Bericht Nr. 83, Januar 1997

Arbeitspapiere des
Sonderforschungsbereichs 340

ISSN 0947-6954/97

Sprachtheoretische Grundlagen für die Computerlinguistik

Tom Cornell:

Representational Minimalism

Universität Stuttgart Universität Tübingen
IBM Deutschland Informationssysteme GmbH

Arbeitspapiere des SFB 340
Bericht Nr. 83, Januar 1997

REPRESENTATIONAL MINIMALISM

TOM CORNELL

1. Introduction

In this paper we examine the problem of X^0 chains from within an explicitly *representational* framework. That is, we will concern ourselves here primarily with the nature of syntactic *structures* and with saying, as directly as possible, what the syntactic structure of a given expression is. Therefore, we will depend more on model theoretic or denotational means, as opposed to proof theoretic or operational means, according to which one defines the valid sound-meaning pairs as the end products (fixed points) of a derivation. There are several reasons for doing this. Two stand out in particular. First, within the corpus of syntactic theory it is widely supposed that X^0 chains are problematic for representational accounts (e.g., Rizzi (1990), Brody (1995)) which appeal to chains as objects, rather than appealing to a series of movement transformations. The reason is that the base position of a head and its target of adjunction are required to obey certain locality conditions. But if the target of the adjunction is itself subsequently moved—and this process can be iterated almost at will—then a head can be carried very far away from either of the two positions on which the head's chain's well-formedness depends. Therefore, stating locality conditions on this structure appears to be quite difficult, since the condition of locality is not meant to apply to the structure intervening between the final position of the head and its trace.

The second reason, which may in fact be interestingly related to the first, comes from more formal research into the theory of grammars. Recently, Rogers (1994) and Kracht (1995) have presented decidable logics

---

To appear in: Kolb, H.-P. & U. Mönnich (eds.), *The Mathematics of Syntactic Structure*.

ISSN 0947-6954/97

within which much of the content of GB-style syntactic theory appears to be formalizable. These two works are completely independent, and the logics developed there (monadic second order predicate logic and modal dynamic logic, respectively) are very different. Nonetheless, both authors turn to Rizzi's *Relativized Minimality* (Rizzi 1990), perhaps the archetypal representationalist presentation of GB theory, as a source of examples to formalize. Since Rizzi's presentation is entirely dependent on well-formedness conditions on representations, it is naturally well-suited to model theoretic approaches to syntax, which take the well-formed syntactic structures of a language to be the set of models for some axiomatization, and identify grammars with such axiomatizations. As noted, both logics are decidable and, remarkably enough, *only describe sets of trees yielding context-free languages* (the so-called *recognizable* sets of trees). In fact, Rogers' system can define exactly the recognizable sets of trees (not just most of them).

However, it is known that the structures proposed in mainstream GB theory for head movement constructions do *not* form a recognizable set (Huybregts 1984, Shieber 1985).[1] They exhibit the kinds of crossing dependency patterns whose archetype is found in the artificial "copy language", the language all of whose strings are of the form ww, for some lexicon with at least two distinct symbols.[2] This language is known not to be context-free, and therefore it cannot be the yield of any recognizable set, and therefore it cannot be the yield of any set of trees definable in either Rogers' or Kracht's logics. To the extent that the formalizations presented in these works are complete and faithful formalizations of Rizzi (1990) (and Rogers' formalization is indeed very comprehensive, in particular in its coverage of a great variety of movement constructions), it could conceivably be argued that the system of *Relativized Minimality*, and by implication representational approaches to transformational grammar, are in principle too weak to define the kinds of structures we

[1]These papers actually demonstrate the stronger result, that *no* analysis of the strings of Swiss German, in particular, is possible with merely context-free power. The weaker result, that these languages require extra power to characterize the particular *structures* that GB theory, for example, would assign to these strings, does not require very exotic data and applies quite generally to head movement constructions.

[2]So, for example, given the "vocabulary" $\{0, 1\}$, the string `010010` would be in the copy language; in our terms we will think of the initial substring `010` as the complex head formed by head raising and the final substring `010` as the arguments of these heads left behind by the moved heads. The key to this construction is that the hierarchical order of the complex head is reflected in the hierarchical order of its collection of stranded arguments.

see in languages such as Dutch and Swiss German head movement constructions. We intend to demonstrate here that such conclusions would be premature; there is no deep connection between representational approaches and context-free languages.

However, from the standpoint of purely formal research into the theory of grammar, these results do strongly suggest that we should focus particular attention on logics which minimally extend Rogers' and Kracht's in such a way that they can define these non-recognizable tree sets. That is, head movement constructions have acquired a position of considerable importance in the study of formal foundations of grammar. To add to the problem, it appears that very little needs to be added to these logics to make them undecidable, so it is a very real problem to develop *decidable* logics which push us beyond the context-free boundary.

In Cornell (1996), we examined a purely derivational definition of the language $\mathcal{L}_{ww}$, the copy language over the set $\{0, 1\}$, couched in the minimalist framework of Chomsky (1995), broadly construed. That account, like the more comprehensive framework for derivational minimalism of Stabler (1996) on which it is largely based, is essentially algebraic, rather than deductive. So this line of research has not yet led to a suitable logic for languages with head movement. If it is the case that formal grammars of well-formedness conditions are inherently weak in this regard, as the derivationalists (see, e.g., Chomsky (1995)) have sometimes proposed, then there may even be an obstacle in principle to the realization of the model theoretic program of syntax. The full realization of this subprogram then depends on the development of a truly deductive version of the derivational system, and on the development of a model theory in terms of which that deductive system can be shown to be at least sound and complete.

In this paper we hope to push this program forward a little by examining in some detail the structures that would be needed to solve the locality problem for X^0 chains in a representationalist framework. We will do so in a manner which shows strong connections with our previous derivational characterization of $\mathcal{L}_{ww}$, suggesting that an equivalence could perhaps be proved for a suitably constrained minimalist calculus. The key to this problem is of course the relation of the operation *Move* to the definition of well-formed chains. If we can show in particular that we can capture $\mathcal{L}_{ww}$ with a system of well-formedness conditions, focusing in particular on the conditions related to X^0 chains, then we should be in a good position to construct a deductive calculus for which derivational minimalism provides the proof theory and representational minimalism the model theory.

1.1. **A Representational Approach to Head Chain Locality.** Our solution to the locality problem for X^0 chains is based on the following very simple observation. Suppose we are given a collection of syntactic *heads* (projected from the lexicon into some sort of multiset or sequence, a "numeration" in the terminology of Chomsky (1995)). Each such head is a categorized sound-meaning pair (with one or the other member possibly null). We take a sound-meaning pair to be any object to which we can apply either of a pair of projection functions π_{pf} or π_{lf} to extract just the phonologically or just the semantically inerpretable content of that structure. A particular choice of numeration acts like a set of coordinates to pick out from among the well-formed syntactic structures a small set whose members are built out of just that lexical material. These structures are again sound-meaning pairs, that is, formal objects to which we can apply π_{pf} or π_{lf}. The problem of evaluating one of these projection functions at a particular structure is the *spell-out problem* (broadly construed to involve "semantic spell-out" as well as phonological spell-out). If the structure involved is an explicit record of a derivation, e.g., a sequence of phrase markers, then the spell-out problem is just the problem of determining which members of this sequence correspond to the PF and LF outputs. In a monostratal representationalist theory, on the other hand, where both LF and PF are embedded in a tree structure of some sort, then the problem presumably involves a recursion on a tree "stripping off" all features relevant to one level or the other, but keeping the hierarchical structure intact.[3]

Given a head X, there are various ways of extending it in the syntax. Chief among these are what we might think of informally as "vertical" extension, so as to attach another structure to X, and "horizontal" extension, so as to attach X in the domain of several distinct heads. Here we will refer to the former as phrasal extension and the latter as chain formation. These are our main syntactic means for regulating the distribution and consumption of lexical resources in a syntactic structure. These tools introduce their own problems, chief among them being the *locality problem* for chains. This problem is central. It really defines the

[3]There are of course other possibilities. For example we could have a multistratal representationalist theory—which is indeed the standard sort—which stated declaratively the definition of tuples of structures $\langle \mathrm{DS}, \mathrm{SS}, \mathrm{PF}, \mathrm{LF} \rangle$, in which case the spell-out problem is solved rather trivially by having π_{pf} of such a tuple equal PF and π_{lf} equal LF.

relation between chains and phrases, in the following sense. Essentially, every link of a chain corresponds to a region of the phrasal structure, and that region cannot be arbitrarily large. So a solution to the locality problem for chains is a mapping from chains into the phrase structure which satisfies an appropriate definition of locality for every chain link.

The key to our treatment of X^0 chains starts from the recognition that these problems—of spell-out and of locality—are in fact distinct. The reason that the locality problem for X^0 chains has appeared difficult for representational theories of syntax is that the abstract notion of an individual syntactic head has been collapsed with the concrete position in which its phonology or orthography appears in the tree structure. In particular, suppose a dependent head X forms a chain with a host head Y, and Y in turn forms a further chain. The "trace symbol" t_{Y} no longer displays any evidence that it contains the upper point of the chain link (X,t_{X}); the information that the trace of the host is relevant to the computation of locality for the chain of the dependent is lost in standard displays of syntactic structure. It is a fact about the syntactic structure of the sentence which is simply missing from traditional representations of syntactic structure. Under these rather artificial circumstances it is indeed challenging to give a simple tree-geometrical account of how the spell-out position and the base position are related in a simultaneously local and nonlocal way. (Though we will offer such an account in §5.) Here we will not try to relate the two positions directly; we will maintain an *abstract* notion of head *through which* the two positions will be related. That is, given a head X we can discuss how or where X is to be spelled out and we can discuss the locality problem for chain links belonging to X and, crucially, we can discuss these problems separately.

Our working assumption is that if there is a constituent of syntactic structure which is worth talking about, it should be represented. Unadorned phrase structure trees are admirably explicit about their constituent structure as long as those constituents are phrases. But when the theoretically relevant constituent parts of a syntactic structure are no longer phrases, per se, but rather XPs (seen not as categorized substrings of the sentence but rather as the markers of headedness in phrase structures) and chains, then we need to make these constituents as explicit in our structures as phrases are in phrase structure trees. Labeled traces are a start, but, as noted, they are inaccurate and unreliable indicators of the structure of head chains. We prefer here to represent X^0 chains and XPs and their relations explicitly. In such structures the solutions to the problems of chain well-formedness and the distribution of features are now straightforward. Moreover, from our slightly unfamiliar structures the more familiar sorts of tree structures can easily be recovered, and having done so a purely tree-based representationalist theory of X^0 chain locality can be extracted, if desired.

Our approach, then, essentially combines ideas from Chomsky (1995) with ideas from Brody (1995). ¿From Chomsky (via Stabler) we take our basic picture of lexical items and in particular the formal features of lexical items as coming in matching pairs which must be placed in *checking configurations*; movement or chain formation is then driven exclusively by the need to place clashing feature pairs in appropriate configurations. From Brody we take the idea that chains can be represented independently of the representations we might consider as our final products (his "pre-syntactic chains"). Finally, all of this is couched in a formal framework derived from Stabler (1996), which emphasizes feature checking as an instance of *linear resource sensitivity* as a characteristic of minimalist syntactic systems: features, once checked, are used up and cannot be copied in any way so as to be checked again somewhere else; in this sense there is a strong connection between the minimalist program and categorial grammars which emphasize the role of "sub-structural" logics (Moortgat 1996, and refs.).

2. The X^0 Chain Locality Problem from a Minimalist Perspective

2.1. Features and Lexical Resources Reviewed. We recall here the basic definitions of Cornell (1996) as they relate to features and lexical items. Features come in two varieties which we think of generally as being *attractors* or *attractands*. Further, attractor features can be sorted into strong and weak. More formally, we begin with a stock of primitive atoms, and apply to a basic atom x either the operation $+$ (forming the attractand) or the operation $-$ (forming the corresponding attractor), and to a feature of the form $-x$ we apply either the operation s (strengthening) or the operation w (weakening). We will use a simplified notation, however, (adapted from Stabler (1996)) in which $+x$ is written simply x, $w(-x)$ is written simply $\overline{\mathsf{x}}$, and $s(-x)$ is written $\overline{\mathsf{X}}$. Normally we would distinguish features which are involved in selection from other features (as is done in Stabler (1996), for example). So, we would distinguish *categorial* attractands from non-categorial attractands (like wh or $\mathsf{nominative\text{-}case}$), and similarly we would distinguish selectional features from other attractors. Here, however, we make use only of categorial attractands and selectional

$\dfrac{\mathsf{s},\overline{\mathsf{F}}}{\epsilon}$ (s)	$\dfrac{\mathsf{f},\overline{\mathsf{F}},\overline{\mathsf{a}}}{0}$ ($\mathsf{f_a}$)
$\dfrac{\mathsf{a}}{0}$ (a)	$\dfrac{\mathsf{f},\overline{\mathsf{F}},\overline{\mathsf{b}}}{1}$ ($\mathsf{f_b}$)
$\dfrac{\mathsf{b}}{1}$ (b)	$\dfrac{\mathsf{f}}{\epsilon}$ ($\mathsf{f_0}$)

TABLE 1. Lexicon for $\mathcal{L}_{ww}$.

attractors. Attractors and attractands together are referred to as *formal features*.

Here our primitives will be just $\{\mathsf{s}, \mathsf{f}, \mathsf{a}, \mathsf{b}\}$, the basic categories from the grammar for the copy language $\mathcal{L}_{ww}$ as presented in Cornell (1996). Intuitively, the category s is the category of complete expressions, i.e., strings of the language being defined; f is a recursive functor category, roughly analogous to the V+Infl complex of natural languages, and a and b are "constant" categories, which themselves take no arguments, and which serve as the primary (nonrecursive) arguments of the f categories.

Complex categories we take to be lists of features containing exactly one category feature (i.e., categorial attractand in our present terminology). We take the order of (selectional) attractors to be significant, essentially encoding the internal/external argument distinction.

Normally, lexical items are categorized sound-meaning pairs. In our more artificial setting, however, we make use only of *orthographic* features, which take the place of both semantic and phonological features. These are our sole *interpretable features*, that is, we take the two projection functions π_{pf} and π_{lf} to be equal, both yielding the orthographic string associated with a structure. Here our orthographic features will simply be characters (one-element strings) drawn from the alphabet $\{0, 1\}$. So here lexical items will simply be categorized orthographic strings (including the empty string ϵ).

The lexicon for $\mathcal{L}_{ww}$ is given in Table 1. Formal features are written above the line, interpretation features are written below it. The symbols in parentheses will serve us later as abbreviations for these lexical items. So we might think of the "word" $\mathsf{f_a}$ as being "pronounced" 0, and being of category $[\mathsf{f},\overline{\mathsf{F}},\overline{\mathsf{a}}]$.

As noted, we regard selectional features as ordered left-to-right. So, for example, all argument-taking categories ("functor" categories) in this grammar take f-phrases as internal arguments, and a- and b-phrases appear only as external arguments.

The fact that the strings 0 and 1 have multiple categories should not appear problematical; this is required to derive the copy language but is not a necessary feature of the crossing dependency structures which are of actual linguistic interest. The reader wishing to make this language seem more intuitive as an abstraction of natural language properties might take the symbols 0 and 1 to be agreement morphemes (in a language in which the V and N systems show identical agreement morphology). For example, 0 might be singular marking and 1 plural marking. What we have done here, then, is simply to remove the stems, leaving only the agreement morphemes behind. Then f_a is the variant of f that agrees with a "singular" specifier and f_b is the variant that agrees with a "plural" specifier.

2.2. **A Brief Review of the Merge/Move Algebra.** The lexicon in Table 1 was used in Cornell (1996) to derive strings in $\mathcal{L}_{ww}$ using derivational machinery modeled on the systems of Stabler (1996). In that algebraic setting, we can represent derivations as terms, with operators Merge and Move operating (recursively) on lexical material. So we can represent the derivation of the string 1010 (presented at a more leisurely pace in that paper) rather tersely as the tree in Figure 1. This tree should be read from the bottom up. So, for example, the first operation in the derivation it depicts is the merging of an f_a with an f_0. Merge is defined as a two-place operation which takes the intended head of the output construction as its first argument. So Merge(f_a,f_0) is intended to be interpreted as constructing a tree in which f_a projects over f_0. Move is a one-place operation, whose action is determined by the features of the current head.

This derivation could be represented in slightly more conventional terms by the tree in Figure 2. In Figure 2 we use the following notation, which implements the phrase structure theory of Chomsky (1995) in a more readable fashion. Following Stabler, we label interior nodes with '$<$' when the head is down and to the right (the right-hand daughter projects over the left, in Chomsky's terminology) and '$>$' when the head is down and to the left (the left-hand daughter projects over the right). Unlike Stabler, we distinguish X^0-internal structures, using the symbol '$\circ$'

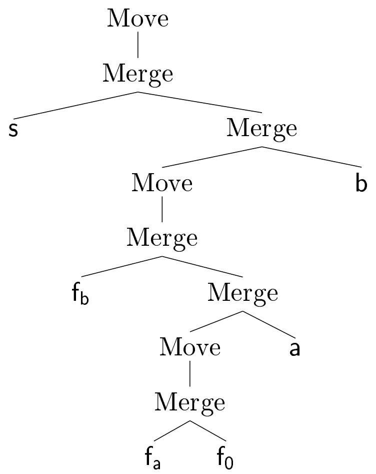


FIGURE 1. A "T-marker" for the derivation of the string 1010.

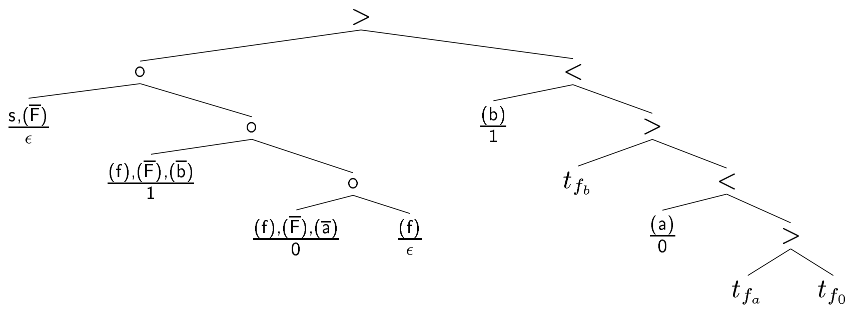


FIGURE 2. A homomorphic image of the T-marker.

to mark X^0's, whose internal head, here, is always to the left.[4] Instead of deleting cancelled features we simply wrap them in parentheses to mark them as cancelled. Since we use only a single instance of each lexical item, we use the names for lexical items from Table 1 as our indices on traces.

[4]In this sense '∘' is essentially '>', only for segments instead of categories. Note however that since in Chomsky's theory an X^0 essentially inherits the features of its "sub-labels", there may be no useful notion of headedness internal to X^0, at least as far as the syntax is concerned.

(We will generally not distinguish heads from the lexical items of which they are instances, using the names for lexical items to name heads as well.)

In slightly more formal terms, we can relate the tree in Figure 1 (let us call it a "T-Marker", recalling the terminology of Chomsky (1975/1955)) to the tree in Figure 2 by interpreting the operation symbols 'Merge' and 'Move' (roughly) in the following way. Given two trees T_1 and T_2,

$$\mathrm{Merge}(T_1, T_2) = \begin{cases} \overset{>}{T_1 \quad\quad T_2} & \text{, if } T_1 \text{ is an } \mathrm{X}^0, \\ \overset{<}{T_2 \quad\quad T_1} & \text{, otherwise.} \end{cases}$$

There are some other side conditions on the application of Merge, but they need not concern us here; see Cornell (1996) for details. For Move we first define an auxiliary function τ. This simply maps moved material to whatever residue it is to leave behind. In the simplest case, this will be a single, constant placeholder symbol, but in the event that the trace of a prior movement can intervene to block some other movement, it may need to preserve some information about the features that used to live there. In the most complex case, it will need to be some unique indicator identifying the trace with a single chain (e.g., a referential index). We leave this open as a parameter along which different formalisms can vary; in the particular case of the derivational system for $\mathcal{L}_{ww}$ (Cornell 1996), the simplest trace function suffices.

Against this background, then, we have that, given a tree T_1 such that

$$T_1 = \overset{>}{\mathrm{head}(T_1) \quad\quad T_3}$$

where T_3 contains a subtree T_2 which can be attracted by the head of T_1, then:

$$\mathrm{Move}(T_1) = \overset{>}{\overset{\circ}{\mathrm{head}(T_1) \quad\quad T_2} \quad\quad T_2\backslash_{\tau(T_2)} T_3}$$

where $A\backslash_x B$ is basically the result of replacing an occurrence of A in B with x. (More accurately, it evaluates to the set of trees which would be just like B if every occurrence of x were replaced by A, that is, it is the inverse of the variable-substitution operation.)

This interpretation of Move is restricted to head movement. A more complex interpretation is required if we wish to capture XP movement as well. Also we have not distinguished between movement to cancel strong vs. weak attractors; in $\mathcal{L}_{ww}$-derivations movement is always to satisfy a strong feature. Feature cancellation is taken to be an automatic and immediate consequence of the construction of a checking configuration. In particular, it is consequent upon the merging in of a specifier and upon any instance of head movement. Finally, we have been completely vague about the meaning of "can be attracted". We will assume here a simple definition: the attracted head must bear an attractand feature which matches the next sequential attractor feature of head(T_1), and there must be no other head which c-commands it which also bears the same attractand feature.

If we interpret feature cancellation as an operation which replaces a cancelled feature ϕ with (ϕ)–instead of replacing all cancelled features with a single constant like 0, which is equivalent to Chomsky's Erasure—and we interpret the trace function as marking the trace position with an index of the moved element, then we derive the tree in Figure 2 as the homomorphic image of the T-marker in Figure 1 under this interpretation homomorphism.

2.3. **The X^0 Chain Locality Problem.** The tree in Figure 2 has no official status either in the derivational system of Cornell (1996) or in the representational system(s) we will develop here. However, it corresponds, mutatis mutandis, to the standard operating procedure of syntacticians in the GB era, relying on annotated traces to represent statically the necessary aspects of the history of a derivation. That is, this tree really represents the practice of syntacticians thinking in essentially derivational terms. In particular, although we consider that there is a chain associated with the head $\mathsf{f_a}$, for example, which corresponds to the derivation step in which $\mathsf{f_a}$ moves to adjoin to $\mathsf{f_b}$, that chain is not represented in this tree. Instead the pair $(\mathsf{f_a}, t_{\mathsf{f_a}})$ spans a region of the representation which superficially at least completely misrepresents the form in which the chain must present itself to the relevant well-formedness conditions. In derivational terms, the original chain link has been so deformed by subsequent movements that the conditions under which it was originally established are rather difficult to recover.

However, we can see immediately that the locality problem for the chain $(\mathsf{f_a}, t_{\mathsf{f_a}})$ is equivalent to another problem, namely the problem of determining whether $t_{\mathsf{f_b}}$ is local to $t_{\mathsf{f_a}}$. In more general terms, this is the problem of determining whether or not *the trace of the host is local to the trace of the dependent.* The geometry of *this* problem is not at all disturbed by subsequent head movements. It is invariant under head movement transformations, in other words.

The hard computational problem then reduces to the problem of finding, given the spell-out positions of the host and dependent heads, their respective trace positions. Again we redefine the problem. We make the concept of Head abstract: a head (the projection into syntax of a lexical item) is an abstract individual which is "extended" by both an XP and an X^0 chain. A position from the dependent head's X^0 chain is attached into the X^0 of the host head. Given a head α, we need to scan its chain, taking adjacent chain positions (chain-links, that is), finding where they have been attached, and measuring the distance between those attachment positions.

As noted, the host in turn can have its own chain positions attached into further hosts, forming what we may informally think of as a chain of chains. But the solution to the spell out problem is again (practically) independent of this complex structure. For PF spell-out we do need to know, for any chain position of a head α, whether it was created to check a strong or a weak attractor feature, but that is all we need to know to tell us which of α's chain positions is relevant to PF spell-out.[5] Where exactly that position appears in the value of π_{pf} applied to the entire syntactic structure is going to depend on the spell-out position of the host head, as we will see, but the problem has a simple recursive structure; in particular, the computation of π_{pf} for an entire structure is not difficult. So the "divide-and-conquer" strategy, which takes the head chain locality problem and splits it into a locality problem and a spell-out problem, pays big dividends for the representational approach to modern transformational grammar. In the remainder of the paper we consider how to implement the solution we have just sketched out, and to extend it to even more problematic cases.

3. A Representational Version of the Minimalist Framework

Suppose that we ask the question "What are trees made of?" What are the significant structural units that make up a tree such as the one in Figure 2? We begin with a rather casual analysis of this tree, and proceed from there to formalize our intuitions. Essentially, the objects

[5]For the moment we take a simple approach to LF spell-out which always takes the highest chain position as the position for LF spell-out.

we are interested in are in the first place *chains*, since the question we began with concerns the adequacy of chains as declarative representatives of transformational derivations. Normally the objects that can move are either maximal phrases or maximal heads (that is, when heads and phrases move they take with them all of their dependents), though in Figure 2, of course, we see only a picture of various head movements. Chains in general, then, are going to be formed from two types of object, which we will term XP's and X^0's. Both of these objects can be internally complex, but they are "type uniform", that is, only XP's can be attached in other XP's and only X^0's can be attached in other X^0's.

We are furthermore particularly interested in a theory of chain formation which harmonizes with the Chomsky/Stabler-inspired theory of transformational derivations we sketched out in §§2.1 & 2.2. In particular, we assume that the structures of XP's and X^0's and their respective chains are decorated with features drawn from the lexical items in such a way as to associate clashing pairs of features with every instance of chain formation, that is, with every chain link. Basically, chains can only exist where there are the relevant featural resources to support them, and the featural resources require the existence of chains to consume them. A well-formed structure is one in which all attractors and attractands are balanced, except for the category of the matrix itself.

If we analyze the tree in Figure 2 along these lines, we decompose it into the "constituents" illustrated in Table 2. Each line in Table 2 corresponds to a single instance of a lexical item, that is, a single numeration item (for which we shall reserve the term "head" in what follows). Essentially, the lines of the table display the contributions of each head to the syntactic structure as a whole, and illustrate the ways in which each head is "extended" in the formation of that structure. The first column displays the associated lexical material and the second column displays the XP and X^0 of the head and its X^0 chain. The '$\square$' symbol marks leaf nodes; otherwise all nodes are labelled as in Figure 2. The leftward pointing arrows represent chain links (as usual, head chains consist of at most a single link). Note that we make no use of XP chains here or elsewhere in this paper. Formally, all our XP chains are trivial: they consist only of a single position (identified here with the maximal XP position). The minimal position of the X^0 chain has been identified here with the minimal position of the XP. We consider this position to be the *head position*. This usage contrasts sharply with traditional terminology for chains, according to which the "head" of a chain is its most remote position, which is typically the spell-out position. In our framework, as already noted,

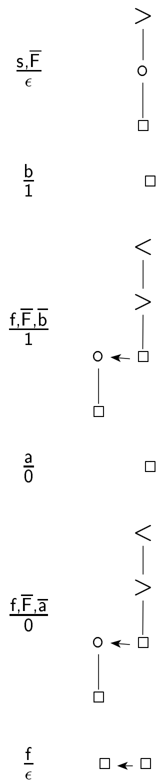


TABLE 2. Constituents of the tree in Figure 2

we do not want to identify the abstract notion of head with the spell-out position of the head, and we view chain formation not as a dislocation of the head but as an extension of the scope of its contribution to the syntax. ¿From this perspective it makes much more sense to identify the "head" of the chain with the (traditional) head of the XP and identify both with the abstract notion of head, that is, a projection into the numeration of a lexical item. However, to avoid as much confusion as possible, we will often refer to the *minimal* position in the XP or the X^0 chain, rather than the head of an XP or X^0 chain.

Any one of the "extensions" X^0, XP, the X^0 chain or the XP chain can be trivial. For example, the structure associated with f_0 has only a trivial XP consisting of a single position which is at once both maximal and minimal. The structures which "extend" a and b in the syntax do so trivially in all dimensions.

Every node in the tree in Figure 2 corresponds to a node somewhere in Table 2. What is missing are the immediate dominance links which associate XP's and X^0's with each other. So, for example, to recover the tree in Figure 2 we would need to add an immediate dominance link connecting the outermost chain position of $\mathsf{f_0}$ with the '$\circ$' node of the outermost chain element for $\mathsf{f_a}$. It is by the establishment of these links that features are put in checking configurations. So in the example of the $\mathsf{f_0}$–$\mathsf{f_a}$ attachment, the establishment of this link allows us to cancel the f categorial feature of $\mathsf{f_0}$ against the (strong) $\overline{\mathsf{F}}$ selectional feature of $\mathsf{f_a}$.

Interestingly, the structure in Table 2 corresponds fairly closely to the proposals in Brody (1995) for "pre-syntactic" chain formation and XP projection, with the exception that we have not required heads to project non-trivially. That is, we allow the identification of XP and X^0 nodes, in the style of Chomsky (1995). The term "pre-syntactic" refers to a technical usage in which the "syntax" is just the assembly of pieces such as those in Table 2 into an LF. Table 2 corresponds to Brody's usage of the term *numeration*: this is supposed to be the guise in which lexical items present themselves to the syntax. Here, however, we find it simpler to distinguish the elementary selection of a multiset of lexical items from the selection of their ultimate syntactic extensions, so we retain Chomsky's usage of "numeration", and distinguish "pre-syntax" (for lack of a better term) from both numeration and syntactic structure.[6] So the reader may (cautiously) consider this system to be (approximately) Chomsky's theory of phrase structure and movement presented in Brody's representational framework.

3.1. **Chains and Phrases.** Turning now to the formalization, our structures are made out of individuals of two sorts: *features* and *positions*. Formal features, as usual, come as either attractors or attractands, and among attractors we distinguish strong and weak varieties. They appear in the syntax bundled into lexical item instances which we will term here *heads*. We assume that we have the projection functions π_{pf} and π_{lf} which

[6]Another way of looking at these "pre-syntactic" structures is as the *partial evaluation* of the Merge and Move operations on the elements of the numeration. Under the assumption of linear resource use those operations are tightly enough restricted to make such an approach feasible, though it is not in general deterministic. Under such an interpretation, this framework could perhaps be seen as providing an alternative *operational* semantics for the basic formalism of Stabler (1996). For reasons of space we do not pursue this here.

1. α^0 denotes a sequence of positions which extend α *internally* in the "vertical" (phrasal) dimension.
2. αP denotes a sequence of positions which extend α *externally* in the "vertical" dimension.
3. α^* denotes a sequence of positions which extend α^0 in the "horizontal" (chain) dimension.
4. $\alpha\mathrm{P}^*$ denotes a sequence of positions which extend αP in a chain.
5. The *syntactic extension* of $\alpha = \langle \alpha^0, \alpha\mathrm{P}, \alpha^*, \alpha\mathrm{P}^* \rangle$.

TABLE 3. Notation for "constituent" structures.

apply to heads to yield their phonological and semantic interpretations, respectively. In this paper we will be perhaps more concrete than we need to be: our sole type of interpretable features, the orthographic features, are drawn from a set disjoint from the set of formal features, and the application of both projection functions is taken to yield as a result whatever is "below the line" in the presentation of lexical items as in Table 1. Our syntactic framework can make use of more sophisticated definitions of the projection functions, however.

Positions are organized in two kinds of phrase structures (head-internal and head-external), and their associated chains. Given a head α, we can extend it (possibly trivially) to an α^0 by the addition of zero or more X^0 attachment sites. We can then extend α^0 in a chain α^* (for the typographically awkward α^{0*}) by the addition of zero or more X^0 chain positions. At the same time we can extend α^0 to an XP αP by the addition of zero or more XP attachment sites. Finally, we can extend αP to an XP chain $\alpha\mathrm{P}^*$. This notation is summarised in Table 3.

As a matter of convention, we consider α to be the minimal member of α^0, we consider $\max(\alpha^0)$ to be the minimal member of both α^* and αP, and we consider $\max(\alpha\mathrm{P})$ to be the minimal member of $\alpha\mathrm{P}^*$. So the following equations hold,

$$\alpha = \min(\alpha^0) \tag{1}$$

$$\max(\alpha^0) = \min(\alpha^*) \tag{2}$$

$$\max(\alpha^0) = \min(\alpha\mathrm{P}) \tag{3}$$

$$\max(\alpha\mathrm{P}) = \min(\alpha\mathrm{P}^*) \tag{4}$$

and as a consequence the structures in the syntactic extension of α, that is $\langle \alpha^0, \alpha\mathrm{P}, \alpha^*, \alpha\mathrm{P}^* \rangle$, are never empty. In the trivial case they all contain one member, which is identical to α itself. We will find it useful, however, to be able to distinguish also those elements which are *proper extensions*, that is, the non-minimal elements of any of the four constituents.

¿From a movement perspective, the equations in (1)–(4) are absolutely counterintuitive, since they have the effect of attaching the XP into its *lowest* chain position. So the structures we are building are not representations of either surface syntax or LF. These levels can easily be recovered by the application of the projection functions π_{pf} and π_{lf}, however, which are in turn very easily computed once the entire syntactic structure has been built. So the differences from traditional approaches are only superficial. The basic motivations for this approach are that, first, once we admit that the PF projection of a structure can place it in a different position from the LF projection, the whole notion of an XP becomes abstract, and it doesn't really matter where we put it, as long as we can get at it when we need it. Second, one readily observes the descriptive generalization that, once a structure has been moved, it becomes opaque as a target for further movement: nothing can any longer attach into it. So our XP's are represented (in derivational terms) in the positions in which they are originally built and in which they take their final form. This will be seen to make calculations of locality and other structural conditions on chain links much simpler.[7]

Within XP we distinguish the first extension of the head position as *complement position* and all other extensions as *specifier positions*. This distinction plays a role in both checking theory and in the theory of word order. As a matter of notation we will continue to mark specifier positions with the symbol ‘$<$’ and complement positions with the symbol ‘$>$’.

We consider the (proper) “phrasal” positions in XP and X^0 to be *attachment sites*, and we consider chain positions in XP^* and X^* to be the elements which are actually attached to these positions. So there is a rather deep sense in which this grammatical framework is “strongly lexicalist”, in that the division of labor which we observe at the lexical level in the distinction between attractor and attractand features is “projected” into the syntax as the distinction between phrasal structures and chains.

Besides the formation of these four head-extensions, our main structure-building operation is *attachment*, which we also interpret as creating an immediate dominance link. By contrast to the immediate dominance links which linearize the phrasal extensions X^0 and XP, which relate positions

[7]This can be seen as one way of formalizing the Main Thematic Condition (MTC) of Brody (1995).

which are all extensions of the same head, attachment assertions relate positions which are extensions of distinct heads. In particular:

Condition 1 (Complementarity of Attachment). *We require that all attachments be of chain positions x extending some head α to phrase-structure positions y (properly) extending a distinct head β.*

Condition 2 (Uniformity of Attachment). *In all attachments of chain positions x to phrase-structure positions y, x and y are of the same type (i.e.,* X^0 *or* XP*).*

We consider syntactic structures, like the lexical items they are built out of, to be categorized sound-meaning pairs, so we will apply the projection functions π_{pf} and π_{lf} to them as well. We can use the projection functions in order to associate only the PF substructure or only the LF substructure with particular chain positions, in this way deriving a chain-formation system reflecting the various movement operations of Stabler (1996). So, for example, given a phrase αP, we can associate $\pi_{pf}(\alpha\mathrm{P})$ with one chain position and then associate $\pi_{lf}(\alpha\mathrm{P})$ with a higher position, simulating covert "LF movement" of the αP. Intuitively, this allows us to construct X^0 and XP without regard to where they will be spelled out in the final syntactic structure. As promised, we thereby separate the issue of which chain position attaches where—which involves among other things the theory of *Locality*—from the problem of what goes in each chain position—which is the *Spell-Out Problem*.

3.2. **Feature Driven Licensing.** In this section we will try to develop a representational view of the feature driven theory of movement from Chomsky (1995). The essential idea of movement theory which we will attempt to adapt here is that movement can only take place if it puts a clashing pair of features in a checking configuration. A checking configuration is essentially any structural configuration involving a phrasal extension of a head α (α^0, αP) and a dependent chain position extending a distinct head β, *unless that dependent is the structural complement*. We will not attempt here to explain the curious shape of this definition, but it is based on the descriptive observation that movement seems to extract structures out of the complement domain and into some other local relation to the head. So we define, by stipulation if necessary, the complement domain as the source of movements and all remaining phrasal positions in the extensions of a head as checking positions, that is, appropriate landing sites for movement. As a matter of terminology, we distinguish the *attraction domain* of a head from its *checking domain*. All movement is from the attraction domain of a head to a position in its checking domain. For our purposes the checking domain can be defined as follows.

Definition 1 (Checking Domain)**.** *The checking domain of a head α is the set of positions properly extending α in either*

1. α^0*, or*
2. *as a specifier in* αP.

Within the more representational framework which we are developing here, we begin from a position in which we are given for each head α a complex of structural positions, namely the tuple $\langle \alpha^0, \alpha\mathrm{P}, \alpha^*, \alpha\mathrm{P}^* \rangle$, and a lexical item, that is, a complex of features consisting of attractors, attractands and interpretable features. In a sense we have already done our chain formation ("pre-syntactically" in the terminology of Brody (1995)), in that α^* and $\alpha\mathrm{P}^*$ already represent proposals for what chains there will be in the syntactic structure as a whole. What we are considering here is more like a theory which licenses these structures, that is, which selects pre-syntactic chain formation hypotheses which will actually work together to yield well formed syntactic structures. The licensing, or feature-distribution, problem is then that we must associate the proper grammatical resources with the constructions that consume them.

3.2.1. *Checking Theory.* The core of our system is naturally going to be Checking Theory, which we take to define a mapping distributing formal features across a syntactic structure in such a way that all features are checked. We will implement this by associating formal features with positions. Intuitively, checking configurations are just attachment arcs, and attractor features belong at the target, attractands at the source of an attachment arc which puts them in a checking configuration.

Our first definition will probably seem out of place here: at first glance it seems to have nothing to do with Checking Theory.

Definition 2 (ϕ-Structure)**.** *Given a categorial feature ϕ, a ϕ-structure is a syntactic structure in which the only unchecked feature is ϕ.*

We take a ϕ-structure Σ to represent the judgment that $\pi_{pf}(\Sigma)$ is a grammatical utterance of category ϕ, under the interpretation $\pi_{lf}(\Sigma)$. When we define a language in this way, we implicitly impose upon it the condition that all (but one) features be placed in checking configurations.

As already noted, we take *checking configurations* to be attachment links where the target is either a specifier position or a head-internal adjunct position, i.e., the target is in the checking domain of its head. We take (proper) X^0 and XP positions to correspond to lexical attractor features, and chain positions to correspond to attractand features, so the feature distribution function, in order to respect this interpretation, must assign attractands to chain positions and attractors to phrasal positions, in particular phrasal positions in the checking domain of the head.

Condition 3 (Uniformity of Distribution)**.** *The distribution function must map attractands to chain positions and attractors to phrasal positions.*

In our terms, Chomsky's principle of Last Resort (Chomsky 1995) requires that every chain position which properly extends some α^0 or $\alpha\mathrm{P}$ must be *licensed* by the possession of an appropriate attractand feature. We add to this a condition of Linear Resource Use, which states, in our terms, that the feature distribution function must be, in fact, a function, so that no feature can be used in more than one checking configuration.[8]

Condition 4 (Last Resort)**.** *For all proper chain positions x in $\alpha^* \cup \alpha\mathrm{P}^*$, the distribution function maps some attractand feature from the lexical item of α to x.*

If we restrict our attention to ϕ-structures, it is clear that now all proper chain positions will have to be attached to phrasal positions with appropriate attractor features. Also all positions in the checking domain of every head will have to host attached chain positions bearing appropriately clashing features. Note in passing that more than one feature can be checked in any given position: we do not require the distribution function to be one-to-one. Note also that the minimal chain positions are not subject to Last Resort, but still *can* appear in checking configurations. For example, in the structure associated with the tree in Figure 2 the heads a and b are extended trivially, so the sole position which exists in $\langle \mathsf{a}^0, \mathsf{a}\mathrm{P}, \mathsf{a}^*, \mathsf{a}\mathrm{P}^* \rangle$ or $\langle \mathsf{b}^0, \mathsf{b}\mathrm{P}, \mathsf{b}^*, \mathsf{b}\mathrm{P}^* \rangle$ is free of Last Resort. Nonetheless, because they are attached as specifiers, they still check their categorial features against the selection features of f_a and f_b, respectively.

3.2.2. *The Theory of Selection.* In the system of Chomsky (1995), a distinction is made between the resource requirements of Merge and Move. In particular, Merge is supposed to be free of resource consumption; it can happen anywhere regardless of the features of the structures involved.

[8]The condition of linear resource use is almost certainly too strong. For example it appears to prevent any sort of successive cyclic movement. However, it endows the framework with such appealing mathematical and computational properties that we prefer to start by assuming it anyway, and then search for appropriate ways of weakening it when that becomes necessary. So for example Stabler (p.c.) suggests that it could be restricted to attractor features, allowing the attractand to be reused.

Our system is not exactly parallel to Chomsky's; in particular Attachment is used here to simulate the effects of both Merge and Move as they act to extend the structure in a derivation. (For example, we use Attachment to create X^0-internal structure; this is something Chomsky never uses Merge for.) Nonetheless, we inherit this asymmetrical resource consumption profile in our stipulation that complement attachment does not create a checking configuration.

However, we found in investigating a derivational minimalist grammar for the copy language $\mathcal{L}_{ww}$ (Cornell 1996) that unless structure-building is subject to the constraints imposed by the argument structure of the head, the grammatical system lacks the power to derive even a simple instance of crossing dependencies like $\mathcal{L}_{ww}$, and, by implication, could never generate a grammar for a language like Dutch where head movement constructions (under standard GB-style analyses, among others) create such patterns of crossing dependencies. It gets worse. Assuming, as is common in minimalist theorizing, that the universal LF hypothesis holds, i.e., that languages differ only in whether the features of a particular category are strong or weak, this system, if it could not generate the LFs for Dutch, could not generate a grammar for any human language.

Therefore, while Attachment need not *check* any features, it is nonetheless subject to the selection features of heads, in the sense that a complement attachment is only allowed if the host head owns an appropriate selection feature for the proposed complement. We must take pains to distinguish the theory of chains, with its associated theory of feature checking, from the theory of selection, in particular as both of these contribute to the definition of the Attach relation for any given structure. We outline the formal underpinnings of selection as follows.

Every XP attachment link has as its source a position from some chain αP^*. If that position is $\min(\alpha P^*)$ then the attachment is subject to the theory of Selection; if not, then the attachment is subject to the theory of Chains. We will distinguish these cases as *root* attachment vs. (proper) *extension* attachment.[9]

Condition 5. *Every root attachment arc must connect the dependent chain position to a host phrasal position whose lexical item has an attractor for the dependent's major category feature. Furthermore, attachment must respect the order of categories in the selectional grid of the host.*

[9]The term "root" is borrowed from Brody (Brody 1995, and refs.), who uses it to refer to $\min(XP^*)$.

In our example of $\mathcal{L}_{ww}$, since we lack non-trivial XP chains, all attachments of XP chain positions are root attachments, and so are subject to this condition.

Condition 6. *All extension attachments must be to positions outside the root attachment positions.*

This latter condition amounts to a condition that chain extensions of an αP must attach to positions which are also extensions of the "core" of the host βP determined by its argument structure. It expresses the centrality of the functor-argument complex and its priority over the properly transformational syntax. If we consider all attractor features for a given lexical item to be ordered, with the selection features first, then we are simply requiring attachment order to correspond to feature order in the lexical item, again, requiring that syntax be essentially lexically driven.

Taken all together these conditions derive the desired consequence that the selection of internal arguments always leads to (overt or covert) head movement. External arguments will attach in specifier positions, which are in the checking domain of the host head, but internal arguments will always attach in complement position, owing to the ordering restrictions in Condition 5. So the categorial feature of the dependent and the internal argument selection feature of the host must be placed in some other checking configuration. Theoretically, the possibility exists that the dependent complement could raise as a phrase and attach one of its XP chain positions to the host (making it at once the complement and specifier of its host). Exactly what causes head movement as opposed to (full) phrasal movement is not at present clear, so there is nothing yet to be formalized. Here we simply stipulate that certain attractor features (categorial selection features, which are all we have in $\mathcal{L}_{ww}$) only attract X^0's, but this is clearly ad hoc and not correct in general.

3.3. Structural Conditions on Phrases and Chains.

We have so far considered attachment as only a relation, but in fact it has more structure than that. For example, distinct chain positions cannot be attached to the same phrasal position, nor can distinct phrasal positions host the same chain position.

Condition 7. *Attachment is a 1-1 function from chain positions to phrasal positions, which is total on the set of chain positions excluding (a) minimal* X^0 *chain positions and (b) one single minimal* XP *chain position.*

The minimal X^0 chain positions are not attached anywhere, but instead identified directly with min(XP). Furthermore, the unique XP chain position associated with the matrix of the structure will not be attached anywhere. (We will see later in this section that the chains associated with the matrix structure must be trivial.)

Consider now the following possibility. Suppose for example that we are given heads α and β, with only trivial XP chains. We can in principle attach the position in αP^* into βP, and also attach the position in βP^* into αP. Typical lexicons will contain selectional restrictions which block this simple example, but it is clear that with a little effort similarly pernicious examples can be found for almost any interesting lexicon. We require then the following condition:

Condition 8 (Tree Condition)**.** *The reduced structure which we get by ignoring all chain extension relations (that is, we retain the positions themselves, but forget how we constructed them), is an unordered tree.*

We have already noted in passing that attachment and phrasal extension links are intended to be interpreted as immediate dominance. Condition 8 just states that, under this intended interpretation, the resulting immediate dominance structure is in fact an unordered tree. This can be weakened to a condition of acyclicity, because Condition 7 already has the connectedness of the entire structure as a consequence. Then there is a root, and acyclicity is enough to guarantee the existence of a unique path from any node to the root, giving us our unordered tree.

We have noted already the descriptive observation that all movements are from within the attraction domain (i.e., complement) to positions in the checking domain. We will see here how to state a corresponding condition on chains. Note that it is traditional in discussing well-formedness conditions on chains to require that each chain element c-command its trace. This is meant to capture in chain-theoretic terms the upward and leftward tendency of individual movement transformations. Now, it has often been observed that this c-command condition is violated in various constructions in natural languages. In German Remnant Topicalization, for example, a DP may scramble out of the VP, and then the VP, including the trace of the scrambled DP, may be topicalized. Each movement is to a c-commanding position, but considering the resulting structure as a whole, the c-command relation between the antecedent DP and its trace no longer holds. Observe, however, that the chain link connecting the scrambled DP to its trace position does preserve the attraction-domain to checking-domain directionality of movement, in the following sense.

The trace of the DP is still to be found inside of some structure (i.e., the syntactic extension of some head) which still owns a chain position somewhere in the complement of the DP's current host. So, simplifying things somewhat, we take the chain position of the DP corresponding to its surface position; we must verify that it stands in the appropriate structural relation with its trace, i.e., the immediately previous DP chain position. We examine the XP of the DP's (surface structure) host to find its complement, which will be a chain position belonging to the V's XP*. We then inspect the XP of the V, looking for the trace of the DP. In this simple example we should find it hanging directly off of the VP. We do not need to consider what chain position the VP will actually surface in; that is not part of the problem. We care only about the internal structure of the VP itself. This is then another example of a structural property which is invariant under movement transformations, which we can make use of to formulate a representational condition equivalent to the corresponding derivational condition.

A little more rigorously, we are interested in paths of the following sort. Given two adjacent chain links x and y, with x immediately following (i.e., higher than) y, we need to find a path through the phrase structure from x to y of the following sort. We need to find the XP which is the host of x—call it αP—then find α's complement, that is, the attraction domain of α, which will be a chain position in some other head's extension. Let us refer to that head as β; then the attraction domain of α will be a position in βP*. We jump to βP, then, and search its dependents for y.

Definition 3 (Subterm)**.** *Given chain positions y and z, y in the extension of a head α, z in the extension of a head β, y is a subterm of z iff either*

1. *$y = z$, or*
2. *there is a chain position w attached to βP and y is a subterm of w.*

This is very similar to the standard definiton of subterm or subtree for trees, except that we can jump directly from any chain position to the associated XP. In effect, we have copied the XP structure to every associated chain position, without having gone to the trouble of actually making copies of the XP.

Condition 9 (Chain Link Condition)**.** *Given two chain positions x and y with x immediately following y in the chain, x in the checking domain of some head α. Then*

y is a subterm of the attraction domain of α.

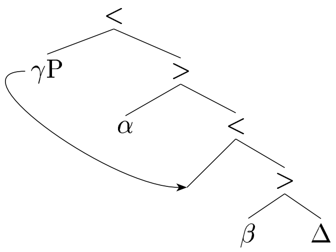


FIGURE 3. A retrograde movement.

Consider how this blocks downwards, "retrograde" movement. Consider the tree in Figure 3; here γP moves down from the specifier of αP to specifier of βP. In our terms, we have the heads α, β and γ, with their associated extensions. In particular, $\gamma\mathrm{P}^*$ contains two positions. The first ($\gamma\mathrm{P}^*_0$, i.e., $\min(\gamma\mathrm{P}^*)$, i.e., $\max(\gamma\mathrm{P})$) is attached to the specifier position of αP. The second ($\gamma\mathrm{P}^*_1$, i.e., $\max(\gamma\mathrm{P}^*)$) is attached to the specifier position of βP. Finally, the sole position in $\beta\mathrm{P}^*$ is attached to the complement position of αP. To verify that $\gamma\mathrm{P}^*$ is (or is not) well-formed we start from $\gamma\mathrm{P}^*_1$; we note that its host is βP, so we need to verify that $\gamma\mathrm{P}^*_0$ is a subterm of the complement of βP (which we have represented here as Δ). Obviously, it is not, so the chain is ill-formed.

Note that there is a more complex possibility that is worth considering: suppose that αP had moved out of Δ in the first place, to attach to some δP, of which βP is the complement. This derivation is indicated in the tree in Figure 4. Now the trace of γP *would* be found somewhere inside Δ: from $\gamma\mathrm{P}^*_1$ we find its host βP, then the attraction domain of β, i.e., Δ, somewhere in which we would find the trace of αP, from which we go to αP itself, where we find the trace $\gamma\mathrm{P}^*_0$ that we were hunting for. So the retrograde movement in this case makes a chain which is well-formed. But this is not a problem! The structure *is* in fact well formed (the reader can verify as an exercise that it is essentially the structure of German Remnant Topicalization). We have constructed it with an illicit derivation, but in fact it also has a good derivation in which γP moves first to specifier of βP and then αP moves to specifier of δP, carrying γP's trace along with it. We conclude that, though the concreteness of a derivational perspective can be very helpful in solving linguistic problems, it is important to keep in mind that a bad derivation can still derive a good structure, and the failure of a particular derivation is not to be taken as an infallible indicator that a particular string is ill-formed. In this respect a

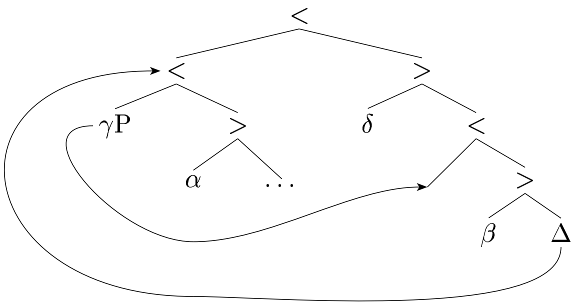


FIGURE 4. A particularly pernicious retrograde movement?

representational approach is more reliable, though its abstractness makes it more difficult to work with.

3.4. **Spell-Out.** We will employ a relatively simplistic theory of spell-out here. We assume that the LF of a syntactic structure is to be computed based on the maximal chain positions. We assume that the PF is to be computed based on the "maximal strong chain positions," to be defined below. We will refer to these positions as the LF and PF chain positions, respectively. In these terms, the action of the projection functions π_{lf} and π_{pf} is easily defined.

The value of $\pi_{lf}(x)$ (resp., $\pi_{pf}(x)$), when x is the LF (resp., PF) chain position, is defined to be equal to $\pi_{lf}(y)$ (resp., $\pi_{pf}(y)$), where y is the maximal corresponding phrasal position. (So, if x is a position in some $\alpha\mathrm{P}^*$, then y is $\max(\alpha\mathrm{P})$, and if x is a position in α^*, then y is $\max(\alpha^0)$.) Otherwise, it is defined to be λ (a meaningless placeholder) if x is not the LF (resp., PF) position of its chain. In this way αP is essentially at one and the same time "LF-attached" to the LF position of its chain and "PF-attached" to the PF position of its chain.

The value of the projection functions at a phrasal structure is given in three cases, corresponding to the node-labels '$<$', '$>$' and '$\circ$'.

If x is a specifier position in some αP, then x immediately dominates some y in αP which it extends, and furthermore there is some chain position z from some $\beta\mathrm{P}^*$ attached to x. The value of $\pi_{lf}(x)$ (resp., $\pi_{pf}(x)$) is defined by the formula

$$\pi_{lf}(x) = (\pi_{lf}(z) < \pi_{lf}(y))$$

(resp., $(\pi_{pf}(z) < \pi_{pf}(y))$). Note that under this definition the projection functions preserve headedness information.

If x is a complement position in αP, then x immediately dominates $\min(\alpha\mathrm{P}) = \min(\alpha^*)$ and a structural complement $z \in \beta\mathrm{P}^*$ (for some β).

$$\pi_{lf}(x) = (\pi_{lf}(\min(\alpha^*)) > \pi_{lf}(z))$$

Again, we treat π_{pf} analogously.

If x is a head-adjunction position, then it immediately dominates an X^0 position y which it extends and also an X^* position z which is attached to it. Then

$$\pi_{lf}(x) = (\pi_{lf}(y) \circ \pi_{lf}(z))$$

Yet again, we treat π_{pf} analogously. Note that y may be the head itself; we have already defined the action of the projection functions on lexical items.

The only problem that remains is the determination of the PF position for an arbitrary chain. As noted, we want this to be the "maximal strong chain position," that is, the position farthest from the minimal chain position which can be reached accross chain elements whose features have been checked against strong attractors.

Definition 4. *A chain extension is called strong if it creates a position in which an attractand is checked against a strong attractor.*

Definition 5 (Maximal Strong Position)**.** *The maximal strong position is defined as the maximal element in the closure of the minimal chain position under strong chain extension.*

This has the desired consequence that the position we want must be reachable along a continuous sequence of strong extensions, and also that it is the minimal chain position if the first extension is weak or nonexistent. On the other hand it is a relatively weak condition since it allows strong positions after weak positions, which would correspond in Chomsky's derivational terms to a derivation that would crash, since it would have had to carry a strong feature into the covert component. We could strengthen our definitions by requiring chains to have a continuously strong "prefix" followed by a continuously weak "suffix", but we do not do so at this point.

4. The Copy Language $\mathcal{L}_{ww}$ Revisited

4.1. **Syntactic Extension of the $\mathcal{L}_{ww}$ Lexicon.** In general, the extension of heads from the numeration into pre-syntactic structures is nondeterministic. However, in the particular case of $\mathcal{L}_{ww}$, extension is deterministic. So in this section we show the "extended lexicon" for $\mathcal{L}_{ww}$, which will make the proof in the following section that we have indeed defined $\mathcal{L}_{ww}$ a good deal easier.

So, let us consider the particular case where we have a head which instantiates the lexical item $\mathsf{f_a}$, i.e.,

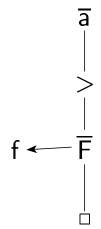
$$\frac{\mathsf{f},\overline{\mathsf{F}},\overline{\mathsf{a}}}{\mathsf{0}}$$

for example. The selectional features $\overline{\mathsf{F}}$ and $\overline{\mathsf{a}}$ will require the attachment of two dependent projections, so they require the construction of a phrasal spine with at least complement and specifier positions. No features of $\mathsf{f_a}$ will require the attraction of an XP, so those will be all the positions which are required in $\mathsf{f_a}$P. The attachment of a specifier from some chain aP* will cancel the $\overline{\mathsf{a}}$ selection feature; that is, $\overline{\mathsf{a}}$ attractor will be mapped to the specifier position by the feature distribution map.

The attachment of a f complement will not allow the cancellation of the $\overline{\mathsf{F}}$ attractor, however. There are in principle two ways in which the $\overline{\mathsf{F}}$ attractor could then be cancelled. First, the complement fP could attach one of its chain positions to a further specifier position of $\mathsf{f_a}$P. Alternatively the f^0 of the complement could attach one of its chain positions into $\mathsf{f_a}^0$. We are not certain what should rule out movement of the complement to (non-thematic) specifier positions; this is a particular problem in a formalism which allows multiple specifiers. One possible solution might be to appeal to the relative simplicity of an X^0 as compared to an XP, invoking some sort of economy principle to the effect that only heads move unless some yet-to-be-clarified theory of pied-piping requires a larger structure to be displaced. This seems to follow the lines laid out in Chomsky (1995), according to which movement is only of individual features wherever possible. Another possible solution would be to develop the intuition that selection is a head-to-head phenomenon (cf. Koopman (1994), for example), implementing the intuition here via a requirement that selectional features are specifically X^0-attractors, while for example case assignment features (if we made use of them) would be XP-attractors.

The important point is that the corresponding derivational formalism sketched out in Cornell (1996) is subject to the same problem, and the same solution can most likely be applied in both cases. Certainly the two possibilities indicated above are equally applicable. So we assume without further argument that the only possibility available for checking the category feature of the complement against a selectional attractor is by head movement or, in our terms, by the attachment of an X^* position belonging to the complement into the X^0 of the host. Accordingly we construct $\mathsf{f_a}^0$ containing one proper extension of the head. That exhausts the stock of attractor features from the lexical item $\mathsf{f_a}$, so we cannot construct any more phrasal positions.

This leaves us with the categorial attractand, the f feature of $\mathsf{f_a}$ itself. As an attractand, it must be associated with some position in either $\mathsf{f_a}\mathrm{P}^*$ or $\mathsf{f_a}^*$. So this question really boils down to the question of how far to extend $\mathsf{f_a}\mathrm{P}^*$ and $\mathsf{f_a}^*$. For example, if $\min(\mathsf{f_a}\mathrm{P}^*)$ is attached as a specifier somewhere, then that will cancel the f feature, but if it is attached as a complement, then the f feature can only be cancelled in a proper extension of either $\mathsf{f_a}\mathrm{P}^*$ or $\mathsf{f_a}^*$. We can in our particular case examine the lexicon and note that f categories are always selected as complements, never as specifiers, so in this particular language the f feature can not be cancelled in $\min(\mathsf{f_a}\mathrm{P}^*)$, and we will have to create a chain extension. Again we examine the lexicon for $\mathcal{L}_{ww}$ and note that all attractors $\overline{\mathsf{f}}$ or $\overline{\mathsf{F}}$ are selection features.[10] We have here assumed that selectional features will only attract X^0's, so we extend $\mathsf{f_a}^*$ with one position and map the f feature to that position. Given the identifications in (1)–(4), we can display the results as follows.



Further inspection of the lexicon shows us that f categories are only ever strongly attracted, so we know also that the PF spell-out position of the chain $\mathsf{f_a}^*$ will be $\max(\mathsf{f_a}^*)$.

As the foregoing example suggests, for our rather restricted case of $\mathcal{L}_{ww}$, the lexicon is projected into syntax deterministically, and the structures in Table 4 are the only possible ones. (In Table 4 we have indicated the spell-out position of each non-trivial chain by placing the interpretable features of the corresponding head there.)[11]

[10] We have here defined selection features as equivalent to categorial attractands, but this is not necessary: we could for example follow Stabler (1996) in marking selection features differently from other types of attractors.

[11] One problem with this formalism is indicated by the structure extending $\mathsf{f_0}$. If $\min(\mathsf{f_0}\mathrm{P}^*) = \max(\mathsf{f_0}\mathrm{P}) = \min(\mathsf{f_0}\mathrm{P}) = \min(\mathsf{f_0}^*)$ then, as the XP is also the X^0, the chain would appear to be both the XP^* and the X^*. But this does not fit well with

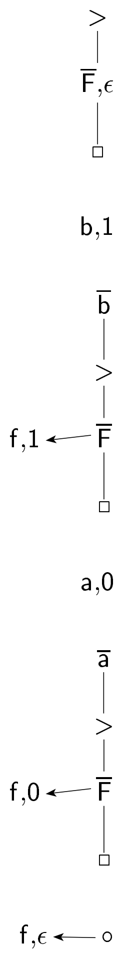


TABLE 4. "Pre-syntactic" structural elements of $\mathcal{L}_{ww}$.

---

our implementation of spell-out, since we would want the (PF) spell-out position of the XP chain to be the position attached to $\mathsf{f_a}$P, which is here $\min(\mathsf{f_0}P^*)$, but the spell-out position of the X^0 chain to be its extension position, which here appears identified with $\max(\mathsf{f_0}P^*)$. This could be a problem for a theory which allows trivial phrasal extension, i.e., the identification of $\max(\alpha P)$ with $\max(\alpha^0)$. However, it is not clear that we need the identifications in (1)–(4) to be as strong as actual equality. In particular we may not need them at all, and if we do we certainly do not need the identification relation

4.2. **The string 1010.** We define the language $\mathcal{L}_{ww}$ to be (strongly) the set of s-trees satisfying the conditions we have laid out here and with lexical items drawn from the lexicon in Table 1, and (weakly) the set of strings yielded by the strong language. Again we consider the string 1010, which is derived from a numeration which uses each lexical item exactly once, so Table 4 can be thought of equally well as the presyntactic form of the numeration for this string.

Because the lexicon for $\mathcal{L}_{ww}$ is so simple and highly structured, and in particular because it supports a deterministic extension into "pre-syntax", the only work left to do is to connect the pieces in Table 4 together with immediate dominance links. Again, assuming we are using exactly one instance of each lexical item, we have only a few options. First, there are no attachment sites for an s category. So there can only be one instance of this lexical item in *any* successful numeration, and it must be the matrix. It provides a (strong) X^0-internal attachment site for an X^* element with attractand f, and a complement position, which also must be filled with an XP^* position belonging to a f head. So only the extensions of f_a, f_b and f_0 can fit here. f_0 has no attachment sites, so if we attach it to s^0 we will derive an inert element, an s-tree in fact, without having used up all of our numeration elements. We can attach the sole position in f_aP^* to the complement position, attaching the maximal position in $f_a{}^*$ into s^0 as well, or we can do the same with f_b. Note that we cannot attach the position from f_bP^* to sP and attach the outer position from $f_a{}^*$ to s^0 (or vice-versa). This represents a case of retrograde movement: to license position ${f_a}_1^*$, which is attached to $f_b{}^0$, we would have to find ${f_a}_0^*$ somewhere as a subterm of the complement of f_bP. We have so far attached nothing in that position, but we have already attached the lowest chain positions belonging to f_a outside of that domain, so nothing we could build there could contain traces of f_a without introducing even more retrograde chains.

The first of the good cases, where we attach the extension of f_a as complement to sand attach its maximal head chain position into s^0, already

to be a congruence for the extension-forming operations, even if it is an equivalence relation. In such a case the equivalence of $\max(\alpha P)$ with $\min(\alpha P)$ and transitively with $\max(\alpha^0)$ would not entail the literal identification of, e.g., $\max(\alpha P^*)$ with $\max(\alpha^*)$. Still, this should be considered a bug, and it has to be worked out in further research on this formalism. Note that Brody considers the identification of an XP with its X^0 to be problematical, and requires non-trivial phrasal extension of all heads. Adopting this stipulation our formal problem goes away, but we lose certain supposed empirical benefits in the theory of clitics which encourage Chomsky to allow trivial projection, the more economical alternative.

yields a structure on which we can begin to compute π_{pf}. Since the X^0 of s contains the PF position of $\mathsf{f_a}^*$, we know already that $\pi_{pf}(\mathsf{s}^0)$ will be a string with prefix $\epsilon\,\mathsf{0}$, i.e., $\mathsf{0}$, so nothing we can do will derive $\mathsf{1010}$ from such a structure. To get our desired interpretation, we must attach $\mathsf{f_b}\mathrm{P}^*_0$ as complement, and attach $\mathsf{f_b}{}^*_1$ into s^0. Selection requires us to attach the sole position belonging to b as specifier of $\mathsf{f_b}\mathrm{P}$, creating the configuration in which b and $\overline{\mathsf{b}}$ are cancelled. So we have derived a structure which contains only s and the $\overline{\mathsf{F}}$ feature of $\mathsf{f_b}$ as unchecked features.

Again, we could attach $\mathsf{f_0}$ to $\mathsf{f_b}$, but that would create another s-tree, leaving us with one numeration item too many. So we attach $\mathsf{f_a}\mathrm{P}^*_0$ to the complement position of $\mathsf{f_b}\mathrm{P}$, and we attach $\mathsf{f_a}{}^*_1$ into $\mathsf{f_b}{}^0$. Attaching the sole position belonging to a as specifier of $\mathsf{f_a}\mathrm{P}$ again leaves us with a structure with only s and $\overline{\mathsf{F}}$ as features, and now we have only one remaining numeration item, namely $\mathsf{f_0}$. We attach $\mathsf{f_0}\mathrm{P}^*_0$ to complement position in $\mathsf{f_a}\mathrm{P}$, and $\mathsf{f_0}{}^*_1$ as head adjunct in $\mathsf{f_a}{}^0$, cancelling both the $\overline{\mathsf{F}}$ feature of $\mathsf{f_a}$ and the f feature of $\mathsf{f_0}$.

To compute π_{pf} of the resulting structure, we compute

$$\pi_{pf}(\mathsf{s}\mathrm{P}^*_0).$$

That is, necessarily, the PF position of $\mathsf{s}\mathrm{P}^*$, so this equals

$$\pi_{pf}(\mathsf{s}\mathrm{P}_1)$$

which in turn equals, by definition,

$$(\pi_{pf}(\mathsf{s}^*_0) > \pi_{pf}(\mathsf{f_b}\mathrm{P}^*_0)).$$

Both of these are trivially the PF positions of their respective chains. Taking the lefthand subterm first we derive

$$\begin{aligned} &(\pi_{pf}(\mathsf{s}^0_1) > \pi_{pf}(\mathsf{f_b}\mathrm{P}^*_0)) \\ = {}&((\pi_{pf}(\mathsf{s}^0_0) \circ \pi_{pf}(\mathsf{f_b}{}^*_1)) > \pi_{pf}(\mathsf{f_b}\mathrm{P}^*_0)) \\ = {}&((\pi_{pf}(\mathsf{s}) \circ \pi_{pf}(\mathsf{f_b}{}^0_1)) > \pi_{pf}(\mathsf{f_b}\mathrm{P}^*_0)) \end{aligned}$$

The last identity follows from the fact that the f feature of $\mathsf{f_b}$ is strongly attracted by the $\overline{\mathsf{F}}$ attractor of s, making $\mathsf{f_b}{}^*_1$ the PF position of that chain.

Note that our interpretation is ultimately in terms of strings (we care only that our system weakly generate $\mathcal{L}_{ww}$). In particular we do not care about headedness information, so we will interpret all three of $>$, $<$, $\circ$ simply as string concatenation, which is associative, so we can drop parentheses freely. We continue.

$$
\begin{aligned}
&((\pi_{pf}(\mathsf{s}) \circ \pi_{pf}(\mathsf{f_b}{}_1^0)) > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*)) \\
&= ((\epsilon \circ (\pi_{pf}(\mathsf{f_b}{}_0^0) \circ \pi_{pf}(\mathsf{f_a}{}_1^*))) > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*)) \\
&= ((\pi_{pf}(\mathsf{f_b}) \circ \pi_{pf}(\mathsf{f_a}{}_1^0)) > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*)) \\
&= ((1 \circ (\pi_{pf}(\mathsf{f_a}{}_0^0) \circ \pi_{pf}(\mathsf{f_0}{}_1^*)))) > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*)) \\
&= ((1 \circ (\pi_{pf}(\mathsf{f_a}) \circ \pi_{pf}(\mathsf{f_0}{}_0^0))) > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*)) \\
&= ((1 \circ (0 \circ \pi_{pf}(\mathsf{f_0}))) > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*)) \\
&= ((1 \circ 0 \circ \epsilon) > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*)) \\
&= (10 > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*))
\end{aligned}
$$

Note in particular that our attachment relation is subject to strictly local licensing conditions. It cares about the features at either end of the arc it builds and looks no further. Similarly, determining the PF position of a chain only requires looking at the attachment arc of each chain to see whether there is a strong or weak attractor at the far end. Yet these local computations come together with a simple recursive descent of the structure to give us the effect of head movement.

We continue the example with the computation of the right-hand subterm.

$$
\begin{aligned}
&(10 > \pi_{pf}(\mathsf{f_b}\mathrm{P}_0^*)) \\
&= (10 > \pi_{pf}(\mathsf{f_b}\mathrm{P}_2)) \\
&= (10 > (\pi_{pf}(\mathsf{b}\mathrm{P}_0^*) < \pi_{pf}(\mathsf{f_b}\mathrm{P}_1)))
\end{aligned}
$$

Despite the fact that b has only a trivial syntactic extension, it takes a few steps to get to its interpretable features as we move from the XP chain position to the XP to the X^* to the X^0. We skip these steps here, and likewise we will skip over most other identification steps in what follows.

$$
\begin{aligned}
&10 > (1 < \pi_{pf}(\mathsf{f_b}\mathrm{P}_1)) \\
&= 10 > (1 < (\pi_{pf}(\mathsf{f_b}\mathrm{P}_0) > \pi_{pf}(\mathsf{f_a}\mathrm{P}_0^*))) \\
&= 10 > (1 < (\pi_{pf}(\mathsf{f_b}{}_0^*) > \pi_{pf}(\mathsf{f_a}\mathrm{P}_2)))
\end{aligned}
$$

Position $\mathsf{f_b}{}_0^*$ is not the PF position of $\mathsf{f_b}^*$, so the value of π_{pf} there is the special empty placeholder symbol λ. In the string interpretation it amounts to another name for the empty string ϵ. So we have the following.

$$
\begin{aligned}
&(\mathsf{10} > (\mathsf{1} < (\pi_{pf}(\mathsf{f_b}{}^*_0) > \pi_{pf}(\mathsf{f_a}\mathrm{P}_2))))\\
&= (\mathsf{10} > (\mathsf{1} < (\epsilon > \pi_{pf}(\mathsf{f_a}\mathrm{P}_2))))\\
&= (\mathsf{10} > (\mathsf{1} < (\epsilon > (\pi_{pf}(\mathsf{a}\mathrm{P}^*_0) < \pi_{pf}(\mathsf{f_a}\mathrm{P}_1)))))\\
&= (\mathsf{10} > (\mathsf{1} < (\epsilon > (\mathsf{0} < (\pi_{pf}(\mathsf{f_a}\mathrm{P}_0) > \pi_{pf}(\mathsf{f_0}\mathrm{P}^*_0))))))
\end{aligned}
$$

Position $\mathsf{f_a}\mathrm{P}_0$ is identified with the non-PF position $\mathsf{f_a}{}^*_0$ of $\mathsf{f_a}{}^*$, and similarly for $\mathsf{f_0}\mathrm{P}^*_0$ ($= \mathsf{f_0}\mathrm{P}_0 = \mathsf{f_0}{}^*_0$). So both are spelled out as the empty string and we have

$$\mathsf{10} > (\mathsf{1} < (\epsilon > (\mathsf{0} < (\epsilon > \epsilon))))$$

which simplifies to

$$\mathsf{1010}$$

as desired.

4.3. **Defining $\mathcal{L}_{ww}$.** As in Cornell (1996) we have defined a ϕ-tree as any tree whose only uncancelled feature is ϕ. We characterize $\mathcal{L}_{ww}$ as the set of s-trees. The same counting arguments as were used in the derivational version apply here: there must be exactly as many instances of ϕ as there are instances of $\overline{\phi}$ in any numeration, for all $\phi \neq \mathsf{s}$. Therefore, for every instance of s in the numeration, which introduces an $\overline{\mathsf{F}}$ attractor without introducing an f attractand, there must be a balancing instance of $\mathsf{f_0}$, which introduces an f attractand without introducing an $\overline{\mathsf{F}}$ attractor. The same reasoning applies in reverse: only an occurrence of s can balance an occurrence of $\mathsf{f_0}$. There can only be one instance of s, since nothing selects s, so there can only be one instance of $\mathsf{f_0}$. So every numeration has the same basic form: an instance of s (intuitively, the left-hand string terminator) and an instance of $\mathsf{f_0}$ (intuitively, the right-hand string terminator), and then some number of instances of $\mathsf{f_a}$ and $\mathsf{f_b}$ (maybe zero), each with a balancing instance of a and b, respectively. Any such numeration will extend to a well-formed structure. It may extend to several well-formed structures, in fact, if we assume that the numeration is unordered. We argue here that each such structure will yield a string from $\mathcal{L}_{ww}$.

First note again that all instances of an f-item will attach as complements, either to another f category or to the root s category, owing to the conditions on selectional features. This yields a right-branching "spine" of f phrases, capped off on the left/top by the s phrase and at the right/bottom with the f_0 phrase. All the a's and b's will attach as specifiers in f_a and f_b categories, respectively. This completes the Attachment relation.

We have not specified any particular theory of locality. For the purposes of dealing with the locality problem for X^0 chains this is not necessary: our solution works for pretty much any particular implementation of the notion of locality. However, to show that we have indeed captured a particular language, we do need to commit to some such theory here. We adopt the same theory of locality as in Cornell (1996), namely, that a head can only attract the closest head with a clashing feature, where "close" means that no intervening c-commander has the same feature as the attractand. Since the features which drive X^0 chain formation are so specific, being senstive to the category of the attractand, we derive hereby the Head Movement Condition, at least for this grammar. In particular, only the heads of complements can be attracted, so every chain will be (strongly) adjoined to the same host to which the corresponding phrase is attached. Therefore the spell-out position for all the orthographic features of the f categories will be inside the head position of the s phrase. The order of spell-out reflects the structure of the attachment relation and so we generate under the s head a copy of the orthograpy associated with the string of specifiers in the "main" tree.

5. Back to Trees

We have examined in the foregoing sections a rather abstract representation of syntactic structure. It was developed with the primary goal of being explicit about the elements and relations which are of particular interest to linguists. This has led us to representations which are more revealing of syntactic structures than the trace-annotated phrase structure trees we started with. At the same time the computations we require are made easier both to formulate and to execute, by making the information which is most important most accessible. However, these structures are rather nontraditional, and will certainly not appeal to all linguists. So we pause for a moment at the end of this work to consider what we have learned of the X^0 chain locality problem, and how we might use that to solve it for a strictly tree based representational system.

The main observation was made at the outset: the geometry of the locality problem for a (head,trace) pair is the same as the geometry of the locality problem for a (host trace,dependent trace) pair. Our main effort was to use the problem of X^0 chain locality to lead us to the properties of X^0 chains which are central in linguistic theorizing and which therefore ought to be central in linguistic structures. We can recover the essential features of our (fairly obvious) solution for use in a universe of phrase structure trees if we consider the locality problem in the following light. The locality problem is essentially a search problem. In fact, for chains, it is the inverse of the search problem embedded in movements. In a theory with the Move operation (in particular the Attract operation of Chomsky (1995)), the attracting head must in effect search for something to attract, with the scope of that search constrained by the theory of locality (e.g., the search can be turned back at the borders of opaque domains, or distracted by invalid material which is too similar to the sought-for item). A representational theory, given in terms of chains and their properties, is of course neutral as to the direction of search, but whether we think of evaluating locality constraints in terms of an antecedent looking for its trace or a trace looking for its antecedent, the essential computational features of a search problem are embedded in the relevant definitions. "Top down," then, the antecedent must in effect search for its trace. The structure of this search is recursive. In the simplest case, the search is trivial: the "antecedent" and its "trace" are identical. Otherwise, the search for a trace must begin at the cite of the antecedent's host's trace. But where is the host's trace? It must be found by the same algorithm. So the definition of locality is recursive: locality for a dependent is defined in terms of locality of its host.

So, to put it another way, the "deformed" link $(\mathsf{f}_0, t_{\mathsf{f}_0})$ in the tree of Figure 2 (on p. 9) is well formed if it is only a certain degree longer than the link $(\mathsf{f}_\mathsf{a}, t_{\mathsf{f}_\mathsf{a}})$ of its host (where the meaning of "a certain degree longer" is determined by the theory of locality). However, in order to verify that, we first have to measure the link $(\mathsf{f}_\mathsf{a}, t_{\mathsf{f}_\mathsf{a}})$ in terms of the link $(\mathsf{f}_\mathsf{b}, t_{\mathsf{f}_\mathsf{b}})$ of *its* host. And so on.

There is a more declarative, structural view of this same scene. We think of the objects formed by head movement as a sort of "meta-chain", a recursively nested collection of chains. The "core" of this structure—the innermost chain—must be local, and then every extension of this structure by an additional chain must be local as well. The way in which this structure recursively elaborates the structure of "standard" chains in the same way that the structure of chains and phrases elaborates heads illustrates at once the extra expressive power that head-movement constructions give to grammars and the underlying unity of the mathematical devices available to them.

6. Conclusion

In conclusion, it could be argued that we have done little more than to steal ideas from the theory of movement, simply redefining chain formation in such a way that it is not essentially different from defining movement steps. This could be true, but it is not an argument. Certainly one goal of syntactic theory should be the elucidation of syntactic structures. Derivations do not do this. Derivations *traverse* syntactic structure, but all they produce is the final (LF,PF) pair. It can be argued in defense of that approach—and in fact I believe this to be true—that the valid (LF,PF) pairs can be enumerated without ever actually building the syntactic structure of which these coordinates are the projections, but this does not mean that syntactic structure does not exist or even that it is pointless to study it. Derivations are themselves structured objects, of course, and therefore they are technically also "syntactic structures". However, these structures are really too highly structured to satisfy minimalist demands for direct research into syntactic structure; for example, several derivations may yield the same (LF,PF) pair, or the operations can be superficially redefined, again without affecting the (LF,PF) pairs derived. For certain kinds of investigation greater abstraction is required, and we are prepared to sacrifice the detailed control we get from highly operationalized, procedural notions like (particular interpretations of) Move to get the leverage we gain with greater abstractness. We should conclude, I believe, that just as proof theory and model theory both have their uses in logic, and operational and denotational semantics both have their uses in computer science, so derivational and representational approaches both have their uses in linguistics. To return to the first point, it should be added that these bipolar worldviews are only truly useful *when they are equivalent*, so it would in fact be good methodological practice to steal ideas from the theory of movement (or, vice versa, from the theory of chain formation) on a regular basis. In an ideal formal setting, to the extent that one person's representational theory gives different results from another person's derivational theory, they must be making different assumptions about syntax that would not change in the least if the former were to adopt representationalist methods or the latter derivationalist. Nonetheless, as I hope to have shown, it takes some serious, technical, formal work to make those translations smooth.

References

Brody, M. (1995). *Lexico-Logical Form*, MIT Press, Cambridge, MA.

Chomsky, N. (1975/1955). *The Logical Structure of Linguistic Theory*, University of Chicago Press, Chicago.

Chomsky, N. (1995). *The Minimalist Program*, MIT Press.

Cornell, T. (1996). A minimalist grammar for the copy language, Ms., University of Tübingen; drafts available at http://www.sfs.nphil.uni-tuebingen.de/~cornell.

Huybregts, R. (1984). The weak inadequacy of context-free phrase structure grammars, *in* de Haan, Trommelen and Zonneveld (eds), *Van Periferie naar Kern*, Foris, Dordrecht, pp. 81–99.

Koopman, H. (1994). Licensing heads, *in* D. Lightfoot and N. Hornstein (eds), *Verb Movement*, Cambridge University Press, Cambridge, pp. 261–296.

Kracht, M. (1995). Syntactic codes and grammar refinement, *Journal of Logic, Language, and Information* **4**: 41–60.

Moortgat, M. (1996). Categorial type logics, *in* J. van Benthem and A. ter Meulen (eds), *Handbook of Logic and Language*, Elsevier, chapter 2.

Rizzi, L. (1990). *Relativized Minimality*, MIT Press, Cambridge, Mass.

Rogers, J. (1994). *Studies in the Logic of Trees with Applications to Grammar Formalisms*, PhD thesis, University of Delaware, Department of Computer & Information Sciences.

Shieber, S. M. (1985). Evidence against the context-freeness of natural language, *Linguistics and Philosophy* **8**: 333–343.

Stabler, E. P. (1996). Acquiring and parsing languages with movement, Available at http://128.97.8.34/utrecht/utrecht.html.

SfS, Universität Tübingen, Wilhelmstr. 113, D-72074 Tübingen, Germany
E-mail address: `cornell@sfs.nphil.uni-tuebingen.de`