substrings is required. We further assume that the parser employs a so-called filler-driven strategy in dealing with movement dependencies, which means that the parser can only postulate a trace (which indicates the base position of a moved element) after it hypothesises that some element it encounters has been moved. Hence, rightward movement requires the insertion of a trace in an already partially analysed string. After all, the moved element follows the part of the string in which the trace should be inserted. We will argue that introduction of a trace in an already analysed string is sometimes incompatible with information stored in short-term memory. If in addition no alternative parse is available, the structure cannot be processed at all. In contrast, in cases of leftward movement the trace can be inserted at the same time that the string in which it should be placed is analysed. Consequently, leftward movement does not cause this kind of processing difficulty.

The advantage of the approach just sketched is that it reconciles the evidence that syntax equally allows left-branching and right-branching structures with differences in the nature of rightward and leftward movement; the correct locality conditions on rightward movement follow from independently motivated properties of the parser.

This chapter is organised as follows. First, we will discuss some evidence for the existence of the structures in (1b-d), and show which difficulties theories based on antisymmetry face in accounting for the relevant data (sections 2 and 3). Then, we will outline the properties of the human parser (section 4). These allow us to explain in which circumstances head movement to the right is impossible (section 5), and in which circumstances it is possible (section 6).[2] Then we will

[2] X-bar theory assigns constituents the following general three-layered structure, which we will adopt for concreteness: $[_{XP} \ldots [_{X'} \ldots X \ldots] \ldots]$. Phrasal movement is movement of

present the empirical evidence that relates to our predictions with respect to rightward head movement (section 7). Finally, we will consider how the parser restricts rightward XP-movement (section 8).

2. SYMMETRIC SYNTAX

Traditional views of syntactic structure allow both left- and rightbranching structures (cf. Chomsky 1986:3). Complements, specifiers and adjuncts can in principle either precede or follow the head.[3] Given that each of these constituents in a phrase is attached at a specific level of that phrase, such a theory of syntactic structure predicts mirror image effects: the order in which two base-generated (unmoved) elements appear to the left of the head is the reverse of the order of those elements if they are generated to the right of the head:[4]

(2) a. $[_{Y''}\ XP_1\ [_{Y'}\ XP_2\ Y]]$ (= (1b))
b. $[_{Y''}\ [_{Y'}\ Y\ XP_2]\ XP_1]$ (= (1c))

We will refer to this as the symmetric theory.

For independent reasons, it is not always possible to generate a specific constituent on either side of the head. In particular, nominal arguments are assigned case by the verb, which, for reasons we cannot discuss here, can have the effect that they must be in a fixed position with respect to the verb (see Neeleman and Weerman 1999 for

---

the entire constituent, XP. Head movement is movement of its head, X. In a Verb Second language like Dutch, for example, the finite verb that heads the VP is moved to the second position in main clauses: *Jan geeft$_i$ Marie een boek t$_i$.* 'John gives Mary a book'. (Compare with the embedded clause *dat Jan Marie een boek geeft* 'that John Mary a book gives').

[3] In X-bar theory, a complement is the sister of a head X and the daughter of an X'. A specifier is the daughter of an XP and the sister of an X'. Adjuncts are elements attached to a recursive category: they are either the sister of an XP and the daughter of an XP, or the sister of an X' and thedaughter of an X'. The former option gives rise to the following structure: $[_{XP}$ adjunct $[_{XP} \ldots] \ldots]$.

[4] For ease of presentation, we will represent the dependents of the head as being generated within the head's projection, abstracting away from possible functional structure. However, the predictions that a symmetric X-bar theory makes with respect to mirror image effects do not change if functional structure is added to (2).

discussion.[5] For elements within VP that *can* be generated on both sides of the head V, however, mirror image effects can indeed be observed. Their distribution therefore confirms the view that base-generated structure to the right of the head is left-branching.

One such mirror image effect occurs with prepositional phrases in Dutch. In Dutch it is possible to have (nonpredicational) PPs on both sides of V. Koster (1974) observed that the order in which the PPs stand when they occur in preverbal position is the reverse of their order in postverbal position. This is illustrated in (3).

(3) a. dat Jan [[tijdens de pauze]$_1$ [[aan zijn vader]$_2$ dacht]]
that John during the break of his father thought
'that John thought of his father during the break'
a'. ??dat Jan [[aan zijn vader]$_2$ [[tijdens de pauze]$_1$ dacht]]
b. dat Jan [[dacht [aan zijn vader]$_2$] [tijdens de pauze]$_1$]
b'. ??dat Jan [[dacht [tijdens de pauze]$_1$] [aan zijn vader]$_2$]]

An argument for the claim that PP_1 is attached higher than PP_2 in (3b) as well as in (3a) can be given on the basis of the examples below. When the VP is moved to the front of the clause ('VP-topicalization'), it is possible to include PP_2 in the moved VP and leave PP_1 behind. The reverse, however, is impossible. This shows that the verb and PP_2 form a constituent which excludes PP_1. In (4) both PPs are generated to the left of the basic V-position, but in (5) they are both generated to the right. The contrast remains the same.

(4) a. ?Aan zijn vader denken heeft Jan alleen tijdens de pauze gedaan.
of his father think has John only during the break done
'Thought of his father, John only has during the break'
b. *Tijdens de pauze denken heeft Jan alleen aan zijn vader gedaan.

[5] With double object constructions, a cross-linguistic anti-mirror image effect can be observed. The order in which the objects show up in English is identical to that in Dutch: V-IO-DO and IO-DO-V, respectively. It is argued in Neeleman & Weerman 1999 that these data follow from the way the case systems of OV and VO languages are organised.

(5) a. ?Denken aan zijn vader heeft Jan alleen gedaan tijdens de pauze.
think of his father has John only done during the break
'Thought of his father, John only has during the break'
b. *Denken tijdens de pauze heeft Jan alleen gedaan aan zijn vader.

Scope relations, too, can be used to illustrate the symmetry of base-generated structures. Since projections are right-branching before and left-branching after the head in the symmetric theory (see (2)), the prediction is that in case two adverbials precede the head, the left one takes scope over the right one, whereas the reverse should be true of two adverbials that follow the head. The data in (6) bear out this prediction. Both (6a) and (6b) receive an interpretation such that *in die dagen* 'in those days' takes scope over *volgens Marleens plan* 'according to Marleen's plan': the sentences mean that, in those days, we followed Marleen's ideas about where to spend the holidays. In both (6a') and (6b') the scope relations are reversed: these sentences mean that, in accordance with Marleen's plan, we went on holiday at a particular time.[6]

(6) a. dat we [in die dagen [volgens Marleens plan op vakantie gingen]]
that we in those days according to Marleen's plan on holiday went
'that we used to go on holiday in accordance with Marleen's plan in those days'
a'. dat we [volgens Marleens plan [in die dagen op vakantie gingen]]
that we according to Marleen's plan in those days on holiday went
'that, in accordance with Marleen's plan, we went on holiday in that period'
b. dat we [[op vakantie gingen volgens Marleens plan] in die dagen]

[6] Of course, the sentences in (6b,b') can be read in such a way that the second PP is an afterthought. This leads to scope ambiguity. However, the sentences are unambiguous when pronounced without a prosodic break.

'as in (a)'

b'. dat we [[op vakantie gingen in die dagen] volgens Marleens plan]
'as in (a')'

It is further predicted that the sentence will be ambiguous in case one adverbial appears to the left of the head and the other to its right. This is confirmed as well. As indicated, the sentences in (7a,a') and (7b,b') are ambiguous in that they can be associated with both scope relations discussed above.

(7) a. dat we [in die dagen [op vakantie gingen volgens Marleens plan]]
that we in those days on holiday went according to Marleen's plan

a'. dat we [[in die dagen op vakantie gingen] volgens Marleens plan]

b. dat we [volgens Marleens plan [op vakantie gingen in die dagen]]

b'. dat we [[volgens Marleens plan op vakantie gingen] in die dagen]

Note that, without further qualification, the antisymmetry theory predicts that in (6) and (7) the left-hand adverbial must always have scope over the right-hand one, since the tree goes 'downward' from left to right.

Mirror image effects can not only be observed within one language, but also cross-linguistically. The order of preverbal adverbials in Dutch, for instance, turns out to be the mirror image of the order of postverbal adverbials in English, see (8). The assumption that in (8b) *yesterday* is attached higher than *passionately*, which occurs to its left, is corroborated by the fact that under VP-preposing *yesterday* can, but *passionately* cannot, be stranded, as illustrated in (9) (see Roberts 1985).

(8) a. dat Jan [[gisteren]$_1$ [[vurig]$_2$ [een meisje kuste]]]
that Jan yesterday passionately a girl kissed
'that Jan kissed a girl passionately yesterday'

a'. ?*dat Jan [[vurig]$_2$ [[gisteren]$_1$ [een meisje kuste]]]

b. John [[[kissed a girl] [passionately]$_2$] [yesterday]$_1$]

b'. ?*John [[[kissed a girl] [yesterday]$_1$] [passionately]$_2$]

(9) a. John wanted to kiss a girl passionately
and [kiss a girl passionately]$_i$ he did t_i (yesterday).
b. John wanted to kiss a girl yesterday
and [kiss a girl yesterday]$_i$ he did t_i (*passionately).

Some further evidence for posthead left-branching structures in English can be based, as in Dutch (cf. (6) and (7)), on scopal relations. Ernst (1994) observes that the examples in (10) differ in interpretation.

(10) a. She kissed him willingly many times.
a'. She kissed him many times willingly.
b. Joe hit him on purpose frequently.
b'. Joe hit him frequently on purpose.

In the sentences (10a,b) the rightmost adverbial unambiguously takes widest scope, as predicted by the symmetric theory. The sentences in (10a',b') are ambiguous. In one reading it is again the rightmost adverbial that takes widest scope, but a reading with reversed scope is possible as well. However, this latter reading may be the result of an alternative structural analysis in which the frequency adverbial is a modifier of the adverb of intention. One argument for this analysis is that, when a PP is inserted between the frequency adverbial and the adverb of intention, the scope relations are unambiguous again, with the rightmost adverbial taking wide scope:

(11) a. She kissed him many times in the bathroom willingly.
b. Joe hit him frequently with a baseball bat on purpose.

In short, when two constituents occur before the head they are dependents of, scope relations go from left to right, but when they occur after the head, scope relations go from right to left; this is exactly what the symmetric theory in (2) predicts.

Mirror image effects do not only occur in the verbal, but also in the nominal domain. One example of this is the order of prenominal adjectives in English in comparison with that of postnominal adjectives in French (see Lamarche 1991):

(12) a. A [[rusty]$_1$ [[white]$_2$ car]]
b. Une [[voiture [blanche]$_2$] [rouillé]$_1$]
a car white rusty
'a rusty white car'

A language-internal example of the same phenomenon can be found in Tagalog, as pointed out to us by Norvin Richards. Tagalog adjectives can appear on either side of the noun. (A morpheme referred to as the 'linker' must appear between them; this takes the form of a velar nasal attached to the first word when this is phonologically feasible and *na* otherwise.) In those cases where two nonconjoined adjectives are generated on the same side of the noun, their order shows the by now familiar mirror image effect:

(13) a. pinakamalapit na pulang bahay
nearest red house
'the nearest red house'
a'. *pulang pinakamalapit na bahay
b. bahay na pulang pinakamalapit
b'. *bahay na pinakamalapit na pula

In fact, the argument holds more generally. Greenberg (1966) observes that in the majority of cases the order of determiners, numerals and adjectives in languages in which these elements follow the noun is the mirror image of the order found in languages in which they precede the noun, as in (14). (See below for a fuller discussion of the cross-linguistic possibilities of the order in a DP, incorporating the possibility of N-movement.)

(14) a. determiner - numeral - adjective - noun
b. noun - adjective - numeral - determiner

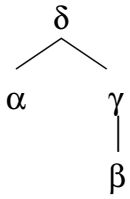
In conclusion, the symmetric theory of phrase structure predicts that elements generated to the right of the head appear in the reverse order of elements generated to its left, if no movement occurs. The data discussed in this section appear to confirm this prediction.

3. MIRROR IMAGE EFFECTS IN AN ANTISYMMETRIC THEORY

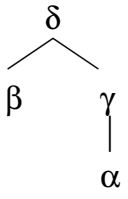
It is, of course, not impossible to account for mirror image effects in a theory such as that of Kayne (1994), which does not allow the structures in (1b-d). However, as we will argue in this section, this can only be done at the cost of some ad hoc assumptions.[7]

Kayne proposes that the linear order of terminal nodes in a tree reflects the hierarchical structure of the nonterminal nodes dominating the terminals. In particular, the c-command relations between the nonterminal nodes are crucial. A category α c-commands a category β if every category that dominates α also dominates β. (Roughly speaking, α is at least as high in the tree as β.) For example, in (15a) α c-commands β, but in (15b) it does not (because here γ dominates α but not β).

(15) a.



b.



What Kayne hypothesises is that, if α c-commands β, the terminal nodes dominated by α precede the terminal nodes dominated by β. One of the consequences of this is that there must be a universal specifier-head-complement order, as in (1a), and, as already discussed, rightward movement cannot exist.

The question of how to derive mirror image effects in such an antisymmetric syntax is addressed by Cinque (1996), in connection with such effects within DPs (Determiner Phrases, i.e. nominal constituents). His proposal straightforwardly carries over to mirror image effects in the verbal domain. Below, we will therefore abstract away from the labels of the particular projection involved (as Cinque in fact does himself).

Suppose that the observed order of three modifiers to the left of a lexical head L is the reverse of those elements to the right of L:

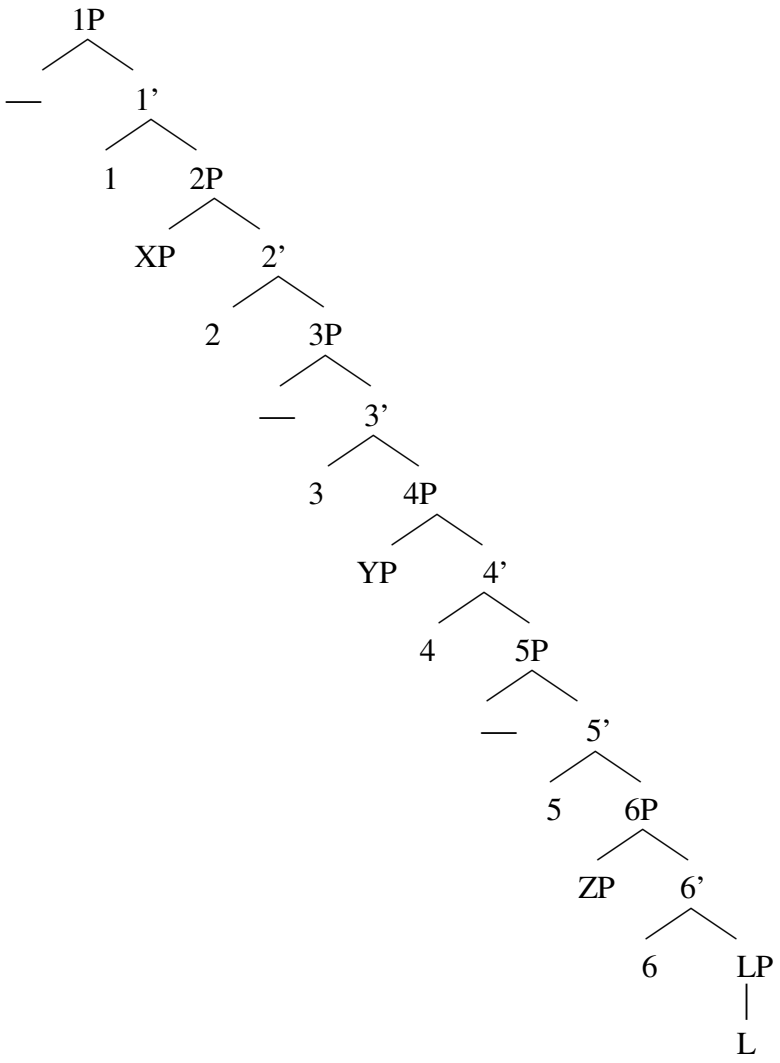
[7] This section by necessity contains a few syntactic technicalities, which a general readership might not be familar with. It briefly reviews an alternative account of the data discussed so far. Readers mainly interested in the nonsyntactic account of movement asymmetries may therefore wish to skip it. This can be done without losing the thread of the argument.

(16) a. XP YP ZP L
b. L ZP YP XP

From Kayne's theory it follows that there is only one specifier or adjunct per head, generated to the left of this head. Modifiers must therefore each be generated in a distinct functional projection.[8] In the structure below, 2P, 4P and 6P accommodate XP, YP and ZP respectively. The problem, then, is how to derive (16b) from (16a). To this end, extra functional projections must be introduced, providing the landing sites for the necessary movements. These are 1P, 3P and 5P in the structure below:

[8] A functional projection is the projection of a functional head like Tense (projecting a TP), Aspect (projecting an AspP), Agreement (projecting an AgrP) and so on. The projection of a lexical head is dominated by functional projections pertaining to that lexical category only (e.g. a verb phrase but not a noun phrase is dominated by a Tense Phrase, whereas a noun phrase but not a verb phrase is dominated by a Determiner Phrase), so functional projections are in some sense parasitic on, or an extension of, a lexical projection. Since the precise content of the functional projections is not relevant for our argument, we have simply numbered them in the example structures.

(17)



The pattern in (16b) can now be derived by movement of LP to the specifier position of 5P, followed by movement of 5P to spec-3P, followed by movement of 3P to spec-1P (the structure is ‘rolled up’ as it were).

However, if mirror image effects are to be accounted for along these lines, motivation must be given for some crucial assumptions. First, the analysis presupposes that LP is dominated by an extensive functional structure, which provides the landing sites for movement. In particular, every functional projection hosting a modifier must be dominated by a functional projection to whose specifier movement is possible. Apart

from the conceptual consideration that such a theory is rather far removed from the ideal of structural economy (cf. Chomsky 1995), it must be noted that independent syntactic evidence for 3P and 5P in (17) cannot be given. That is to say, distributional data can never distinguish movement of L to 2 from movement to 3, or movement of L to 4 from movement to 5. The reason is that no adverbials or other material can be generated in between these positions, given that spec-3P and spec-5P must be empty in order to function as landing sites for movement.

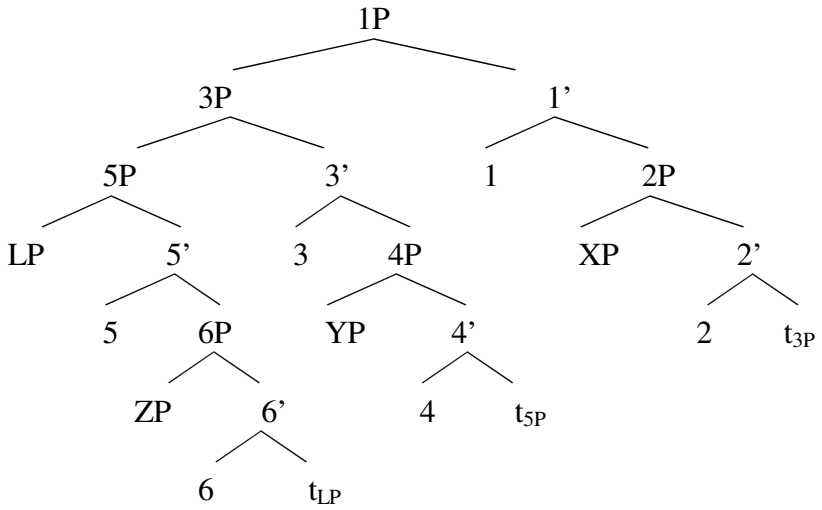
Second, the movements needed to derive (16b) should all be motivated. In the minimalist program (Chomsky 1995) it is assumed that movement occurs to 'check' a 'strong' feature of a functional head. For (17) this implies some feature that triggers movement must be present on the heads of the odd-numbered projections, and these features must be 'checked' by movement of LP, 5P and 3P to the relevant specifier positions. Again, such triggering features cannot be introduced without further argumentation. The content of the odd-numbered projections must motivate their presence. However, as we have seen above, it is impossible to find independent evidence for the existence of these projections, and consequently for their content. Moreover, if the odd-numbered projections are assigned such features, these should not attract the even-numbered projections. Given that the even-numbered projections are closer to the relevant specifier positions, they would otherwise block the movements necessary to derive (16b). Finally, the problem of finding triggers is complicated by the fact that many mirror image effects occur language-internally (compare the Dutch and Tagalog data in (3)-(6) and (13) discussed above). In order to account for this, the various movements would have to be optional, which would imply that the attracting features have a variable strength.

Third, even if the analysis is adopted as it stands, orders can be derived which the analysis is intended to exclude. One would expect that the lexical head in (17) can undergo successive head-to-head movement (from L to 6 to 5 and so on) - both in verbal and nominal contexts such movements are well-attested. Suppose that phrasal movement of the type under discussion is combined with head movement. Suppose, more specifically, that head movement of L to 3 is followed by phrasal movement of 3P to spec-1P. The resulting order would be L-YP-ZP-XP. If this order is to be excluded, as in the case at hand, there must be a condition that forbids phrasal movement if head

movement has taken place – a condition for which there seems to be no independent motivation.

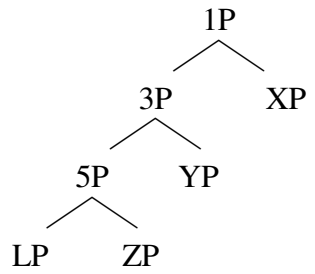
Finally, note that the structure assigned to (16b) in the analysis under discussion is, in fact, a left-branching structure as far as the lexical elements in it are concerned. This structure is the following:

(18)



The structural relations between the lexical material in this tree become apparent when we remove all nodes that do not branch into nodes that each dominate lexical material (that is, when we remove all structure assumed .for theory-internal reasons). The result is the tree in (19).

(19)



The structural configuration in (19) is identical to the one in a base-generated left-branching structure. In other words, the antisymmetric theory imitates, at the cost of introducing potentially problematic movements, the structure directly derived in the symmetric phrase structure theory.

We conclude that the mirror image phenomena are captured more satisfactorily in the symmetric theory than in the antisymmetric one. This means we have to look for an alternative explanation of the different properties of leftward and rightward movement than the one the antisymmetric theory offers. We now turn to this issue.

4. ASYMMETRIC PARSING

In addition to the mirror image effects discussed in section 2, some specific anti-mirror image effects are attested. In a number of noun-initial languages, for instance, the dependent elements show up in the same order as in noun-final languages, see (20c). This can be explained by head movement of the noun to the left, starting with the basic structure in (20a) (= (14a)). (As noted, such head movement is well-motivated in both the verbal and the nominal domain). However, a fundamental asymmetry can now be observed. Head movement of the noun to the right appears to be impossible. Alongside (20b) (= (14b)), no languages exist in which the order in (20d) is found (Greenberg 1966).[9]

(20) a. determiner - numeral - adjective - noun
b. noun - adjective - numeral - determiner
c. noun - determiner - numeral - adjective
d. *adjective - numeral - determiner - noun

The asymmetry of head movement, as opposed to the symmetry of base generation, is also attested in the verbal domain. The assumption that subjects can be generated on either side of V' and objects on either side of V gives rise to an acceptable language typology. In addition to this, some languages display head movement of V to the left. Examples are VSO languages and languages with Verb Second. There are no

[9] There are also languages in which the order determiner - numeral - adjective is attested and in which the noun occurs between two of these elements, instead of at the beginning or end of the sequence. This can of course also be explained by noun movement to the left, starting from the basic structure in (20a). As far as we know, however, there are no languages with the order adjective - numeral - determiner with the noun in between, showing again that noun movement to the right, starting from the structure in (20b), is impossible.

languages, however, in which there is straightforward distributional evidence that the verb moves to the last or one but last position in the clause.

In view of the discussion in sections 2-3, we take it that syntax is not fundamentally asymmetric. This implies that we must either assume a principle of asymmetry specific to movement processes, or look for the explanation of the observed asymmetries elsewhere. We will pursue the latter option, because we think that the grammar as a whole can be simplified if a nonsyntactic explanation is sought. In particular, we would like to suggest an explanation in terms of universal parsing strategies.

The parser is generally assumed to be a mechanism that assembles structures, whereas the grammar is the knowledge base which the parser uses to do this processing. It is likely that certain structures are grammatical but cause difficulties for the processing mechanism (cf. Kimball 1973). Such difficulties may lead to so-called garden path effects, in which a hearer is forced to abandon his initial analysis of a string and pursue an alternative. Below we will argue that some instances of rightward movement lead to a more serious processing problem: the hearer's initial analysis again turns out to be incorrect, but this time there is no alternative. The result is that the sentence in question is not just difficult to process, but ruled out altogether.

As explained in the introduction, leftward movement and rightward movement differ in that rightward movement requires the introduction of a trace in a partially analysed string, whereas leftward movement allows the trace to be introduced at the same time that the string is analysed (provided that first the moved element is recognized as such, of course). Hence, rightward movement may lead to difficulties if further computation involving insertion of the trace is incompatible with information stored in short-term memory. Before discussing the details of our proposal, let us briefly consider some basic properties of the human parser.

First, it is obvious that, due to the temporal order in which the input is received, the parser scans this string from left to right, computing a representation of the sentence as it goes along. During this process, it can store information concerning already analysed parts of the string in short-term memory (see Kaan & Stowe, this volume, on the distinction between the computational and storage functions of the parser).

Second, we assume that there are severe limitations on holding unstructured linguistic material in short-term memory (Just and Carpenter 1992). The consequence of this is that the parser must have limited look-ahead capacity. The most strict assumption is that it has no look-ahead at all, as has been argued by Frazier & Rayner 1982, Gorrell 1995, and others. This means that at any stage, the parser can only consider the current input symbol and (a limited amount of) the already parsed material when it is deciding what to do next. Crucially, information from its right context is not available.

Third, the parser builds a syntactic tree. Of course, a syntactic tree is nothing more than a set of dominance and precedence relations between nodes. What the parser does, then, is hypothesise such relations for a particular input string. For our purposes, precedence relations will be crucial. In order to maintain a maximally economical use of its storage function, the relations that the parser hypothesizes will be kept to the minimum that is necessary to construct a full, coherent, tree. For instance, if A dominates B and B dominates C, the parser will not also notate that A dominates C, as this is superfluous. For precedence relations between XPs and heads this implies that these are only notated with respect to the head that the XP is a dependent of, since, combined with the precedence relations between heads, this is the minimal information that is needed to describe the order in a tree. (This way of notating information about the structure that is being parsed will be amply illustrated below).

Fourth, we assume that the parser handles antecedent-gap relations by applying a so-called filler-driven strategy (cf. Frazier 1987, 1993, Frazier & Flores d'Arcais 1989, Gibson 1998). This means that the postulation of a gap depends on the presence of an antecedent. Once an antecedent is identified as a moved element (on the basis of the grammar of the language in question, which the parser uses as a knowledge base), a position to insert a gap in is looked for. This is done by considering selectional information of heads as well as by applying 'gap-finding' strategies such as Frazier's (1987) Active Filler Strategy (which says that the gap should be inserted as soon as is possible). The filler-driven strategy differs from so-called gap-driven strategies, which assume that postulation of a gap depends on lexical (for instance selectional) information and then look for the antecedent in the left context. Experimental evidence (Frazier & Flores d'Arcais 1989) as

well as the possibility of adjunct movement (i.e. movement of unselected elements) seem to support the first assumption.[10]

The above assumptions are more or less standard (see also Davis & Alphonce 1992:88). More controversy arises in connection with the issue of locally ambiguous inputs. What does the parser do when it encounters material that can be integrated into the existing parse in more than one way? Various answers have been proposed. Some authors have argued that analysis is delayed until disambiguating information becomes available (Berwick & Weinberg 1985, Weinberg 1988). Others have argued that the parser pursues one analysis (preferred on grounds that need not concern us here) and only returns to the second if it fails (Frazier & Clifton 1995). Yet others claim that the parser computes all possible analyses in parallel, ranking them according to some preference principle(s) (Gibson 1991).

All these approaches appear to share a basic assumption, namely that it is impossible to alter established structural relations *within a single parse*. The point of delaying analysis is exactly to guarantee that no established information is destroyed ("informational monotonicity", cf. Marcus et al. 1983). In both ranked parallel and serial parsers the preferred parse is abandoned when new input is incompatible with it. The next best alternative is pursued. Crucially, what does not happen in any of these models is that the parser changes structure it has already postulated.[11] We assume that this is indeed impossible. So, if at some point P the parser cannot decide between analyses, it may first try one of the analyses and then (in case of failure) shift to another. But if at P the input string allows only one analysis, each continuation of the parse after P (i.e. each parse that can be pursued) must comply with this analysis. If material encountered later is incompatible with it, the sentence is not parsable.

Summarising, the parser has the following properties:

[10] For example, there is nothing in *You think Mary fixed the bike* which suggests that it contains a gap. Therefore, in *How do you think Mary fixed the bike*, postulation of a gap (a trace) is dependent on the presence of the fronted element how.

[11] One exception is Pritchett 1992, where it is argued that under restricted circumstances altering syntactic relations in a parse is possible, namely in case the new position of an element is dominated or governed by its original position. This condition, however, can easily be adapted so as to apply between parses (whether parallel or serial) rather than within a parse.

(21) a. It scans the input string from left to right.
b. It constructs a tree, that is, a set of dominance and precedence relations.
c. It has no look-ahead.
d. It can only postulate a trace after having encountered an antecedent.
e. It cannot alter information (dominance and precedence relations) stored in short-term memory for a given parse.

As said, we will argue that the problem with most instances of head movement to the right concerns (21e). Moreover, we will show that, exactly when it does not lead to problems with respect to (21e), head movement to the right is in fact possible. XP-movement to the right does not violate (21e) and is therefore possible, at least in principle. We will suggest that the fact that it is more restricted than XP-movement to the left follows from (21d) in combination with closure of already parsed material.

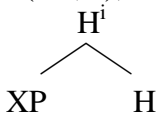
5. INFELICITOUS RIGHTWARD HEAD MOVEMENT

Let us now consider the effects of (21). Suppose that the first part of an input string has been analysed as an XP.[12] The parser then postulates that this maximal projection precedes the head that it is a dependent of. Formally, this means that the parser hypothesises the following syntactic relations: the XP is immediately dominated by a projection H^i of some head H, XP precedes H and, trivially, H^i is projected from H. Crucially, the parser cannot assume that XP follows H, since it has not encountered a head yet. It cannot assume that H is the trace of a moved head either, with XP following this trace, because traces cannot be inserted unless an antecedent has been identified (cf. (21d)). Thus, if (22a) represents the current stage of the parse, the additional information

[12] For the moment, we abstract away from the internal structure of XP. The maximal projections we will talk about are specifiers, complements or adjuncts of the heads under discussion. (We will use the general term 'dependent' of a head to refer to such elements.) We also abstract away from the option that such a projection has moved. We turn to movement of maximal projections in section 8.

in (22b) becomes available (where P means 'precede', ID means 'immediately dominate' and Proj means 'project'). The minimal tree compatible with this information is given in (22c).

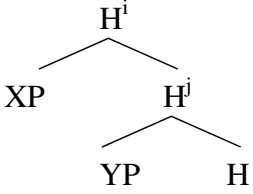
(22) a. XP
b. P(XP,H); ID(H^i,XP); Proj(H,H^i)
c.



As will be obvious, H is an abstract head at this point, which can only be given content when an actual head is encountered or when the trace of a moved head is inserted.

The parser will continue to build a right-branching tree. That is, if it again has analysed a substring as a constituent, it will assume that this constituent precedes H and is immediately dominated by a projection of H.[13] This continues until the parser encounters that head which it hypothetises (e.g. on the basis of selectional information) to be the head of which the XPs analysed thus far are dependents. In other words, if the parser identifies two maximal projections XP and YP before it encounters this head, the relations in (23b) are postulated (where D means 'dominate'). These are compatible with the minimal tree in (23c).

(23) a. XP YP
b. P(XP,H); ID(H^i,XP); Proj(H,H^i)
P(YP,H); ID(H^j,YP); D(H^i,H^j); Proj(H,H^j)
c.



Note that the extension in (23b) of the parse in (22b) does not involve alteration of relations already stored in short-term memory (cf. (21e)), it

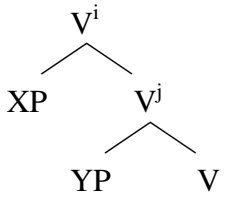
[13] Strictly speaking, the heads of which the first and the second constituent are dependents need not be the same. We abstract away from this matter here, as it does not affect the argument we are about to present. See below for discussion.

only involves adding new relations. Note also that the parser need not lay down precedence relations between maximal projections, as these follow from the dominance relations given in (23b) in combination with the information that XP and YP precede H.

When the parser encounters a head, there are two options. It can assume that this head has moved or is in its base position. The latter case is by far the simplest: the parser simply assumes that this head is H. So, if the input contains a verb, the parse can be extended as in (24).

(24) a. XP YP V

b. P(XP,H); ID(H^i,XP); Proj(H,H^i)
P(YP,H); ID(H^j,YP); D(H^i,H^j); Proj(H,H^j)
H = V

c. 

For any further maximal projection the parser identifies, it will now notate that it follows V and is immediately dominated by a projection of V.[14] Suppose that the parser identifies two more maximal projections. Then the parse is extended to (25).

(25) a. XP YP V ZP WP

b. P(XP,H); ID(H^i,XP); Proj(H,H^i)
P(YP,H); ID(H^j,YP); D(H^i,H^j); Proj(H,H^j)
H = V
P(H,ZP); ID(H^k,ZP); Proj(H,H^k)
P(H,WP); ID(H^l,YP); D(H^l,H^k); Proj(H,H^l)

These relations do not define a complete tree yet, since the dominance relations between H^i and H^j on the one hand and H^k and H^l on the other are not specified yet. This leads to local ambiguities which will partly be resolved on the basis of the grammar and partly on the basis of pragmatic considerations (for instance if the context favours certain

[14] There is another option: the constituent can be parsed as part of the projection of a not yet identified following head. Such structures are similar to those involving leftward head movement, which are discussed below.

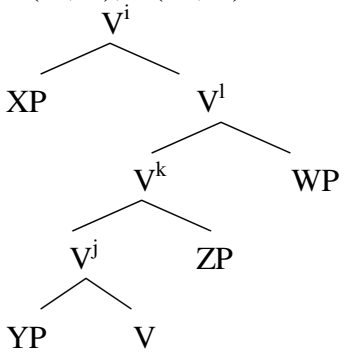
scopal relations between preverbal and postverbal material, compare section 2). Consider, for instance, the example in (6b'), repeated here as (26).

(26) dat we op vakantie gingen in die dagen volgens Marleens plan
that we on holiday went in those days according to Marleen's plan
'that, in accordance with Marleen's plan, we went on holiday in that period'

In this example, the string following the complementizer can be parsed along the lines sketched above, where *we* is XP, *op vakantie* 'on holiday' is YP, *in die dagen* 'in those days' is ZP and *volgens Marleens plan* 'according to Marleen's plan' is WP. The scopal relations between the PPs suggest that both postverbal PP adverbials are attached higher than the preverbal selected PP (see section 2). Suppose that the grammar tells the parser that the subject must be attached higher than the PP adverbials. Then, in addition to the relations in (25), the ones in (27a) are added to the parse, giving the complete tree in (27b).

(27) a. $D(H^i,H^l); D(H^k,H^j)$

b.



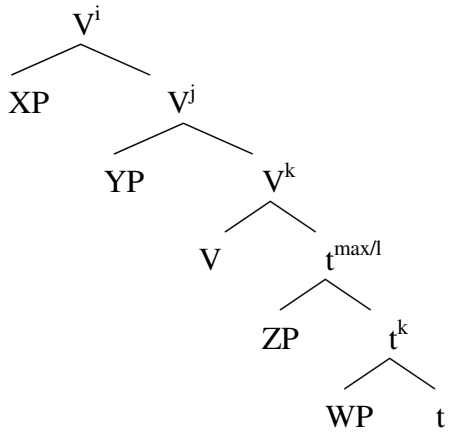
When the parser encounters the verb, there is a second option. It may again assume that the verb equals H, but that it has been moved (leftward). Which option it chooses - movement or not - depends on the grammar of the pertinent language. For instance, in a verb second language like Dutch, the parser will choose this option in a main clause like (28).

(28) Overigens gingen we op vakantie *t* in die dagen volgens Marleens plan
By-the-way went we on holiday in those days according to Marleen's plan
'By the way, in accordance with Marleen's plan, we went on holiday in that period'

If the verb has moved, the tree must remain right-branching until the parser introduces the verb's trace. Otherwise this trace would fail to be c-commanded by the moved verb. Subsequent maximal projections identified before the verb's trace is inserted can therefore not be attached to H's projection. Instead, they must be part of the projection of a lower head, say H2, which hosts the verb's trace. The parser will notate that a projection of H immediately dominates $H2^{max}$ and that $H2^{max}$ follows H. Internal to $H2^{max}$, precedence relations are notated in the usual way. Consider once more the string in (25a), repeated here as (29a), and suppose that this time the parser hypothesises that the verb has moved (this is indicated by M(V) below). Assuming the trace is inserted in final position, the parse is extended as in (29b); (29c) is the resulting minimal tree.

(29) a. XP YP V ZP WP
b. $P(XP,H)$; $ID(H^i,XP)$; $Proj(H,H^i)$
$P(YP,H)$; $ID(H^j,YP)$; $D(H^i,H^j)$; $Proj(H,H^j)$
H = V; M(V)
$P(H,H2^{max})$; $ID(H^k,H2^{max})$; $ID(H^k,H)$; $Proj(H2,H2^{max})$
$P(ZP,H2)$; $ID(H2^l,ZP)$; $Proj(H2,H2^l)$
$P(WP,H2)$; $ID(H2^k,WP)$; $D(H2^l,H2^k)$; $Proj(H2,H2^k)$
H2 = t

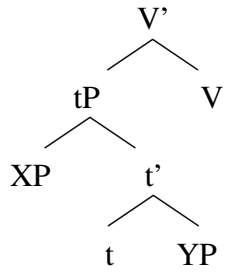
c.



For any maximal projections the parser identifies after having inserted the verb's trace, local ambiguities arise again. If they are attached to the projection of H2 (the verb's trace), c-command relations with respect to ZP and WP must be determined. If they are attached to the projection of H (the verb), c-command relations with respect to XP and YP. Which analysis is chosen is again determined by grammatical and pragmatic factors, but we will not pursue this here.

So, the parser can analyse the string under discussion either as a basic structure or as a structure involving head movement of the verb to the left. However, an analysis in which the verb has been moved rightward across one of its dependents (that is, a maximal projection contained in the projection of its trace) cannot be pursued successfully. In other words, a tree like (30) is ruled out.

(30) *



Consider why. According to (21d), a trace can only be inserted after the parser has encountered a moved element, an antecedent. This implies that if the parser hypothesises rightward verb movement, insertion of the trace must be in accordance with the information in (23b), repeated in

(31). The reason is that, at the time the parser encounters the verb, this information has already been stored (see the discussion above).

(31) P(XP,H); ID(H^i,XP); Proj(H,H^i)
P(YP,H); ID(H^j,YP); D(H^i,H^j); Proj(H,H^j)

Suppose that the parser tries out the analysis in (30) and inserts the verb's trace between XP and YP. In (30), YP is immediately dominated by a projection of the trace. This is only in accordance with the statements in (31) if the projection in question is H^j. (31) states ID(H^j,YP), and a node cannot be immediately dominated by two different nodes. If H^j is a projection of the trace, the trace must be H. However, if the trace is H, a conflict arises with another statement in (31), namely P(YP,H): YP does not precede the trace in the attempted analysis. Hence, this analysis is incompatible with already established information. Since such information may not be destroyed according to (21e), this analysis is in fact impossible. Hence, rightward head movement across one of the dependents of the moving head cannot be parsed. Put informally, the problem that can arise with rightward head movement is this: for all XPs the parser encounters before it encounters a head, it notates that they precede the head they are a dependent of. It is then impossible to insert a trace of the moved head before any such XP, because that would mean the XP in fact follows the head it is a dependent of, which conflicts with the previously established information about this XP.

Note that the impossibility of parsing this type of rightward head movement is very different from difficulties that may originate in local ambiguities. In the case of local ambiguities the parser can at some point pursue two possible analyses (which are computed either in parallel or serially, an issue on which we remain agnostic). However, because of the filler-driven strategy to gap postulation, the parser does not face a local ambiguity at the point where the trace of the rightward-moved head should be inserted. Hence, before it encounters the moved verb, the parser must notate that the XPs it identifies precede the head of their projection. As just discussed, this leads to unparsability in case one or more of these XPs in fact follow the trace.

The situation that arises under rightward head movement differs, therefore, from situations in which an XP can be put on a right branch of the projection of a preceding head, as well as on a left branch of the

projection of a head that is still to follow. Such cases do involve a local ambiguity. Consider, for instance, the following Dutch examples (cf. Frazier 1987):

(32) a. dat [$_{VP}$ [$_{NP}$ het meisje] [$_{V'}$ [$_{PP}$van de buren] houdt]]]
that the girl from the neighbours likes
'that the girl likes the neighbours'
b. dat [$_{VP}$ [$_{NP}$ het meisje [$_{PP}$ van de buren]] [$_{V'}$ glimlacht]]
that the girl from the neighbours smiles
'that the girl next door smiles'

In (32a), the PP *van de buren* is on a left-branch of the verbal projection, since it is a PP-complement selected by the verb. In (32b) the same PP is on a right branch of the projection of the noun *meisje*, it being a modifier of this noun here. In this case, the string is locally ambiguous when the parser postulates the PP. Consequently, two possibilities arise. If the PP is analysed as a dependent of the noun, the parser notates the relations ID(N^i,PP) and P(N,PP). If the PP is analysed as a dependent of the following verb, the parser notates the relations ID(H^i,PP) and P(PP,H) (where H will later be identified with the verb). This ambiguity is resolved in whatever way ambiguities are resolved in general. But in the case of (attempted) head movement to the right, the parser has already fixed the structure of the string for which it should have pursued a different analysis.

6. FELICITOUS RIGHTWARD HEAD MOVEMENT

The result that a head cannot move rightward across one of its dependents explains the problem that we started out with. As we have seen in section 2, basic syntactic structures are symmetric, but typologically head movement is not. N-to-D and V-to-C (verb second) movement are systematically leftward. They can, indeed, not be rightward since that would lead to unparsable structures. For example, the absence of languages with the order in (20d) (*adjective - numeral - determiner - noun) follows. The order of adjective, numeral and determiner indicates that the NP is head-initial underlyingly (as in (20b)). The illegitimate order must hence have been derived by

rightward movement of the noun across its dependents, which we have just shown to be impossible.

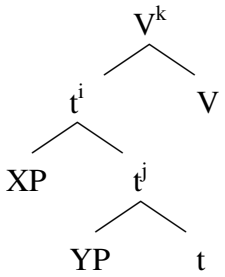
The argumentation in section 5 implies that head movement to the right should be possible if the syntactic relations already established do not have to be altered when the trace is inserted. In this section we will spell out under what circumstances this is possible. In the next section we present empirical evidence that bears out this prediction.

The simplest case involves rightward head movement in which the head does not cross any maximal projection. Consider once more the situation in which the parser has identified two maximal projections, XP and YP, but has not yet encountered the head. As noted in section 5, the information already stored at that point consists of the statements in (23b), repeated here in (33).

(33) P(XP,H); ID(H^i,XP); Proj(H,H^i)
P(YP,H); ID(H^j,YP); D(H^i,H^j); Proj(H,H^j)

If the parser now encounters the verb and hypothesises that this element has moved rightward, it can insert a trace without violating the statements in (33) by inserting it to the right of both XP and YP and identifying it with H. Thus, string-vacuous rightward head movement is possible: the tree in (34) is in accordance with (33) if *t* is H.

(34)



Indeed, string-vacuous rightward head movement (namely movement of the verb to a sentence-final C(omplementizer) position in the strictly head-final languages Japanese and Korean) is argued for by Whitman (1991). Whitman shows that certain properties shared by Verb Second languages (like Dutch) and Japanese/Korean can be explained by assuming that there is V-to-C movement in both.

Interestingly, there is a type of structure in which rightward head movement can be parsed in which it is not string-vacuous. Let us once

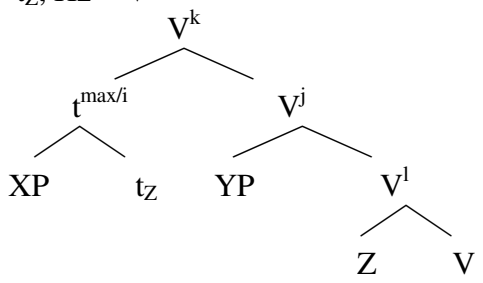
more consider the situation in which the parser has identified two maximal projections but has not yet encountered a head. Thus far we have assumed that these projections are dependents of the same head H. However, it is possible that they belong to different projections, which are projected from two distinct heads, H1 and H2. In that case, the following information will be notated (compare with (33)):

(35) P(XP,H1); ID($H1^i$,XP); Proj(H1,$H1^i$)
P(YP,H2); ID($H2^j$,YP); Proj(H2,$H2^j$)

Suppose that at this stage the parser encounters a verb, which it identifies as H2, plus another head, say Z, which occurs in an adjoined position to the verb. In that case no problems arise if the parser assumes that this adjoined head has undergone rightward movement across YP. The trace of this rightward moved head can be identified with H1 without leading to conflicts with the information already stored (given in (35)). This is so because the statement that YP precedes the head of its projection is with respect to H2 (the verb), not H1 (the trace of the adjoined Z head). The complete set of relations the parser establishes is given in (36a), with the corresponding minimal tree in (36b).

(36) a. P(XP,H1); ID($H1^i$,XP); Proj(H1,$H1^i$)
P(YP,H2); ID($H2^j$,YP); Proj(H2,$H2^j$)
ID($H2^k$,$H1^{max}$); D($H2^k$,$H2^j$); Proj(H2,$H2^k$);
Proj(H1,$H1^{max}$)
ID(V^l,Z); D(V^j,V^l); Proj(V,V^l); M(Z)
H1 = t_Z; H2 = V

b.



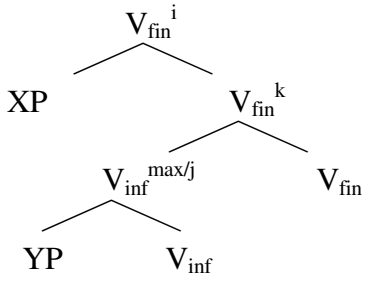
Thus it should be possible to have non-string-vacuous rightward head movement as long as the elements crossed are not dependents of the moved head, but of another head. In many cases, the requirement that

movement be to a c-commanding position (a 'higher' position in the tree, see section 3) will make it impossible to identify H1 in (36) with the trace of a head adjoined to H2. (Z in (36b) may not be 'lower' than its trace). We will show in the next section, however, that there is a limited set of structures which allow such identification without violating the ban on downward movement.

Note that the parser must be allowed to hypothesise that XP and YP belong to the projections of different heads for independent reasons. Consider the case of parsing embedded clauses in strictly head-final languages like Japanese. In that case, too, the maximal projections the parser identifies before encountering a head do not all belong to the projection of the same head. The difference with the case discussed above is that in these cases the head of which XP is a dependent is the higher head (i.e. the projection of H1 contains the projection of H2). So, if the embedded clause is an infinitival for example, the parser does not extend the set of statements in (35) to (36a) in this case, but to (37a). These are compatible with the minimal tree in (37b).

(37) a. $P(XP,H1)$; $ID(H1^{i},XP)$; $Proj(H1,H1^{i})$
$P(YP,H2)$; $ID(H2^{j},YP)$; $Proj(H2,H2^{j})$
$ID(H1^{k},H2^{max})$; $D(H1^{i},H1^{k})$; $Proj(H1,H1^{k})$;
$Proj(H2,H2^{max})$
$H1 = V_{fin}$; $H2 = V_{inf}$

b.



Thus, independently of rightward movement structures, the parser must be able to adopt the statements in (35) when it identifies two maximal projections before having encountered a head.

7. RIGHTWARD P-INCORPORATION AND V-TO-V RAISING IN DUTCH

In the previous section we have derived the following theorem:

(38) *Rightward Head Movement Theorem* (RHMT)
Rightward head movement is possible as long as no dependent of the moving head is crossed

In this section we will provide empirical evidence that confirms (38).

The first type of rightward head movement we will discuss concerns stranded prepositions in Dutch. Prepositions are said to be stranded when the NP complement of the P undergoes movement on its own, leaving the P behind. It can be argued that such stranded prepositions must incorporate into the verb in Dutch (see Sturm & Kerstens 1978 and Hoeksema 1991). The observation on which this claim is based is that they must be adjacent to the verb:[15]

(39) a. Daar$_i$ wil ik tijdens de lunch [t_i t_j] [over$_j$ praten]
that want I during the lunch about speak
'I want to speak about that during lunch'
b. *Daar$_i$ wil ik [t_i over] tijdens de lunch praten
that want I about during the lunch speak

This restriction cannot be reduced to the distribution of PPs in general, since PPs do occur in positions preceding adverbials (so nonadjacent to the verb), see (40a). It can also not be reduced to some prohibition on moving elements out of constituents that are not adjacent to the verb, since this is allowed in constructions like (40b).

[15] There are a few exceptions to this generalisation: locational PPs can intervene between the verb and a very small group of stranded prepositions. For other prepositions the generalisation is robust. Whether or not P-incorporation is obligatory or optional is irrelevant to the argumentation in this section, since the fact remains that it can be non-string-vacuous, as we will show below.

(40) a. Ik wil daarover tijdens de lunch praten
I want that-about during the lunch speak
b. Wat_i heb je [t_i voor mensen] tijdens de lunch gesproken?
what have you for people during the lunch spoken
'What kind of people did you talk to during lunch?'

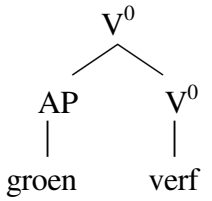
(Presumably, the trigger for P-incorporation into the verb is that the trace of the element that is extracted out of PP must be properly governed; this is a general demand on traces, see for instance Chomsky 1986 and Rizzi 1990. P is not itself a proper governor for elements to its left. However, after incorporation of P into V, V governs the trace; see Baker 1988).

The example of P-incorporation in (39a) is a case of string-vacuous rightward head movement, parsable on a par with (34). It is remarkable, however, that in other cases the movement may be non-string-vacuous: the preposition may cross a dependent of the verb. The relevant examples are given in (41).

(41) a. dat ik de deur daarmee groen verf
that I the door that-with green paint
'that I paint the door green with that'
b. *dat ik de deur groen daarmee verf
that I the door green that-with paint
c. de verf $waar_i$ ik de deur [t_i t_j] [$_V$ mee_j [$_V$ groen verf]]
the paint that I the door with green paint
'the paint with which I paint the door green'
d. de verf $waar_i$ ik de deur [t_i t_j] [$_V$ groen [$_V$ mee_j verf]]
the paint that I the door green with paint
'idem'

There is reason to believe that *groen verven* 'green paint' (i.e. 'paint green') is a complex predicate, that is, a complex V^0 category, generated by adjunction of the resultative to the verb, as in (42) (see Neeleman 1994 for motivation).

(42)



This immediately accounts for the possibility of (41c): although the stranded preposition is not adjacent to the verb, it is adjacent to the verbal complex. It has incorporated into the higher V^0 node of the complex predicate. Crucially, the lower V^0 segment is also a target for P-incorporation. This accounts for (41d). Note that the P-incorporation in (41d) does not involve lowering, since the incorporated preposition still c-commands its trace according to the definitions of c-command and domination of Chomsky 1986, given in (43).

(43) a. α c-commands β iff α does not dominate β and every γ that category-dominates α category-dominates β
b. α is category-dominated by β iff α is dominated by every segment of β

In (41d), the incorporated preposition is not category-dominated by V^0, since it is not dominated by every segment of V^0. This means there is no category that category-dominates the preposition but not its trace. Hence, the preposition c-commands its trace, as required.

What (41d) shows, then, is that rightward head movement may cross a dependent of the head into which incorporation takes place, in accordance with the RHMT in (38). The case in (41d) can be parsed as in (36) (Z is *mee*, YP is *groen*, V is *verf*, and XP is (the trace of) *waar*).

There is, in other words, no absolute adjacency condition on rightward incorporation. Given this, it is remarkable that rightward incorporation may not cross dependents of the incorporating head itself. Yet, this is what the RHMT states, and indeed it can be shown that movement of this type is impossible. In order to do so, we use a construction that involves non-string-vacuous V-movement to the right, namely Dutch V-to-V raising (see Evers 1975). V-to-V raising involves adjoining the head verb of an embedded infinitival clause to the right of the matrix verb. An example is given in (44).

(44) dat Jan [Marie de samba t_i] [zag dansen$_i$]

that Jan Marie the samba saw dance
'that Jan saw Marie dance the samba'

From the perspective of the parser, (44) involves a number of substrings to be analysed as XPs, followed by two verbs. This string can be analysed by the parser on a par with (37b), except that the infinitive is moved and H2 is identified with its trace. The movement of the infinitive does not cross any maximal projections, and hence it can be parsed without problems (compare the discussion of (33) and (34)).

Although we have shown that there is no absolute adjacency condition on rightward head movement, the embedded verb in a V-to-V raising construction cannot move across any of its own dependents. Evidence for this claim can be based on an observation by Reuland (1990). In Dutch, a postverbal PP can take scope over a preverbal adverbial (as expected given the symmetry of basic structures). The example in (45) can mean both that it was the case for some time that Jan frequently hampered the project or that it was frequently the case that Jan hampered the project for some time (see also section 2).

(45) dat Jan het project regelmatig hinderde gedurende een tijdje
that Jan the project frequently hampered for some time

Consider now an infinitival version of (45) and suppose the infinitive heading it adjoins to a higher verb, across the postverbal PP. We would then still expect the same ambiguity. However, in a V-to-V raising construction like (46) only one reading is available, namely the reading in which the adverbial takes scope over the PP.

(46) dat ik Jan het project regelmatig gedurende een tijdje zag hinderen
that I Jan the project frequently for some time saw hamper

This means that, apparently, the trace of the raised verb in (46) cannot precede the PP, as in (47a). The interpretation of (46) is only compatible with the analysis in (47b), with the trace following both the adverbial and the PP. As noted in section 2, in case of two prehead modifiers, scope relations go from left to right. Hence the unambiguous scopal

relation in (46). (For more evidence for the adjacency requirement between the two verbs in cases of V-to-V raising see Van Riemsdijk 1998).

(47) a. *dat ik [Jan het project regelmatig t_i gedurende een tijdje] [zag hinderen$_i$]
b. dat ik [Jan het project regelmatig gedurende een tijdje t_i] [zag hinderen$_i$]

So, while there is no adjacency condition as such on rightward incorporation, as shown above (see (41d)), adjacency is required in (47). It seems fairly problematic to derive this difference from syntactic principles, but it follows directly from the parsing strategies we have assumed. From these strategies we have derived the theorem in (38): a head cannot move rightward across one or more of its own dependents. This, however, is precisely what has happened in (47a). Hence, (47a) is unparsable. No such problems arise in (47b), where no maximal projections are crossed, or in (41d), where the maximal projection crossed is not a dependent of the moving head.

8. RIGHTWARD XP-MOVEMENT

The kind of parsing problems that make rightward head movement impossible under some circumstances do not arise in the case of rightward XP-movement. The reason for this is that precedence relations which mention a particular XP are not notated until that XP has been identified. This implies that in the case of rightward XP-movement statements must be added, but no already stored statements ever would have to be revised. In contrast, abstract heads are mentioned in precedence relations before an actual head is encountered (in head final constructions at least), which may lead to the problems discussed above.

Consider for example a situation in which the parser has identified two maximal projections and a head. As explained in section 5, the parse at that stage contains the information in (24b), repeated in (48).

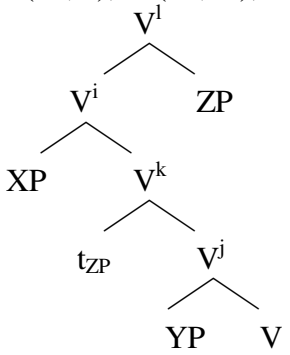
(48) P(XP,H); ID(H^i,XP); Proj(H,H^i)
P(YP,H); ID(H^j,YP); D(H^i,H^j); Proj(H,H^j)

H = V

If the parser now identifies a third maximal projection, ZP, it may hypothesise that ZP has moved from a position between XP and YP and insert a trace there without contradicting any of the statements in (48). After all, these statements do not mention ZP or its trace. One of the possible resulting parses is that in (49).

(49) a. P(XP,H); ID(H^i,XP); Proj(H,H^i)
P(YP,H); ID(H^j,YP); D(H^i,H^j); Proj(H,H^j)
H = V
P(H,ZP); ID(H^l,ZP); D(H^l,H^i); Proj(H,H^l); M(ZP)
P(t_{ZP},H); ID(H^k,t_{ZP}); D(H^i,H^k); D(H^k,H^j)

b.



In conclusion, rightward XP-movement across a maximal projection need not lead to parsing difficulties. Following Rochemont & Culicover (1990), we assume that such movement indeed exists. Rochemont & Culicover argue that, whereas extraposition of PPs and CPs should be analysed in terms of base generation and an interpretational rule, heavy NP shift is an instance of rightward adjunction.[16]

If rightward movement indeed exists, it is unexpected that, as already shown by Ross (1967), it is much more local than leftward movement. Culicover & Rochemont (1990) show that (once extraposition of PPs

[16] Extraposition is the phenomenon that a complement PP or CP, or a relative CP, occurs further to the right instead of directly after the noun it is a dependent of; examples are given in (i)-(ii). Heavy NP shift involves movement to the right of a 'heavy', focused, nominal constituent; an example is given in (iii).

(i) John saw the girl yesterday with the red umbrella
(ii) John saw the girl yesterday that has a red umbrella
(iii) (Who did John see yesterday?) John saw yesterday THE GIRL WITH THE RED UMBRELLA

and CPs is discarded as an instance of movement) the proper locality condition on rightward movement is that it is phrase-bound. For example, heavy NP shift is not possible out of clauses, as shown by the data in (50), and it is not possible out of PPs, as shown by (51) (from Rochemont 1992:387).

(50) a. John wanted [$_S$ PRO [$_{VP}$ to study t_i carefully [the entire book of Revelation]$_i$]]
b. *John wanted [$_S$ PRO [$_{VP}$ to study t_i]] dearly [the entire book of Revelation]$_i$

(51) a. *Mary put the money [on t_i] yesterday [a table that was standing at the entrance to the hall]$_i$
b. *John threw a look [at t_i] as he was walking by [a man who was standing outside his office]

An advantage of the antisymmetry framework (see sections 1-3) is that this contrast between leftward and rightward XP-movement can be made to follow from the assumption that the apparent cases of rightward movement do not, in fact, involve such movement at all (but rather leftward movement of other material, see Kayne 1994:72-73). However, the contrast can be made to follow from the design of the human parser as well, as suggested before by Fodor (1978), Rochemont (1992) and Davis & Alphonce (1992). Consider how.

It is generally assumed that in analysing an input string, the parser closes off certain units of already parsed structure and removes them from short term memory (see Kimball 1973). What this means is that once the internal structure of a syntactic unit has been determined, and its semantics has been calculated, the parser treats that unit as an atom. It removes all statements that describe the internal structure of the unit in question from the set of statements in its storage component. It replaces all this information with a single symbol with the appropriate semantics. As will be clear, this procedure reduces the pressure on short-term memory.

Closure implies that the restriction on inserting traces in already parsed structure is even stricter than we already argued it to be. It is not only impossible to introduce traces if this would require alteration of already established information about the structure, but it is also impossible to introduce traces into those syntactic units that have been

closed. If the unit of closure is the phrase, as Kimball (1973) and Rochemont (1992) suggest, the locality of rightward XP-movement follows straightforwardly from the filler-driven strategy the parser applies in dealing with filler-gap dependencies.

The ban on reopening phrases that have been closed discriminates sharply between rightward XP-movement and leftward XP-movement. Recall that the parser can only insert traces after it has hypothesised that some constituent is an antecedent. If the trace follows its antecedent, in principle it can be inserted in any phrase, since insertion will coincide with the building up of that particular phrase. If the trace is to precede its antecedent, however, the antecedent and its trace must be in the same phrase, or else a closed phrase would have to be reopened, which is impossible.

Consider the structures in (52), which correspond to the examples in (50a) and (50b) respectively.

(52) a. $[_S$ NP V $[_S$ $[_{VP}$ V t_i AdvP $NP_i]]]$
b. *$[_S$ NP V $[_S$ $[_{VP}$ V $t_i]]$ AdvP $NP_i]$

If the parser hypothesises that the sentence-final NP is an antecedent, it may insert a trace in the phrase that it puts the NP in, but not in an already parsed, and therefore closed, phrase. The parser treats closed phrases as atoms, and can therefore not modify their internal structure. However, this is exactly what would be necessary if a trace is to be inserted in a closed phrase.

So, the parser can handle a structure like (52a), since the embedded VP has not been closed at the moment the shifted NP is encountered. A trace may consequently be inserted in this VP. In (52b), however, the moved NP follows a constituent that must be part of the matrix VP. The adverb separating the heavy NP and its trace, *dearly* in (50b), must be a modifier of the matrix verb *want* for semantic reasons. This implies that the parser must already have closed the embedded VP when it reaches the heavy NP. This in turn implies that it can only insert the trace of the heavy NP in the matrix VP, which will not result in a grammatical structure (since the shifted NP is not thematically related to the matrix verb).

The same line of reasoning applies to the examples of heavy NP shift out of PPs in (51). The parser has only one option when it encounters the adverbial following the preposition in these examples. The grammar

determines that the adverbial cannot be a complement of the preposition. It must go with the matrix verb. Again, this means that the parser must have closed the PP when it encounters the heavy NP. It cannot reopen this PP, and therefore the trace of the heavy NP must be inserted in the matrix clause. Hence, heavy NP shift out of PPs is impossible.

As noted, leftward XP-movement is not restricted in this way. Consider, for example, a string that should be assigned the structure in (53).

(53) $[_S\ XP_i\ YP\ V\ [_S\ ZP\ V\ [_S\ WP\ V\ t_i]]]$

Once the parser hypothesises that XP has been moved, it must introduce a trace somewhere. It can do so in any of the phrases, however deeply embedded, that it is working on. The introduction of the trace finds place at the moment the structure of that particular phrase is determined, and so it does not require the reopening of closed material.

Note that from this proposal it follows that a type of movement that necessarily crosses a phrasal boundary can only take place to the left. This type of movement, when taking place to the right, would be unparsable. Hence, it follows that movement to the specifier position of CP, for example, must be leftward because such movement necessarily crosses the VP boundary. As far as we know, operator movement to spec-CP (like wh-movement in questions) is indeed invariably leftward.

So, certain instances of rightward movement cannot be parsed, and are therefore impossible. This phenomenon must be differentiated from the phenomenon of garden paths, which involves constructions that are difficult but not impossible to parse. Garden paths can occur when there is a local ambiguity in the analysis of a string. As noted in sections 4 and 5, the parser postulates two possible analyses in such cases, which are either pursued in parallel or serially. One might think this option affects the explanation of the boundedness of rightward XP-movement just given. Suppose that in an example parallel to (50) heavy NP shift crosses material that induces a local ambiguity. For example, suppose that the adverbial it crosses can either be attached in the matrix or in the embedded clause, as in (54).

(54) John expected Bill to meet his favourite uncle from Cleveland yesterday

(i) expect yesterday; (ii) meet yesterday

If heavy NP shift takes place, the adverbial must be construed as part of the embedded clause, indicating that even in cases like this rightward XP-movement is phrase-bound:

(55) John expected Bill to meet yesterday his favourite uncle from Cleveland
(i) *expect yesterday; (ii) meet yesterday

Data like (55) do not really form a problem for the analysis above, however. The point is that the local ambiguity induced by *yesterday* gives rise to two possible parses, neither of which can accommodate heavy NP shift into the matrix clause. In the first parse, *yesterday* is a matrix adverbial. This implies that the embedded clause will have been closed when the parser identifies the shifted NP. Consequently, in this parse the NP must have been shifted from within the matrix clause (which, as noted, is impossible since this NP is not selected by the matrix verb). In the second parse, *yesterday* is attached to the embedded clause. This means that when the parser identifies the shifted NP the embedded clause need not have been closed yet, and hence a trace can be inserted in it. (Of course, if the embedded clause *has* been closed at that time, the trace must again be inserted in the matrix clause, which leads to the problem just mentioned).

9. CONCLUSION

We have argued that base-generated structures are in principle symmetrical. With respect to movement, however, two asymmetries can be observed. First, head movement to the right seems to be impossible when the head crosses its own dependents, while for head movement to the left (for instance in cases of Verb Second) this is not a problem. Second, XP-movement to the right is extremely local (namely phrase-bound), while for XP-movement to the left this is not necessarily the case. We have argued that both these asymmetries do not impede the basic symmetry of syntax, but are consequences of parsing strategies. In particular, both types of asymmetry can be traced back to the parser's inability to adapt already stored information to the consequences that

the introduction of a trace would have in computing certain cases of rightward movement.

Our analysis implies that the apparent ungrammaticality of certain syntactic structures should not be accounted for by syntax proper (that is, by the theory of competence), but rather by the theory of performance. The latter acts as a filter on possible linguistic representations. It might seem that an explanation in these terms is less desirable than an explanation that invokes syntactic principles only. However, the assumption that there is a system of performance which renders part of the output of the system of competence unacceptable allows for an overall simplification of the theory (see Kluender 1998 for a similar argument in connection with other restrictions on movement). The general rationale for such a modular approach is that it is better to have two simple modules than it is to have a single complicated one. In the case at hand, we believe that the theory of syntax is simplified by transferring some of its load to the parser.

ACKNOWLEDGEMENTS

For useful comments we are indebted to Carl Alphonce, Arthur Dirksen, Lyn Frazier, Edward Gibson, Hans van de Koot, Jan Odijk, Colin Phillips, Tanya Reinhart, Eddy Ruys, Fred Weerman, and audiences at Utrecht University (Linguistics in the Netherlands), University of California at Santa Cruz (WECOL) and University of Cologne (Linguistisches Arbeitskreis).

REFERENCES

Baker, M.: 1988, *Incorporation*, University of Chicago Press, Chicago.
Berwick, R. and A. Weinberg: 1985, 'Deterministic parsing: a modern view', in S. Berman, J.-W. Choe and J. McDonough (eds.) *Proceedings of NELS 18*, GLSA, Amherst, pp. 15-33.
Büring, D. and K. Hartmann: 1997, 'The Kayne mutiny', in D. Beerman, D. LeBlanc and H. van Riemsdijk (eds.) *Rightward movement*, Benjamins, Amsterdam, pp. 59-80.
Chomsky, N.: 1986, *Barriers*, MIT Press, Cambridge, Mass.
Chomsky, N.: 1995, *The minimalist program*, MIT Press, Cambridge, Mass.
Cinque, G.: 1996, 'The 'antisymmetric' programme: theoretical and typological implications', *Journal of Linguistics* **32**, 447-464.

Culicover, P. and M. Rochemont: 1990, 'Extraposition and the complement principle', *Linguistic Inquiry* **21**, 23-47.
Davis, H. and C. Alphonce: 1992, 'Parsing, linear asymmetry, and *Wh*-movement', in *Proceedings of NELS 22*, GLSA, Amherst, pp. 87-100.
Ernst, T.: 1994: 'M-command and precedence', *Linguistic Inquiry* **25**, 327-335.
Evers, A.: 1975, *The transformational cycle in Dutch and German*, Ph.D. dissertation, Utrecht University.
Fodor, J.: 1978, 'Parsing strategies and constraints on transformations', *Linguistic Inquiry* **9**, 427-473.
Frazier, L.: 1987, 'Syntactic processing: evidence from Dutch', *Natural Language and Linguistic Theory* **5**, 519-559.
Frazier, L.: 1993, 'Processing Dutch sentence structures', *Journal of Psycholinguistic Research* **22**, 83-108.
Frazier, L. and C. Clifton: 1995, *Construal*, MIT Press, Cambridge, Mass.
Frazier, L. and G. Flores d'Arcais: 1989, 'Filler driven parsing: a study of gap filling in Dutch', *Journal of Memory and Language* **28**, 331-344.
Frazier, L. and K. Rayner: 1982, 'Making and correcting errors during sentence comprehension: eye movements and the analysis of structurally ambiguous sentences', *Cognitive Psychology* **14**, 178-210.
Gibson, E.: 1991, *A computational theory of human linguistic processing: memory limitations and processing breakdown*, Ph.D. dissertation, Carnegie Mellon University, Pittsburgh.
Gibson, E.: 1998, 'Linguistic complexity: locality of syntactic dependencies', *Cognition* **68**, 1-76.
Gorrell, P.: 1995, *Syntax and parsing*, Cambridge University Press, Cambridge.
Greenberg, J.: 1966, 'Some universals of grammar with particular reference to the order of meaningful elements', in J. Greenberg (ed.), *Universals of Language* (2nd edition), MIT Press, Cambridge, Mass., pp. 73-113.
Hoeksema, J.: 1991, 'Complex predicates and liberation in Dutch and English', *Linguistics and Philosophy* **14**, 661-710.
Just, M. and P. Carpenter: 1992, 'A capacity theory of comprehension: Individual differences in working memory', *The Psychological Review* **99**, 122-149.
Kaan, E. and L. Stowe: 'Storage and computation in sentence processing: a neuroimaging perspective', This volume.
Kayne, R.: 1994, *The antisymmetry of syntax*, MIT Press, Cambridge, Mass.
Kimball, J.: 1973, 'Seven principles of surface structure parsing in natural language', *Cognition* **2**, 15-47.
Kluender, R.: 1998, 'On the distinction between strong and weak islands: a processing perspective', in P. Culicover and L. Mcnally (eds.), *Syntax and semantics 29: The limits of syntax*, Academic Press, San Diego, pp. 241-279.

Koster, J.: 1974, 'Het werkwoord als spiegelcentrum', *Spektator* **3**, 601-618.
Lamarche, J.: 1991, 'Problems for N0-movement to NumP', *Probus* **3**, 215-236.
Marcus, M., D. Hindle and M. Fleck: 1983, 'D-theory: talking about talking about trees', in *Proceedings of the 21st annual meeting of the Association for Computational Linguistics*, MIT Press, Cambridge, Mass., pp. 129-136.
Neeleman, A.: 1994, *Complex predicates*, Ph.D. dissertation, Utrecht University.
Neeleman, A. and F. Weerman: 1999, *Flexible syntax*, Kluwer, Dordrecht.
Pritchett, B.: 1992, *Grammatical competence and parsing performance*, University of Chicago Press, Chicago.
Reuland, E.: 1990, 'Head movement and the relation between morphology and syntax', in G. Booij and J. van Marle (eds.), *Yearbook of morphology 1990*, Foris, Dordrecht, pp. 129-161.
Riemsdijk, H. van.: 1998, 'Head movement and adjacency', *Natural Language and Linguistic Theory* 16, 633-678.
Riemsdijk, H. van and E. Williams: 1986, *Introduction to the theory of grammar*, MIT Press, Cambridge, Mass.
Rizzi, L.: 1990, *Relativized minimality*, MIT Press, Cambridge, Mass.
Roberts, I.: 1985, *The representation of implicit and dethematized subjects*, Ph.D. dissertation, University of Southern California, Los Angeles.
Rochemont, M.: 1992, 'Bounding rightward A'-dependencies', in H. Goodluck and M. Rochemont (eds.), *Island constraints*, Kluwer, Dordrecht, pp. 373-397.
Rochemont, M. and P. Culicover: 1990, *English focus constructions and the theory of grammar*, Cambridge University Press, Cambridge.
Ross, J.: 1967, *Constraints on variables in syntax*, Ph.D. dissertation, MIT.
Sturm, A. and J. Kerstens: 1978, 'Over verklaren in de taalkunde', in P. van Putte and H. Verkuyl (eds.), *Nieuwe tegenstellingen op Nederlands taalgebied*, Instituut de Vooys, Utrecht, pp. 130-172.
Weinberg, A.: 1988, *Locality principles in syntax and in parsing*, Ph.D. dissertation, MIT, Cambridge, Mass.
Whitman, J.: 1991, 'String vacuous V to COMP', *GLOW Newsletter* **26**, 62-63.