11

Generalized Monotonicity for Reanalysis Models*

Patrick Sturt & Matthew W. Crocker

Abstract

A common assumption in psycholinguistic theory is that reanalysis is constrained by a preference to preserve certain aspects of the representation built in response to previous input. In this chapter, we discuss this notion of representation-preservation in the wider context of models of reanalysis as a whole, and point out that in order to define a representation-preserving constraint on reanalysis, we must specify not only which aspects of representation should be preserved, but what is meant by the notion of preservation. We propose that the appropriate notion of preservation is that which is assumed in monotonic models of parsing, where structural relations between linguistic elements are updated totally non-destructively from state to state. Previous monotonic theories of parsing have limited themselves to consideration of phrase structure representations. In contrast, we propose a general framework within which one may formulate models which apply the same notion of preservation to other representation types. The framework is discussed with reference to a model which preserves thematic structure.

* Much of the material in this chapter was presented at seminars given at the University of Freiburg, Germany, and the University of Torino, Italy, during July and September 1996. We would like to thank the seminar participants for their instructive comments, especially Barbara Hemforth, Lars Konieczny, Leonardo Lesmo, Vincenzo Lombardo, and Gerhard Strube. We would also like to thank Suzanne Stevenson and an anonymous reviewer for their comments and suggestions which have greatly improved the quality of this chapter. Any remaining errors are our own. The research reported here was supported by the award of ESRC studentship No. R00429334338 to the first author.

J.D. Fodor and F. Ferreira (eds.), Reanalysis in Sentence Processing, 365–400.
© 1998 *Kluwer Academic Publishers.*

1 Introduction

One of the goals of reanalysis models is to predict differential costs associated with different types of revision. In attempting to achieve this goal, a number of models have been proposed which we call representation-preserving (Pritchett, 1988; Weinberg, 1993; Gorrell, 1995; Sturt & Crocker, 1996). These models constrain revision by appealing to a preference encoded in the processor to preserve certain features of the representation built up to the point when reanalysis occurs. Representation-preserving models contrast with theories which constrain reanalysis directly in terms of limitations on working memory, inferential ability, and other computational resources.

Our goal in this chapter is not to argue that representation-preserving models are better than alternative accounts—in fact, we have no doubt that representation, resource boundedness, and many other factors will turn out to be relevant to reanalysis. Instead, we have settled on two less ambitious goals. Firstly, we aim to situate the notion of representation-preservation within the context of reanalysis models in general. Secondly, we aim to provide a general framework within which precisely formulated representation-preserving models can be built and evaluated. Specifically, we will propose a general framework for building monotonic models, a particular subclass of representation-preserving models. The idea of monotonicity provides us with a notion of 'preservation' which can be applied to different types of representation.

2 What Is Reanalysis?

2.1 A Definition of 'Reanalysis'

Loosely speaking, 'reanalysis' may be taken to mean 'discarding one reading of an ambiguity in favor of another. However, exactly what counts as 'reanalysis' varies considerably from model to model. For example, in Gibson's (1991) restricted parallel system, 'reanalysis' refers only to recovery from 'parsing breakdown' situations, which are predicted to occur exactly in those cases where a previously discarded parse has to be re-introduced. The re-ordering of alternatives still under parallel consideration, which presumably must occur at some level, is not regarded as reanalysis. Similarly, in 'constraint satisfaction' models (e.g., MacDonald *et al.*, 1994), which are characterized by the parallel competitive activation of multiple alternatives, minor changes in relative activation levels are not usually classed as 'reanalysis', but some researchers working in this tradition (Trueswell *et al.*, 1993) do not deny the existence of 'true garden path' effects, which are indeed associated with 'reanalysis'. Presumably, reanalysis in a constraint satisfaction model corresponds to the re-activation of an alternative whose activation has dropped below some threshold to a level close to zero.

By contrast, within a serial framework, the Garden Path model (Frazier, 1978; Frazier & Rayner, 1982) assumes that normal processing is accompanied by any number of minor reanalyses, as initial attachments based on restricted grammatical knowledge are discarded by higher level processing modules. However, even within the Garden Path tradition, it is not always clear which varieties of structural change count as reanalysis. For example, in models influenced by the Garden Path tradition a distinction is often made between the attachment of postmodifiers of verbs, which are incorporated into the representation via sister adjunction, and the attachment of postmodifiers of nouns, which are incorporated via Chomsky adjunction (see Clifton *et al.*, 1991, for an example of this distinction). Adams (1995) uses this distinction to argue that the attachment of NP modifiers is, in some sense, more destructive than the attachment of VP modifiers, and therefore we should presumably see attachment of NP modifiers as involving some form of reanalysis, and attachment of VP modifiers as not.

Given this wealth of subtly different nuances associated with the term, we propose to clarify what *we* mean by 'reanalysis', before discussing the notion any further. We believe that it is useful to view the processing of a string as a series of states, where each word induces a transition from one state to the next. For the sake of simplicity of definition, we assume a processing model in which a single preferred analysis is always unambiguously available at each state.[1] Now, we assume that whatever syntactic representation is used in the model, it will include information about grammatical dependencies, which could be modelled using licensing relations, coindexation, structure sharing, unification, argument cancellation or any number of alternative means, depending on the variety of grammar that the processing model uses.[2] We will use the

[1] Our definition does not necessarily exclude parallel models; it merely insists that, in the case of a parallel model, of the alternative analyses available, one has a higher *value* (which could be determined by activation, processing load units, structural preferences, or other means) than all the others.

[2] We see dependencies as separate from the structural conditions that license them; for example, we would include "X is a modifier of Y", or "X is a complement of Y" as a dependency relation, but not "X is a daughter of Y", or "X is a sister of Y".

abbreviation Dep^S to stand for the following proposition: "*Dep* is the set of dependencies holding in the preferred analysis at state *S*". We will abbreviate the notion of 'a transition from state *S* to state *T*' as $S \Rightarrow T$. Now, we define reanalysis as follows:[3]

A transition $S \Rightarrow T$ involves reanalysis iff $Dep^S \not\subseteq Dep^T$.

From the definition, it can be seen that by 'reanalysis' we mean 'breaking at least one dependency'. From this it follows that, for example, attaching a postmodifier is not a case of reanalysis, since it involves adding rather than breaking a dependency. On the other hand, in a ranked parallel system, changing the relative preference order of analyses in such a way that the preferred analysis at *S* is no longer the preferred analysis at *T* will count as reanalysis, if there are any dependencies at *S* which do not hold at *T*.

2.2 The Nature of Reanalysis

Until recently, the phenomenon of reanalysis has been of concern to experimental psycholinguistics chiefly for the light it can shed on preferences for initial attachment. For example, in the following sentence, the detection of processing disruption at the word *was*, in an on-line task, may be taken as evidence that the processor has decided to attach *the cake* as a direct object of *saw*, rather than as the subject of an embedded clause, on the assumption that the disruption is the result of reanalysis.

(1) The wedding guests saw the cake was still being decorated.

However, reanalysis has rarely been treated as anything more than a diagnostic test for mis-attachment. Experimental research on ambiguity resolution has shown us *when* reanalysis occurs, but there have so far been very few experimental studies examining the question of *how* reanalysis occurs, or what preferences exist in reanalysis (but for examples of experimental studies which examine reanalysis for its own sake see Frazier & Rayner, 1982; Rayner *et al.*, 1983; Warner & Glass, 1987; Ferreira & Henderson, 1991; Sturt, in preparation).

---

[3] Note that the definition makes the simplifying assumption that it will always be possible to identify one particular transition where the preference for one alternative is replaced by the preference for another.

In the remainder of this section, we give a general discussion of the abstract problem of reanalysis, as compared with initial attachment. The aim of this discussion is simply to point out that reanalysis is an inherently more complex process than initial attachment, and that this complexity has consequences both for the human language processing system that has to perform the task, and for the models which psycholinguists need to build in order to understand the phenomenon. To understand the complexity of reanalysis as compared to initial attachment let us restrict our attention to a maximally incremental processing system, where each word is incorporated into single totally connected syntactic representation before the processor moves on to consider the next word (see Sturt & Crocker, 1996, and Lombardo, this volume, for examples of processing systems which has this property). Consider the problem of incorporating a new word into the representation, as informally illustrated in Figure 1.

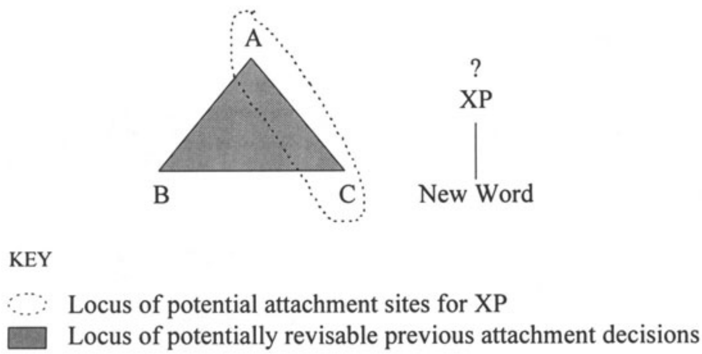

FIGURE 1: Incorporating a new word into the tree.

Assuming that the processor attaches in accordance with the theory of trees, then the locus of initial attachment possibilities for the projection of the new word corresponds with the right frontier of the tree built so far—that is, depending on the processing system used, the constituent XP could be attached as a daughter of any of the nodes on the path from C to A, or alternatively, the root A could be attached as the left daughter of XP (see Stevenson, 1994a, 1994b and Sturt & Crocker, 1996, for particularly clear examples of 'right-frontier based' attachment). However, no other attachments are possible if the processor wants to obey the theory of trees, retain a connected structure, and preserve all dependencies.

Now consider the possibilities for reanalysis. It may be the case that XP cannot find a felicitous attachment site on the right frontier. If this is true, then, according to most models, the processor will attempt to revise a previous decision. Recall example (1), repeated below:

(1) The wedding guests saw the cake was still being decorated.

Assume that *the cake* is initially attached as the direct object of *saw*, and that the processor attempts to reanalyze this decision at *was*. Assume also that the processor is serial to the extent that it does not have the sentential complement reading available.[4] Then, the reanalysis must involve at least the following operations:

1. *the cake* must be 'detached' from its current position as the direct object of *saw* (we will call this position the *detachment site*).

2. *the cake* must be re-attached as the subject of *was* (we will call this position the *target site*).

3. The (projection of the) new word must be attached into a position in the representation (we will call this position the *attachment site* of the new word).

Items 2 and 3 are each tasks comparable in complexity to standard initial attachment. Thus, in general, reanalysis must involve at least as much search as initial attachment. Note also that, while in this particular case the detachment site is on the right frontier of the tree at the point when reanalysis occurs (and incidentally, this is part of the reason why both Sturt & Crocker, 1996, and Stevenson, 1994a, predict example (1) to be unproblematic for the human language processor), in the general case, the *a priori* locus of candidate detachment sites corresponds to the entire tree, as illustrated in Figure 1, and not just the right frontier. This means that, generally speaking, reanalysis is likely to involve considerably more search than initial attachment.

Of course, all this is a statement of the obvious. However, we think it is a good idea to reflect seriously on the nature of reanalysis, particularly as, from reading the literature on psycholinguistic experiments, we could be forgiven for assuming that reanalysis is a much simpler phenomenon than it really is. Most experiments that are written to test hypotheses about ambiguity resolution are

4 In a parallel system, such as that of Gibson (1991), the processor simply has to discard the structure corresponding to the direct object reading from its working set, and attach *was* into the structure corresponding to the sentential complement reading.

carefully controlled to contain only the one ambiguity under investigation. This means that, if evidence for reanalysis is found, the researcher can be relatively sure that this reanalysis involves revision of the single choice point involved. However, the sentences which readers have to process in real life are unlikely to conform to this ideal. In fact, at any given point where reanalysis is necessary, there may be any number of candidate choice points—corresponding to suitable detachment sites throughout the tree representation—which could be revised in any number of ways. Given this, one of the major aims of a theory of reanalysis must be to make predictions about how the processor deals with reanalysis ambiguity. That is to say, when the processor is faced with a choice of more than one possibility for reanalysis, we want to know what, if anything, influences the choice of detachment site and target site; (see Sturt & Crocker, 1996, and Sturt, in preparation, for a discussion of reanalysis ambiguity).

As far as theoretical models of reanalysis are concerned, we believe that the complexity of reanalysis motivates models that are computationally implementable. In models of initial attachment, where the task is to predict how the current word incorporates itself into the right edge of the current partial phrase marker, it is often very easy to ascertain the predictions of models. One of the reasons why the principles of Minimal Attachment and Late Closure (Frazier, 1978) have been so influential (and so acrimoniously debated) is that, in any given situation, it is usually clear what predictions these principles make. By contrast, when we are attempting to model the more complex process of reanalysis, we are much more likely to miss loopholes, where the model either over-predicts or under-predicts, unless we have a foolproof way of testing the consequences of a model. Furthermore, given the complex nature of the reanalysis problem, we would expect to find that the human language processing system, which has to perform this daunting task, imposes constraints on the search involved in reanalysis. Given that the number of analyses for a string grows exponentially in the length of the input, it is unlikely that the processor considers every alternative when the current analysis is discarded. This leads us to consider a further aim of reanalysis models—to define the exact nature of the constraints which the human language processor imposes on reanalysis search. It is this aim that has occupied most of the attention of researchers working on reanalysis, and in the next section we will give a brief review of how these constraints have been defined.

3 Accounting for Constraints on Reanalysis

As we stated above, recent models of reanalysis have concentrated on the aim of accounting for differential costs associated with different reanalyses by postulating various constraints on the possibilities for revision. In this section we will take a very brief look at some of these proposals, and identify two dimensions within which such constraints may vary; firstly, models vary in the extent to which they constrain the *diagnosis* versus the *cure* of garden paths (terms we have borrowed from Fodor & Inoue, 1994). Secondly, models vary in the nature of the computational limitations which are assumed to be the ultimate motivation for the constraints. Specifically, while some models attribute their constraints directly to limits in the computational resources available to the processor (such as bounded working memory, for example), others express the computational limitations less directly, by appealing to a preference encoded in the processor to preserve certain core aspects of representation. We will call the former class resource bounded and the latter class representation-preserving models. It is with the problem of defining representation-preserving models that the remaining sections (4-7) will concerned.

3.1 Diagnosis versus Cure

As we mentioned above, there is still not a great deal of experimental evidence available on the nature of reanalysis, and many researchers have relied on intuitive distinctions, between degrees of difficulty associated with different types of reanalysis. An example of such a distinction is that between 'conscious' garden paths (which, typically are claimed to cause noticeable processing difficulty) and 'unconscious' garden paths (which intuitively cause only a minor disruption to processing). Contrast (1), repeated below, which would on most accounts be classed as an unconscious, or easy garden path, with (2), which on many would be classed a conscious, or difficult garden path (c.f. Pritchett, 1988, 1992; Gorrell, 1995).

(1) The wedding guests saw the cake was still being decorated.

(2) While the wedding guests ate the cake was still being decorated.

Using the intuitive distinctions such as these as a clue, researchers have attempted to define constraints on reanalysis which exclude garden paths from which recovery is difficult, but include easy garden paths. Recall from the previous section that reanalysis typically involves finding a detachment site, finding a target site to re-attach the detached constituent, and finding a new attachment site for (the projection of) the new word. The constraints in the following models can be defined in terms of one or more of these notions:

> *Pritchett (1988)*: The target site must be in the same *theta-domain* as the detachment site.[5]
>
> *Pritchett (1992)*: The detachment site must govern or dominate the target site.
>
> *Gorrell (1995)*: The detachment site must dominate the target site. (N.B., this is a consequence of the indelibility of dominance and precedence relations, rather than a stipulation.)
>
> *Lewis (1993)*: The detachment site must be local to the new attachment site of the incoming word (where the notion of local corresponds to 'within the same maximal projection').
>
> *Fodor & Inoue (this volume)*: The detachment site must be accessible to the new attachment site of the incoming word through a chain of grammatical dependencies.
>
> *Sturt & Crocker (1996)*: The detachment site must be on the right frontier of the tree description built so far. The detachment site, reattachment site and new attachment site must be local to each other, as defined by the *tree-lowering* parsing operation. (The result is a system which obeys Gorrell's constraints.)
>
> *Stevenson (1994a, 1994b)*: Similar constraints to Sturt & Crocker (1996), but derived from the architecture of a hybrid Competitive Activation model.

In addition to the above, a *recency preference* has been proposed for the detachment site. This is captured, for example, in Stevenson (1994a, 1994b),

[5] α is in the γ θ-domain of β iff α receives the γ θ-role from β or α is dominated by a constituent that receives the γ θ-role from α (Pritchett, 1988, p. 555).

via decay of activation, and in Sturt & Crocker (1996) via an explicit search strategy.[6]

It can be seen from the above that the models differ with respect to the relative importance attached to the *detachment site* and the *target site*. For example, Lewis (1993) and Fodor & Inoue (this volume) impose no constraints on the *target site*, other than the constraints that apply to initial attachment, but impose strong constraints on the *detachment site* with respect to the position where the projection of the new word is attached. In terms of search, this implies that the search for the attachment error that is the ultimate cause of the need for reanalysis is more tightly constrained than the search for an alternative attachment. On the other hand, Pritchett (1988, 1992) imposes no constraints on the position of the detachment site itself, but does impose constraints on the position of the *target site* relative to the detachment site. Pritchett's model implies that the search for an alternative attachment is more tightly constrained than the search for the error. We can call models of the former kind, which constrain the search for the error more heavily than the search for alternative attachments, 'diagnosis-constrained models', and models of the latter kind, which constrain the search for alternative attachments more heavily than the search for the error, 'cure-constrained models'. Of course, many models contrain elements of both diagnosis and cure; for example, Sturt & Crocker (1996) impose a right-frontier accessibility constraint on the search for a *detachment site*, and to this extent is *diagnosis*-constrained. On the other hand, the model also strongly constrains the space of possible *target positions*, and to this extent it is also *cure*-constrained. This shows that diagnosis-constrained models and cure-constrained models are not mutually exclusive. Rather, diagnosis and cure should be seen as two aspects of search which, in any particular processing model, may independently be constrained to a greater or lesser extent. To the extent that a model can be seen to constrain diagnosis at the expense of cure, that model can be called diagnosis-constrained, and vice versa.

3.2 Resource-Boundedness versus Representation-Preservation

It is likely that the ultimate motivation for the constraints on reanalysis is computational. Given the combinatorial explosion of ambiguity in natural language, and the *a priori* complexity of search involved in reanalysis which we

---

6 Ferreira & Henderson (1991) show evidence that this recency preference should be sensitive to the point at which the *head* of the detachment site is read.

discussed in the previous section, it is unlikely that all alternatives to an initially chosen analysis are equally available to the processor. However, models vary with respect to how these limitations manifest themselves in processing behavior. Specifically, while some models express the computational limitations directly in terms of, for example, bounds on working memory, or constraints on inferential ability, others propose that the processor acts in such a way as to preserve, to as great an extent as possible, the representation that it has built up to the point where reanalysis occurs. We will call models of the first type *resource-bounded* models, and models of the second type *representation-preserving* models.

Gibson (1991) is a clear example of a *resource-bounded* model, since in this model, the bounds on computational resources play a direct role in the constraints imposed on reanalysis. In Gibson's model, the processor maintains multiple analyses in parallel, but, to avoid memory overload, prunes away analyses which exceed a certain processing cost relative to the preferred analysis. The necessity for re-introducing a pruned structure is predicted to correspond to the conscious garden path effect in human processing. In this system, the constraint on reanalysis is clearly motivated by working memory limitations. Conserving working memory also plays a key part in the more serial model of Lewis (1993), where attachment sites waiting to be filled are pruned from the parser's memory under certain conditions.

By contrast, *representation-preserving* models attribute to the processor an unwillingness to destroy representation, or propose that there is a difficulty involved in doing so. The ultimate motivation for this may very well be limitation in computational resources; after all, representations presumably require computational resources to build and modify. However, the link between bounds on computational resources and constraints on processing is much less direct in these models than in the resource bounded models. Instead, the claim is that there is something about destroying structure which is intrinsically dispreferred. The model described in Fodor & Inoue (this volume) contrasts strongly with a representation-preserving model, since it claims that there is no difficulty involved in destroying structure *per se*, just so long as the limited inferential capabilities of the processor allow it to determine *which* structure needs to be destroyed.

An example of a principle which is clearly motivated by representation-preservation is Frazier & Clifton's (this volume) *Minimal Revisions* Principle (c.f. Frazier, 1990a):

> *Minimal Revisions*: Don't make an unnecessary revision. When revision is necessary, make the minimal revision consistent with the error signal, maintaining as much of the already assigned structure and interpretation as possible.

Of course, there are various ways in which a representation-preserving principle like this could be interpreted, depending on precisely what is meant by 'minimal revision', and 'maintaining as much of the already assigned structure and interpretation as possible'. The question is, what general metric would be appropriate to quantify destructiveness to representations? One possibility, which would be consistent with our dependency-based definition of reanalysis, is to count the number of dependencies which have to be broken by the revision in question. This metric was behind the explanation we suggested for differences, described in Sturt & Crocker (1996), between reanalysis preferences in head-initial versus head-final languages, and is also consistent with the data which Frazier & Clifton discuss in their chapter of this volume.

However, the simple *number* of dependencies which are broken is unlikely to provide a *complete* answer to the question—in many cases, both 'hard' and 'easy' garden paths involve breaking only one dependency. In the reputedly 'easy' garden path in (1) repeated below, one dependency (between *saw* and *the cake* is broken, and the same is true of the 'hard' garden path in (2), where the broken dependency is between *ate* and *the cake.*

(1) The wedding guests saw the cake was still being decorated.

(2) While the wedding guests ate the cake was still being decorated.

Furthermore, there is evidence that the preferred interpretation of certain relative clause attachment ambiguities in Italian (De Vincenzi & Job, 1995) and Japanese (Kamide & Mitchell, 1997) may be derived via the breaking of one dependency even in the absence of any syntactic or semantic error signal (see Sturt & Crocker, to appear, for discussion).

This leads us to another side to this question, which is the consideration that there might be certain *aspects* of representation that are, in some sense, more sacred to the processor than others, and which the processor is therefore more reluctant to destroy. This has been the basic assumption behind many of the reanalysis models that have been recently proposed. Thus, Pritchett's (1988) *Theta Reanalysis Constraint* encodes a preference to preserve certain aspects of *thematic* structure, as defined by his 'thematic domains'. Meanwhile, Gorrell

(1995) and Weinberg (1993, 1995), encode a preference to preserve (description based) *constituent* structure representations. On the other hand, Bader (1996) argues that *multiple* representation types (a mixture of syntactic, lexical-morphological and prosodic representations) should be preserved.

As with the diagnosis/cure distinction, resource-boundedness and representation-preservation are not mutually exclusive characteristics. That is, a processor may be resource bounded in some way, say, due to limitations on working memory, but may use representation-preservation as a heuristic for keeping the computational cost of reanalysis within these bounds. Similarly, a processor may be primarily motivated to preserve representations, but be further limited by rather modest inference capabilities.

4 The Monotonicity Framework

As we mentioned in the previous section, the assumption of a representation-preserving processor leaves open two questions:

1. How should we formalize the notion of preservation?

2. Which aspects of representation should be preserved?

As an answer to the first question, we will argue that the most natural notion of 'preservation' is provided by the *monotonicity* paradigm, by which we mean the family of parsing models descended from Description theory (henceforth D-theory, Marcus *et al.*, 1983). However, D-theory has hitherto been applied to purely *constituent structure* representations. Therefore, we propose to generalize the paradigm in a way such that it can be applied to other representation types. This yields a framework within which models can be built to test proposed answers to the second question (see Sturt & Crocker, to appear, for a model which uses the generalized monotonicity framework to define monotonicity at a *thematic* level of representation).

What, precisely, do we mean by 'representation-preservation'? We can regard a representation as a set of (possibly binary) linguistic or structural relations. Now, recalling that a parse is viewed as a series of transitions from one state to the next, the notion of 'preservation' implies that, once a relation has been added to the representation, it is not removed. Monotonic models (Gorrell, 1995; Weinberg, 1993; Sturt & Crocker, 1996) use a form of representation called a *description* which can be 'preserved' in this manner, while tolerating certain classes of *reanalysis*. Recalling our definition of reanalysis, this means that, while at the level of linguistic *dependencies*, a relation holding at some state *S* may fail to hold at the next state *T*, at the level of the *description*, all relations holding at state *S* also hold at state *T*.

4.1 The Essence of Monotonicity

We can think of the dynamics of processing in monotonic models on three levels, as illustrated in Figure 2. On the first level, the *description*, is a set of statements describing certain aspects of the syntactic representation. Following Marcus *et al.* (1983), the description standardly consists of a set of *dominance* relations; for example, at State 1, the description consists of the statement "*a* dominates *b*". As we shall see, the fact that the dominance relation is *transitive* (i.e., if *X* dominates *Y* and *Y* dominates *Z*, then *X* dominates *Z*) gives D-theory its flexibility and power. At any state, the description can be interpreted as an infinite set of 'possible trees', which we have depicted in the middle level of the diagram, corresponding to all the trees which are consistent with the dominance statements, along with the axioms of the theory of trees. At any point in processing, a minimal 'default' tree can be generated from the description. In the simplest case, the 'default' tree at some state *S* is simply the result of assuming all dominance relations in the description at *S* encode *immediate* dominance, and assuming all nodes not mentioned in the description do not exist, that is, assuming the description is a 'closed world'. If we take state 1 in Figure 2, for example, the 'possible trees' comprise the infinite set of trees corresponding to the statement "*a* dominates *b*", including trees in which other nodes exist on the domination path between *a* and *b*. However, at this state, the default tree contains only the unique member of the set of possible trees for which there is evidence, namely, the tree in which *a* dominates *b*, and no other nodes exist; in other words, where *a directly* dominates *b*.

The constraint of monotonicity says that the description must be preserved—that is, once a dominance relation has been added, it cannot be deleted. This means that the description can be seen as a *monotonically increasing* set, where the description at each state will be a subset of all descriptions at subsequent states. Conversely, the 'possible trees' comprise a *monotonically decreasing* set, in that adding new relations to the description removes 'possible trees', but never adds them. Consider the effect of adding the new relation *dom(a,c)* at state 2. This has the effect of removing from the set of possible trees all structures in which *c* does not exist, for example, or all trees in which *c* dominates *a*. However, all the trees which are compatible with the description at state 2 are also compatible at state 1.

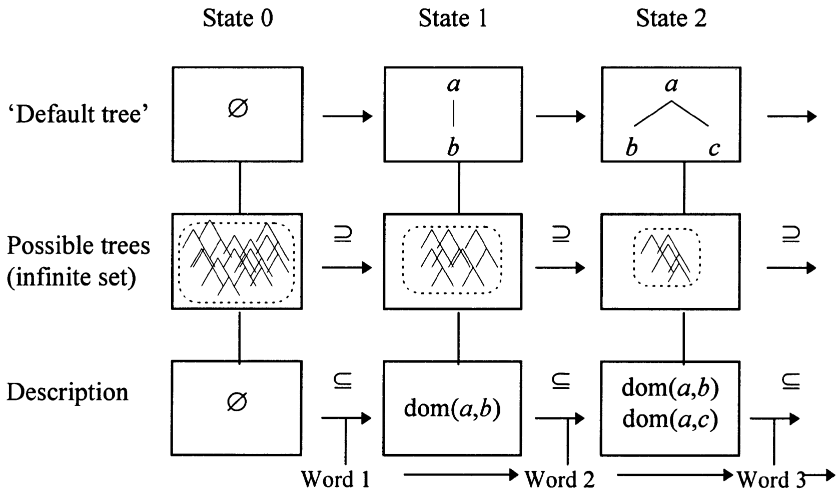

FIGURE 2: The essence of monotonicity.

In this sense, the monotonicity framework shares common features with the classical *cohort* model of word recognition (Marslen-Wilson, 1987). The 'cohort' in Marslen-Wilson's system is simply the set, at a given state of word recognition, of all word candidates consistent with the string of phonemes processed up to that point. As subsequent phonemes are recognized, word candidates which become inconsistent with the input are removed. If it subsequently becomes necessary to reintroduce a candidate that has been eliminated at a previous state, then a processing cost is predicted. Thus, the cohort is a monotonically decreasing set, and is equivalent to the level of 'possible trees' in the monotonicity framework. Of course, one of the major differences between syntax and word recognition is that, unlike the set of word candidates in a cohort, the set of continuations of a *syntactic* prefix is infinite, and thus impossible to enumerate.[7] The *description* in a monotonic processing model may therefore be viewed as an underspecified representation which, at

7 This distinction does not hold in languages which exhibit cyclical agglutinative morphology, where the number of word candidates for a prefix is potentially infinite.

any state *S*, encodes the infinite locus set of possible changes which the representation may undergo as new material is processed subsequent to state *S*.

4.2 Infinite Local Ambiguity

Infinite local ambiguity may arise in circumstances such as the following. Imagine the following string of words has been processed:

(3) I saw the man...

Now, let us say that the minimal 'default' interpretation contains the following information:

(4) (State S:) I $[_{VP}$ saw $[_{NP_x}$ the man]]

Because of the transitivity of the dominance relation, the set of 'possible trees' will include trees in which various numbers of nodes intervene on the domination path between VP and NP_x. However, the minimal default tree will include only the single tree indicated above, in which VP *immediately* dominates NP_x. Now imagine that the input proceeds so that at state *T*, the default tree includes the following information:

(5) (State T:) I $[_{VP}$ saw $[_{NP_y}$ $[_{NP_x}$ the man]'s wife]]

By the time of state *T*, the default tree at state *S* has been removed from the set of possible trees, and, since the dependency between the verb *saw* and NP_x has also been removed, we class this change as a *reanalysis*. However, the new minimal interpretation at state *T* has existed in the set of possible trees at all previous states, including that of state *S*. In fact, the input could continue indefinitely in the same manner, and the set of possible trees would at each stage be monotonically *narrowed down*, with the description simultaneously monotonically increasing.

(6) I saw the man's wife's friend's sister's...

The type of local ambiguity exemplified in (3)-(6) is more pervasive than might be thought. We can expect to find similar examples in all cases where right branching and left-branching structures meet—a situation which, as pointed out by Inoue & Fodor (1995), is extremely pervasive in head-final languages such as Japanese.

It should be clear at this point that the D-theory formalism is extremely powerful, in that it allows a compact encoding of an infinite number of possible trees. This property makes the monotonicity paradigm an interesting alternative to parallelism. In contrast to, for example, ranked parallel models (e.g., Gibson, 1991), in which alternative analyses are explicitly built, and structures pruned when they exhibit a particular relative or absolute processing cost, in the monotonic framework, multiple analyses are simply implicit in the description, and the 'pruning' is simply the by-product of adding new relations to the description.

4.3 Problems with Monotonicity

Despite the conceptual attraction of the monotonicity framework, we have argued (Sturt & Crocker, to appear), that existing models that have been proposed within the framework have been too inflexible to provide an account of the available data. One of the examples which we gave in that paper concerns the attachment of modifiers. Consider the following well-known sentence from Cuetos & Mitchell (1988).

(7) The journalist interviewed the daughter of the colonel who had had the accident.

There are two possible interpretations for sentences such as (7), corresponding to the attachment of the relative clause as a modifier of *daughter* on the one hand (high attachment) and as a modifier of *colonel* on the other (low attachment). Notice that, if the relative clause is initially attached in the low site, and subsequently reanalyzed to the high site, then, at the level of constituent structure, this reanalysis will break a dominance relation. This is because, in the low site, it will be dominated by the NP headed by *colonel*, while reanalysis to the high site will remove the relative clause to a position where it is no longer dominated by this NP. Thus, standard, constituent structure based D-theory models make a clear prediction that reanalysis of this type should cause conscious processing difficulty.

However, recent experimental evidence from Italian (De Vincenzi & Job, 1993, 1995) and Japanese (Kamide & Mitchell, 1997), suggests that, in sentences similar to (7) in these languages, although the initial, on-line preference is for for the *low* site, the final, off-line preference is for the *high* site. Specifically, questionnaire studies in both Italian (De Vincenzi & Job, 1993) and Japanese (Kamide & Mitchell, 1997) show a high attachment preference for globally ambiguous sentences similar to (7). However, self-paced reading studies in both languages (De Vincenzi & Job, 1993, 1995; Kamide & Mitchell, 1997) show evidence for an initial on-line commitment to the low attachment site.

This suggests that the revision from low attachment to high attachment, far from causing conscious processing difficulty, as predicted by standard monotonic models, may actually be the preferred course of action for the processor.[8]

5 A General Definition of Monotonicity

Despite this problem and others which we mention in Sturt & Crocker (to appear), we argue that the monotonicity framework should not be rejected as a paradigm for studying reanalysis. The reason for this is that previous monotonic models represent only a small subset of possible monotonic models. Specifically, the representation which they preserve using monotonicity is a description which mentions all nodes in the phrase structure tree at each state. Because of this, we can see these models as defining monotonicity over constituent structure representations. However, there is nothing to prevent us from defining similar constraints over other representation types. If we can provide a general definition of monotonicity capable of being applied to different types of representation, then we can use this framework to examine specific versions of the representation-preserving hypothesis. That is, we can keep the notion of preservation constant (using the general definition of monotonicity) while varying the type of representation to which it applies.

5.1 Monotonicity at the Constituent Structure Level

In what follows, we will give a brief review of the constraints imposed in one monotonic model (Gorrell, 1995). This model predicts processing cost associated with any reanalysis which deletes either precedence or dominance relations. Since the dominance and precedence relations constrained by this definition of monotonicity are between all nodes, this can be seen as a definition of monotonicity at the constituent structure level of representation. In the following, we again use the abbreviation '$S \Rightarrow T$' to stand for the transition from state S to state T.

[8] Due to certain characteristics of left-branching languages a monotonic account of the Japanese results is possible within the type of framework we have been discussing here. However, we do not go into the details of this, due to lack of space. The (right-branching) Italian results remain problematic for the type of models we have discussed here.

Thus, Gorrell's constraint can be defined as follows:[9]

> *Dominance Constraint*: For each transition $S \Rightarrow T$, if X dominates Y at state S, then X dominates Y at state T.
>
> *Precedence Constraint*: For each transition $S \Rightarrow T$, if X precedes Y at state S, then X precedes Y at state T.[10]

We will take these two constraints in turn, to see how they predict some intuitive distinctions. The Dominance Constraint predicts the intuitive difference between the 'easier' sentence in (1) and the 'harder' one in (2), repeated below:

(1) The wedding guests saw the cake was still being decorated.

(2) While the wedding guests ate the cake was still being decorated.

Assuming that *the cake* is initially attached as the direct object of the immediately preceding verb, then the reanalysis required to convert this NP to the subject of the complement clause in (1) involves only the *addition* of (at least) one node dominating the NP (i.e., at least the maximal sentential node dominating the complement clause). However, in the case of (2), the NP has to be 'moved' to a position from where it is no longer dominated by any of the nodes in the preposed clause, thus there exist dominance relations which hold at the state before reanalysis occurs but fail to hold at following states.

As an example of a violation of Gorrell's version of the Precedence Constraint, consider the following well-known garden path, which is 'hard' in the null (unbiasing) context:[11]

---

9 Although this constraint is equivalent to Gorrell's, it is not stated in identical terms.

10 To avoid cluttering these definitions, we avoid explicit mention of the universal quantifiers over the variables X and Y. We continue to make this omission in subsequent definitions, where we believe there is no danger of misinterpretation.

11 Clifton (to appear) describes an eye-tracking experiment which investigated garden path sentences of this type. Though the experiment was intended to investigate a separate issue, Chuck Clifton informs us (p.c.) that, in addition to garden path effects found in the eye-movement data, response accuracy to comprehension questions was extremely low for the dispreferred relative clause continuation, in comparison with the preferred complement clause continuation (34% versus 85% correct). This indicates that reanalysis is extremely hard and often impossible in this type of garden path.

(8) The doctor persuaded the woman that he was having trouble with to leave.

Assuming that the *that*-clause is initially attached as an argument of *persuaded*, (and, on standard analyses, is therefore preceded by *the woman*), the reanalysis necessary to reinterpret the that-clause as a relative clause will result in the that-clause being *dominated* by *the woman.* Hence the reanalysis requires the deletion of the precedence relation between *the woman* and the *that*-clause, and is therefore predicted to be difficult—that is, at least one precedence relation holds at the state before reanalysis occurs, but fails to hold after it.

5.2 Generalizing Monotonicity in Terms of Properties

In what follows, we will present two methods of generalizing monotonicity. The inspiration for both of these methods comes from Barker & Pullum (1990), who introduce two similar methods of generalizing *command relations*, based on *properties* on the one hand, and *relations* on the other. In this section, we describe a definition of monotonicity which is generalized in terms of *properties* of nodes. The central idea is that representation types can be defined by picking out, from a phrase structure tree, subsets of nodes which share a certain property, which we will call the *Generator Property* (borrowing some terminology from Barker & Pullum, 1990).

We will illustrate this with reference to the tree for (7), given in Figure 3. The most fine-grained representation is the phrase-structure tree (marked 'PS tree' in the diagram). It is the most fine-grained representation because it specifies all the nodes. However, we can also pick out less-fully specified representations by picking out only *subsets* of nodes which share a particular generator property. In the diagram, we have drawn a circle around all the nodes which we call *thematic assigner* nodes—that is, nodes which share the property of being the (extended) maximal projection of a theta-assigning head. (We make the assumption—along with Frazier & Clifton, 1996; De Vincenzi & Job, 1995; and Gilboy *et al.*, 1995—that the noun *daughter* assigns a thematic role to NP3, *the colonel.* Hence NP2, the maximal projection of *daughter* is a thematic assigner node.) The nodes which have been enclosed in squares correspond to the set of what we call *thematic receiver* nodes. These are nodes which have received a thematic role from some theta-assigning head. Let us say that the set of *thematic nodes* consists of the union of the thematic assigner and thematic receiver nodes. Then we can envisage three different levels of specification, according to which generator property we consider: the phrase-structure tree consisting of all the nodes, the thematic tree consisting of all and only the thematic nodes, and the thematic assignment tree consisting of all and only the thematic assigner nodes. Notice that the trees become 'flatter' as the membership of the relevant set of nodes becomes smaller.[12] It is possible to define monotonicity specifically, in terms of any one of these three levels, or in general, in terms of any arbitrary generator property.

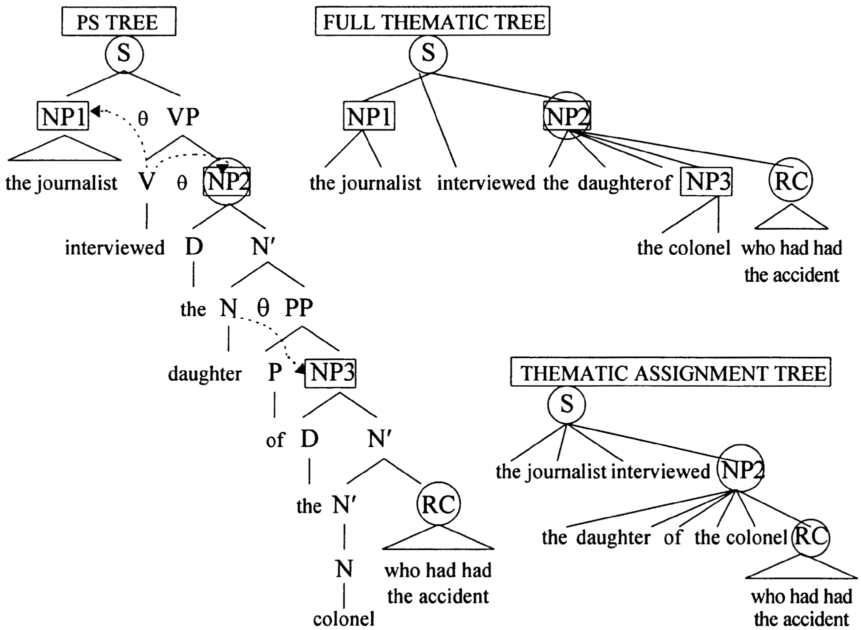

FIGURE 3: A representation showing thematic *assigner* nodes (enclosed in circles) and thematic *receiver* nodes (enclosed in squares) for a modifier attachment ambiguity.

What, then, is the role played by the structural relations of dominance and precedence, and how does this role relate to the notion of generator properties?

The statement "A dominates B" can be interpreted as a statement that domain A contains element B. On the other hand, a precedence relation between C and D

[12] Of course, given a phrase structure tree labelled with the necessary thematic feature information, the 'thematic trees' are purely derivative constructs and do not need to be built separately by a parser. We depict them in the diagram (including terminal symbols for clarity) purely for expository convenience.

can be interpreted as a statement that C and D are independent domains (i.e., neither one contains the other), since, by the theory of trees, a precedence relation between two nodes implies that the two nodes are not related by dominance.[13] The requirement that structural relations be updated monotonically can therefore be seen as a dynamic constraint on the construction of domains; that is, the constraint that both the membership (dominance) and the independence (precedence) of domains should be preserved.

When we move on to consider monotonic models defined in terms of some arbitrary generator property P, we simply use P to define the relevant notion of domain; the domains whose membership and independence we wish to preserve consist of subtrees rooted in nodes of which P holds (we will call such nodes P-nodes). Thus, in a system defined by the generator property P, the *membership* of a domain is the set of nodes dominated by some P-node, while the *independence* of two domains is captured by a precedence relation between two P-nodes. So, for example, if we were to take the generator property to be that of *thematic nodes* (i.e., the union of thematic assigner and thematic receiver nodes), then, in Figure 3, NP1 and NP2 would be independent domains (since NP1 and NP2 are each thematic nodes, and a precedence relation holds between them), while NP3 would be a member of the domain of NP2, because NP2 dominates NP3.

We now move on to a general definition in which any arbitrary generator property can be used to define a domain. In the following definitions, the P-Dominator Monotonicity Constraint ensures that although each node can become a member of a new domain (where domains are defined in terms of the generator property P), no node can cease to be a member of an existing domain. On the other hand, the P-Precedence Partner Monotonicity Constraint ensures that once a pair of independent domains has been established, (i.e., two nodes, each with property P are in a precedence relation), that independence must be preserved.

> *P-Dominators*: The *P-dominators* for Y at state S (abbreviated as $D(Y, P)^S$) is the set of all nodes X, such that property P holds of X at state S, and X properly dominates Y at state S

---

13 The exclusivity condition of the theory of trees forbids two nodes from being simultaneously in a dominance and a precedence relation. See Wall (1972) for an introduction to the theory of trees.

(where X *properly dominates* Y iff X dominates Y and $X \neq Y$).[14]

P-Dominator Monotonicity Constraint: For each transition $S \Rightarrow T$, and for each node X, $D(X, P)^S \subseteq D(X, P)^T$.

P-Precedence Partners: The set of *Precedence Partners* at state S (written $PRP(P)^S$) is the set of all ordered pairs of nodes X and Y, such that property P holds of both X and Y at state S, and X precedes Y at state S.

P-Precedence Partner Monotonicity Constraint: For each transition $S \Rightarrow T$, $PRP(P)^S \subseteq PRP(P)^T$

This generalized definition allows one to recreate Gorrell's (1995) constraints by simply setting the generator property P to the trivial property of being a node. This will have the effect of preserving dominance and precedence relations between all nodes. However, it also allows any number of alternative definitions to be made. For example, we can set the generator property to be the property of being a thematic assigner node. Recalling Figure 3, imagine that, as in De Vincenzi & Job's (1995) Italian experiments, we have evidence for an initial on-line commitment to the low attachment of the relative clause (the reading in which the relative clause modifies *colonel*, as illustrated in the Figure), followed by a final preference for the high attachment site (where the relative clause modifies *daughter*), and that this reanalysis is therefore of the 'easy' kind. Let us say that this reanalysis occurs in the transition $S \Rightarrow T$. Then, in Gorrell's (1995) system, the relative clause will lose at least one dominator (e.g., NP3), and the reanalysis will be incorrectly predicted to be 'difficult'. However, if we alter the generator property so that it picks out, say, only the set of thematic assigner nodes, then the reanalysis will be correctly predicted to be unproblematic, since both dominators for the relative clause (i.e., both S and NP2) remain dominators for RC in both attachment positions.

5.3 Generalizing Monotonicity in Terms of Relations

The second, more powerful way in which we can generalize monotonicity is in terms of *relations*. In the previous section, where monotonicity was defined in

---

[14] The use of the predicate 'Properly Dominates' rather than 'Dominates' allows a close convergence with the generalized command theory of Barker & Pullum (1990).

terms of *properties*, the P-dominators for a node N were defined exclusively in terms of the generator property P, along with the predicate 'properly dominates'. By contrast, when we define monotonicity in terms of *relations*, the membership of the set of dominators of N may depend partly on features of N itself. This allows us to make different types of nodes sensitive to different types of domains.

Before providing a concrete example, we will give the definitions. In what follows, we use the the abbreviation $R(X, Y)^S$ to stand for "the relation R holds between X and Y at state S". We will refer to R as the *Generator Relation.*

> *R-Dominators*: The R-dominators for Y at state S (abbreviated as $D(Y, R)^S$) is the set of all nodes X such that $R(X, Y)^S$, and X properly dominates Y at state S.
>
> *R-Dominator Monotonicity Constraint*: For each transition $S \Rightarrow T$, and for each node X, $D(X, R)^S \subseteq D(X, R)^T$.
>
> *R-Precedence Partners*: The set of *R-Precedence Partners* at state S (written $PRP(R)^S$) is the set of all ordered pairs of nodes X and Y, such that $R(X, Y)^S$ and $R(Y, X)^S$, and X precedes Y at state S.[15]
>
> *R-Precedence Partner Monotonicity Constraint*: For each transition $S \Rightarrow T$, $PRP(R)^S \subseteq PRP(R)^T$.

An example of a monotonicity constraint defined in terms of a relation is 'Thematic Monotonicity' (Sturt & Crocker, to appear). This constraint is defined in terms of a generator relation which we call 'visibility'. The visibility relation captures the idea, proposed in recent years, that the attachment of modifiers is, in some sense, less constrained than the attachment of obligatory constituents (see Frazier, 1990b; Frazier & Clifton, 1996). The visibility relation yields a model which behaves as though it were based on two generator properties simultaneously, by making modifiers and nonmodifiers sensitive to different kinds of domains. Specifically, the model behaves as though the

[15] Note that, for the sake of generality, we have defined R-Precedence Partners 'symmetrically', assuming no difference in priority between the left and right members of a pair of nodes in a precedence relation. An anti-symmetric definition would also be possible, where only one of $R(X, Y)$ or $R(Y, X)$ would have to hold. This would require independent justification of the directionality of the relation.

reanalysis of modifiers is constrained by the generator property of *thematic assignerhood*, while the reanalysis of non-modifiers is constrained by the generator property of being a *thematic node*. Thus, all and only the *thematic assigners* are visible to modifiers, while (depending on certain conditions, which we will discuss below) both thematic assigners and thematic receivers are visible to *non*-modifiers.

The definition of the visibility relation, taken from Sturt & Crocker (to appear) is as follows:

Visibility:

1. A is visible to a modifier M iff A is a thematic assigner node.[16]
2. A is visible to a non-modifier N iff

   either there is no modifier that dominates N but not A, and A is a thematic node (i.e. a receiver or assigner),
   or there is a modifier that dominates N but not A, and A is a thematic assigner node (see below).

The 'either-or' statement in part 2 of the definition requires some explanation. The actual processing system which employs the thematic monotonicity constraint propagates configurational information relevant to visibility in a strictly local fashion, with the consequence that modifiers act as what can informally be thought of as 'visibility filters' to their descendants.[17] Recall Figure 3, for example. Because the relative clause, (labelled RC) is a modifier, only thematic assigner nodes are visible to the RC, and, moreover, none of the nodes *inside* the RC can 'see' anything other than thematic assigner nodes 'beyond' the RC, even though, if they are non-modifiers, they can 'see' all thematic nodes 'up to' the most immediately dominating modifier. Thus, NP3, because it is not a thematic assigner node, is not visible to the RC or to any nodes dominated by the RC. On the other hand, both S and NP2 are thematic assigner nodes, and so *are* visible to the RC and to nodes within the RC.

---

[16] This and the remainder of the definitions in this section are intended to be relativized to *states*. Strictly speaking, this particular clause of the definition of visibility should read "A is *visible* to a modifier M at State S iff A is a thematic assigner node at state S". We omit mentioning states to avoid cluttering the definitions.

[17] We lack the space to discuss the details of implementation here, but sese Sturt (in preparation) for further details, including a discussion of the inheritance mechanism used.

However, *all* thematic nodes, that is both assigners and receivers, are visible to non-modifiers, unless they are 'filtered out' by a modifier in this way. Thus, for example, NP1 is visible to NP3, and vice versa, but neither NP1 nor NP3 are visible to any nodes inside the RC.

Using the visibility relation as a generator relation in the above definitions we can see that the *R*-dominators for the relative clause (which is a modifier) consist of NP2 and S, the two thematic assigner nodes properly dominating the RC. Meanwhile, the *R*-dominators for a non-modifier, such as the head noun immediately dominating *colonel*, for example, consist of NP3, NP2, and S, the three thematic nodes properly dominating this node. Thus, the *Thematic Monotonicity Constraint*, which defines monotonicity using visibility as a generator relation has the effect of 'anchoring' each node into representations, which are more or less specified according to the features of the node in question. This allows us to apply to modifiers a constraint which is similar to *Construal* (Frazier & Clifton, 1996), while non-modifiers are subject to a constraint which is similar to Pritchett's (1988) *Theta Reanalysis Constraint*. We need the extra power of relations to do this; it would be impossible to reproduce the same behavior in a system which only used properties as generators.

We will now illustrate the visibility relation with reference to two concrete examples. The first example demonstrates how we handle the effect of preposition type which has been found in the processing of sentences similar to that illustrated in Figure 3. Consider Figure 4, which shows the translation of an Italian attachment ambiguity from De Vincenzi & Job (1995). The relative clause is a modifier, so, assuming the notion of thematic structure which we discussed in section 5.2, its dominators are the two circled nodes: the NP, which is the maximal projection of *father*, and the Matrix S node, which is the (extended) maximal projection of the verb *suspects*. Even if the relative clause is initially attached in the low site, its attachment can later be revised to the high site (modifying *father*) without violating the monotonicity constraint.

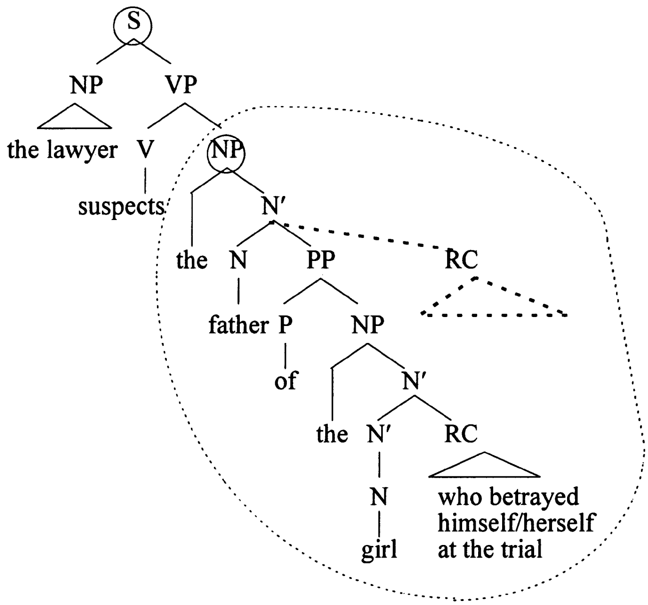

FIGURE 4: A relative clause attachment ambiguity. The dotted line encloses the locus of possible re-attachment sites for the relative clause once it has been attached in the low site. The high site is available here. (After De Vincenzi & Job, 1995.)

Now let us consider the situation in which the preposition inside the complex NP assigns a thematic role in its own right, as shown in Figure 5. In this case, the PP is a thematic assigning node, and therefore acts as a thematic dominator for the relative clause. This means that, once the relative clause has been attached in the low site, it cannot be re-attached to the high site without violating the Monotonicity Constraint. This pattern is consistent with De Vincenzi & Job's finding that, although an initial low attachment was found for the relative clause, for conditions including both non-thematic and thematic prepositions (illustrated by Figures 4 and 5 respectively), a final, off-line preference was found only for the condition including the *thematic* preposition (illustrated by Figure 5).

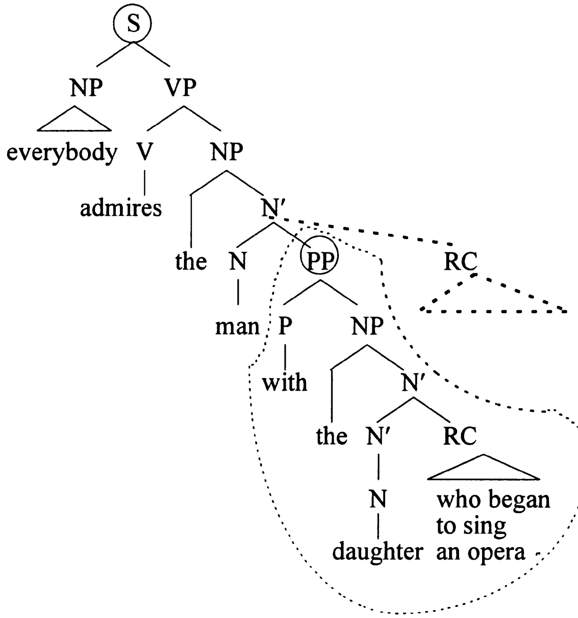

FIGURE 5: An illustration of a relative clause attachment ambiguity involving a preposition which assigns a thematic role. The locus of possible attachment sites for the relative clause now excludes the high site. (After De Vincenzi & Job, 1995.)

The second example applies to the reanalysis of a *non*-modifier. Consider again sentence (8), repeated below, which we cited as an illustration of Gorrell's (1995) Precedence Ponstraint, which, we recall, is known to cause considerable 'difficulty' in a null context:

(8) The doctor persuaded the woman that he was having trouble with to leave.

Assuming that the *that*-clause is originally attached as an argument of *persuaded*, and assigned a thematic role, then the clausal node will be a thematic receiver. The NP dominating *the woman* will also be a thematic receiver. Given also that neither the NP nor the clausal node are modifiers, they will be visible to each other. Hence, taking visibility as a generator relation, the NP and the clausal node will be *R*-Precedence Partners on the initially preferred attachment. However, reanalyzing the *that*-clause as a relative modifying the NP will result in a configuration in which the NP *dominates* the clause, and therefore, the precedence partnership will cease to hold, against *R*-Precedence Partner Monotonicity.

6 Reflections on the Monotonicity Framework

At this point we would like to step back and consider the nature of the monotonicity framework from a wider perspective. As we have pointed out, our purpose in this paper has not been to motivate any one particular instantiation of monotonicity, but rather to point out how the general concepts of D-theory can be applied to a wider range of models than has previously been considered. In this section we will consider some of the more general properties that are implied by the framework.

6.1 Consequences of Defining Monotonicity in Terms of Arbitrary Properties and Relations

We have seen in the preceding sections that monotonicity defined at the level of *constituent structure* (as in Gorrell, 1995) is equivalent to defining monotonicity with respect to the property that holds of all nodes, and thus has the effect of preserving all dominance and precedence relations in a tree. However, we have also seen that it is possible to define monotonicity with respect to properties that do not necessarily hold of all nodes. For monotonic models defined in terms of *properties*, it is true to say that there is an inverse correlation between the restrictiveness of the generator property and the restrictiveness of the resulting processor. That is to say, generator properties which are more general, in the sense that they hold of more nodes, will impose more restrictions on the possibilities for reanalysis, while generator properties which are more restrictive, in the sense that they hold of fewer nodes, will impose less strong constraints on the possibilities for reanalysis.[18] This means that monotonicity defined at the level of *constituent structure* yields the monotonic model which has the most restricted possibility for reanalysis, simply because the generator property necessarily holds of all nodes. Conversely, a generator property which necessarily holds of no nodes (i.e., the empty set) will impose no constraints on reanalysis at all. Generator properties which are intermediate between these two extremes will 'pick out' more or less underspecified representations, and subsequently allow more or less freedom for reanalysis. One of the consequences of defining a model in terms of a generator property P, for example, is that it will be impossible to 'raise' a constituent past the nearest dominating node of which P holds. In Gorrell's system, this means no nodes can be raised at all. In a system in which the generator property P is *thematic*

[18] This is related to the fact that, as proved by Barker & Pullum (1990), intersection over command relations corresponds to union over their generating properties.

assignerhood, no node can be raised past the nearest dominating thematic assigner. In a system in which the generator property is clause-hood, no constituent can be raised past the nearest clausal node, and so on.

In this paper, we have given some statically defined definitions but said very little about the *dynamics* of processing. In particular, we have not considered the question of when properties are 'assigned' to nodes. In a constituent-based model of monotonicity, where the generator property is simply *nodehood*, this question does not arise, as a node trivially cannot exist unless it has the generator property. However, in systems with smaller generator properties, there exists the possibility that a node may be attached into a phrase structure tree at some state, but acquire the generator property at a subsequent state. This allows monotonic models to capture some notion of 'delayed commitment', since a node may be in a position in which it could *potentially* have the generator property, but the processor may not necessarily *commit* itself to assigning the property to this node immediately. In Sturt & Crocker (to appear), for example, we assumed a processor which *attached* nodes immediately, but in which the *confirmation* of nodes holding the relevant properties was *head-driven* (in this case, the relevant properties were thematic assigner-hood and thematic receiver-hood, properties which are used in the definition of the 'visibility' relation on which the model is based).

6.2 Consequences of a Binary Monotonic/Non-monotonic Distinction

In this chapter, we have implicitly presented monotonicity as though it were a necessary condition for parse success—we have made the implicit assumption that a revision which violates monotonicity (in whichever monotonic model is proposed) is not possible for the parser. However, it is clear that people *do* manage to make sense of a number of 'conscious' garden paths, and, moreover, it is well known that the effort required to (re)process these structures can be lessened by manipulating factors such as subcategorization frequency and semantics. There are various ways of viewing the conscious/unconscious garden path distinction. The stronger view, due to Pritchett (1992), is that reanalysis of hard garden paths is possible, but that, unlike reanalysis of easy garden paths, it is not performed within the language module. A monotonic model does not necessarily commit itself to this position. The framework is equally consistent with a view that sees monotonicity not as a necessary condition for parsability but rather as a condition that the processor attempts to meet if it can. On this view, the processor is capable, if necessary, of destroying parts of its favored representation (as defined by the monotonicity model under consideration), but *prefers*, all other things being equal, to reanalyze in such a way that this representation is preserved.

Whichever view one takes, we believe a useful way to evaluate reanalysis models is to test what must be, after all, their most central prediction: the *preference* for one reanalysis over another, when more than one possibility is available. For example, a monotonic model would make clear predictions about the pair of sentences in (9):

(9) a. Once the students had understood the homework was easy they quickly finished it.
b. Once the students had understood the homework was easy and they quickly finished it.

In (9), the verb *understood* can either be intransitive, or subcategorize for either a clause or an NP. We would assume that in both (9a) and (9b), the NP *the homework* is initially attached as the direct object of *understood.* However, on the appearance of *was*, there are two possibilities for reanalysis: either *the homework* can be reanalyzed as the subject of a complement clause (a monotonic revision) or *the homework* can otherwise be reanalyzed as the subject of the main clause (which would violate monotonicity under most models). The former reanalysis is supported by the continuation in (9a) and the latter by the continuation in (9b). According to our intuitions, (9a) is indeed easier. Though we recognize the necessity for carefully controlling such materials (for example, in (9), we would need to consider the relative subcategorization frequencies for *understand*), we believe this general tactic is a useful one for reanalysis research.

One phenomenon for which monotonic models appear to be ill-suited is that of *graded* difficulty. That is, even *within* the class of either monotonic or non-monotonic revisions, there may be discernible *levels* of difficulty. For example, consider the following sentences, from Fodor & Inoue (1994):

(10) a. She put the candy in her mouth on the table. HARDER
b. She put the candy in her mouth onto the table. EASIER

Experiments investigating structures similar to this (Adams, 1995) suggest that *both* of the sentences in (10) cause processing disruption, and in fact, both are predicted to require non-monotonic revision in both constituency-based models (Gorrell, 1995; Sturt & Crocker, 1996) and the model based on the thematic notion of visibility described in Sturt & Crocker (to appear). However, as Fodor & Inoue (1994) point out, there is an intuition that (10a) is harder to resolve than (10b). How can we account for this in a model which treats both examples on a par? Assume that (as predicted, for example, in Sturt & Crocker, 1996), the preposition *on* in (10a), is initially attached as a modifier of *mouth.* However, assume also that, for lexical reasons, the processor knows that *onto* cannot be similarly attached (as assumed by Fodor & Inoue, 1994). This means that, in (10a), the processor not only needs to reattach *in her mouth* as a modifier of *candy*, but also to reattach *on the table* as the locative argument of put (both of these count as non-monotonic revisions in Sturt & Crocker, to appear). However, in (10b) only the former of these revisions is necessary. It is fairly natural to assume that a representation-preserving processor would find it more painful to destroy two pieces of its favorite representation rather than just one. If this is the case, then we might expect degree of difficulty in a monotonic parser to correlate with the number of violations of monotonicity.

6.3 Monotonicity Applied to Non-Syntactic Representations

While we have defined conditions for monotonicity that can be applied to representations other than phrase structure, the reader will have noted our implicit assumption that the generator property for each monotonic model will pick out a *subset* of nodes from a phrase structure tree (for example, the set of *thematic nodes* is subset of all the nodes in a phrase structure tree). This means that, according to our definitions, monotonicity is applied to representation types which are, in some sense, *subsumed* by phrase structure trees. We can call such monotonic models *phrase structure based.* Clearly, a phrase structure-based monotonic model cannot 'preserve' representations that are non-isomorphic to phrase structure. One example of a representation which is standardly assumed to be non-isomorphic to phrase structure (though not by everybody—c.f. Steedman, 1991) is intonation structure. On the standard view of constituency, there is no guarantee that a node in a phrase structure tree will correspond to a node in an intonational tree (for example, an intonational phrase may dominate a string of terminals that spans the final half of one phrase structure constituent, and the initial half of a second phrase structure constituent), and so no generator property applied to *phrase structure* trees could be used to define intonational phrases. However, there is no reason why a *non*-phrase structure-based monotonic model cannot be constructed—one could equally well define generator properties and apply them to intonational structure. Thus it would be possible, for example, to construct a model in which a lower level phonological unit could not be 'raised' past its dominating intonational phrase.

7 Concluding Remarks

In this paper, we have been concerned with the notion, common in reanalysis models, that there are certain core aspects of representation which the processor attempts to preserve. In particular, we have considered monotonicity as a useful general framework for the investigation of this question, and have shown how it can be generalized to allow the investigation of *which* representations are treated as the most sacred.

In closing, we would like to make some comments on the place that representation-preserving behavior may occupy with relation to the comprehension system as a whole. In particular, we would like to point out that representation-preservation should not be seen as a global strategy that guides *all* aspects of processing. If this were the case, then we would never expect the processor to make commitments in advance of disambiguation, since there is always a chance that any commitment may be wrong. Yet, thirty years of experimental research on sentence processing have told us that people *do* make commitments in advance of disambiguation. Rather, we think, the global strategy is to make the most sense of the input as it becomes incrementally available. This strategy produces conflicting demands on initial attachment and reanalysis. As far as *initial attachment* is concerned, the strategy implies that interpretable structures should be built quickly, which inevitably requires making commitments that are possibly wrong. As far as *reanalysis* is concerned, however, the strategy implies that the processor should try to make the most of the structure it has already built. The representation- preserving approach to reanalysis is concerned with the question of what is meant by 'making the most' of existing structure, and which aspects of structure this applies to. Generalized monotonicity provides a framework in which such questions can be considered.

Centre for Cognitive Science, University of Edinburgh

References

Adams, B.C. 1995. A model for strategic reanalysis in sentence processing. Unpublished manuscript, University of Virginia, VA.

Bader, M. 1996. On reanalysis: Evidence from German. Unpublished manuscript, Friedrich-Schiller-Universität Jena.

Barker, C. & Pullum, G.K. 1990. A theory of command relations. *Linguistics and Philosophy*, 13, 1, 1-34.

Clifton, C. (to appear). Evaluating models of human sentence processing. In M.W. Crocker, M.J. Pickering, & C.E. Clifton (eds.), *Architectures and Mechanisms for Language Processing*. Cambridge, UK: Cambridge University Press.

Clifton, C., Speer, S., & Abney, S.P. 1991. Parsing arguments: Phrase structure and argument structure as determinants of initial parsing decisions. *Journal of Memory and Language*, 30, 251-271.

Cuetos, F. & Mitchell, D.C. 1988. Cross-linguistic differences in parsing: Restrictions on the use of the Late Closure strategy in Spanish. *Cognition*, 30, 72-105.

De Vincenzi, M. & Job, R. 1993. Some observations on the universality of the Late-Closure strategy. *Journal of Psycholinguistic Research*, 22, 2, 189- 206.

De Vincenzi, M. & Job, R. 1995. An investigation of Late Closure: the role of syntax, thematic structure and pragmatics in initial and final interpretation. *Journal of Experimental Psychology. Learning, Memory and Cognition*, 21, 5, 1303-1321.

Ferreira, F. & Henderson, J.M. 1991. Recovery from misanalyses of garden-path sentences. *Journal of Memory and Language*, 30, 725-745.

Fodor, J.D. & Inoue, A. 1994. The diagnosis and cure of garden pāths. *Journal of Psycholinguistic Research*, 23, 4, 405-432.

Fodor, J.D. & Inoue, A. (this volume). Attach Anyway.

Frazier, L. 1978. *On Comprehending Sentences: Syntactic Parsing Strategies*. Unpublished doctoral dissertation, University of Connecticut, Storrs, CT. Distributed by the Indiana University Linguistics Club, Bloomington, IN.

Frazier, L. 1990a. Identifying structure under X^0. *Yearbook of Morphology*, 3, 87-105.

Frazier, L. 1990b. Parsing modifiers: Special purpose routines in the HPSM? In D.A. Balota, G.B. Flores d'Arcais, & K. Rayner (eds.), *Comprehension Processes in Reading*. Hillsdale, NJ: Lawrence Erlbaum Associates, 303-331.

Frazier, L. & Clifton, C. 1996. *Construal*. Cambridge MA: MIT Press.

Frazier, L. & Clifton, C. (this volume). Sentence reanalysis, and visibility.

Frazier, L. & Rayner, K. 1982. Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. *Cognitive Psychology*, 14, 178-210.

Gibson, E.A.F. 1991. *A Computational Theory of Human Linguistic Processing: Memory Limitations and Processing breakdown*. Unpublished doctoral dissertation, Carnegie Mellon University, Pittsburgh, PA. Available as Center for Machine Translation Technical Report CMU-CMT-91-125.

Gilboy, E.A.F., Sopena, J.M., Clifton, C., & Frazier, L. 1995. Argument structure and association preferences in Spanish and English compound NPs. *Cognition*, 54, 131-167.

Gorrell, P. 1995. *Syntax and Parsing*. Cambridge, UK: Cambridge University Press.

Inoue, A. & Fodor, J.D. 1995. Information-paced parsing of Japanese. In R. Mazuka & N. Nagai (eds.), *Japanese Sentence Processing*. Hillsdale, NJ: Lawrence Erlbaum Associates, 9-63.

Kamide, Y. & Mitchell, D.C. 1997. Relative clause attachment: Nondeterminism in Japanese parsing. *Journal of Psycholinguistic Research*, 26, 2, 247-254.

Lewis, R. 1993. *An Architecturally-based Theory of Human Sentence Comprehension*. Unpublished doctoral dissertation, Carnegie Mellon University, Pittsburgh, PA. Available as Technical Report CMU-CS-93-226 from reports@cs.cmu.edu.

Lombardo, V. (this volume). A computational model of recovery.

MacDonald, M.C., Pearlmutter, N.J., & Seidenberg, M.S. 1994. The lexical nature of syntactic ambiguity resolution. *Psychological Review*, 101, 4, 676- 703.

Marcus, M., Hindle, D., & Fleck, M. 1983. D-theory: Talking about talking about trees. *Association for Computational Linguistics*, 21, 129-136.

Marslen-Wilson, W.D. 1987. Functional parallelism in spoken word recognition. *Cognition*, 25, 71-102.

Pritchett, B.L. 1988. Garden path phenomena and the grammatical basis of language processing. *Language*, 64, 3, 539-576.

Pritchett, B.L. 1992. *Grammatical Competence and Parsing Performance*. Chicago, IL: University of Chicago Press.

Rayner, K., Carlson, M., & Frazier, L.. 1983. The interaction of syntax and semantics during sentence processing: Eye movements in the analysis of semantically biased sentences. *Journal of Verbal Learning and Verbal Behavior*, 22, 358-374.

Steedman, M.J. 1991. Structure and intonation. *Language*, 67, 2, 260-297.

Stevenson, S. 1994a. Competition and recency in a hybrid network model of syntactic disambiguation. *Journal of Psycholinguistic Research*, 23, 4, 295-321.

Stevenson, S. 1994b. *A Competitive Attachment Model for Resolving Syntactic Ambiguities in Natural Language Parsing*. Unpublished doctoral dissertation, University of Maryland, College Park, MD. Available as Technical Report TR-18 from Rutgers Center for Cognitive Science, Rutgers University.

Sturt, P. (in preparation). *Syntactic Reanalysis in Human Language Processing*. Unpublished doctoral dissertation, Centre for Cognitive Science, University of Edinburgh, Edinburgh, Scotland.

Sturt, P. & Crocker, M.W. 1996. Monotonic syntactic processing: a cross-linguistic study of attachment and reanalysis. *Language and Cognitive Processes*, 11, 5, 449-494.

Sturt, P. & Crocker, M.W. (to appear). Thematic monotonicity. *Journal of Psycholinguistic Research*.

Trueswell, J.C., Tanenhaus, M.K., & Kello, C. 1993. Verb-specific constraints on sentence processing: Separating effects of lexical preference from garden paths. *Journal of Experimental Psychology: Learning, Memory and Cognition*, 19, 3, 528-553.

Wall, R. 1972. *Introduction to Mathematical Linguistics*. Englewood Cliffs, NJ: Prentice Hall.

Warner, J. & Glass, A.L. 1987. Context and distance-to-disambiguation effects in ambiguity resolution: Evidence from grammaticality judgements of garden path sentences. *Journal of Memory and Language*, 26, 714-738.

Weinberg, A. 1993. Parameters in the theory of sentence processing: Minimal committment theory goes East. *Journal of Psycholinguistic Research*, 22, 3, 339-364.

Weinberg, A. 1995. Licensing constraints and the theory of language processing. In R. Mazuka & N. Nagai (eds.), *Japanese Sentence Processing*. Hillsdale, NJ: Lawrence Erlbaum Associates, 235- 255.