Long-Distance Scrambling and Tree Adjoining Grammars[*]

Tilman Becker[†] Aravind K. Joshi, Owen Rambow
University of Pennsylvania, Department of Computer and Information Science, Philadelphia, PA 19104-6389
tilman@cs.uni-sb.de, {joshi,rambow}@linc.cis.upenn.edu

1 Introduction

Scrambling, both local and long-distance, has recently attracted considerable attention among linguists and computational linguists. In this paper, we will explore the adequacy of the Tree Adjoining Grammar (TAG) formalism for dealing with long-distance scrambling[1] in German. We will show that TAGs cannot capture the full range of constructions derived by scrambling. However, Multi-Component TAGs (MC-TAG), an extension of TAGs introduced earlier [Joshi 1987a, Weir 1988] and utilized for linguistic purposes (e.g. for extraposition [Kroch and Joshi 1986]), can indeed capture the full range of constructions derived by scrambling. We will also present an ID/LP variant of TAG to capture the same constructions, and then comment on the relationship between the two systems.

2 Some Linguistic Data

In German (and in many other SOV languages, such as Korean, Hindi and Japanese), a constituent of an embedded clause may be moved from that clause into the matrix clause. Consider, for example, sentences (1) in Figure 1. In German, the object of the embedded clause can be "moved"[2] to any position in the matrix clause, as in sentences (1b) and (1c).

A striking feature of scrambling is its freedom: there appear to be no systematic syntactic restrictions on the number of verbal arguments that undergo "movement," nor on the distances over which they may move[3]. Thus, any ordering of the arguments from all clauses is possible. To illustrate this freedom we will present two additional examples in which scrambling of a more complex nature occurs.

1. More than one constituent may undergo movement into higher clauses. The scrambled constituents need not retain their original relative order to each other after scrambling. In sentence (2b), two NPs are scrambled out of the embedded clause into the top-level clause.

2. Constituents may be moved over an unbounded number of clauses. In sentence (3b), NP *die Witwen* has been moved into its immediately dominating clause, while NP *der Opfer* has been moved from the most deeply embedded clause into the top-level clause, beyond the intermediate clause.

3 A TAG Analysis

The TAG formalism (for a recent introduction, see [Joshi 1987a]) is well suited for linguistic description because (1) it provides a larger domain of locality than a CFG or other augmented CFG-based formalisms such as HPSG or LFG, and (2) it allows factoring of recursion from the domain of dependencies. This extended domain of locality, provided by the elementary trees of TAG, allows us to "lexicalize" a TAG grammar: we can associate each tree in a grammar with a lexical item [Schabes et al 1988, Schabes 1990][4]. The tree will contain the lexical item, and all of its syntac-

[*] This work was partially supported by NSF grants DCR-84-10413, ARO grant DAAL03-89-0031, and DARPA-ONR grant N0014-85-K0018. We are indebted to Tony Kroch, Young-Suk Lee, Beatrice Santorini, Yves Schabes and David Weir for discussions and clarifications related to this paper.

[†] Now at University of Saarbrücken, Fachbereich Informatik, D-W6600 Saarbrücken.

[1] Intra-clausal scrambling and string-vacuous scrambling will not be discussed in this paper, since they do not pose any particular problem for the TAG formalism.

[2] We use "traces" only to indicate the unmarked order; we do not mean to imply any particular theory of movement. In fact, analyses have been proposed (going back to [Evers 1975]) based on a process of "verb cluster formation", which avoid inter-clausal movement altogether. However, since embedding is recursive, the verb clusters cannot all be listed in the lexicon. Hence, from a formal point of view, a lexical or morphological analysis of verb cluster formation poses exactly the same problems as scrambling interpreted as syntactic movement.

[3] Some verbs allow scrambling out of their complements more freely than others. It appears that all subject-control verbs and most object-control verbs governing the dative allow scrambling fairly freely, while scrambling with object-control verbs governing the accusative is more restricted (cf. [Bayer and Kornfilt 1989]). From a formal point of view, these restrictions are not relevant for the present argument.

[4] The associated lexical item is called the *anchor*, and is either the head or the functional head of the tree.

(1a)	...daß	ich$_i$	dem Kunden	[PRO$_i$	den Kühlschrank	zu reparieren]	versprochen	habe
	...that	I	the client (dat)		the refrigerator (acc)	to repair	promised	have

...that I have promised the client to repair the refrigerator

(1b)	...daß	ich$_i$	[den Kühlschrank]$_j$	dem Kunden	[PRO$_i$ t_j	zu reparieren]	versprochen	habe
	...that	I	the refrigerator (acc)	the client (dat)		to repair	promised	have

... that I have promised the client to repair the refrigerator

(1c)	...daß	[den Kühlschrank]$_j$	ich$_i$	dem Kunden	[PRO$_i$ t_j	zu reparieren]	versprochen	habe
	...that	the refrigerator (acc)	I	the client (dat)		to repair	promised	have

... that I have promised the client to repair the refrigerator

(2a)	...daß	der Detektiv$_i$	dem Klienten	[PRO$_i$	den Verdächtigen
	...that	the detective (nom)	the client (dat)		the suspect (acc)

des Verbrechens	zu überführen]	versprochen	hat
the crime (gen)	to indict	promised	has

...that the detective has promised the client to indict the suspect of the crime

(2b)	...daß	[des Verbrechens]$_k$	der Detektiv$_i$	[den Verdächtigen]$_j$
	...that	the crime (gen)	the detective (nom)	the suspect (acc)

dem Klienten	[PRO$_i$ t_j t_k	zu überführen]	versprochen	hat
the client (dat)		to indict	promised	has

...that the detective has promised the client to indict the suspect of the crime

(3a)	...daß	der Rat	dem Pfarrer	[die Witwen$_i$	[PRO$_i$	der Opfer
	...that	the council (nom)	the priest (dat)	the widows (acc)		the victims (gen)

gedenken]	zu lassen]	versprochen	hat
commemorate	to let	promised	have

...that the council has promised the priest to let the widows commemorate the victims

(3b)	...daß	[die Witwen]$_j$	[der Opfer]$_i$	[dem Pfarrer]$_k$
	...that	the widows (acc)	the victims (gen)	the priest (dat)

der Rat t_k	[t_j [PRO$_j$ t_i	gedenken]	zu lassen]	versprochen	hat
the council (nom)		to commemorate	let	promised	have

...that the council has promised the priest to let the widows commemorate the victims

Figure 1: Example Sentences

tic dependents. As has been shown previously, certain long-distance phenomena such as topicalization and wh-movement can be handled naturally within TAG [Kroch and Joshi 1985]. Here, "naturally" means that dependencies are stated within the larger domain of locality (the elementary tree), i.e., each clausal tree still contains a verb and all of its arguments. Thus, in a lexicalized TAG, the effects of long-distance movement are achieved by adjunction. The word order freedom possible in the context of unconstrained scrambling, however, eludes the scope of TAGs. In this section, we will informally argue this formal result.

By an argument very similar to Shieber's argument for Swiss German [Shieber 1985], it can be shown that the string language of scrambled High German is not a context-free language. However, the linguistic facts of German do not allow an extension of the argument: we cannot show that the string language is not a Tree Adjoining Language. From a linguistic perspective, the existence of *some* grammar that generates the string language of German scrambling is not in itself of much interest. For example, if we define a TAG grammar that generates the strings of scrambled German in which, however, some trees pair a verb with the arguments of some other verb, then we have not adequately described the linguistic facts. We are really only interested in linguistically motivated grammars, namely those that exploit the extended domain of locality and whose trees obey the constraint of containing a lexical item and all of its syntactic dependents (and nothing else). We will refer to such restrictions as "co-occurrence constraints". We can show that no TAG meeting the co-occurrence constraints can generate the sentences of German scrambling. We will argue this point in two complementary ways. First, we will consider the case of clauses with two overt nominal arguments. Then, we will consider the case of clauses with one overt nominal argument.

In the first case, the verb of the embedded clause subcategorizes for three NPs, one of which is an empty subject (PRO). There is no verb in German that subcategorizes for three NPs and an S, so in this case a recursively embedded structure is impossible, and we have only one level of embedding. We show that the language $\{\sigma(NP_1^1, NP_1^2, NP_2^1, NP_2^2)V_2V_1 \mid \sigma \text{ a permutation}\}$

cannot be generated by a TAG that contains only elementary trees obeying the co-occurrence restraints. A linguistically plausible set of two such trees is shown in Figure 2. Consider the string $NP_2^2 NP_1^1 NP_2^1 NP_1^2 V_2 V_1$, which corresponds to the ordering in sentence (2b). It can easily be seen that this string cannot be generated by a TAG of the specified sort: after an adjunction the yield of the adjoined tree is segmented into at most two sections, while the yield of both trees would need to be segmented into three sections in order to be composed into the desired string.

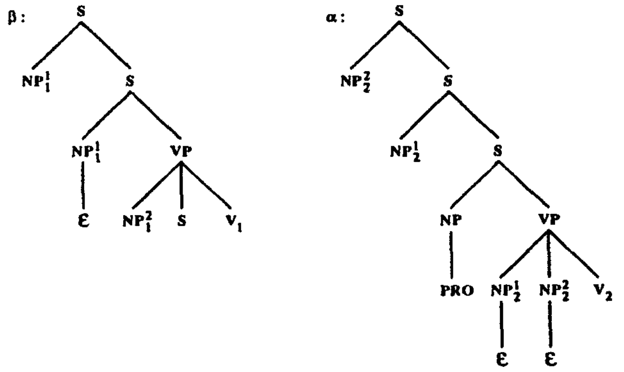

Figure 2: Initial trees with two verbal arguments

In the second case, the verbs of the embedded clauses subcategorize for two NPs, one of which is again an empty subject (PRO), and an S. We will argue that the language $\{\sigma(NP_1, \ldots, NP_k)V_k \ldots V_1 \mid k \in \mathsf{N} \text{ and } \sigma$ a permutation$\}$ cannot be generated by a TAG which obeys the co-occurrence constraints, i.e., whose elementary trees have only two (non-vacuous) terminal leaves, NP_i and V_i[5]. The idea in selecting this language is as follows: we keep the verbs at the end in the inverted order required by embedding, and then consider all possible permutations of the NPs. For $k \leq 4$ TAGs that generate the possible permutations can be constructed; for $k = 4$ the construction is not obvious, but we will not give the details here. However, for $k = 5$ it is impossible. Consider the string $w = NP_3 NP_1 NP_5 NP_2 NP_4 V_5 V_4 V_3 V_2 V_1$. For this string, it can be shown that it is impossible to construct a TAG which meets the co-occurrence constraints discussed above and that generates the string. The proof is fairly involved; for details, see [Becker and Rambow 1990].

In deciding whether scrambling as a linguistic phenomenon can adequately be described by a TAG or a TAG-equivalent formalism, it is crucial to decide whether or not sentences corresponding to the strings given above are indeed grammatical. In the case of the embedded two-argument clauses, examples are readily available, as in sentences (2a) and (2b). In the case of the embedded one-argument sentences, it is more difficult (but not impossible) to construct an adequate example because of the great depth of embedding. However, one might argue that there is a limit on the number of clauses over which a scrambled NP may move. If this were true, the number of resulting structures would be finite, so that they could be handled trivially by simple formalisms. Sentences (3a) and (3b) show scrambled NPs can move over two clauses, and we know of no evidence that convincingly establishes such a limit for higher numbers. The reluctance that some native speakers show to accept the more complex sentences is due mainly to processing difficulties, rather than to the ungrammaticality of the sentences. A similar phenomenon occurs when native speakers reject multiply center-embedded sentences as "ungrammatical".

In summary, long-distance scrambling provides linguistic evidence that shows that scrambling is beyond the formal generative capacity of TAGs. In the next two sections, we will investigate two ways of extending the TAG formalism in order to achieve the necessary power. In the first approach, we will relax the immediate dominance relation of the elementary trees. In the second approach, we will relax the linear precedence relations of the elementary trees. In both cases, our concern will be to preserve the key properties of TAGs, namely their extended domain of locality, and the factoring of recursion from dependencies.

4 A Multi-Component TAG (MC-TAG) Approach: Relaxed ID

One approach is to relax the ID (Immediate Dominance) relation within one elementary tree. Even in a standard TAG, the ID relation between a mother and a daughter node is not necessarily an *immediate* dominance relation because of the possibility of adjoining another tree at the daughter node. We propose to relax some of the ID relations of the auxiliary tree when it is adjoined. This can be seen as splitting up the auxiliary tree into parts, while still keeping a dominance constraint between the parts. As we will show, such a splitting of the elementary trees, interestingly, leads to a previously defined extension of TAGs: namely, that of Multi-Component TAGs (MC-TAG)

[5] Note that the indices are not actually part of the alphabet over which we have defined the language, which is simply $\{NP, V\}$. The indices only serve the purpose of indicating which terminals are supposed to be contributed by which tree, exploiting the co-occurrence constraints.

[Joshi 1987a, Weir 1988].[6]

As shown in Section 3, a TAG meeting the co-occurrence constraints cannot derive the string $NP_2^2 NP_1^1 NP_2^1 NP_1^2 V_2 V_1$. It is obvious that in order to get this variation from the trees in Figure 2, the yield of the adjoined tree β has to be broken up into three segments, which means that β has to be broken up into two parts that are then adjoined to different nodes of α. This is exactly what relaxation of the ID relation can achieve. If the tree β in Figure 2 is split at the interior S node, i.e. by relaxing the ID relation between the two S nodes, we can construct a pair of auxiliary trees as shown in Figure 3, where the dashed line indicates a dominance relation.

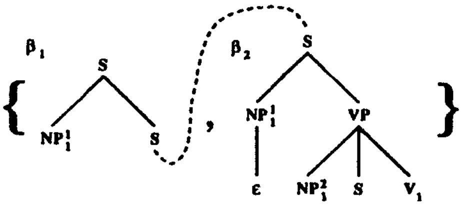

Figure 3: Splitting an elementary tree into two parts.

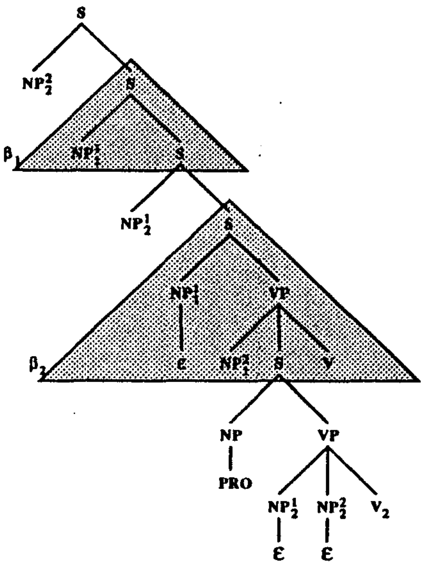

Figure 4: Adjunction of an MC-TAG tree set

Sets of trees are exactly what the MC-TAG formalism introduces. In an MC-TAG, instead of auxiliary trees being single trees we have auxiliary sets, a set containing more than one (but still a fixed number) of auxiliary trees. For details of the definition of MC-TAG see [Joshi 1987a, Weir 1988]. In an MC-TAG, adjunction is defined as the simultaneous adjunction of all trees in a set to different nodes. It is not possible to adjoin trees from the same set into each other. Thus, we can interpret TAGs with relaxed dominance as MC-TAGs, by identifying subtrees containing only ID relations with trees in an MC-TAG tree set. However, we need to introduce an additional constraint: the foot node of the first tree (β_1) in the tree set of Figure 3 has to dominate the root node of the second tree (β_2) after adjoining the tree set. This is indicated by the dashed link between the foot node of β_1 and the root node of β_2. For example, Figure 4 shows the effect of adjoining the tree set of Figure 3 into tree α of Figure 2, which yields the ordering of scrambled sentence (2b), $NP_2^2 NP_1^1 NP_2^1 NP_1^2 V_2 V_1$.[7]

In defining adjunction for MC-TAGs, an issue arises that is irrelevant in the case of simple TAGs: do we restrict adjunction in such a way that members of a tree set must be adjoined into the trees of an elementary tree set, or do we allow adjunction into derived tree sets as well? With the restricted definition of adjunction (called "local MC-TAG"), MC-TAG has been shown to have a slightly greater generative power than TAG; however, local MC-TAGs still belong to the class of Mildly Context Sensitive Grammar formalisms (MCSG). Weir [Weir 1988] has also shown that MC-TAGs are equivalent to the Linear Context Free Rewriting Systems (LCFRS), which are the best known characterization of the MCSG formalism (though they are not an exhaustive characterization of MCSG). In particular, we know that local MC-TAGs are polynomially parsable. However, it can be shown that local MC-TAGs are not adequate for deriving all possible scrambled sentences in German (for a detailed discussion, see [Becker and Rambow 1990]). In fact, no LCFRS is powerful enough to capture scrambling. It is obvious that MC-TAG with the more liberal definition of adjunction (called "nonlocal MC-TAG") can produce all the possible versions of scrambled embedded sentences for any level of embedding. However, nonlocal MC-TAG has not yet been studied in detail, and it is currently not known whether nonlocal MC-TAGs are polynomially parsable.

[6] MC-TAGs have already been used by Kroch and Joshi [Kroch and Joshi 1986] for the analysis of extraposition. It is interesting to note that the additional requirement that the foot node of one of the components of an auxiliary set dominate the root node of the other component in the same auxiliary set was also used by them.

[7] This particular example can be derived with a weaker formalism; the point of the example is merely to illustrate the proposed formalism. It can easily be seen how it can handle scrambling from arbitrary levels of embedding.

5 A Free-Order Approach: Relaxed LP

An alternative formalism, which we will call FO-TAG (Free Order TAG), is closely based on the LD/LP-TAG framework presented in [Joshi 1987b]. As does an LD/LP-TAG grammar, a FO-TAG grammar consists of a set of elementary structures. Each elementary structure is a pair consisting of one linear dominance (LD) structure (i.e., an unordered tree) and corresponding LP rules. The LD structure (which will, imprecisely, be referred to as a "tree" in this paper) is either an initial or an auxiliary structure. The LP rules may relate any two nodes of the tree unless one linearly dominates the other. However, these precedence rules can only be stated with respect to the nodes of an elementary tree; it is impossible to relate nodes in different trees. When an auxiliary tree β is adjoined into an initial tree α, the nodes of β are not ordered with respect to the nodes of α. However, even in languages with relatively free word order there are restrictions on movement. In order to capture these, we introduce two linguistically motivated constraints, the ***integrity constraint*** and the ***inheritance constraint***. The integrity constraint, written Δ, allows us to express the fact that German does not allow scrambling into or out of certain constituents, such as NPs and CPs (tensed clauses). If we have ΔX for some node X, then any node which is not in the subtree rooted at X and which does not dominate X must either precede or follow every node in the subtree rooted in X. The inheritance constraint, written \$, forces inheritance and allows us to capture the clause-final position of the verb in German. If we have $\$X$ for a node X, then when the tree of which X is a node is adjoined into another tree at node A, X inherits all LP rules specified for A.

As an example, consider sentences (2a) and (2b) given in Section 2. The initial trees along with the LP rules and constraints are shown in Figure 5. Adjunction yields the structure shown in Figure 6. Note that only one of the possible orderings of the nodes, corresponding to sentence (2b), is shown.

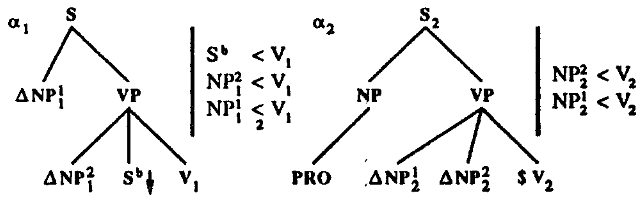

Figure 5: The initial trees in the FO-TAG formalism.

It is easy to see that FO-TAG can generate all scrambled configurations, while obeying the co-occurrence constraint. As in the case of nonlocal MC-TAGs, it is immediately obvious that FO-TAG is not an LCFRS; the question of polynomial parsability remains open, as does the question of the generative power of FO-TAG. We are currently investigating these issues.

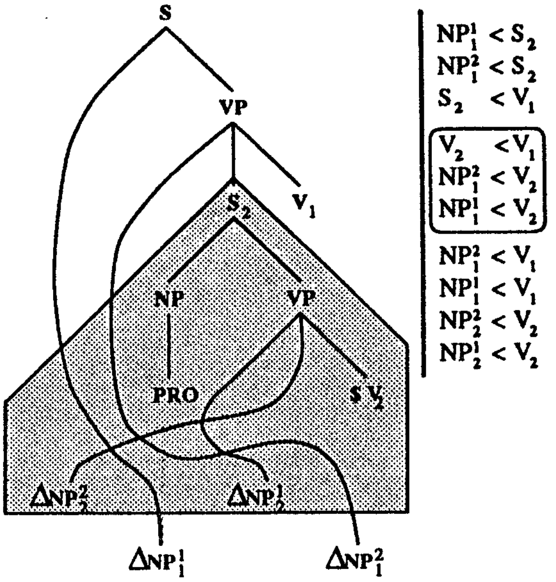

Figure 6: Sentence (2b) in FO-TAG

From a linguistic point of view, it is interesting that the same linguistic phenomenon can be handled by two very different formalisms. Scrambling is currently attracting much attention from syntacticians working in the GB framework. One question as yet unresolved is whether clause-internal scrambling is the same type of syntactic movement as long-distance scrambling. In FO-TAG, both types of movement are created by the same formal device, namely the underspecification of LP rules. In the case of MC-TAG, only long-distance scrambling can be simulated by multicomponent adjunction; clause-internal scrambling must be handled by some other means (such as metarules, which function as an abbreviatory device for listing a finite set of elementary trees), since it is impossible to adjoin one tree into another tree of the same tree set. There are several other syntactic issues which are currently being debated in the linguistic literature, and for which the two formalisms make different predictions. For details, see [Rambow and Becker 1990].

6 Comparison with Other Work

Kroch, Santorini and Joshi's analysis [Kroch et al 1990] of sentences like (2a) and (2b) is similar to the approach proposed in section 4. They also make use of a splitting of the auxiliary tree β of figure 2, though they split the elementary tree into different tree sets. In particular, the verb and its arguments are no longer contained within the same domain of locality, a key requirement of the TAG formalism. Their approach is essentially motivated by linguistic considerations; however, it is easy to show that their analysis can be expressed in our proposed variant of MC-TAG, thus supporting our purely formal analysis, and also showing that the locality of TAGs can be preserved.

The proposed FO-TAG formalism is close in spirit to GPSG, in that ID and LP relations are stated separately. However, none of the work done on free word-order languages in the GPSG framework that we are aware of [Uszkoreit 1987, Lee 1985] deals with long-distance scrambling.

7 Conclusion

We have shown that long-distance scrambling, a syntactic phenomenon exhibited by German and some other languages, cannot be adequately described with a TAG. We have proposed two more powerful extensions of TAG: a variant of the well-studied MC-TAG, and a TAG formalism with free node order, FO-TAG. We have shown that both are descriptively adequate. The linguistic descriptions that these formalisms give rise to, however, are quite different, and they make different predictions about the nature of long-distance scrambling.

Some key formal properties of the two formalisms are still under investigation, in particular the issues of polynomial parsability and generative power. We conjecture that FO-TAG and MC-TAG with dominance links (or some slight definitional variants of the two systems) are weakly equivalent to each other.

References

[Bayer and Kornfilt 1989] Bayer, Josef and Kornfilt, Jaklin, 1989. Restructuring Effects in German. In *Parametric Variation in Germanic and Romance*, Centre for Cognitive Science, University of Edinburgh.

[Becker and Rambow 1990] Becker, Tilman and Rambow, Owen, 1990. Formal Aspects of Long Distance Scrambling. Unpublished Paper, University of Pennsylvania.

[Evers 1975] Evers, Arnold, 1975. *The transformational cycle in Dutch and German.* PhD thesis, University of Utrecht. Distributed by the Indiana University Linguistics Club.

[Joshi 1987a] Joshi, Aravind K., 1987. An Introduction to Tree Adjoining Grammars. In Manaster-Ramer, A. (editor), *Mathematics of Language.* John Benjamins, Amsterdam.

[Joshi 1987b] Joshi, Aravind K., 1987. *Word-Order Variation in Natural Language Generation.* Technical Report, University of Pennsylvania.

[Kroch and Joshi 1985] Kroch, Anthony and Joshi, Aravind K., April 1985. *Linguistic Relevance of Tree Adjoining Grammars.* Technical Report MS-CIS-85-18, Department of Computer and Information Science, University of Pennsylvania.

[Kroch and Joshi 1986] Kroch, Anthony and Joshi, Aravind K., 1986. Analyzing extraposition in a Tree Adjoining Grammar. In Huck, G. and Ojeda, A. (editors), *Syntax and Semantics: Discontinuous Constituents.* Academic Press, New York, NY.

[Kroch et al 1990] Kroch, Anthony; Santorini, Beatrice; and Joshi, Aravind, August 1990. A TAG Analysis of the German Third Construction. In *First International Workshop on Tree Adjoing Grammars.* Schloß Dagstuhl, Germany.

[Lee 1985] Lee, Ik-Hwan, 1985. Toward a Proper Treatment of Scrambling in Korean. In Kuno, Susumo; Whitman, John; Lee, Ik-Hwan; and Kang, Young-Se (editors), *Harvard Studies in Korean Linguistics.* Hanshin Publishing Company, Seoul, Korea.

[Rambow and Becker 1990] Rambow, Owen and Becker, Tilman, 1990. Scrambling and Tree Adjoining Grammars. Unpublished Paper, University of Pennsylvania.

[Schabes 1990] Schabes, Yves, August 1990. *Mathematical and Computational Aspects of Lexicalized Grammars.* PhD thesis, University of Pennsylvania, Philadelphia, PA. Available as technical report (MS-CIS-90-48, LINC LAB179) from the Department of Computer Science.

[Schabes et al 1988] Schabes, Yves; Abeillé, Anne; and Joshi, Aravind K., August 1988. Parsing Strategies with 'Lexicalized' Grammars: Application to Tree Adjoining Grammars. In *Proceedings of the 12^{th} International Conference on Computational Linguistics (COLING'88).* Budapest, Hungary.

[Shieber 1985] Shieber, Stuart B., 1985. Evidence against the context-freeness of natural language. *Linguistics and Philosophy* 8:333–343.

[Uszkoreit 1987] Uszkoreit, Hans, 1987. *Word Order and Constituent Structure in German.* CSLI, Stanford, CA.

[Weir 1988] Weir, David J., 1988. *Characterizing Mildly Context-Sensitive Grammar Formalisms.* PhD thesis, Department of Computer and Information Science, University of Pennsylvania.