SUBCATEGORIZATION LISTS IN IMMEDIATE DOMINANCE RULES*

Thomas E. Hukari

Department of Linguistics
University of Victoria

1. INTRODUCTION.

This paper explores the use of subcategorization lists or 'stacks' in a version of Generalized Phrase Structure Grammar (GPSG) which is otherwise conceptually very close to Gazdar, Klein, Pullum and Sag (1985, hereafter, GKPS). While the use of list structures in extensions of GPSG is not new (cf. Pollard, 1984; Pollard and Sag, 1987), the present paper develops list-like feature structures in the context of Gazdar et al. (1988), where categories are set-theoretic objects—partial functions—which are constrained in various ways by statements in a formal constraint language L_C.

Before turning to the specifics of the proposal for incorporating list structures into GPSG, I sketch out analyses of two syntactic phenomena which receive more perspicuous treatments in a grammar with subcategorization lists: subject-(auxiliary) verb inversion and lexically determined case assignment. This discussion not only gives some justification for the use of list structures—a point which the reader may explore more fully in Pollard and Sag, 1987—but it provides a context for an informal exposition of the GPSG extension set out in more detail in subsequent sections.

1.1. Subject-Verb Inversion.

The English subject-verb inversion structures proposed by GKPS are not consistent with their formulation of agreement involving control and the Control Agreement Principle (see Hukari and Levine, 1987). They propose the following inversion metarule.

(1) **Subject Auxiliary Inversion Metarule.**
V^2[-SUBJ] $\rightarrow W \Rightarrow V^2$[+INV, +SUBJ] $\rightarrow W$, NP

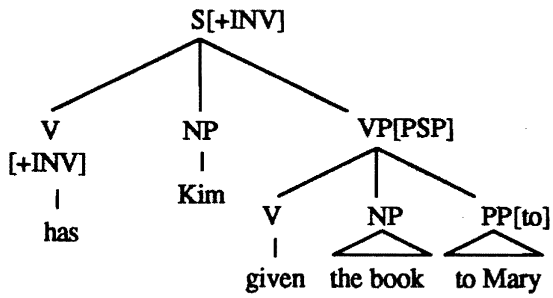
This 'liberates' the constituents of VP into S by expanding the set of licensing immediate dominance (ID) rules[1] to include S-expansion counterparts, such as (3), to certain VP rules, such as (2)

(2) VP $\rightarrow$ H[n], VP[PSP]

(3) S[+INV] $\rightarrow$ H[n], NP, VP[PSP]

yielding sentences like (4).[2] However nothing in the control mechanisms postulated in GKPS establishes a link between the subject and the auxiliary verb. Their Control Agreement Principle does not apply between the subject (*Kim*) and lexical V (*has*), nor does it link the subject with its VP sister. Thus nothing in the feature instantiation system rules (5).

(4)



(5)*Have Kim given the book to Mary?

Hukari and Levine (1987) note this problem and propose an extension of GPSG to include liberation rules (cf. Zwicky, 1985, 1986), where the liberation operation takes two immediate dominance rules, one which expands a daughter of the other, and merges them into one rule by replacing the daughter with its daughters:

(6) a. $S_0 \rightarrow NP_1, VP_2$

b. $VP_2 \rightarrow V_3, VP[PSP]_4$

c. $S_0[+INV] \rightarrow NP_1, V[+INV]_3, VP[PSP]_4$ (= liberation of (a) and (b))

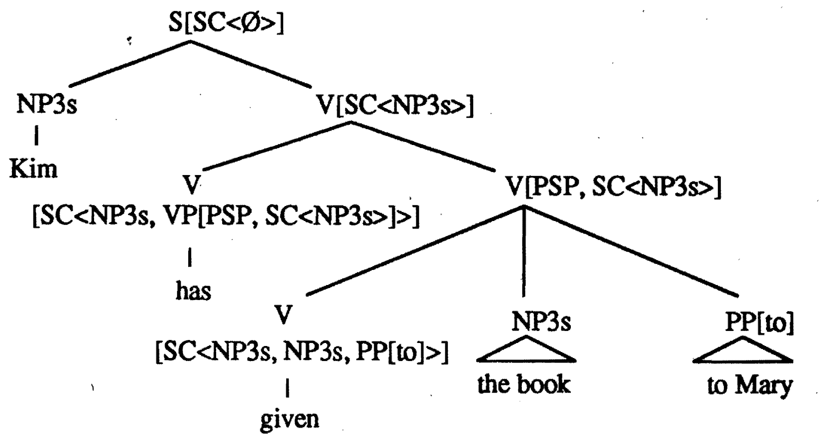
While this offers an account of subject-verb agreement in inverted sentences, it does so at the cost of introducing considerable complexity in the grammar. In fact, the resulting theory can be thought of as being multistratal. Since liberation is, in effect, structure-destroying, Hukari and Levine propose that it operates on fully instantiated immediate dominance rules. The feature instantiation principles apply to these sets of instantiated pre-liberation ID rules, such as (6) (a) and (b), where the control agreement principle links NP_1 and VP_2 in (a), and the Head Feature Convention passes the agreement feature specification down from VP_2 to V_3. Given a set of ID rules representing the structural/grammatical relations of an entire sentence, liberation maps this to a new set of rules in which liberation ID rules such as (c) replace original rules such as (a) and (b).

I consider here a very different approach to inversion, following in the spirit of Pollard (1985), and Pollard and Sag (1987), in which list-like feature structures are employed. These can be defined in a way which is compatible with the set-theoretic formulation of categories as (partial) functions in Gazdar, Klein, Pullum, and Sag (1985) and, more specifically, as in Gazdar et al. (1988), as discussed below. For the moment, subcategorization lists will be informally noted as a sequence of categories, where the rightmost is the most oblique and the leftmost, e.g., the subject, is the least oblique (cf. Dowty, 1982a and b; Pollard and Sag, 1987), so the subcategorization specification for *give* is roughly SC<NP, NP, PP[to]>, where the leftmost NP corresponds to the subject.[3] The basic ID rule for S-expansions is the following.

(7) $V[SC<\emptyset>] \rightarrow XP, H$

This licenses local trees in which the mother (i.e., S) is 'saturated', that is, its subcategorization list contains no categories, and the head daughter's list contains one category corresponding to the non-head daughter (XP), as in the topmost local tree of (8).

(8)



The correspondences between the head's list and the mother's, on one hand, and the head's list and the non-head daughters, on the other, are not determined directly by the immediate dominance rule, but by two feature instantiation principles whose first approximations are as follows.

(9) **List Condition.**
The mother's subcategorization list is a portion of the head daughter's: if $C_H[SC<C_1, ..., C_n>]$ then $C_0[SC<C_1, ..., C_i>]$, where $0 \leq i \leq n$.

(10) **Subcategorization Condition.**
Given a licensing ID rule $C_0[SC<...>] \rightarrow C_H, C_1, ..., C_n$, each C_i daughter in the tree, $1 \leq i \leq n$, corresponds to one category C'_i in the portion of the head's subcategorization list which is not passed up to the mother's list, and vice versa.

Subject-verb agreement follows as a consequence of the subcategorization condition and the list condition as they apply to the topmost local tree in (8). Since the mother's subcategorization list is empty,[4] the subcategorization condition says that all categories in the heads must 'cancel' by matching non-head daughters in the tree. But the licensing ID rule mentions only one such daughter, XP, thus the head's list must be of length one, and the category in it must correspond to a third person singular NP in order to match the non-head daughter in the tree.[5]

The immediate dominance rule licensing the next lower local tree in (8) is the following general schema, or more specifically, the version of it in (12).

(11) V[SC<XP>] → H[+LEX], XP*
(12) V[SC<XP>] → H[+LEX], XP

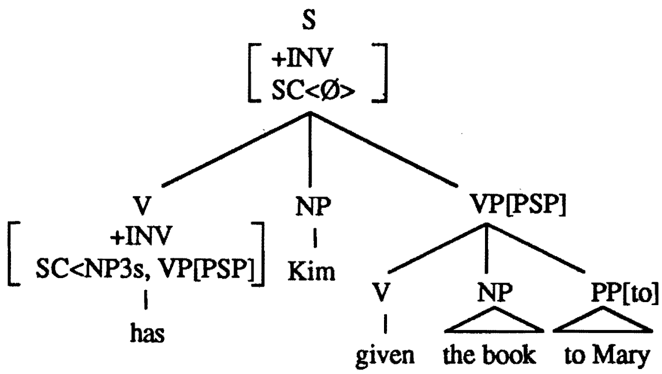
Again, the list condition and the subcategorization condition apply. In this case, the mother's list must contain one category, as determined by the ID rule. By the list condition, this must correspond to the left-most category in the head's list. By the subcategorization condition, the remaining categories in the head's list must cancel with the non-head daughters in the tree. Since the head's list is of length two—roughly, <NP3s, VP>—just the last element in the list corresponds to a non-head daughter. The next lower local tree is headed by *given*, whose subcategorization list is roughly <NP, NP, PP[to]>, and the relevant case of (11) is the following.

(13)V[SC<XP>] → H[+LEX], XP, XP

The inversion ID rule stipulates that the subcategorization list of the mother must be empty. The principles above ensure that all categories in the head verb's subcategorization list which are not also in the mother's are matched by constituents in the tree, so the entire list must cancel.

(14) V[SC<Ø>, +INV] → H[+LEX], XP*

(15)



And subject-verb agreement still follows from the subcategorization condition, since the subject NP (*Kim*) must match the left-most category in the head's list.

1.2 Lexically Governed Case Assignment.

A second problem area for GPSG is the assignment of case to subjects and objects. GKPS force nominative case in subjects of tensed clauses via the agreement system. This is accomplished by means of a feature co-occurrence restriction saying that tensed VP must agree with a nominative subject. But if this approach is extended to lexically determined subject case assignment in languages like Icelandic (cf. Andrews, 1982; Zaenen, Maling and Thráinsson, 1985), we arrive at a strange asymmetry. Lexically conditioned subject case assignment will presumably involve inherent specification for the agreement feature AGR, so Icelandic *hvolfdi* 'capsized' is specified [AGR: NP[CASE: DAT]], as in

(16) Bátnum(D) hvolfdi. 'The boat capsized'. [Andrews, example 50d]

But lexically conditioned object case assignment would presumably be imposed in immediate dominance rules such as the following for dative case assignment to objects of such verbs as Icelandic *björguðu* 'rescued'.

(17) VP → H[n], NP[CASE: DAT]

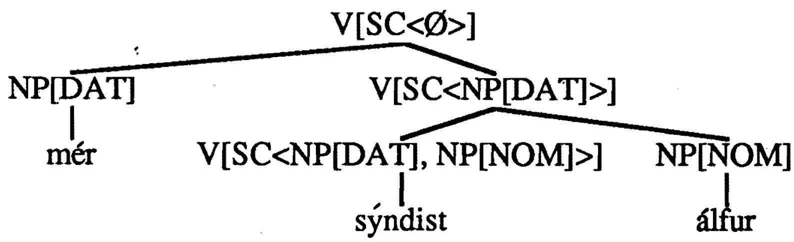
But a single account of lexically governed case assignment is readily available in a grammar which employs subcategorization lists (cf. Pollard and Sag,1987). We can say, for example, that a verb which assigns dative case to its subject and nominative case to its object has a list of the form

(18) SUBCAT<NP[CASE: DAT], NP[CASE: NOM]>

If we assume that immediate dominance rules along the lines of (7) and (11) extend to basic sentences in Icelandic, the subcategorization and list conditions noted above guarantee that the appropriately case-marked noun phrases will be selected, as in

(19) Mér(D) sýndist álfur(N). 'I thought I saw an elf.' [Andrews, example 50h]
me thought-saw elf

(20)



Furthermore, though a treatment of Icelandic passives is well beyond the scope of the present paper, an approach which employs subcategorization lists seems promising. For example, active verbs which select dative objects in Icelandic have corresponding passives selecting dative subjects (see Zaenen, Maling and Thráinsson, 1985).

(21) Þeir björguðu stúlkunni(D). they rescued the-girl [Andrews, example 56a]
(22) Stúlkunni(D) var bjargað. the-girl was rescued [Andrews, example 58a]

Assume for the sake of argument that passive is a lexical rule which eliminates the least oblique (left-most) argument:

(23) $[\ \mathrm{V},\ \mathrm{SC}{<}C_1, C_2, \ldots, C_n{>}] \rightarrow [\mathrm{V},\ +\mathrm{PASS},\ \mathrm{SC}{<}C_2, \ldots, C_n{>}]$

It follows automatically that—barring any language-specific constraints to the contrary—whenever an active verb governs a special object case, this property should be transferred to passive subjects, as in (22).

2. THE STRUCTURE OF LISTS.

Lists can be constructed in set-theoretic terms roughly along the lines discussed for indexed grammars in Gazdar et al. (1988). While simple sequence notation was used in the previous discussion, this form of notation, as in (24)

(24) $\mathrm{SUBCAT}{<}C_1, C_2, C_3{>}$

stands for the feature structure

(25) $$\left[\mathrm{SUBCAT}: \left\{\begin{array}{l} \mathrm{ARG}: C_3 \\ \mathrm{SUBCAT}: \left\{\begin{array}{l} \mathrm{ARG}: C_2 \\ \mathrm{SUBCAT}: \left[\begin{array}{l} \mathrm{ARG}: C_1 \\ \mathrm{SUBCAT}: \mathrm{NIL} \end{array}\right] \end{array}\right\} \end{array}\right\}\right].$$

where we can think of C_3 as being the most oblique complement and C_1 the least. The value of SUBCAT is a set, a (highly specialized) category whose content is restricted to a list-like structure by a feature co-occurrence restriction (see footnote 14) which has the effect of guaranteeing that the category value of SUBCAT contains either specifications for ARG(gument), a category-valued feature, and SUBCAT and nothing else, or just the element NIL (which can be taken as an abbreviation of {+NIL}).

If lists are treated as category-valued features rather than some entirely different feature structure, this makes that the constraint language of Gazdar et al. (1988) available and, furthermore, the definitions of extension (subsumption) and unification found in GKPS hold without modification for categories containing subcategorization lists. Suppose, for example, we use the following definition of extension, where type-0 features take atomic values (e.g., +, -, 1, 2, 3) and type-1 features take category values.

(26) **Extension.** $A \sqsubseteq B$ (B extends A) iff
a. if $\tau(f) = 0$, then if $A(f)$ is defined, then $A(f) = B(f)$,
b. if $\tau(f) = 1$, then if $A(f)$ is defined, then $A(f) \sqsubseteq B(f)$.

It is easy to see that C[SC<C_1, ..., C_n>] extends C[SC<C'_1, ..., C'_n>] just in case C_i extends C'_i for each i, $1 \leq i \leq n$. Further, given that the list feature structure always terminates in the singleton set {<+, NIL>}, it turns out that neither of two categories containing list-valued features can extend the other unless the lists are of exactly the same length. For example, suppose two categories differ only in their subcategorization lists, which are as follows:

(27) SC<C_1, C_2>:

$$\left[\text{SC: }\left\{\begin{array}{l}\text{ARG: } C_2\\ \text{SC: }\left\{\begin{array}{l}\text{ARG: } C_1\\ \text{SC: \{NIL: +\}}\end{array}\right\}\end{array}\right\}\right]$$

(28) SC<C_1, C_2, C_2>:

$$\left[\text{SC: }\left\{\begin{array}{l}\text{ARG: } C_3\\ \text{SC: }\left\{\begin{array}{l}\text{ARG: } C_2\\ \text{SC: }\left\{\begin{array}{l}\text{ARG: } C_1\\ \text{SC: \{NIL: +\}}\end{array}\right\}\end{array}\right\}\end{array}\right\}\right]$$

Neither will extend the other, since the value of SUBCAT in latter does not extend the value in the former (nor, of course, vice versa). That is, the category {ARG: C_3, SC: ...} clearly does not extend the category {ARG: C_2, SC: ...}, unless possibly $C_3 = C_2$. Suppose for the sake of argument that $C_3 = C_2 = C_1$. Even so, the value of SC in (28) does not extend that in (27) and this is because the value of the next-to-lowest token of SUBCAT in (28)—{ARG: C_1, SC: {NIL: +}}—would have to extend the lowest token of SUBCAT in (27)—SC: {NIL: +}—and it does not. This property of lists makes it possible to write immediate dominance rules in which the length of the subcategorization list can be specified. So, for example, the 'VP' rule (11) in 1.1 above

(11) V[SC<XP>] $\rightarrow$ H[+LEX], XP*

says that a verbal category with a subcategorization list of length one dominates a lexical verbal category and any number of phrases XP, where the category labelling the mother node in the local tree must be an extension of the left-hand side category in the rule (hence the restriction on the length of its subcategorization list).

3. IMMEDIATE DOMINANCE RULES.

Consider the following two types of immediate dominance rule.

(29) $C_0 \rightarrow H, C_1, \ldots, C_n$
(30) $C_0[SC<C_1, \ldots, C_n>] \rightarrow H, C^*$

These follow the ID/LP format of GKPS, where ID rules license hierarchical order but not linear order. Rules of the first type are like those in GKPS: the categories on the right-hand side constitute a multiset; there is a one-to-one correspondence between these and the daughters, such that each daughter extends exactly one; and the mother in the tree extends C_0. Rules of the second type—actually a rule schema—involve cancellation or 'off-loading' of the head's subcategorization list.[6] Both species of rule can be subsumed under the definitions in GKPS, provided feature instantiation principles regulate the relationship between the lists in the head and the mother, on one hand, and the relationship between the head's list and the non-head sisters in the tree, on the other. So the only additions we need in order to augment GPSG with a list-valued subcategorization feature are the List Condition and the Subcategorization Condition (see section 1.1.), which are given below in their full form.

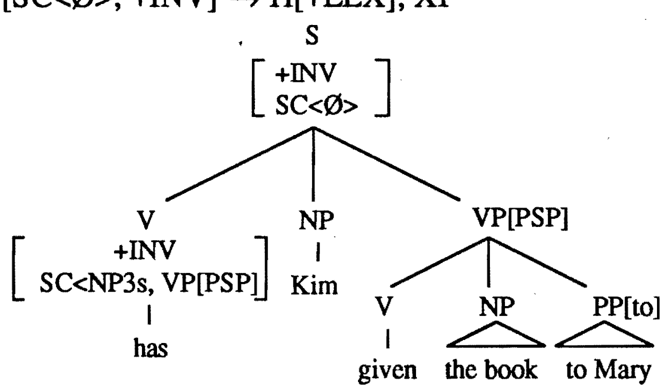
But certain notational conventions require explication before we turn to the constraints.[7] The relationship between an ID rule r and a tree is described in GKPS as a projection function ϕ, where categories as node-labels in a tree extend corresponding categories in the rule. Following their notation, I write $\phi(C_i)$ to denote the projection in the tree of C_i in the ID rule (so $C_i \sqsubseteq \phi(C_i)$). Expressions in square brackets are statements in the category constraint language L_c of Gazdar et al. (1988). Given a constraint $[\psi]$, $[\psi](C)$ is to be interpreted as meaning '$[\psi]$ is true of category C'. And a constraint $[\Diamond\psi]$ 'ψ is possible' is true of a category just in case ψ is true at some level of inclusion (but see below a necessary revision of the semantics of the modal operators).

(31) **List Condition.** In a rule $r = C_0 \rightarrow C_H, C_1, \ldots, C_n$ and a tree $\phi(r)$, if SUBCAT is specified in C_0,[8] then the subcategorization list in the mother $\phi(C_0)$ is contained in the value of SUBCAT in the head $\phi(C_H)$ at some level of inclusion:[9]

$$[SC](C_0) \supset [SC: \Diamond[\phi(C_0)(SC)]](\phi(C_H))$$

An example of this is the inversion rule from 1.1 above.

(14) $V[SC<\emptyset>, +INV] \rightarrow H[+LEX], XP^*$
(15)



Since C_0 in the licensing rule contains SUBCAT, the List condition says that the head's list in the tree must contain the mother's list at some level of inclusion. The two lists are as follows:

(32) [SC: {NIL: +}]

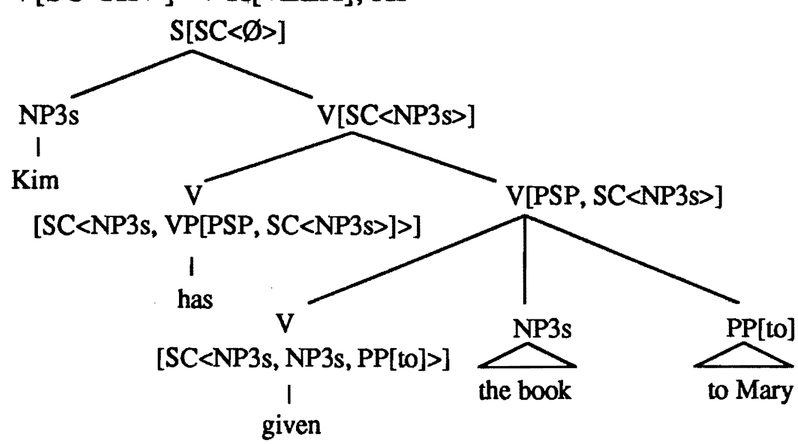
(33) $$\left[\text{SC: }\left\{\begin{array}{l}\text{ARG: VP[PSP]}\\ \text{SC: }\left\{\begin{array}{l}\text{ARG: NP3s}\\ \text{SC: \{NIL: +\}}\end{array}\right\}\end{array}\right\}\right]$$

And the value of SUBCAT in (32) is contained at some level of inclusion—the deepest—in the value of SUBCAT in (33).

A second example is the VP-expansion rule which applies in the uninverted counterpart to the sentence above.

(11) V[SC<XP>] → H[+LEX], XP*

(8)



In the local tree headed by *has*, the subcategorization lists for the mother and the head are as follows.

(34) $$\left[\text{SC: }\left\{\begin{array}{l}\text{ARG: NP3s}\\ \text{SC: \{NIL: +\}}\end{array}\right\}\right]$$

(35) $$\left[\text{SC: }\left\{\begin{array}{l}\text{ARG: V[PSP, SC<NP3s>]}\\ \text{SC: }\left\{\begin{array}{l}\text{ARG: NP3s}\\ \text{SC: \{NIL: +\}}\end{array}\right\}\end{array}\right\}\right]$$

And the mother's list is contained at some level of inclusion in the head's.

The Subcategorization Condition stipulates the relationship between the head's subcategorization list and the non-head sisters in the tree.

(36) **Subcategorization Condition**. In a rule $r = C_0 \rightarrow C_H, C_1, \ldots, C_n$ and a tree $\phi(r)$, if $[SC](C_0)$, then for all i, $1 \leq i \leq n$,

a. $\phi(C_i) = C'_i$,
b. $[SC: \Diamond[[ARG: C'_i] \;\&\; [SC: v_i]]](\phi(C_H))$, and
c. $[SC: \neg\Diamond[[ARG: C'_i] \;\&\; [SC: v_i]]](\phi(C_0))$.

That is, there is (a) a one-to-one correspondence of identity between each non-head daughter $\phi(C_i)$ and each C'_i which is (b) at some level of inclusion in the head's subcategorization list and (c) not in the mother's.[10] Consider again the uninverted sentence in (8). In the second local tree the subcategorization list of the head (*has*) is as in (35) and thus it contains—at one level of inclusion or another—two structures satisfying (b) in (36):

(37)
i. [[ARG: V[PSP, SC<NP3s>] & [SC: {ARG: NP3s, SC: {NIL: +}}]]
ii. [[ARG: NP3s] & [SC: {NIL: +}]]

But the second (ii) also appears in the subcategorization list of the mother, so only the first (i) is relevant, and the Subcategorization Constraint stipulates that the category-value V[PSP, SC<NP3s>] must correspond to a sister-constituent of the head.The inversion structure is similar, except the mother's subcategorization list is 'empty', so the values of ARG in both (i) and (ii)—V[PSP, SC<NP3s>] and [NP3s]—must correspond to sisters of the head in the tree.

4. THE MODAL OPERATORS.

There is a problem in formulating modal statements about the composition of subcategorization lists using the language of category constraint L_c in Gazdar et al. (1988). In particular, we want to be able to state constraints on the composition of subcategorization lists which pertain to what we intuitively think of as categories in the list, but not to values deeply embedded in those categories. For example, given a constraint

(38) [SC: $\Diamond$[NIL]]

it turns out that we want this to be interpreted in such a way that the evaluation stays on the recursive SUBCAT path, that a modal constraint is not satisfied (or falsified) by, say, looking deeply into a particular category on the list.[11] In other words, finding a specification for NIL deeply inside PP in SUBCAT< NP, PP, VP> should not satisfy the constraint, while terminating the list in NIL (see (25)) should. (See also the discussion of (41) below.)

The definition of the semantics of the modal necessity operator '$\Box$' in Gazdar et al. (1988) is as follows (where F^1 are the category-valued features and '$\Delta(C)$' denotes the domain of C, i.e., the features which are specified in C):

(39) A category α satisfies a constraint of the form '$\Box\phi$':

$\|\Box\phi\|_{\Sigma,\alpha} = 1$ just in case

(i) $\|\phi\|_{\Sigma,\alpha} = 1$, and

(ii) for all $f \in F^1 \cap \Delta(\alpha)$, $\|\Box\phi\|_{\Sigma,\alpha(f)} = 1$

I revise the semantics of the modal operator here so that the evaluations take into account accessibility relations between features. Atomic-valued features are type 0 (belonging to the set F^0), category-valued features are type 1 (they are in F^1), and list-valued features are type 2 (in F^2).[12] Features of a given type—for types 1 or 2—are accessible to each other, meaning that they can see into each other's values. Type 2 features may be accessible to type 1 features (I leave the matter open), but type 1 features are not accessible to type 2 features. The accessibility relation R is the set of ordered pairs <f, f'> such that the second is accessible to the first. And the definition of the modal operator invokes *R* as follows:[13]

(40) A category α satisfies a constraint of the form '$\Box\phi$':

$||\Box\phi||_{\Sigma,\alpha} = 1$ just in case
(i) $||\phi||_{\Sigma,\alpha} = 1$, and
(ii) $||\Box\phi||_{\Sigma,\alpha(f)} = 1$ for all f, f' such that
a. $f \notin F0$
b. if $\alpha = \beta(f')$, then $f \in R(f')$, and
c. $f \in \Delta(\alpha)$.

Consider an example step-by-step. The FCR—which is an approximation of the restriction on list-valued features—says that if a category has a specification for SUBCAT, then its value must have, at all levels of inclusion, specifications for SUBCAT and ARG, or for NIL.[14]

(41) FCR: $[SC] \rightarrow [SC: \Box[[SC \ \& \ ARG] \vee NIL]]$

Let us confine ourselves to the case where $||SC||_{\Sigma,\alpha} = 1$, so the consequent must be true. Then we need one further semantic rule from Gazdar et al. (1988):

(42) $||f : \phi||_{\Sigma,\alpha} = 1$ just in case $||\phi||_{\Sigma,\alpha(f)} = 1$

Consider the evaluation of the following feature structure with respect to (41).

(43)
$$\left[SC^0\!: \left[\begin{array}{l} ARG^1\!: VP \\ SC^1\!: \left[\begin{array}{l} ARG^2\!: PP \\ SC^2\!: \left[\begin{array}{l} ARG^3\!: NP \\ SC^3\!: [NIL: +] \end{array} \right] \end{array} \right] \end{array} \right] \right]$$

(44) $||SC: \Box[[SC \ \& \ ARG] \vee NIL]||_{\Sigma,\alpha} = 1$ iff

a. $||\Box[[SC \ \& \ ARG] \vee NIL]||_{\Sigma,\alpha(SC^0)} = 1$[by (42)], iff

b. $||[[SC \ \& \ ARG] \vee NIL]||_{\Sigma,\alpha(SC^0)} = 1$ [which it does], and

c. $||\Box[[SC \ \& \ ARG] \vee NIL]||_{\Sigma,\alpha(SC^0)(f)} = 1$ for all f, $f \in R(SC)$ and $f \in \Delta(\alpha(SC^0))$ [f = SC^1 only], iff

e. $||[SC \ \& \ ARG] \vee NIL||_{\Sigma,\alpha(SC^0)(SC^1)} = 1$ [which it does], and

f. $||\Box[[SC \ \& \ ARG] \vee NIL]||_{\Sigma,\alpha(SC^0)(SC^1)} = 1$ for all f such that $f \in R(SC)$ and f $\Delta(\alpha(SC^0)(SC^1))$ [f = SC^2 only], iff

g. $||[SC \ \& \ ARG] \vee NIL||_{\Sigma,\alpha(SC^0)(SC^1)(SC^2)} = 1$ [which it does], and

h. $||\Box[[SC \ \& \ ARG] \vee NIL]||_{\Sigma,\alpha(SC^0)(SC^1)(SC^2)(SC^3)} = 1$ for all f such that $f \in R(SC)$ and f $\Delta(\alpha(SC^0)(SC^1)(SC^2))$ [f = SC^3 only], iff

i. $||[SC \ \& \ ARG] \vee NIL||_{\Sigma,\alpha(SC^0)(SC^1)(SC^2)(SC^3)} = 1$[which it does], and

j. $\neg\exists f[f \in R(SC) \ \& \ f \in \Delta\alpha(SC^0)(SC^1)(SC^2)(SC^3)$.

Clearly the constraint will not be satisfied unless at *every* relevant level of inclusion the feature SUBCAT and ARG, or the feature NIL occurs, which is true in (43).

Suppose from SC^0 we were allowed to go into ARG^1 and evaluate

(45) $\|\Box[[SC \,\&\, ARG] \vee NIL]\|_{\Sigma,\alpha(SC^0)(ARG^1)}$

While the value of AGR^0, which is VP, will actually contain a specification for SUBCAT, it will not contain one for ARG. That is, the VP value of ARG^0 might be realized as something along the following lines:

(46)
$$\begin{bmatrix} \text{-N} \\ \text{+V} \\ \text{VFORM: INF} \\ \text{SC: } \begin{bmatrix} \text{ARG: NP} \\ \text{SC: NIL} \end{bmatrix} \end{bmatrix}$$

Thus if the value of ARG^0 were accessible to SUBCAT, the structure would fail the constraint. But, given the constraint on accessibility, the evaluation does not involve ARG since <SC, ARG> $\notin R$. Therefore the evaluation proceeds down through successively more deeply embedded values of SUBCAT, never evaluating the value of ARG, and the constraint is satisfied.

The modal possibility operator '$\Diamond$' is analogous. While we can think of it simply as the dual of the necessity operator—$\Diamond\phi =_{def} \neg\Box\neg\phi$—its semantics can be defined independently as follows:

(47) A category α satisfies a constraint of the form '$\Diamond\phi$':

$\|\Diamond\phi\|_{\Sigma,\alpha} = 1$ just in case

(i) $\|\phi\|_{\Sigma,\alpha} = 1$, or

(ii) $\|\Diamond\phi\|_{\Sigma,\alpha(f)} = 1$ for some f, f' such that

a. $f \notin F^0$

b. if $\alpha = \beta(f')$, then $f \in R(f')$, and

c. $f \in \Delta(\alpha)$.

5. VARIABLES.

As a final, minor point, consider the use of variables for partial representations of lists in the list notation.

(48) V[SC<XP>] $\rightarrow$ H[SC<XP,VP[INF], *W*>]; X*

If such partial representation is desirable, then *W* must be provided with an interpretation. Clearly we want it to correspond to the upper portion ('...') of the feature structure:

(49)
$$\left[\text{SUBCAT: }\left\{ \ldots \left\{\text{SUBCAT: }\begin{Bmatrix}\text{ARG: V[INF]} \\ \text{SUBCAT: }\begin{Bmatrix}\text{ARG: XP} \\ \text{SUBCAT: NIL}\end{Bmatrix}\end{Bmatrix}\right\} \ldots \right\}\right]$$

A perfectly straightforward interpretation exists for the *W*-variable notation, namely, as the modal operator of possibility '$\Diamond$' in statements in the category constraint language L_c, as in the following examples.

(50) a. $SC<C, W> = [SC: \Diamond[ARG: C \;\&\; [SC: NIL]]$
b. $SC<W, C> = [SC: [ARG: C]]$[15]
c. $SC<W, C, W> = [SC: \Diamond[ARG: C]]$
d. $SC<W, C_1, W, C_2> = [SC: [ARG: C_2 \;\&\; \Diamond[ARG: C_1]]$
e. $SC<C_1, W, C_2> = [SC: [[ARG: C_2] \;\&\; \Diamond[[ARG: C_1] \;\&\; [SC: NIL]]]]$

Leaving aside the matter of whether such partial representation of lists is needed in immediate dominance rules, consider the following approximation of a constraint on reflexives, inspired by work in progress by Carl Pollard and Ivan Sag.

(51) **Reflexive Rule.**
$SUBCAT<W, XP, W, XP[RE: \alpha], W> \supset SUBCAT<W, XP[\alpha], W, XP[RE: \alpha], W>$

That is, if the subcategorization list contains an anaphor (a category specified for RE) in non-final (non-subject) position, then it agrees with a less oblique category in the list (i.e., a more deeply embedded category). This can be stated in L_c as follows, where subscripted numbers are provided on some brackets for the reader.

(52) **Reflexive Rule.** For any category C,
$[_1[_2SC: [_3\Diamond[_4ARG: [RE: \alpha] \;\&\; [SC: \Diamond[ARG]]_4]_3]_2] \supset$
$[_5SC: \Diamond[_6[ARG: \alpha] \;\&\; [SC: \Diamond[ARG: [RE: \alpha]]]_6]_5]_1]$

A number of considerations arise in the treatment of reflexivization, and, while these are beyond the scope of this paper, I will at least mention them. One issue is whether a constraint along these lines should pertain just to reflexive NPs (i.e., anaphors) or also to constituents containing, at some depth, a reflexive pronoun. This, in turn, raises the matter of whether or not RE is a foot feature. Also related to these matters, is the status of anaphors when there is no higher element on the subcategorization list, as in the following, under the assumption that *for himself to have done better* and *a picture of herself* are constituents corresponding to the 'saturated' categories V[SC<Ø>] and N[SC<Ø>].

(53) Kim would have preferred for himself to have done better.
(54) Lee saw a picture of herself in the newspaper.

The constraint proposed above exempts cases where the anaphor is highest in the subcategorization list. If, however, RE is a foot feature, then the saturated categories will be specified for RE, and the constraint will apply in the matrix clause. Finally, I have not made explicit the feature content of α in the rule, which may or may not involve a syntactic binding feature (i.e., an index).

NOTES

* This project was supported in part by grant 410-88-0435 from the Social Sciences and Humanities Research Council of Canada and by a University of Victoria faculty research grant. I

also wish to thank Robert Levine and the editors of the Working Papers in Linguistics—Andrea Giles and Michael McGovern—for their comments. Needless to say, the responsibility for any remaining faults is mine alone.

[1]Immediate dominance rules, like phrase structure rules, state structural relations (sister-of, daughter-of) but, unlike phrase structure rules, they say nothing about the linear (sequential) order of daughters. Insofar as daughters are linearly ordered, their order is determined by linear precedence rules. See GKPS; Gazdar and Pullum, 1981; and Pullum, 1982.

[2]As INV is a head feature, it will appear on the lexical head, and a feature co-occurrence restriction prevents -AUX verbs from containing a +INV specification, thus the metarule yields useful ID rules only in constructions in which the head verb is an auxiliary verb. This use of INV carries over to the other analyses discussed below.

[3]This departs from the notational convention in Pollard and Sag (1987), where the least oblique complement is rightmost (e.g., *give* is SC<PP[to], NP, NP >).

[4]SC<Ø> does not denote an empty list; rather, the list contains no categories for cancellation. As discussed below, <Ø> denotes a value containing a feature specification [+NIL], not the empty set. The fact that list structures terminate in [+NIL] guarantees that whenever a list is mentioned in an ID rule, the matching category in the tree will have a list of exactly the same length.

[5]I assume a somewhat more complex account of agreement, though this has no bearing on the matters at hand. That is, I assume there is an agreement feature AGR, whose value must match the subject (leftmost category) in the subcategorization list. Matching here may be confined to a small set of agreement features, such as person, number, and possibly an index.

[6]See Evans (1987) for an interesting alternative formulation of the relationship between ID rules and trees, where rules contain statements in L_c and the category labels in trees are models of these.

[7]SUBCAT is a HEAD feature and the List Condition should over-ride the Head Feature Convention, but when SUBCAT is not mentioned in an ID rule, the HFC applies.

[8] We might wish to also stipulate that SUBCAT is not mentioned in the head in the ID rule: SUBCAT $\notin \Delta(C_H)$. I leave this matter open. See the discussion of (43) in section 5.

[9]Strictly speaking, a constraint of the form '$\Diamond[\phi(C_0)(SC)]$' is not a statement in L_c, where '$\phi(C_0)(SC)$' denotes the category-value of SUBCAT in $\phi(C_0)$. But this can be written as a statement in L_c augmented with variables: $[[SC](C_0)\ \&\ [SC{:}\ \alpha](\phi(C_0))] \supset [SC{:}\ \Diamond\alpha](\phi(C_H))$.

[10]Alternatively, this correspondence involves extension rather than identity: $C'_i \sqsubseteq \phi(C_i)$. Note the crucial use of v_i here. Two tokens of the same category may occur at different levels of inclusion in the list, and these are distinguished here by the fact that at different levels, the value of v_i in $\{<ARG{:}\ C_i>, <SC, v_i>\}$ will be different, given the geometry, so to speak, of the feature structure.

[11]This is given strictly for exemplification; (38) is not offered here as a constraint.

[12]Strictly speaking, list-valued features are also category-valued here, since lists are a highly specialized sort of category (one which never labels nodes in trees). But I assume that type-2 features are not type-1 and conversely.

[13]Recall that the possibility operator '$\Diamond$' is the dual, and can be defined derivatively as '$\neg\Box\neg$'.

[14]This of course will not suffice to characterize the content of the value of SUBCAT, since no other features should occur at any level. Perhaps we could say that for all features f and g, f $\neq$ SC, ARG, g $\neq$ NIL

FCR: [[SC] $\rightarrow$ [SC: $\Box$[[SC & ARG& $\neg$f] $\vee$ [NIL & $\neg$g]]].

[15]Since nothing is specified in (b) beyond the stipulation that the top of the list contains C, a modal operator is not needed here. For the same reason, (c) contains only one modal operator.

REFERENCES

Andrews, Avery D. (1982). The representation of case in modern Icelandic. In Joan Bresnan (ed.), *The mental representation of grammatical relations* (pp. 427-503). Cambridge, MA: MIT Press.

Dowty, David R. (1982a). Grammatical relations and Montague Grammar. In Pauline Jacobson and Geoffrey Pullum (eds.), *The nature of syntactic representation* (pp. 79-130). Dordrecht: Reidel.

Dowty, David R. (1982b). More on the categorial analysis of grammatical relations. In Annie Zaenen (ed.), *Subjects and other subjects* (pp. 115-153). Bloomington: Indiana University Linguistics Club.

Evans, Roger. (1987). Towards a formal specification for defaults in GPSG. *Cognitive Science Research Paper* no. CSRP 084, University of Sussex.

Gazdar, Gerald, Ewan Klein, Geoffrey K. Pullum, and Ivan Sag. (1985). *Generalized Phrase Structure Grammar*. Cambridge, MA: Harvard University Press.

Gazdar, Gerald and Geoffrey K. Pullum. (1981). Subcategorization, constituent order, and the notion "head". In M. Moortgat, H. v.d. Hulst, and T. Hoekstra (eds.), *The Scope of Lexical Rules* (pp. 107-123). Dordrecht: Foris.

Gazdar, Gerald, Geoffrey K. Pullum, Ewan Klein, Robert Carpenter, Thomas E. Hukari, and Robert D. Levine. (1988). Category structures. *Computational Linguistics 14*, 1-19.

Hukari, Thomas E. and Robert D. Levine. (1987). *Generalized Phrase Structure Grammar* : A review monograph. To appear in *Linguistic Analysis*.

Hukari, Thomas E. and Robert D. Levine. (1988). Liberation and inversion in GPSG. In Hagit Borer (ed.), *Proceedings of the West Coast Conference on Formal Linguistics 7* (pp. 159-170). Stanford: Stanford Linguistics Association.

Pollard, Carl J. (1984). Generalized Phrase Structure Grammars, Head Grammars, and natural language. Ph.D. Dissertation, Stanford University.

Pollard, Carl J. (1985). Phrase structure grammar without metarules. In Jeffrey Goldberg, Susannah MacKaye, and Michael Wescoat (eds.), *Proceedings of the West Coast Conference on Formal Linguistics 4* (pp. 246-261). Stanford: Stanford Linguistics Association.

Pollard, Carl J. and Ivan Sag. (1987). *Information-based syntax and semantics, volume 1: Fundamentals*. CSLI Lecture Note Series no. 13. Chicago: University of Chicago Press.

Pullum, Geoffrey K. (1982). Free word order and phrase structure rules. In James Pustejovsky and Peter Sells (eds.), *Proceedings of the North Eastern Linguistic Society 12* (pp. 209-220). Amherst, MA: Graduate Linguistic Student Association, University of Massachusetts.

Zaenen, Annie, Joan Maling, and Höskuldur Thráinsson. (1985). Case and grammatical functions: The Icelandic passive. *Natural Language and Linguistic Theory 3*, 441-483.

Zwicky, Arnold M. (1985). Free word order in GPSG. Paper presented at the Linguistic Society of America summer meetings, Georgetown University.

Zwicky, Arnold M. (1986). Concatenation and liberation. In Anne Farley, Peter Farley and Karl-Erik McCullough (eds.), *Papers from the Regional Meeting of the Chicago Linguistic Society 22* (pp. 65-74). Chicago: Chicago Linguistic Society, University of Chicago.