September 1989

Phrase Structure and Intonational Phrases: Comments on the Papers by Marcus and Steedman

Aravind K. Joshi
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-89-56.

Phrase Structure and Intonational Phrases: Comments on the Papers by Marcus and Steedman

Abstract

This paper is a commentary on two papers presented at the *Workshop on Cognitive Models of Language*, Sperlonga (Italy), May 1988. These two papers are: "Description Theory and Intonation Boundaries" by Mitch Marcus, and "Syntax and Intonation Structure in Combinatory Grammar" by Mark Steedman. Their approaches to the interface between syntax and metrical phonology are compared and certain problems with their approaches have been pointed out. A phrase structure-like approach, which is able to support some flexible phrasing and permit a set of analyses to hold *simultaneously* has been discussed also. This approach is based on structured objects as syntactic types rather than on curried functional syntactic types as in Steedman's work.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-89-56.

PHRASE STRUCTURE AND INTONATIONAL PHRASES: COMMENTS ON THE PAPERS BY MARCUS AND STEEDMAN

Aravind K. Joshi

MS-CIS-89-56
LINC LAB 160

Department of Computer and Information Science
School of Engineering and Applied Science
University of Pennsylvania
Philadelphia, PA 19104

September 1989

ACKNOWLEDGEMENTS:
Papers presented at the Sprlonga(Italy) Workshop on Computational and Cognitive Models of Speech, May 1988. This work was partially supported by the NSF grants MCS-8219196-CER, IRI84-10413-A02, DARPA grant N00014-0018, ARO grant DAA29-84-K0061.

PHRASE STRUCTURE AND INTONATIONAL PHRASES: COMMENTS ON THE PAPERS BY MARCUS AND STEEDMAN[1]

Aravind K. Joshi
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, Pa 19104
U.S.A.

August 25, 1989

To appear in Computational and Cognitive Models of Speech, G. Altmann (Ed.), MIT Press, 1990.

1. Papers presented at the Sperlonga (Italy) Workshop on Computational and Cognitive Models of Speech, May 1988. This work was partially supported by the NSF grants MCS-8219196-CER, IRI84-10413-AO2, DARPA grant NOO14-85-K-OO18, and ARO grant DAA29-84-KOO61.

ABSTRACT

This paper is a commentary on two papers presented at the *Workshop on Cognitive Models of Language*, Sperlonga (Italy), May 1988. These two papers are: "Description Theory and Intonation Boundaries" by Mitch Marcus, and "Syntax and Intonation Structure in Combinatory Grammar" by Mark Steedman. Their approaches to the interface between syntax and metrical phonology are compared and certain problems with their approaches have been pointed out. A phrase structure-like approach, which is able to support some flexible phrasing and permit a set of analyses to hold *simultaneously* has been discussed also. This approach is based on structured objects as syntactic types rather than on curried functional syntactic types as in Steedman's work.

This paper is based on my notes prepared for oral comments on the two papers presented earlier in the conference. The two papers are: Description Theory and Intonation Boundaries by Mitch Marcus, and Syntax and Intonation Structure in a Combinatory Grammar by Mark Steedman. Since the reader will have read their papers before reading these comments, I will not reproduce their detailed discussions but only refer to some of their key points as I need them.

What are they trying to do ?

Both Marcus and Steedman are concerned with certain aspects of intonational phrasing and their relationship to the syntactic structure . Marcus subscribes to the more conventional X-bar syntax. His parser however does not produce phrase structure trees but 'descriptions' of trees, which, on occasion, may underspecify the structure. Steedman has developed the combinatory categorial framework (CG) which leads to non-standard constituents (giving rise to flexible phrase structure).

Intonational phrases (IP) and intonational boundaries (IB) do not necessarily line up with the conventional phrase structure and phrase boundaries. This is especially true for the so called optional IBs. The term 'optional' makes sense only in the case where there is a fixed phrase structure specified for a sentence, as is the case for Marcus. Marcus completely ignores these optional boundaries, because for him, these boundaries do not affect the syntactic structure. He assumes that these boundaries are specified by some mechanism that relates pragmatics to the intonational contours, Marcus is concerned with obligatory boundaries, i.e., those boundaries which if deleted would result in an unacceptable sentence or his parser would give a wrong analysis. Steedman is more concerned (at least in this paper) with the so-called optional boundaries. Strictly speaking, the optional/obligatory distinction is not relevant for Steedman's grammatical framework, as there is often no unique analysis for a sentence but rather a class of grammatically equivalent derivations. I will continue to use these terms for convenience. Marcus and Steedman are not talking about exactly the same aspects of intonational phrasing. They have focussed on certain aspects of intonational phrasing that relate in interesting ways to certain key features to their theories of grammatical/processing systems. We will discuss these in the next section.

How do they do it ?

Marcus's D-theory parser produces descriptions of trees and not the phrase structure trees directly. In general, it underspecifies the structure. Marcus's parser is deterministic, which on occasion gives the wrong reading. Marcus assumes that the obligatory IBs will be marked by some input that is available to his parser. He then uses this extra input to make sure that his parser does not give a wrong reading, which it would give without this extra input. Thus the obligatory IBs help Marcus's parser, in other words, the obligatory IBs and the D-theory fit together well.

Steedman's CG, in general, provides a multiplicity of analyses for a sentence (which are all equivalent, semantically). This aspect of CG is crucial to the way it provides nice analyses of several coordination phenomena. Steedman wants to suggest that the multiplicity of analyses CG provides for a sentence are also relevant to intonational phrasing. Out of all the possible equivalent analyses for a sentence, Steedman will pick the analysis that gives the syntactic phrasing that lines up with the intonational phrasing. Thus Steedman uses the IPs to help select one of the many equivalent analyses. The nonstandard analyses that a CG provides are thus relevant because the IPs, in general, cut across the standard syntactic boundaries in various ways. Steedman claims that the flexibility of structure possible in a CG is adequate for capturing the IPs. Thus a separate mechanism is not needed for specifying the IPs, as is the case if standard constituency is assumed, resulting in a simpler architecture for metrical phonology.

Although both Marcus and Steedman adopt some notion of flexible constituent structure (D-theory for Marcus and CG for Steedman), their architectures for relating the systems of constituent structure and the generation of intonational phrases are quite different. The major difference between these architectures is described in Fig. 1 and 2 below.

Architecture implicit in Marcus's work.

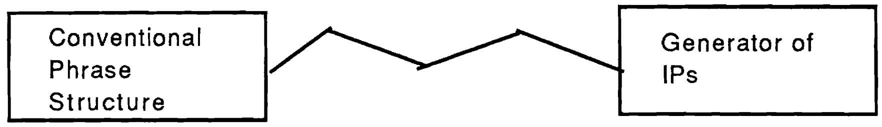


FIG. 1

The two components, phrase structure component and the generator of IPs

are separate components. The relationship between these two components is not direct, rather lose and not strictly defines, hence shown by a wavy line in Fig 1. An appeal to some additional condition such as the 'sense-units' condition (refereed to in Steedman's paper, attributed to Selkirk) is made to explicate this relationship. These assumptions
are not made directly by Marcus, he is however assuming a conventional architecture for metrical phonology.

Architecture proposed by Steedman

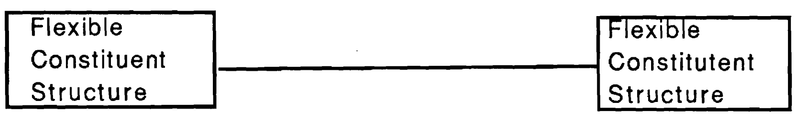


Fig. 2

The two components are not really distinct. The system of flexible constituent structure given by CG also provides the IPs. Thus the two components are really identical and therefore the relationship between them is clearly well-defined, hence shown by a solid straight line on Fig.2.

Later I will describe an architecture which as a phrase structure like component for the constituent structure and a generator for IPs that is well-defined and directly linked to the first component but not identical to the first component as in Steedman's work. But first I will return to the works of Marcus and Steedman.

Some problems with what they have done.

Marcus has used the obligatory IBs to help his D-theory parser so that some unwanted readings are prevented. Since Marcus's D-theory is a processing account, it is difficult to formalize it in the sense that it is not possible to state precisely all the analyses that are supported by his system and those that are not supported. It is clear from his paper how some of the obligatory IBs prevent the parser from getting wrong readings, however, it is difficult to see just exactly what is the set of all wrong readings that are prevented by the obligatory IBs. In other words, although the relationship between the obligatory IBs and the D-theory parser is clear for the specific examples discussed, the general character of this relationship is not obvious. This is so, to a large extent, due to the fact that the distinction between obligatory IBs and optional IBs seems to be tied with the D-theory parser itself.

Steedman's CG formalism is a grammar formalism, so in principle, it is possible to figure out the set of possible (equivalent) analyses for a sentence. Each one of these analyses corresponds to different syntactic phrasing. Each phrase is semantically coherent as a "sense unit". This follows from the theory of CG itself. Steedman identifies the required IP with one of these phrasings. This part of Steeedman's argument is very clear. It is not clear, however, whether each one of the possible analyses corresponds to some IP in some appropriate context, or whether only some subset of these analyses corresponds to IPs in appropriate contexts. If the latter situation holds then it is necessary to give an independent characterization to this subset and it is not clear to me how this can be done. Steedman's CG provides a collection of analyses in which it is possible to find one which corresponds to the required IP. The claim for the flexibility provided by CG would be stronger and more interesting if it could be shown that either all the analyses are relevant to IPs in suitable contexts, or that only some are, and that this subset of analyses has some natural characterization. In the case of coordinate constructions Steedman has taken the position that all these analyses are relevant for coordination, Thus, it is important to raise a similar question in the case of IPs. If such a claim could be supported it would make the case for Steedman's revised version of the architecture for metrical phonology much stronger.

Since the multiplicity of analyses is exploited by Steedman for both coordination and for the specification of IPs, there is a potential problem when coordination and IP would require different phrasing. Consider the following situation:

(1) Q: What does Mary prefer ?

(2) R: Mary prefers oysters.

(1) suggests the open proposition

λx . Mary prefers x

and this corresponds to the IP as in (3) below

(3) (Mary prefers) (oysters)

which is one of the analyses provided by the CG. However, let us suppose that R's response is

(4) R: Mary prefers oysters but hates clams.

Now the coordination in (4) requires (in CG) that the syntactic phrasing must be

(5) (Mary) (prefers oysters)

for the left conjunct in (4). This is different from (3) so there is a potential conflict between the phrasing required in (3) and in (5). Of course, we could say in this case (as Steedman suggested in his response to my comments at the workshop) that R has a different open proposition in mind when responding as in (4). In the discourse (1) and (2), the syntactic phrasing of (1) (in accordance with CG) suggests the open proposition

λx . Mary prefers x

which then serves to pick the appropriate phrasing for (2). This is an attractive part of Steedman's proposal. If (4) can be a response to (1) then the open proposition suggested by phrasing of (1) is irrelevant to the phrasing of (4). The syntactic phrasing required for (4) can be related to some appropriate open proposition, which is, of course, not the same as that suggested by (1), but then this makes the open proposition suggested by (1) not relevant in predicting the phrasing in (4). If (4) can be uttered with the first conjunct in (4) receiving the IP as in (Mary prefers) (oysters) then, of course, there is real trouble. If one adopts the proposal that there can be multiple analyses *simultaneously* holding for the sentence, then there is no problem. (I don't think Steedman would subscribe to this position.) Note that this proposal of simultaneously holding multiple analyses is not the same proposal as is implied in the conventional architecture for metrical phonology. The conventional architecture requires that there is a phrase structure component giving the syntactic structure and there is some other machinery that provides the IP, the phrase structure component does not provide the IP. The proposal suggested above is different. The different analyses which hold simultaneously are all given by the same grammatical system. I will pursue this point later in a little more detail.

Nonstandard syntactic analyses

Both Marcus and Steedman have used nonstandard syntactic analyses; Marcus only marginally, and Steedman substantially. As I pointed out earlier, Marcus uses the standard X-bar syntax and his parser constructs (partial) descriptions of trees rather than the trees themselves. He assumes as in the case of conventional architecture for metrical phonology, that the syntax specifies a fixed phrase structure, and the IPs are specified by some other mechanism, IPs are not specified by the phrase structure component. In contrast, Steedman's CG formalism gives a flexible phrasing, in fact, a sentence has multiple analyses (all semantically equivalent), all of which are given by the grammar itself. An appropriate phrasing can be selected then as determined by the required IP.

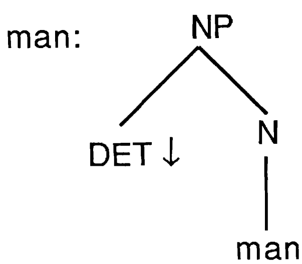
The space of possibilities of nonstandard syntactic analyses provided by different formalisms is not limited to those considered by Marcus and Steedman. I will briefly discuss some of the implications of the TAG (Tree Adjoining Grammar) formalism for some of the issues raised in these two papers, especially the issue of flexible phrasing raised by Steedman. TAGs are interesting to look at in this context because it has been shown that a number of different grammatical formalisms are equivalent to TAG; for example, Head Grammars (HG) of Pollard, Linear Indexed Grammars, as discussed by Gazdar, and more importantly for the present discussion, Combinatory Grammars (CG) of Steedman (Joshi (1987) and Joshi, Weir, and Vijay-Shanker (1989)).

The elementary trees of TAG provide an extended domain of locality (as compared to Context-free Grammars), which allows factoring recursion from the domain of dependencies. Dependencies such as agreement, subcategorization, and even the so-called long distance dependencies such as topicalization and wh-movement are all defined on the elementary trees and are thus local. The long distance nature of the dependencies is then a consequence of the adjoining of auxiliary trees to elementary trees or derived trees. The extended domain of locality and the consequent encapsulation of the predicate argument structure in each one of the elementary trees is relevant to the issue of 'flexible structure'.

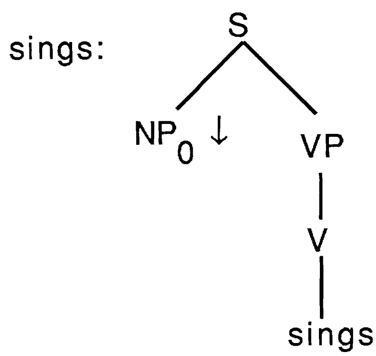
I will consider a lexicalized TAG, which consists of a finite set of structures (trees) associated with each lexical item which is intended to be the 'head' of these structures,[2] and two operations for composing these

2. The notion of 'head' here really corresponds to that of a functor in CG. The elementary tree associated with a lexical item can be regarded as a *structured* object which is its syntactic

structures. These operations are substitution and adjoining. Instead of giving formal definitions for these two operations they will be illustrated by examples later.The finite set of trees consists of two disjoint sets of trees: initial trees and auxiliary trees. Each elementary tree encapsulates the predicate argument structure. The following example illustrates a lexicalized TAG. Some elementary trees (initial trees and auxiliary trees) are shown below.



(1)

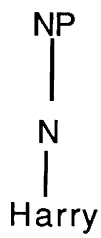


(7)

---

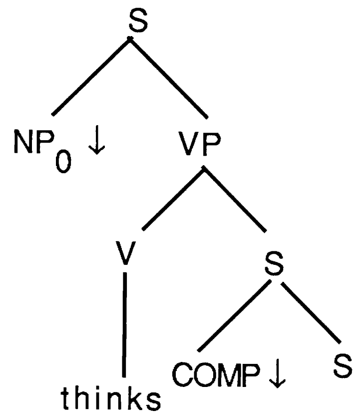
type.

Harry:



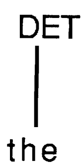
(2)

thinks:



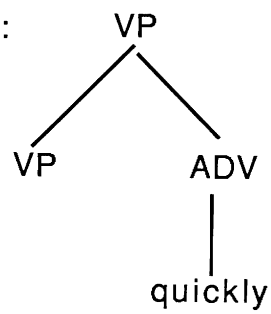
(8)

the:



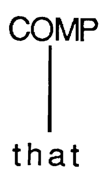
(3)

quickly:



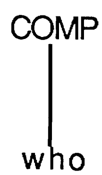
(9)

that:

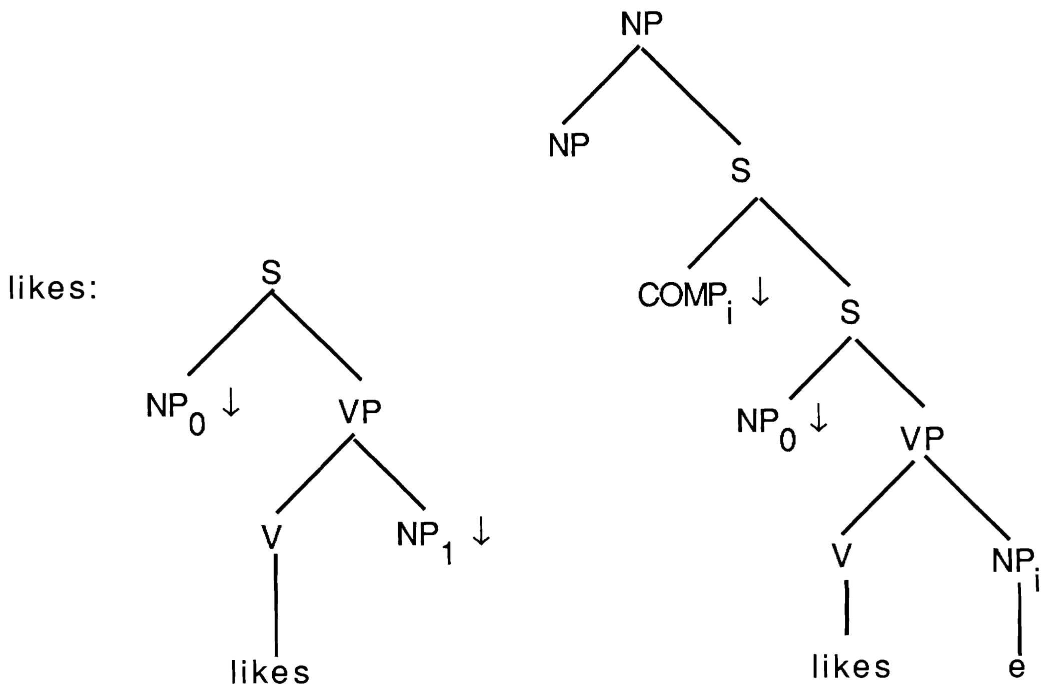


(4)

who:



(5)



(6)

(10)

Trees (1) - (8) are initial trees. Trees 9 and 10 are auxiliary trees. For X $\neq$ S, X-type initial trees correspond to the trees which can be substituted for one of the argument positions of elementary trees. Trees 6, 7, and 8 are examples of of S-type initial trees. They are structures that encapsulate

all arguments of the verb including the subject. The $\downarrow$ near a node indicates that an appropriate substitution has to be made at that node. Nodes without $\downarrow$, unless they are terminals, are possible sites for ad-joining. In particular, note that the foot node of an auxiliary tree (e.g. tree (10)) does not have $\downarrow$ because this tree is an auxiliary tree which can be adjoined at some appropriate node in a some tree.

Trees (9) and (10) are auxiliary trees (these are modifiers or predicates taking sentential complements), Tree (9) corresponds to an adverbial modifier, and tree (10) corresponds to a relative clause. Tree (9) has VP as the root node and VP as the foot node, tree (10) has NP as the root node and NP as the footnode.

Complex structures are built by substitution and adjoining. Substituting tree (3) in tree (1) for DET, substituting tree (1) at the NP_0 node in tree (6), and tree (2) at the NP_1 node of tree (6), we get

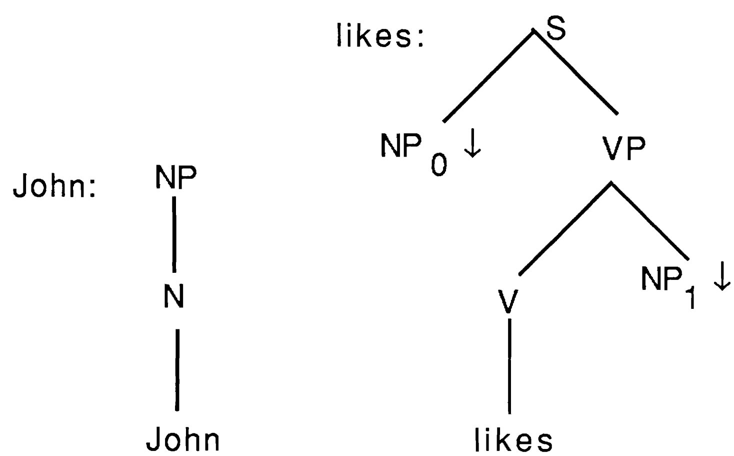
(6) The man likes Harry

With appropriate substitutions in tree (10) and then adjoining tree (10) to the tree corresponding to (6) above (at the subject NP node) we get

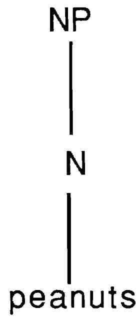
(7) The man who Mary likes likes Harry

Adjoining can be thought of as excising the subtree at a node , inserting an auxiliary tree of the right type and then attaching the excised subtree at the foot node of the auxiliary tree. It has been shown that by using adjoining together with substitution we can obtain a lexicalized TAG as shown above, with each elementary tree encapsulating the predicate and its arguments. In this representation, it is easy to see the relationship between lexicalized TAG and CG. Although a TAG provides a phrase structure at the level of elementary trees and at each step of the derivation a new tree is derived, these trees are *object language* trees and not derivation trees as in a context-free grammar. The derivation structures of TAG are in terms of these elementary trees, i.e., they record the history of the derivation of the *object language* tree in terms of the elementary trees and the nodes where substitution or adjunctions are made. Therefore, the resulting system is not a standard phrase structure grammar. Thus although CGs are weakly (i.e., in terms of the strings they generate) equivalent to TAGs , TAGs represent a system intermediate between the standard phrase structure grammars and CG.

The question now is: How can we assign structures to strings which correspond to nonstandard constituents (in terms of CG)? Given the following elementary trees:

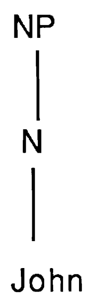


peanuts:



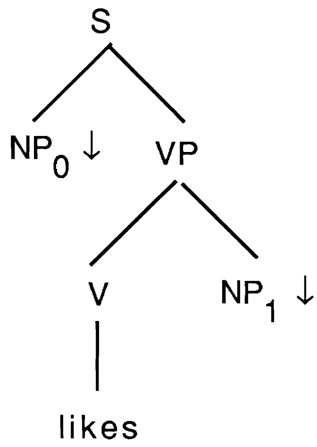
we can associate structures to the following strings. On the right hand side of each structure we have shown its associated syntactic (functional) type to bring out the relationship to CG.

(1) John:



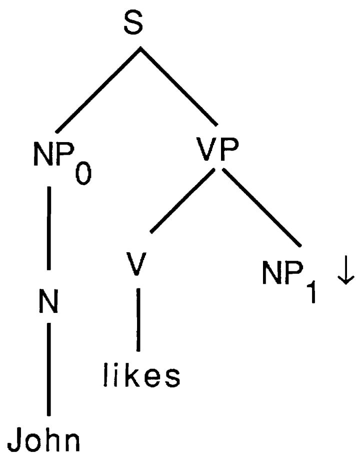
NP

(2) likes:



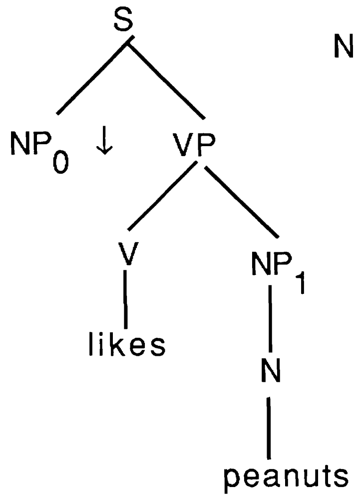
$$NP_1 \rightarrow NP_0 \rightarrow S$$
$$=$$
$$NP_0 \rightarrow NP_1 \rightarrow S$$

(3) John likes:



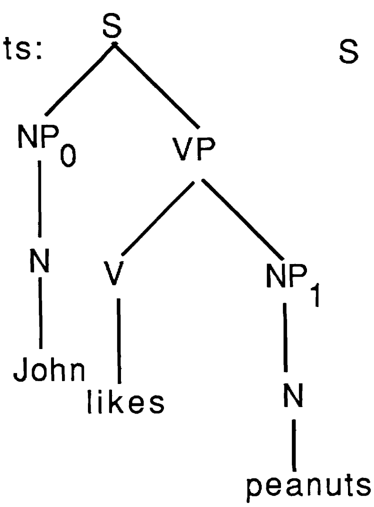
$$NP_1 \rightarrow S$$

(4) likes peanuts:



$NP_0 \rightarrow S$

(5) John likes peanuts:



S

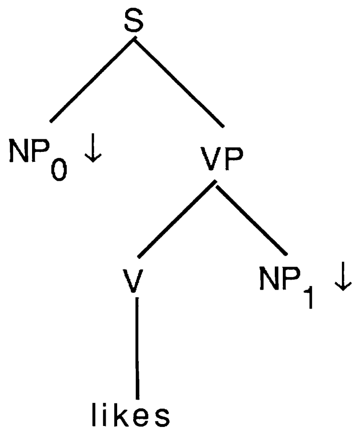
Thus *John likes* is of the type $NP_1 \rightarrow S$ (the indices are for convenience only, the tree addresses distinguish the NPs), *likes peanuts* is of the type $NP_0 \rightarrow S$, and *John likes peanuts* is of the type S. Note that *likes* is of the type $NP_1 \rightarrow NP_0 \rightarrow S$ or $NP_0 \rightarrow NP_1 \rightarrow S$, and, in fact, we must assert an equality between these two types.

The *Curry* notation of CG captures the argument structure (e.g., for *likes* one has (S\NP)/NP corresponding to the elementary tree for *likes* as shown above); however, with this *Curry* notation comes the requirement that the arguments must be bound in a specific order. What we have shown in the above example is that if we work with structures (as shown above) then we can bind the arguments in any order, thus allowing the assignment of types to strings such as *John likes* , without letting go the phrase structure at the level of elementary trees. Thus in this representation there appears to be a way of assigning types to such strings as *John likes* as well as to *likes peanuts* , i.e., to the possible intonational phrasings.[3]

We can regard the elementary trees of the lexicalized TAG, for example, trees (1) - (10) as a representation of categories as *structured objects*. Thus the representation of the category for *likes* we have the structured object (tree (6)).

3. The representation provided by a lexicalized TAG is crucial here. It can be shown that substitution alone cannot lexicalize a context-free grammar, but substitution and adjoining together can lexicalize a context-free grammar. The resulting system is then a lexicalized TAG! For details, see Schabes, Abeille, and Joshi (1988).

likes:



(6)

Thus with operations of substitution and adjoining as defined before a lexicalized TAG is like a *categorial* grammar , where the categories are structured objects, where the basic categories are structured objects and the operations of substitution and adjoining derive structured objects from structured objects. A basic structured object can be associated with a functional type or types as described earlier. Derived structured objects can also be associated with a functional type or types in the obvious manner. Thus the operations of substitution and adjoining play a role analogous to function composition.[4]

We have associated functional types with the structured objects to bring out the relationship to CG. However, since one of our major objectives is to avoid the Curry notation, it is better to associate a relational type to a

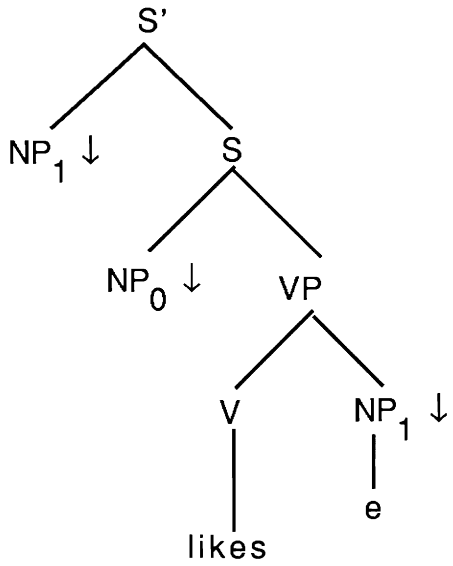
4. Substitution corresponds directly to function composition. Adjoining can be shown to be a kind of function composition also. An auxiliary tree is function of the type $X \rightarrow X$. Adjoining is then a composition of a function obtained by abstracting on the node in a tree where adjunction is to be made and the function corresponding to the auxiliary tree to be adjoined at that node. Function application is a special case of substitution.

structured object. Thus for the tree (6) above, we can associate the relational type

(eat NP_0 , NP_1)

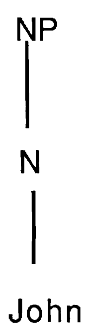
The derived objects can be associated with relational types also. Substitution and adjoining give rise to derived structured objects and the derived relational types are obtained by composition of relations. This relational type approach is the appropriate one to pursue in the context of lexicalized TAGs where the elementary trees are viewed as categories represented as structured objects. However, since the immediate concern is the relationship between TAG and CG, we will continue to talk in terms the associated functional types.

Since we have avoided the Curry notation,. the order in which the arguments of a function are can filled in is arbitrary and this freedom gives us the ability to assign (functional) syntactic types to lexical strings that are not constituents in the conventional sense. Of course, this freedom will give rise to to some functions that we would like to rule out. Thus, for example, consider the elementary tree corresponding to a topicalized sentence, the (12) below. Tree (12) can be regarded as another category (structured object) associated with *likes.*

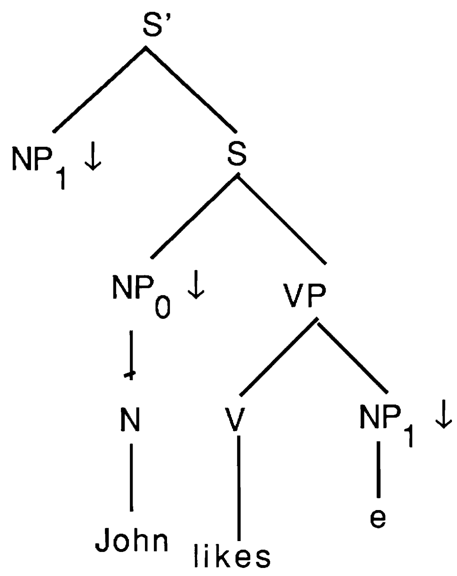


(12)

substituting



for NP_0 in (12), we have



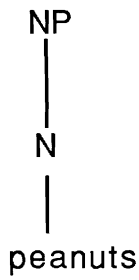
$NP_1 \rightarrow S'$

(13)

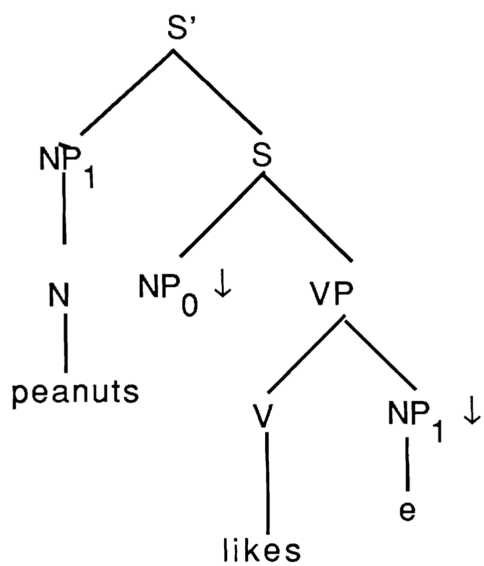
The lexical string corresponding to (13) is *John likes.* Let us call *John likes* as the string 'spelled out' by (13) or by the function corresponding to (13). Note that *John likes* can be coordinated with , say, *Bill hates* as in

(14) Apples John likes and Bill hates.

Thus *John likes and Bill hates is of the functional type* $NP_1 \rightarrow S'$. The idea is that the lexical strings spelled out by the structured objects are the fragments that can be coordinated (and therefore also serve as appropriate intonational phrases, following Steedman's idea). Of course, sometimes the lexical string spelled out by a structured object will not be a fragment that can be coordinated (or serve as an intonational phrase). Thus in the tree (13) above, if we first substitute



for NP_1 then we have the derived object, tree (15) below.



(15)

The lexical string spelled out by (15) is *peanuts likes,* however, this string is not an appropriate fragment for coordination (or for serving as an intonational phrase). If it were, we would get

(16) * Peanuts John likes and almonds hates

(meaning *John likes peanuts and John hates almonds).* So we want to rule out the functional type corresponding to (15) as a possible candidate for coordination. We can rule out (15) by stating a requirement on the lexical string spelled out by the structured object. The lexical string spelled out by (15) is not a contiguous string in the sense that it is interrupted by NP_0, i.e., in the frontier of the tree (15) *peanuts* and *likes* are not contiguous.

It appears that we can develop an account of coordination (and therefore for the possible intonational phrases as suggested by Steedman) in a lexicalized TAG by requiring that the only possible functional types that are appropriate for coordination (or can serve as functional types associated with intonational phrases) are those which spell out a lexical string that is contiguous[5]. This requirement of contiguity of the lexical string appears to be wee motivated if we want to treat these fragments as possible candidates for IPs. This contiguity condition can be thought of as a phonological condition and not as a syntactic condition.

I have thus shown (albeit rather briefly and informally) that if we start with categories as structured objects (trees of lexicalized TAGs, for example) then we can assign (syntactic) functional types to strings which

5. A detailed development of this approach is being worked out at present and will be described elsewhere later. It is only after reading Steedman's paper that I began to pursue the approach briefly described above.

are nonconstituents (as well as as those which are constituents, of course) in the conventional sense, without letting go the phrase structure at the level of the elementary trees. Thus in contrast to the two architectures described in Fig. 1 and Fig. 2 earlier, we have now a new architecture as described in Fig. 3 below.

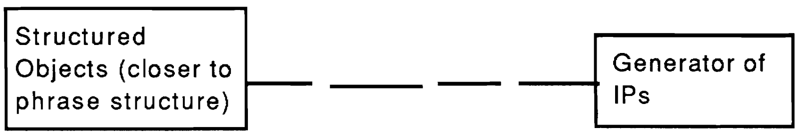


Fig. 3

The two components are systematically related, however they are not identical as in CG. Hence, we have shown this relationship between these two components by a dashed line.

In conclusion, in this section, I have suggested that the space of nonstandard phrase structure analyses is large. I have indicated that it may be possible to have a nonstandard syntactic analysis which combines standard phrase structure and intonational phrasing in a systematic manner, and not by having a separate independent component for intonational phrasing, as is the case in the standard architecture for metrical phonology.

Conclusion

Both Marcus and Steedman have made very interesting contributions to the interface between syntax and metrical phonology by showing how their respective theories of grammar/processing interact with IPs. I have commented briefly on their respective approaches, pointing out the different aspects of IPs they are concerned with, and bringing out some of the problems with their respective theories. I have also discussed briefly a phrase structure-like approach, which is able to support some flexible phrasing, thus permitting a set of analyses holding *simultaneously*. It is clear that there are a number of different ways to set up the interface between syntax and metrical phonology, and there will be theory-internal criteria for selecting one theory over another. There are, of course, two external criteria that are highly relevant from the point of view of the topic of this workshop. These are psycholinguistic relevance and computational efficiency. Both Marcus and Steedman have said very little in their papers with respect to these issues. Marcus's commitment to determinism and Steedman's commitment to incremental processing in their other papers reflect clearly their concern for these issues. I expect them to develop these issues further in their future work concerning the interface between syntax and metrical phonology.

References

Joshi, A.K. An introduction to tree adjoining grammars. In A. Manaster-Ramer, editor, *Mathematics of Language* John Benjamin, Amsterdam, 1987.

Joshi, A.K., Vijay-Shanker, K., and Weir, D. The convergence of mildly context-sensitive grammar formalisms. In S. Shieber and T. Wasow, editors, *The Processing of Linguistic Structure* MIT Press, Cambridge, MA, to appear in 1989.

Schabes, Y., Abeille, A., and Joshi, A.K. Parsing strategies with 'lexicalized' grammars: application to tree adjoining grammars. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING 88)* Budapest, August 1988.