CLS 22.1.65-74
(1986)

Concatenation and Liberation*

Arnold M. Zwicky
Ohio State University and Stanford University

1. Wrapping and obligatorily discontinuous constituents. The immediate dominance (ID) portion of a phrase structure rule specifies that certain daughter constituents, belonging to the categories $D_1 \ldots D_n$, can combine to form a construct belonging to the mother category M. In standard conceptions of phrase structure grammar the combining operation is concatenation, but (as many have noted) other operations are conceivable; separate approaches to wrapping operations as methods of combination have been proposed, for instance, by Bach (1979, 1981, 1983) for syntax and morphology, Pollard (1984) for syntax, and Hoeksema (1984, ch. 2) in morphology.

When M_1 is wrapped around M_2, the daughters of M_1 serve as sisters of M_2, some on either side of it. Wrapping then allows obligatorily discontinuous constituents, as in (1), to be treated as syntactic, and semantic, constructs.

(1) a. English V^1:
elect president IN elect Robin president
b. V+O in VSO languages, e.g. Welsh:
gwelodd gath 'saw a cat' IN Gwelodd Jac gath 'Jack saw a cat'

This treatment separates two functions of an ID rule that are in principle independent of one another: the assignment of material to constructs that can function in the statement of syntactic and semantic rules; and the licensing of branchings in tree structures associated with particular sentences.[1] On this treatment, the fact that the combination *elect president* is analyzed as a construct, let us say a construct of category V^1, means that there can be pro-forms for the combination (as in (2a)) and that we expect there to be idioms of category V^1 (for instance, *keep company* as in (2b,c)). And compositional semantics can treat *Robin* as the direct object of *elect president*. On the other hand, the wrapping analysis assigns a tripartite, or flat, structure to branchings of VP in the tree structures associated with sentences like *We elected Robin president*.

(2) a. We elected Robin president long before we did Sandy.
b. We kept Kim company.
c. *We kept company Kim.

2. Liberation and free word order. When the reference to linear precedence ('some on either side of it') is removed from the informal description of wrapping I gave above, we have in fact an operation, *liberation*, that has played a role in theorizing about a very different syntactic issue, namely the analysis of free word order.

In Pullum's (1982) and my (1985) GPSG treatments, constituent structure is flattened by a metarule permitting the daughters of M_1 to serve as sisters of M_2; freedom of order follows, not directly from flattening, but as a result of flattening plus the

absence of conditions on the linear precedence (LP) of sisters. For Pullum, such metarules are parochial statements. That is, a metarule like (3), which liberates the daughters of VP into S and so permits 'Satzglieder' to be treated as sister constituents, is stipulated as part of the grammar of any language with free constituent order (which is to say, as part of the grammar of most languages). My 1985 approach posits instead a Universal Liberation Metarule liberating the daughters of B into A, as in (4), with parochial stipulation localized in the list P of pairs (A, B) subject to this universal metaprinciple (so that free constituent order is described by stipulating that the pair (S, VP) is in P).[2]

(3) *IF* VP ---> V, X *THEN* S ---> NP, V, X

(4) Given a pair-set P for a language, then for any pair (A, B) in P,
   a. A and B are phrasal categories,
   b. B ≠ S, and
   c. *IF* A ---> B, X *AND* B ---> Y *THEN* A ---> X, Y,
      where X and Y are any sets of categories.

Both proposals combine constituents by concatenation and then liberate one or more of them, but in fact the desired effects can be obtained straightforwardly by treating liberation as an alternative operation to concatenation. That is, instead of saying that V, NP, and PP can be concatenated into a VP and are then liberated into the S dominating this VP, as Pullum and I did in our earlier approaches, it is possible to say that VP is combined with its subject NP by liberation rather than concatenation. I will refer to the Pullum/Zwicky proposals as <u>indirect liberation</u> approaches, and to the one I am advocating here as <u>direct liberation</u>.

Depending on other assumptions about syntactic rules, direct liberation analyses might be consistent with a wider range of word order facts than indirect liberation analyses. The crucial additional variable is whether (as in early developments of GPSG, such as Gazdar (1982)) there can be distinct ID rules with identical categorial content - separate ID rules of the form in (5),[3] for instance, for verbs of the *expect*, *ask*, *promise*, and *believe* classes - or not (as in Gazdar et al. (1985)). If there can be, then a direct liberation framework would allow for the possibility that some of these rules combined constituents via liberation while others employed only the concatenation operation, so that the word order facts for the one set of constructions could be different from the word order facts for the others. If, on the other hand, there can be only one such ID rule, then the direct liberation framework makes the same predictions as the indirect liberation frameworks, namely that the word order facts are identical for all such constructions.

(5) VP ---> V, NP, VP:+INF

Thus far we have seen two uses for the operation of liberation: (1) in the analysis of obligatorily discontinuous constituents; (2) in the analysis of free word order. There are

at least three further uses: (3) flat structures in rigid order languages (section 3 below); (4) alternative hierarchical and flat structures (section 4); and (5) what I will refer to as 'daughter borrowing' (section 5).

3. <u>Flat structures in 'configurational' languages</u>. The liberation operation permits constructions (especially in rigid word order languages) which for semantic, and some syntactic, purposes are binary (and therefore hierarchical) to have flat structures nevertheless. The multiple verb constructions of Germanic languages, as in English *must have been being observed*, are a case in point.

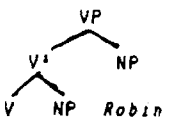
In early transformational grammar, the auxiliary verbs in these constructions were assigned a flat structure, without explicit argument. In works like *Syntactic Structures*, flat structures seem to have been taken to represent the null hypothesis; indeed, ordinary distributional evidence does not suggest a hierarchical structure here, nor do the facts of phonological phrasing. Nevertheless, the way VP Deletion, as in (6a), and VP Fronting, as in (6b), work clearly points to a binary right-branching structure. (The situation in German is essentially the same, except that there is left branching in a construction like (6c).)

(6) a. Robin must (have (been (being))).
   b. Being observed Robin must have been.
   c. (dass er) gesehen geworden haben muss '(that he) must have been seen'

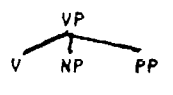
Liberation provides a way of having things both ways. An ID rule can introduce an auxiliary V and its complement VP as daughters of VP, thus making the complement VP available as a constituent for the purposes of VP Deletion (in which the VP is empty) and VP Fronting (in which VP serves as the value of SLASH). However, if VP is combined with V by liberation rather than concatenation, its daughters will appear as sisters, not nieces, of V in the branchings described by the ID rule.

4. <u>Alternative flat and hierarchical structures</u>. Liberation also provides a way of treating hierarchical and flat structures as alternative combinations of the same constituents. The English dative constructions, hierarchical *show Kim Robin* and flat *show Robin to Kim*, as in (7), can be analyzed in this way, as can the ordering of the major sentence constituents in Chinese, hierarchical SVO and SOV versus flat OSV.

(7)



show Kim



show Robin to Kim

In both structures in (7), the direct object NP *Robin* has been combined with a V¹, the V¹ itself involving a combination of

the V *show* with its indirect object *Kim*. In the hierarchical structure on the left in (7), V^1 and its direct object have been combined by concatenation. In the flat structure on the right, V^1 has been combined with its direct object by liberation, with the result that there are branchings in which the verb, its direct object, and its indirect object are sisters of one another. I assume that the trappings of dative case in English, involving a PP with the *P to*, appear when an indirect object is a daughter in a branching of the category VP (rather than V^1); I will return to this matter in section 7.

5. Daughter borrowing. Finally, liberation allows us to say that a construction has the external syntax of an X but the internal syntax (and in part the semantics) of a Y, without having to assign instances of the construction to both the categories X and Y. The standard analysis of such phenomena requires double category assignment, via a rule X ---> Y, and so fails to distinguish them from single branchings that result from the nonappearance of an optional daughter (as is the case for NP ---> N^1, with no Det, and N^1 ---> N, with no AP).

Now there are a great many phenomena that might fall under this rubric, and it is not clear to me that an analysis appropriate for one is appropriate for all. Here I will mention only two possible instances from English, without claiming that the analyses I suggest are necessarily the right ones. The examples I have in mind are (a) PPs like *with trepidation* serving as an AdvPs, as in (8b), and (b) sentential constructs like *for Robbie to sing* serving as NPs, as in (9b).

(8) a. Stacy moved the microtome carefully.
    b. Stacy moved the microtome with trepidation.
(9) a. A surprise party would distress me.
    b. For Robbie to sing would distress me.

Liberation gives us a way to describe branchings of AdvP into daughters P and NP, without an intervening node PP in the branchings; thus AdvP 'borrows' the daughters of PP. And it gives us a way to describe branchings of NP into daughters PP and VP, without an intervening node S[-FIN] in the branchings; thus NP borrows, in a variant form, the daughters of S. What we need to say (if these are the consequences we want) is that in the rules in (10) the (only) daughters are liberated, so that their daughters will appear in branchings of the mother categories mentioned in the rules.

(10) a. AdvP ---> PP
     b. NP ---> S[-FIN]

6. Formal matters. I now sketch a formalization of ID rules within a framework embracing liberation as well as concatenation.

The syntactic portion of an ID rule is an ordered triple (*M*, *C*, *L*). *M* is a set of conditions on the mother category, *C* is a multiset,* each member of *C* being a set of conditions on one concatenated daughter category, and *L* is a multiset, each member of *L* being a set of conditions on one liberated daughter

category. The familiar rule in (11a) corresponds to the ID rule in (11b), where the square brackets are used to name multisets; there is no liberation here. A free constituent order language like German or Finnish, in which the daughters of VP are liberated, has instead of (11b) the ID rule in (11c). And a free word order language, in which the daughters of both subject NP and VP are liberated, has instead of (11b) or (11c) the ID rule in (11d).

(11) a. S ---> NP, VP
     b. (S, [NP, VP], ∅)
     c. (S, [NP], [VP])
     d. (S, ∅, [NP, VP])

I now turn to the other classes of phenomena for which a liberation treatment might be suitable.

First, obligatorily discontinuous constituents, for instance the VP in VSO languages. Let me suppose for this discussion that (certain instances of particular) grammatical relations are treated as features on NPs, so that a subject is something belonging to the category NP[GR:1]. Then the relevant ID rules are as in (12a,b), with the LP condition in (12c) stipulating the rigid order of the Satzglieder.

(12) a. (S, [NP[GR:1], [VP])
     b. (VP, [V, NP[GR:2], ∅)
     c. V < NP[GR:1 < NP[GR:2

Next, flat structures in generally rigid word order languages, in particular multiple verb constructions in the Germanic languages. These can be described via the ID rule in (13), in combination with suitable LP rules (ordering modals before perfective *have* before progressive *be* before passive *be*).

(13) (VP, [V], [VP])

Flat and hierarchical structures can be described as alternatives via ID rules like those in (14a), for the flat VP of *show Robin to Kim*, and (14b), for the hierarchical VP of *show Kim Robin*. (Here I have decomposed the major categories into features, so that V is +V, -N; P is -V, -N; and N is -V, +N. And I indicate bar level with superscripts.) In addition to (14a,b) we need the ID rule in (14c) for V^1, and the LP rule in (14d).

(14) a. ($+V^2$, [$-V^2$[GR:2], [$+V^1$])
     b. ($+V^2$, [$-V^2$[GR:2, $+V^1$], ∅)
     c. ($+V^1$, [$-V^2$[GR:3], ∅)
     d. $+V^n$ < -V *for* n < 2

Formulating the rules this way predicts variable order, V+DO+IO and V+IO+DO, for the flat structure (since (14d) does not order DO and IO with respect to one another), as in (15a), but rigid order, V+IO+DO, for the hierarchical structure (since (14d) orders IO after V within V^1 and orders DO after this V+IO combination within V^2), as in (15b).

(15) a. give a marvelous accolade to them,
give to them a marvelous accolade
c. give our generous benefactors thanks,
*give thanks our generous benefactors

Note also that since the flat and hierarchical structures are described by distinct (though similar) ID rules, there is no guarantee that all the verbs occurring in one construction also occur in the other - a welcome consequence, given the familiar facts indicating the partial independence of the two constructions.

Finally, daughter borrowing. This will be formalized with an empty C and a singleton L, as in (16), an ID rule for NP borrowing the daughters of nonfinite S.

(16) ⟨NP, ∅, {S[-FIN]}⟩

7. Rules versus branchings again. The approach I have been sketching here dissociates the contents of ID rules from the contents of the branchings these rules describe. We must then inquire as to how this move interacts with other aspects of the GPSG framework, in particular with the treatment of semantic interpretation, subcategorization, agreement, government, and linear precedence.

7.1. Semantic interpretation and subcategorization. Both semantic interpretation and subcategorization must in this framework be associated with individual ID rules. To see this, consider the situation in which the rules and the branchings they describe are most divergent, namely a free word order language.

Suppose we are confronted with a sentence composed of a verb along with five NPs, representing a subject, direct object, indirect object, benefactive, and locative, each NP having not only its head N but also a modifying A; such a sentence might translate into English as 'The tall cook gave the fearless hunter a blue pot for the industrious farmer in a dark spot'. In the branching for this sentence in a free word order language there are eleven sister constituents (five Ns, five As, and a V), and nothing to indicate how the various constituents 'go together'. I cannot imagine how subcategorization requirements could be checked in such configurations, nor how semantic interpretations could be composed from the semantic interpretations of the parts.

The information about which constituents go together, and in what way, is available, however, in the ID rules of the language. That is what ID rules are for.

7.2. Feature instantiation and linear precedence. Matters are more complex here. At first glance, it would seem that agreement and government conditions - and in general all feature instantiation (FI) conditions (sister-sister or mother-daughter conditions on the distribution of features) - have the configurations in ID rules as their domains, while LP conditions have branchings as their domains. Certainly these assumptions represent what must be taken as the usual mode of application of

these conditions.

Consider, again, a free word order language. If the Control Agreement Principle of GPSG is to work as intended, it must apply to sister categories as defined by ID rules, for in the actual branchings there will be, for example, numerous As and Ns that are candidates to agree with one another. Similar observations hold for the distribution of morphological case features. But the typical LP condition in such a language - requiring, for instance, that certain items occur in second position - takes cognizance of what is in particular branchings, regardless of which rules licensed those branchings.

Unfortunately, not all the analyses I put forward in my earlier discussion can be accommodated in this simple scheme. For instance, I said in section 4 that 'the trappings of dative case in English, involving a PP with the P *to*, appear when an indirect object is a daughter in a branching of the category VP (rather than V^1)'. But this assumes that case assignment looks at the context of $-V^2$|GR:3 in branchings, not in ID rules (where such constituents always appear with the mother category $+V^1$, as in rule (14c) above).

There might also be circumstances in which we would want certain LP conditions to apply to the daughter-sets specified in ID rules, rather than to the constituents appearing in branchings. I will mention one such circumstance in section 8 below, but here it should suffice to consider again the (familiar) facts about English auxiliaries rehearsed in section 3 and rule (13) of section 6. The linear order of the auxiliaries reflects the order in which they are combined by the ID rules. If only hierarchical structures are assigned to VPs with auxiliaries in them, as in the standard GPSG treatments, then this ordering is fully described by a single LP condition requiring that V precede VP; this LP condition applies within each of the binary branchings licensed in such an analysis. However, if flat structures are assigned to VPs with auxiliaries in them, as in my suggestions above, then the ordering must be stipulated in an LP condition that has no visible relationship to the ID rules. We could capture this relationship in the liberation framework by having the LP condition requiring V before VP apply to the daughter-sets of ID rules, rather than to the flat branchings these rules license. An analysis like this would indeed be forced in the liberation framework for a language that permitted (say) multiple occurrences of infinitival complements to verbs but rigidly ordered sequences of verb forms so that the V[+INF] head of a complement had to follow its governing V immediately.

Yet another set of problems arises when we consider agreement rules that take cognizance of linear ordering, in particular the German and Russian rules mentioned by Hinrichs (1985) that license agreement with the nearest suitable controller constituent.[8]

Hinrichs' general proposal is that FI conditions 'can be viewed as wellformedness constraints on possible mappings f from constituents of ID rules to nodes of trees' and that 'LP rules can be conceived of as constraints on the set of possible mappings between ID rules and ordered trees' (127). Seeing both FI conditions and LP conditions as constraints on the association between ID rules and branchings extends the set of possible

grammars from what I described above as the 'usual' situation, in which FI conditions serve only as conditions on ID rules and LP conditions only as conditions on branchings. I would not want to embrace an unconstrained extension of the GPSG framework here, but it seems clear to me that some extension is called for. At the very least, I believe, it must be possible for some FI conditions to serve as conditions on branchings and for some LP conditions to serve as conditions on the daughter-sets of rules.

A few words on parochial (language-particular, nonuniversal) metarules. In a syntactic framework without the liberation operation, it would make no difference whether LP conditions are checked on the daughter-sets of ID rules or on the branchings those rules describe - unless there are parochial metarules, in which case the metarules could take account of ordering differences in the daughter-sets of ID rules; consequently it is ordinarily assumed that LP conditions apply to branchings and so 'follow' the application of all syntactic rules, whether ID rules, FI conditions, or parochial metarules. In a syntactic framework with the liberation operation, it clearly makes a descriptive difference which way LP conditions are checked, and parochial metarules would add still another degree of freedom, producing a system of unknown properties. It would therefore be a great metatheoretical blessing to be without parochial metarules, especially in a framework that includes the liberation operation.*

B. On the limitations of liberation. What sorts of stringsets are described by phrase structure syntaxes permitting liberation as an alternative operation to concatenation? Obviously, trading some or all instances of concatenation for liberation does not affect the symbols that occur in licensed strings. But can the LP conditions that apply within hierarchical structures be mirrored within flat structures, and can the freedom of order that goes with the flat structures be reproduced within hierarchical structures? I believe the answer to both these questions is yes, and that proofs can be constructed using the fact that the number of ID rules is finite, so that the daughter nodes licensed by a particular rule can be indexed as the 'products' of that rule. But I have not worked through the details of the proofs.

In any event, I suppose that the stringsets described by this version of phrase structure syntax are in fact the context-free (CF) languages, despite the novel form of the syntax. Liberation consequently cannot do all the work done by Bach's and Pollard's wrapping operations, which were explicitly intended to make the class of languages described trans-CF (though not wildly so). What would it take to extend the framework that I have discussed here so that it will be mildly trans-CF in the appropriate way?

There is now a considerable literature on appropriate extensions of or replacements for CF syntax. For instance, Gazdar (1985), citing a range of earlier work, discusses the indexed grammars, in which a CF syntax is supplemented by a single stack-valued feature that can be manipulated by the rules of the syntax. And in a series of recent papers, Manaster-Ramer and his associates have maintained that the central function of CF rules - to create nested and mirror-image constructions - is unnecessary,

and that such rules should be abandoned in favor of rules that copy material; see, for instance, Manaster-Ramer (to appear).

What I propose here is that CF syntax should be extended à la Gazdar rather than abandoned à la Manaster-Ramer, but that the appropriate extension, assuming that liberation is already available, is a third operation that has the effect of copying material and so can describe sublanguages involving crossed dependencies. A Pollard-style wrapping operation ! will do the trick; when ! is applied to constituents X and Y, the head of X follows the head of Y and the remainder of X (that is, X minus its head) follows the remainder of Y. Given that one type of VP has the head constituent V and the remainder NP, if we say that another type of VP is formed by applying ! to the constituents VP and VP, we can describe a set of VPs of the form in (17), exhibiting unlimited crossed dependencies.

(17) $V_1V_2 \ldots V_nNP_1NP_2 \ldots NP_n$

For this analysis to work, however, LP conditions must be checked on each output of !, not on the (flat) branchings that would be described by repeated applications of an order-free variant of !. Once again, branching conditions, FI conditions, and LP conditions interact in complex ways, as they do when liberation is added to the framework of GPSG.

Notes

* Special thanks to the institutions (the Ministry of Education of the People's Republic of China, the Committee for Scholarly Communication with the PRC of the U.S. National Academy of Sciences, and the College of Humanities of the Ohio State University) whose support enabled me to spend the autumn of 1985 teaching at the Beijing Language Institute, where most of the theoretical ideas in this paper were developed. And to the many people who commented on earlier versions of this paper, presented 9 December 1985 at Ohio State and 17 April 1986 at the CLS, and so precipitated a number of substantial revisions in its form and content. This is the version of 17 May 1986.

1. This use of wrapping then provides an alternative to the analysis of discontinuities via explicitly discontinuous tree structures, as in McCawley (1982).
2. There are other relevant differences between the two approaches. In particular, Pullum's approach predicts that an SVO clause in a language with free constituent order has both hierarchical and flat structures, while mine assigns only the flattest structure available.
3. The vertical bar '|' separates major category and bar features from other features of a constituent.
4. See Gazdar et al. (1985, 53f) for a discussion of multisets in the formalization of ID rules.
5. I would not want to rule out in principle the possibility that a language might have government rules of a similar sort,

stipulating (say) that a verb governs a particular morphological case on the candidate NP that is closest to it.

6. Here I echo the general sentiment among GPSG enthusiasts that parochial metarules are a bad thing and ought to be eliminated from the framework. I believe in fact that parochial metarules are eliminable, in favor of various other descriptive mechanisms, including not only the use of liberation as in (12), but also lexical redundancy rules, reference to grammatical relations, and the use of various SLASH-like feature-valued features.

References

Bach, Emmon. 1979. Control in Montague Grammar. LingI 10.515-31.

Bach, Emmon. 1981. Discontinuous constituents in generalized categorial grammars. NELS 11.1-12.

Bach, Emmon. 1983. On the relationship between word-grammar and phrase-grammar. NLLT 1.65-89.

Gazdar, Gerald. 1982. Phrase structure grammar. In Pauline Jacobson and Geoffrey K. Pullum (eds.), On the nature of syntactic representation (Dordrecht: Reidel), 131-86.

Gazdar, Gerald. 1985. Applicability of indexed grammars to natural languages. Report CSLI-85-34. Center for the Study of Language and Information, Stanford Univ.

Gazdar, Gerald, Ewan Klein, Geoffrey Pullum, and Ivan Sag. 1985. Generalized phrase structure grammar. Oxford: Basil Blackwell.

Hinrichs, Erhard. 1985. A new approach to feature instantiation in GPSG. OSU WPL 31.122-9.

Hoeksema, Jacob. 1984. Categorial morphology. Ph.D. dissertation, Rijksuniversiteit te Groningen.

Manaster-Ramer, Alexis. To appear. Dutch as a formal language. L&P.

McCawley, James D. 1982. Parentheticals and discontinuous constituent structure. LingI 13.91-106.

Pollard, Carl J. 1984. Generalized phrase structure grammars, head grammars, and natural language. Ph.D. dissertation, Stanford Univ.

Pullum, Geoffrey K. 1982. Free word order and phrase structure rules. NELS 12.209-20.

Zwicky, Arnold M. 1985. Free word order in GPSG. Paper presented at Linguistic Institute summer meetings, Georgetown Univ. To appear in OSU WPL.