

## D-Theory: Talking about Talking about Trees

Mitchell P. Marcus

Donald Hindle

Margaret M. Fleck

Bell Laboratories

Murray Hill, New Jersey 07974

Linguists, including computational linguists, have always been fond of talking about trees. In this paper, we outline a theory of linguistic structure which talks about talking about trees; we call this theory *Description theory* (D-theory). While important issues must be resolved before a complete picture of D-theory emerges (and also before we can build programs which utilize it), we believe that this theory will ultimately provide a framework for explaining the syntax and semantics of natural language in a manner which is *intrinsically* computational. This paper will focus primarily on one set of motivations for this theory, those engendered by attempts to handle certain syntactic phenomena within the framework of deterministic parsing.

### 1. D-Theory: An Introduction

The key idea of D-theory is that a syntactic analysis of a sentence of English (or other natural language) consists of a *description* of its syntactic structure. Such a description contains information which differs from that contained in a standard tree structure in two crucial ways:

1) The primitive predicate for indicating hierarchical structure in a D-theory description is "dominates" rather than "directly dominates". (A node A is said to *dominate* a node B if A is some ancestor of B; A is said to *directly dominate* B if A is the immediate parent of B.) A D-theory analysis thus expresses directly only what structures are contained (somewhere) within larger structures, but does indicate *per se* what the immediate constituents of any particular constituent are.

A tree structure, on the other hand, encodes which nodes are directly dominated by other nodes in the analysis; it indicates directly the immediate constituents of each node. In a standard parse tree, the topmost S node might directly dominate exactly a Noun Phrase node, an Aux node and a Verb Phrase node; it is thus made up of three subparts: that NP, that Aux, and that VP.

2) A D-theory description uses *names* to make statements about entities, and does not contain the entities themselves. Furthermore, there is no distinguished set of names which are taken to be standard names or rigid designators; i.e. given only a name, one cannot tell what particular syntactic entity it refers to. (This is the primary reason that we view D-theory representations as descriptions and not merely as directed acyclic graphs.)

Because there are no standard names, if one is presented with two descriptions, each in terms of a different name, one can tell with certainty only if the two names refer to *different entities*, but never (for sure) if they refer to the same entity. In the latter case, there is always potential ambiguity. To take a commonplace example, given that "John has red hair" and "Mr.

Jones has black hair", one can be sure that John is not Mr. Jones. But if one is told "John has red hair" and "Mr. Jones wears glasses" and nothing more about either John or Mr. Jones, then it is impossible to tell whether John is or is not Mr. Jones. In the domain of syntax, if a D-theory description says that

X is an NP; Z is an NP  
Y is an Adjective Phrase  
W is a noun

X dominates Y  
Z dominates W

and nothing else is stated about W, X, Y or Z, then it cannot be determined whether X and Z are aliases for the same NP node or are names for two distinct nodes. If an additional statement is added to the description that "Y dominates Z", then it must be the case that X and Z name distinct entities. We will show in what follows that the use of names has important ramifications for linguistic theory and the theory of parsing.

The structure of the rest of this paper is roughly as follows: We will first sketch the computational framework we build on, in essence that of [Marcus 80], and explore briefly what a parser for this kind of grammar might look like; in appearance, its data structures and grammar will be little different from that developed in [Berwick 82]. A series of syntactic phenomena will then be explored which resist elegant account within the earlier framework. For each phenomenon, we will present a simple D-theoretic solution together with exposition of the relevant aspects of D-theory.

One final introductory comment: That D-theory expresses syntactic structure in terms of dominance rather than direct dominance may be reminiscent of [Lasnik & Kupin 1977] (henceforth L-K), but our use of the dominance predicate differs fundamentally from the L-K formulation both in the primacy of the predicate to the theory, and in the theory of syntax implied. Lasnik and Kupin's formalization of the Extended Standard Theory *derives* domination relations from their primary representation of linguistic structure, namely a set of strings of terminals and nonterminals with specified properties. D-theory structures are expressed directly in terms of dominance relations; the linear order of constituents is only directly expressed for items in the lexical string. Despite appearances, D-theory and the Lasnik-Kupin formalization are *not* inter-definable. We discuss the properties of the Lasnik-Kupin formalization at length in a forthcoming paper.

## 2. Deterministic Tree-Building: The Old Theory

D-theory grows out of earlier work on deterministic parsing as deterministic tree building (as in e.g. [Marcus 1980], [Church 80] and [Berwick 82]). The essence of that work is the hypothesis that natural language can be analyzed by some process which builds a syntactic analysis *indelibly* (borrowing a term from [McDonald 83]); i.e. that any structure built by the parser is part of the correct analysis of the input. Again, in the context of this earlier theory, the form of the indelible syntactic analysis was that of a *tree*.

One key idea of this earlier tree-building theory that we retain is the notion that a natural language parser can *buffer* and examine some small number (e.g. up to three) unattached constituents before being forced to add to its existing structures. (In D-theory, the node named X is *attached* to Y if the parser's description of the existing structure includes a predication of the form "Y dominates X", or, as we will henceforth write, "D(Y,X)." X is *unattached* if the parser's description of the existing structure includes no predication of the form "D(Y, X)", for any name Y.) We thus assume that such a parser will have the two principle data structures of these earlier deterministic parsers, a stack and a buffer. However, the stack and the buffer in a D-theory parser will contain *names* rather than constituents, and these data structures will be augmented by a *data base* where the description of the syntactic structure itself is built up by the parser. (While this might sound novel, a moment's reflection on LISP implementation techniques should assure the reader that this structure is far *less* different from that of older parsers like Parsifal and Fidditch [Hindle 83] than it might sound.)

As we shall see below, however, a parser which embodies D-theory can *recover* (in some sense) from some of the constructions which would terminally confuse (or "garden path") a parser based on the deterministic tree-building theory. For D-theory to be psychologically valid, of course, it must be the case that just those constructions which *do* garden path a D-theory parser garden path people as well. (We might note in passing that recent experimental paradigms which explore online syntactic processing using eye-tracking technology promise to provide delicate tests of these hypotheses, e.g. [Rayner & Frazier 83].)

Another goal of this earlier work was to find some way of procedurally representing grammars of natural languages which is brief and perspicuous, and which allows (and perhaps even forces) grammatical generalizations to be stated in a natural way. As is often argued, such a representation must be embodied by our language understanding faculty, given that the grammar of a language is learned incrementally and quickly by children given only limited evidence. (To recast this point from an engineering point of view, this property is also a prerequisite to writing a grammar for a subset of some given natural language which remains extensible, so that new constructions can be added to the grammar without global changes, and so that these new constructions will interact robustly with the old grammar.)

Following [Shipman 78], as refined in [Berwick 82], we assume that the grammar is organized into a set of context free rules, which we will call *base templates*, and a set of pattern-action rules. As in Parsifal, each pattern consists of up to four elements, each of which is a partial description of an element in the buffer, or the accessible node in the stack (the "current active node"). Loosely following [Berwick 82], we assume that

the action of each rule consists of exactly one of some small set of limited actions which might include the following:

- Attach a node in the buffer to the current active node.
- Switch the nodes in the first two buffer positions.
- Insert a specified lexical item into a specified buffer slot.
- Create a new current active node.
- Insert an empty NP into the first buffer slot.

(Where "attachment" is as defined above, and "create" means something like coin a new node name, and push it onto the active node stack.) Each rule is associated with some position in one of the base templates. So, for example, in figure 1 below, one base template is given, a highly simplified template for a sentence. Associated with the NP in the subject position of the sentence are several rules. The first rule says that if the first buffer position holds a name which is asserted to be an NP (informally: if there is an NP in the first buffer slot), then (informally) it is dominated by the S. The second says that if there is an auxiliary verb in the first slot followed by an NP, then switch them. And so on.

Note that while a D-theory parser itself has no predicate with which to express direct dominance, the base templates explicitly encode just such information. Insofar as the parser makes its assertions of dominance on the basis of the phrase structure rules, the parser will behave very similarly to deterministic tree

```

S      ->   NP          VP   PP*
          {[NP => Attach}
          {[auxv][NP => Switch}
          {[v, tenseless] => Insert(NP, 0)}
  
```

Figure 1. A simplified base template for S, with associated NP rules.

building parsers. In fact, the parser will typically (although, as we will see below, not always) behave in just such a fashion.

## 3. The Problem of Misleading Leading Edges

By and large, we believe that a significant subset of the grammar of English has been successfully embedded within the deterministic tree-building model. However, a residue of syntactic phenomena remain which defy simple explication within this framework. Some of these phenomena are particular problems for the deterministic tree-building framework. Others, for example coordination and gapping phenomena, have defied adequate explication within any existing theory of grammar.

In the remainder of this paper we will explore a range of such phenomena, and argue that D-theory provides a consistent approach which yields simple accounts for the range of phenomena we have considered to date. We will first argue for taking "dominates", not "directly dominates" as primitive, and then later argue why the use of *names* is justified. (Our view that this representation should be viewed as a description hangs on the use of names. In this section and in section 5 we argue only for a representation which is a particular kind of directed acyclic graph. Only with the arguments of section 7 is the position that this is a kind of description at all defensible.)

One particularly interesting class of sentences which seems to defy deterministic accounts is exemplified by (2).

- (2) I drove my aunt from Peoria's car.

Sentences like (2) contain a constituent which has a misleading "leading edge", an initial right-embedded subconstituent which could itself be the next constituent of whatever structure is being built at the next level up. For example, while analyzing (2), a parser which deterministically builds old-fashioned trees might just take "my aunt" to be the object of "drove", attaching it as the object of the VP, only to discover (too late) that this phrase functions instead as genitive determiner of the full NP "my aunt from Peoria's car".

In fact, the existing grammar for Parsifal causes exactly this behavior, and for good reason: This parser constructs NPs only up to the head noun before deciding on their role within the larger context; only after attaching an NP will Parsifal construct the post-modifiers of the NP and attach them. (This involves a mechanism called *node reactivation*; it is described in [Shipman & Marcus 79].) One reason for this within the earlier framework is that, given a PP which immediately follows the head of an NP, it cannot be determined whether that PP should be attached to the preceding NP or to some constituent which dominates the NP until the role of that NP itself has been determined. In the specific case of (2), the parser will attach "my aunt" as the object of the verb "drove" so that it can decide where to attach the PP beginning with "from". Only after it is too late will the parser see the genitive marker on "Peoria's" and boggle. While one could attempt to overcome this particular motivation for the two-stage parsing of NPs with some variant of the notion of *pseudo-attachment* (first used in [Church 80]), this and related approaches have their problems too, as Church notes.

Potential pseudo-attachment solutions aside, the upshot is that sentences like (2) will cause deterministic tree building parsers to garden path. However, it is our strong intuition that such cases are not "garden paths"; we believe that such cases *should* be analyzed correctly by a deterministic parser rather than by the (putative) mechanism which recovers from garden paths.

The D-theoretic solution to the problem of misleading "leading edges" hinges on one formal property of this problem: The initial analysis of this class of examples is incorrect only in that some constituent is attached in the parse tree at a *higher* point in the surrounding structure than is correct. Crucially, the parser neither creates structures of the wrong kind nor does it attach the structure that it builds to some structure which does not dominate it. In the misanalysis of (2), the parser initially errs only in attaching the NP "my aunt", which is indeed dominated by the VP whose head is "drove", too high in the structure.

This class of examples is handled by D-theory without difficulty exactly because syntactic analyses are expressed in terms of domination rather than direct domination. The developing description of the structure of (2) in a D-theory parser at the point at which the parser had analyzed "my aunt", but no further, might include the following predications:

- (3.1) D(vp1, np1)
- (3.2) D(vp1, v1)

where the verb node named v1 dominates "drove", and the NP node named np1 dominates the lexical material "my aunt".

Let us assume for the sake of simplicity that while building the PP "from Peoria's", the parser detects a genitive marker on the proper noun "Peoria's" and knows (magically, for now) that "Peoria's car" is not the correct analysis. Given this, the genitive must mark the entire NP "my aunt from Peoria" and thus "my

aunt from Peoria" must serve not as the object of the verb "drove" but as the determiner of some larger NP which itself must be the object of "drove". (Unless *it* is followed by a genitive marker, in which case....) The question we are centrally interested in here is not *how* the parser comes to the realization that it has erred, but rather *what* can be done to remedy the situation. (Actually how the parser must resolve this first problem is a complex and interesting story in and of itself, with the punchline being that exactly one (but only one) of (2) and

(4) I drove my aunt from Peoria's suburbs home.

*must* cause a garden path. The details of this await further research on the control of D-theory parsing.)

The description (3) is easily fixed, given that "D" is read "dominates", and not "directly dominates". Several further predications can merely be *added* to (3), namely those of (5), which state that np1 is dominated by a determiner node named det1, which itself is dominated by a new np node; np2, and that np2 is dominated by vp1.

- (5.1) D(np1, det1)
- (5.2) D(det1, np2)
- (5.3) D(np2, vp1)

Adding these new predications does not make the predications of (3) false; it merely adds to them. The node named np1 is still dominated by vp1 as stated in (3.1), because the relation "D" is transitive. Given the predications in (5), (3.1) is redundant, but it is not false.

The general point is this: D-theory allows nodes to be attached initially by a parser to some point which will turn out to be higher than its lowest point of attachment (for the more general sense of attachment defined above) without such initial states causing the parser to garden path. Because of the nature of "D", the parser can in this sense "lower" a constituent without falsifying a previous predication. The earlier predication remains indelible.

#### 4. Semantic Interpretation: The Standard Referent

But how can such a list of domination predications be interpreted? It would seem that compositional semantics *must* depend upon being able to determine exactly what the *immediate* constituents of any given structure are: if the meaning of a phrase determined from the meanings of its parts, then it must be determined exactly what its parts are.

We assume that semantic interpretation of a D-theory analysis is done by taking such an analysis as describing the minimal tree possible, i.e. by taking "D" to mean *directly* dominates wherever possible *but only for semantic analysis*. For example, if the analysis of a structure includes the predications that X dominates Y, Y dominates Z and X also dominates Z, then the semantic interpreter will assume that X directly dominates Y and that Y directly dominates Z. We will call such an interpretation of a D-theoretic analysis the *standard referent* of the analysis. (We further assume that the description produced by a D-theory parser will have at each stage of the analysis one and only one standard referent, and the complex situation where two or more chains of domination must be merged to arrive at a single standard referent will not arise in the operation of a D-theory parser. Substantiation of these assumptions awaits the construction of a parser and a sizable grammar.)

This notion of "standard referent" means that *adding* predications to the (partial) analysis of a sentence may very well

*change* the standard referent of that analysis as viewed by the semantic interpreter. The key idea here is that from the point of view of semantics, the structure built by the parser may appear to change, but from the parser's point of view, the description remains indelible.

The situation we describe is not far from that which occurs as the usual case in the communication of descriptions of objects between individuals. Suppose Don says to you, standing before you wearing a brown tweed jacket, "My coat is too warm". The phrase "my coat" can refer to *any* coat that Don owns, yet you will undoubtedly take the phrase to refer to the brown tweed jacket. Given that descriptions are always necessarily partial, there must always be a conventional standard referent for a description. But now suppose that Don says "My blue coat is too warm". He merely *adds* "blue" to the phrase "my coat", but the set of possible referents changes, and in fact shrinks. More to the point, you will now take the referent of the phrase "my blue coat" to mean some blue coat or other which Don owns; i.e. *adding* to the description *changes* the standard referent.

The key notion here is that because descriptions are always underspecified, there must be some set of conventions for choosing the intended single referent out of the often large (and sometimes infinite) class of objects that any given description is true of. Thus, once we claim that the output of syntactic analysis is a description, it is not surprising that there must be some restrictive conventions to determine exactly what such a description refers to. Given this, the convention we assume seems a simple and natural one.

##### 5. On the Reanalysis of Indelible Structures

Another problematic class of constructions for deterministic tree-building theories are those for which it is argued that some kind of active *reanalysis* process must occur. For each of these constructions, there is linguistic evidence (of varied force) which suggests (recast in processing terms) that different syntactic structures must be assigned to that construction at different points during grammatical processing. In other words, it can be demonstrated that each of these constructions has properties which provide evidence for one particular structure at one stage of processing, while displaying properties which argue for a quite different structure at a later stage of processing. But if this reanalysis account is the correct account for *any* of these constructions, then the deterministic tree building theory must be wrong somewhere, for changing a structural analysis is the one thing that indelible systems cannot do, *ex hypothesi*.

One class of examples widely assumed to involve some kind of reanalysis is the class of verb complement structures which have so-called "pseudo-passives". These verbs seem to have two passive forms, one of which has an NP in subject position which serves in the same role as that served by the seeming object of the active form, while the other passive form seems to have an underlying prepositional object in subject position. For example, there are two passives which correspond to the active sentence (6.1), a "normal" passive (6.3), and a passive which seems to pull the object of "of" into subject position, namely (6.2).

- (6.1) Past owners had made a mess of the house.
- (6.2) The house had been made a mess of.
- (6.3) A mess had been made of the house.

One fairly common view is that the phrase "made a mess of" functions as a single idiomatic verb, so that "the house" in (6.1) and (6.2) can be simply viewed as the object of the verb "made a mess of". But then to account for (6.3), it must be assumed

that "made" is *first* treated as a normal verb with "a mess" as object. This means that either (6.3) has a different underlying syntactic structure than (6.1-2), or that the syntactic analysis assigned to the string "made of" (or perhaps "made <trace> of") *changes* after the passive is accounted for. To get a consistent syntactic analysis for these sentences, one can argue either that reanalysis *always* or *never* takes place. The position that we find most tenable, given the evidence, is that reanalysis *sometimes* takes place. (Of course, the fact that purely lexical accounts (see, e.g. [Bresnan 82]) seem plausible leaves the older tree-building theories on not entirely untenable ground.) But how can any reanalysis at all be reconciled with the determinism hypothesis?

Consider the analysis that a D-theory parser will have built up after having parsed "made a mess", but before noticing "of". At this point the parser should assign the sentence a non-idiomatic reading, with "a mess" the real object of "made". Some of the predications in the analysis will be

- (7.1) D(vp1, v1)
- (7.2) D(vp1, npl)

where vp1 is a vp node dominating "made" and npl is an np node dominating "a mess". (Note that in

- (8.1) The children made a mess, but then cleaned *it* up.

"it" refers to a mess, but that one cannot say

- (8.2) \*The children made a mess of their bedrooms,  
but then cleaned it up.

which seems to indicate that the phrase "a mess" is opaque to anaphoric reference in the idiomatic reading, and that therefore (8.1) is not idiomatic in the same sense.)

We assume here that the preposition "of" is lexically marked for the idiomatic verb "make a mess", i.e. it is lexically specified for the idiom, but it is not itself a part of the idiom. Evidence for this includes sentences like (9), in which the preposition *cannot* be reanalyzed into the verb, given D-theory, as we will see below.

- (9) Of what did the children make a mess?

From a parsing point of view, this means that the presence of the preposition "of" will serve as a *trigger* to the reanalysis of "make a mess", without being part of the reanalysed material itself. (Thanks to Chris Halverson for pointing out a problem caused by (9) for an earlier analysis.)

Returning to the analysis of (6.1), the preposition "of" triggers exactly such a reanalysis. Given D-theory, this can be effected simply by adding the additional predication (10) to (7.1-2) above:

- (10) D(v1, npl)

Given this new predication, the standard referent of the description now has npl directly dominated by v1, i.e. it is now part of the verb. And now when "a house" is noticed by the parser, it will be attached as the first NP *after* the verb v1, i.e. as its object. Once again, the predications (7.1-2) are not falsified by the additional predication; they remain indelibly true - npl remains dominated by vp1, although no longer *directly* dominated by it. But, to repeat the point, the parser is (blissfully) unaware of this notion; the standard referent is a notion meaningful only to semantics.

The analysis of (6.2) proceeds as follows: After parsing "made" as a verb and "a mess" as its object and noticing the trigger "of" sitting in the buffer, the parser will add an extra predication effecting just the same "reanalysis" as was done for (6.1). We assume that the passive rule inserts a trace either immediately after a verb, or *after the preposition immediately following a verb, if that preposition is lexically specified for that verb.* We will not argue for this analysis here; suffice it to say that this analysis is motivated by facts which also motivate recent somewhat similar analyses of passive, e.g. [Hornstein and Weinberg 81] and [Bresnan 82]. Given this analysis, the parser will now drop a passive trace for the subject "the house" into the buffer after the lexically specified preposition "of", and the parse will then move to completion. (One issue that remains open, though, is exactly how the parser knows not to drop the passive trace after "made". The solution to this particular problem must interact correctly with many such control problems involving passive. Resolving this entire set of issues in a consistent fashion awaits the pending implementation of a parser to serve as a tool in the investigation of these control issues.)

How is (6.3) parsed? Here we assume that the parser will drop a passive trace after the verb "made". Because we assume that the parser cannot access the binding of the trace, and therefore cannot access the lexical material "a mess", it must be the case that reanalysis will *not* take place in this case. While this asymmetry may seem unpleasant, we note that there is no evidence that syntactic reanalysis has taken place here. Instead, we assume that semantic processing will simply add an additional domination predicate after *it* notices the binding of the passive trace. Thus, the reanalysis here is semantic, not syntactic. (Note that there are other cases, e.g. right dislocation, where it is clear that additional domination predicates are added by post-syntactic processes. We believe that semantics can add domination predicates, but cannot construct new nodes.)

As an example of the kind of operation that is ruled out by D-theory, let us return to our assertion above that the preposition "of" cannot always be part of the idiomatic verb "make a mess". Consider (9) above. In this sentence, the analysis will include some assertions that "of" is dominated by a PP, which itself is dominated by COMP. But if an assertion is then added to this description asserting that "of" is also dominated by a verb node, then there is no consistent interpretation of this structure at all, since the COMP cannot dominate the verb node and the verb node cannot dominate the COMP. Put more simply, there is no way something can merely be "lowered" from a COMP node into the verb.

Another possibility similarly ruled out by D-theory is that in sentences like (6.1) there is initially a PP node which dominates both "of" and the NP "the house", but that "of" is reanalyzed into the idiomatic verb. For "of" to be dominated by a verb node, given that it is already dominated by the PP node, either the PP node must be dominated by the verb or the verb by the PP node, if the dominance relations are to be consistent. But it makes no sense for the PP node to have a standard referent where it immediately dominates only a verb and an NP, but no preposition. And if the verb dominates the PP, then the verb also dominates the NP which serves as the object of the VP, which is impossible.

In this sense, D-theory is clearly more restrictive than the theory of [Lasnik and Kupin 77], at least as interpreted by [Chomsky 81], where reanalysis is done by adding an additional monostring to the existing Restricted Phrase Marker and eliminating others.

In this case, the domination relations implied by the new analysis need not be consistent with those implicit in the pre-reanalysis RPM.

#### 6. Constraints on D-theory: a brief discussion

While we will not discuss this issue here at length, our current account of D-theory includes a set of stipulated constraints that further restrict where new domination predications can be added to a description. These constraints include the following: *The Rightmost Daughter Constraint*, that only the rightmost daughter of a node can be lowered under a sibling node at any given point in the parsing process; and *The No Crossover Constraint*, that no node can be lowered under a sibling which is not contiguous to it, and some others.

As viewed from the point of view of the standard referent, we believe that a D-theory parser will appear to operate, by and large, just like a tree building deterministic parser, until it creates some structure whose standard referent must be changed. From the parser's point of view, it will scan base templates left-to-right for the most part, initiating some in a top-down manner, some in a bottom-up manner, until it finds itself unable to fill the next template slot somehow or other. At this point some mechanism must decide what additional predications to add to allow the parser to proceed. The functional force of the stipulations discussed above is to severely restrict the range of possibilities that can be considered in such a situation. Indeed, we would be delighted if it turned out to be the case that the parser can never consider more than several possibilities at any point that such an operation will be performed.

It is particularly worthy of note that these two constraints interact to predict that the range of constructions that can be reanalyzed in the manner discussed in the last section is severely circumscribed, and that this prediction is borne out (see [Quirk, Greenbaum, Leech & Svartvik 72], §12.64). These two constraints together predict that verb reanalysis is possible only when a single constituent precedes the trigger for reanalysis:

Suppose that there were two constituents which preceded the trigger for reanalysis, i.e. that the order of constituents in the VP is

V C1 C2 T

where C1 and C2 are the two constituents, and T is the trigger. Then these two constituents would be attached to the VP whose head is V before T is encountered, causing the parser (before attaching T) to assert two new predications which would have the force of shifting the two constituents into the verb. But which predication could be parser add first? If it asserts that D(V, C1), this violates the Rightmost Daughter Constraint, because only C2 can be lowered under a sibling. But if the parser first asserts D(V, C2) then C2 crosses over C1, which is prohibited by the No Crossover Constraint. Therefore, only constituent can have been attached before the reanalysis occurs.

#### 7. A DETERMINISTIC APPROACH TO COORDINATION

We now turn from the consequences of expressing syntactic structure in terms of domination to the use of *names* within D-theory. As stated above, it is this use of names which really makes D-theory analyses descriptions, and not merely directed acyclic graphs. The power of naming can be demonstrated most clearly by investigating some implications of the use of names

for the representation of coordinate constructions, i.e. conjunction phenomena and the like.

### 7.1 The Problem of Coordinate Structure

Coordinate constructions are infamous for being highly ambiguous given only syntactic constraints; standard techniques for parsing coordinate structures, e.g. [Woods 73], are highly combinatoric, and it would seem inherent in the phenomenon that tree-building parsers must do extensive search to build all syntactically possible analyses. (See, e.g. the analysis of [Church & Patil 1982].)

One widely-used approach which eliminates much of this seemingly inherent search is to use extensive *semantic and pragmatic interaction* interleaved with the parsing process to quickly prune unpromising search paths. While Parsifal made use of exactly such interactions in other contexts, e.g. to correctly place prepositional phrases, such interactions seem to demand at least implicitly building syntactic structure which is discarded after some choice is made by higher-level cognitive components. Because this is counter to at least the spirit of the determinism hypothesis, it would be interesting if the syntactic analysis of coordinate structures could be made autonomous of higher-level processes.

There are more central problems for a deterministic analysis of conjunction, however. Techniques which make use of the look-ahead provided by buffering constituents can deterministically handle a perhaps surprising range of coordinate phenomena, as first demonstrated by the YAP parser [Church 80], but there appear to be fundamental limitations to what can be analyzed in this way. The central problem is that a tree building deterministic parser cannot examine the context necessary to determine what is conjoined to what without constructing nodes which may turn out to be spurious, given the (ultimate) correct analysis.

In what follows, we will illustrate each of these problems in more detail and sketch an approach to the analysis of coordinate structures which we believe can be extended to handle such structures deterministically and without semantic interaction.

### 7.2 Names and Appropriate Vagueness

Consider the problem of analyzing sentences like (11.1-2). These two sentences are identical at the level of preterminal symbols; they differ only in the particular lexical items chosen as nouns, with the schematic lexical structure indicated by (11.3). However, (11.1) has the favored reading that the apples, pears and cherries are all ripe and from local orchards, while in (11.2), only the cheese is ripe and only the cider is from local orchards. From this, it is clear that (11.1) is read as a conjunction of three nouns within one NP, while (11.2) is read as a conjunction of three individual NPs, with structures as indicated by (11.1a,2a). We assume here, crucially, that constituents in coordination are all attached to the same constituent; they can be thought of as "stacking" in a plane orthogonal to the standard referent, as [Chomsky 82] suggests. The conjunction itself is attached to the rightmost of the coordinate structures.

(11.1) They sell ripe apples, pears, and cherries from local orchards.

(11.1a) They sell [<sub>NP</sub> ripe [<sub>N</sub> apples], [<sub>N</sub> pears], [<sub>N</sub> and cherries] from local orchards].

(11.2) They sell ripe cheese, bread, and cider from local orchards.

(11.2a) They sell [<sub>NP</sub> ripe cheese], [<sub>NP</sub> bread], [<sub>NP</sub> and cherries] from local orchards].

(11.3) They sell ripe N1, N2, and N3 from local orchards.

Thus, it would seem that to determine the level at which the structures are conjoined requires much pragmatic knowledge about fruit, flowers and the like.

Note also that while (11.1-2) have particular *primary* readings, one needs to consider these sentences carefully to decide what the primary reading is. This is suggestive of the kind of syntactic *vagueness* that VanLehn argues characterizes many judgements of quantifier scope [VanLehn 78]. Note, however, that most evidence suggests that quantifier scope is not represented directly in syntactic structure, but is interpreted from that structure. For the readings of (11.1-2) to be vague in this way, the structures of (11.1a-2a) must be interpreted from syntactic structure, and not be part of it. It turns out that D-theory, coupled with the assumption that the parser does *not* interact with semantic and pragmatic processing, provides an account which is consistent with these intuitions.

But consider the D-theoretic analysis of (11.1); there are some surprises in store. Its representation will include predications like those of (12.1-8), where we are now careful to "unpack" informal names like "np1" to show that they consist of a content-free identifier and predications about the type of entity the identifier names.

(12.1) D(vp1, np1); VP(vp1); NP(np1)

(12.2) D(vp1, np2); NP(np2)

(12.3) D(vp1, np3); NP(np3)

(12.4) D(np1, ap1); D(ap1, adj1); ADJ(adj1)

(12.5) D(np1, n1); NOUN(n1)

(12.6) D(np2, n2); NOUN(n2)

(12.7) D(np3, n3); NOUN(n3)

(12.8) D(np3, pp1); D(pp1, prep1); PREP(prep1)

(12.9) adj1 < n1 < n2 < n3 < prep1

Here vp1 is the name of a node whose head is "sell", ap1 an adjective phrase dominating "ripe", and pp1 the PP "from local orchards." The analysis will also include predications about the left-to-right order of the terminal string, which has been informally represented in (12.9); "X < Y" is to be read "X is the left of Y". We indicate the order of nonterminals here only for the sake of brevity; we use

np1 < np2

as a shorthand for

D(np1, 'cheese'); D(np2, 'bread'); 'cheese' < 'bread'.

In particular, a D-theory analysis contains *no* explicit predications about left-right order of non-terminals.

But given only the predications in (12), what can be said about the identities of the nodes named np1, np2, and np3? Under this description, the descriptions of np1, np2 and np3 are compatible descriptions; they are *potentially descriptions of the same individual*. They are all dominated by vp1, and each is an

NP, so there is no conflict here. Each dominates a different noun, but several constituents of the same type can be dominated by the same node if they are in a coordinate structure (given the analysis of coordinate structures we assume) and if they are string adjacent. N1, n2 and n3 are string adjacent (given only (12)), so the fact that the nodes named np1, np2 and np3 dominate nouns which may turn out to be different does not make the descriptions of the NPs incompatible. (Indeed, if the nouns are viewed as a coordinate structure, then the structure of the nouns is the same as that of (11.1).) Furthermore, adj1 is immediately to the left of and ppl is immediately to the right of all the nouns, so these constituents could be dominated by the same single NP that might dominate n1, n2 and n3 as well. Thus there is no information here that can distinguish np1 from np2 from np3.

The fact that the conjunction "and" is dominated by np3 does not block the above analysis. The addition of one domination predicate leaves it dominated by n3 (as well as np3, of course), thereby making n1, n2 and n3 a perfect coordinate structure, and leaving no barrier to np1, np2 and np3 being co-referent.

But this means that the D-theory analysis of (11.1) has as standard referents *both* it and (11.2)! (This modifies our statement earlier in this paper about the uniqueness of the standard referent; we now must say that for each possible "stacking" of nodes, there is one standard referent.) For if np1, np2 and np3 corefer, then the analysis above shows that the structure described is *exactly* that of (11.2). There is also the possibility that just np1 and np2 corefer, given the above analysis, which yields a reading where np2 is an appositive to np1, with np1 and np3 coordinate structures (the structure of appositives is similar to that of coordinate structures, we assume); and the possibility that just np2 and np3 corefer, yielding a reading with np1 and np2 coordinate structures, and np3 in apposition to np2. (The fact that we use a simplified phrase structure here is not an important fact. The analysis goes through equally as well with a full X-bar theoretic phrase component; the story is just much longer.)

The upshot of this is that upon encountering constructions like (11), the parser can proceed by simply assuming that the structures are conjoined at the highest level possible, using different names for each of the potential highest level constituents. It can then analyze the (potentially) coordinate structures entirely independently of feedback from pragmatic and semantic knowledge sources. When higher cognitive processing of this description requires distinguishing at what level the structures are conjoined, pragmatics can be invoked where needed, but there need be no interaction with syntactic processes themselves. This is because, once again, it turns out if it is syntactically possible that structures should be conjoined at a lower level than that initially posited, the names of the potentially separate constituents simply can be viewed as aliases of the one node that does exist in the corresponding standard referent; in this case all predications about whatever node is named by the alias remain true, and thus once again no predications need to be revoked.

We now see how it is that D-theory gives an account of the intuition that the fine structure of coordinations is vague, in the sense of VanLehn. For we have seen that pragmatics does not need to determine whether (e.g.) all the fruits in (11.1) are ripe or not for the syntactic analysis to be completed deterministically, exactly because the D-theory analysis leaves all (and, we also claim, only) the syntactically correct

possibilities open. Thus the description given in (12) is *appropriately vague* between possible syntactic analyses of sentences like those schematized in (11.3). Thus, this new representation opens the way for a simple formal expression of the notion that some sentences may be *vague* in certain well defined ways, even though they are believed to be understood, and that this vagueness may not be resolved until a hearer's attention is called to the unresolved decision.

### 7.3 The Problem of Nodes That Aren't There.

While we can give only the briefest sketch here (the full story is quite long and complicated), exactly this use of names resolves yet another problem for the deterministic analysis of coordinate structures: To examine enough context (in the buffer) to decide what kind of structure is conjoined with what, a tree-building parser will often have to go out on a limb and posit the existence of nodes which may turn out not to exist after all. For example, if a tree-building parser has analyzed the inputs shown in (13.1-2) up to "worms" and has seen "and" and "frogs" in the

(13.1) Birds eat small worms and frogs eat small flies.

(13.2) Birds eat small worms and frogs.

buffer, it will need to posit that "frogs" is a full NP to check to see if the pattern

[conjunction] [NP] [verb]

is fulfilled, and thus if an S should be created with the NP as its head. But if the input is not as in (13.1), but as in (13.2), then positing the NP might be incorrect, because the correct analysis may be a noun-noun conjunction of "worms" and "frogs", (with the reading that birds eat worms and frogs, both of which are small).

Of course, there is a *second* problem here for a tree-building parser, namely that (13.2) has a second reading which is an "NP and NP" conjunction. As we have seen above, there is no corresponding problem for a D-theory parser, because if it merely posits an NP dominating "frogs", the structure which will result for (13.2) is *appropriately vague* between *both* the NP reading and the noun reading of "frogs" (i.e. between the readings where the frogs are just plain frogs and where the frogs are small.)

But the solution to the second problem for a D-theory parser is also a solution to the first! After seeing "and" and "frogs" in its buffer, a D-theory parser can simply posit an NP node dominating "frogs" and continue. If the input proceeds as in (13.1), then the parser will introduce an S node and assert that it dominates the new NP. This will make the descriptions of the NPs dominating "worms" and dominating "frogs" incompatible, i.e. this will assure that there really *are* two NPs in the standard referent. If the input proceeds as in (13.2), a D-theory parser will state that the node referred to by the new name is dominated by the previous VP, resulting in the structure described immediately above. To summarize, where a tree-building parser might be misled into creating a node which might not exist at all, there is no corresponding problem for a D-theory parser.

## 8. SUMMING UP: D-Theory on One Foot

This paper has described a new theory of natural language syntax and parsing which argues that the proper output of syntactic analysis is not a tree structure *per se*, but rather a *description* of such structures. Rather than constructing a tree, a natural language parser based on these ideas will construct a

single *description* which can be viewed as a partial description of each of a family of trees.

The two key ideas that we have presented here are:

(1) An analysis of a syntactic structure consists primarily of predications of the form "node X dominates node Y", and not the more traditional "node X immediately dominates node Y"; syntactic analysis never says more than that node X is *somewhere* above node Y.

(2) Because this is a description, two *names* used to refer to syntactic structures can always co-refer if their descriptions are compatible, and furthermore, it is impossible to block the possibility of coreference if the descriptions are compatible.

These two ideas, taken together, imply that during the process of analyzing the structure of a given utterance, merely *adding* to the emerging description may *change* the set of trees ultimately described (just as adding "honest" to the phrase "all politicians" may radically change the set described). We have also sketched some implications of this theory that not only suggest a new analysis of coordinate structures, but also suggest that coordinate structures might be much easier to analyze than current parsing techniques would suggest.

We are currently working to flesh out the analyses presented above. We are also working on an analysis of gapping and elision phenomena which seems to fall naturally out of this framework. This new analysis is surprising in that it makes crucial use of descriptions even less fully specified than those we have discussed in this paper, by using the notations we have introduced here to fuller advantage. These emerging analyses move yet further away from the traditional view of either trees or phrase markers as an appropriate framework for expressing syntactic generalizations.

## 9. References

- Berwick, R. (1982) *Locality Principles and the Acquisition of Syntactic Knowledge*, MIT PhD thesis.
- Bresnan, J. (1982) "The Passive in Lexical Theory," in J. Bresnan (ed.) *The Mental Representation of Grammatical Relations*, MIT Press, pp. 3-86.
- Chomsky, N. (1981) *Lectures on Government and Binding*, Foris Publications.
- Chomsky, N. (1982) *Some Concepts and Consequences of the Theory of Government and Binding*, MIT Press.
- Church, K. (1980) "On Memory Limitations in Natural Language Processing," MIT Masters thesis, MIT/LCS/TR-245.
- Church, K. and R. Patil (1982) "Coping with Syntactic Ambiguity or How to Put the Block in the Box on the Table," MIT/LCS/TM-216.
- Hindle, D. (1983) "Deterministic Parsing of Syntactic Non-fluencies," this proceedings.
- Hornstein, N. and A. Weinberg (1981) "Case Theory and Preposition Stranding," *Linguistic Inquiry*, 12.1, pp. 55-91.
- Lasnik, H. and J. Kupin (1977) "A Restrictive Theory of Transformational Grammar," *Theoretical Linguistics*, vol. 4, pp. 173-196.
- McDonald, D. (1983) "Natural Language Generation as a Computational Problem: an Introduction," in M. Brady and R. Berwick (eds.) *Computational Models of Discourse*, MIT Press, pp. 209-265.
- Marcus, M. (1980) *A Theory of Syntactic Recognition for Natural Language*, MIT Press.
- Quirk, R., S. Greenbaum, G. Leech and J. Svartik (1972) *A Grammar of Contemporary English*, Longman.
- Shipman, D. (1979) "Phrase Structure Rules for Parsifal", MIT AI Lab Working Paper 182
- Shipman, D. and M. Marcus (1979) "Towards Minimal Data Structures for Deterministic Parsing," IJCAI79.
- VanLehn, K.A. (1978) "Determining the Scope of English Quantifiers", MIT AI-TR-483.
- Woods, W.A. (1973). "An Experimental Parsing System for Transition Network Grammars." in R. Rustin, ed., *Natural Language Processing*, Algorithmics Press.