8

Reanalysis and Limited Repair Parsing: Leaping off the Garden Path*

Richard L. Lewis

The Ohio State University

Abstract

This chapter develops a theory of reanalysis called limited repair parsing. Repair parsers deal with the problem of local ambiguity in part by modifying previously built structure when the chosen structure later proves to be inconsistent. This modification of existing structure distinguishes repair parsing from parallel or multi-path parsing, least-commitment parsing, backtracking, or reparsing strategies. Parsers with a limited capability for repair are psycholinguistically important because they can potentially explain the contrasts between difficult garden path structures (when repair fails) and unproblematic local ambiguities (when repair is successful or easy). Although the idea of repair has been implicit in some psycholinguistic work (and emerged explicitly in the diagnosis model of Fodor & Inoue, 1994, and the NL-Soar model of Lewis, 1993), there has been no clear formulation of the general class of repair parsers. This chapter makes a first step toward such a formulation, shows how repair parsing offers significant computational advantages over other alternatives for reanalysis, and proposes a particular repair mechanism, *snip*, that explains a wide range of cross-linguistic reanalysis phenomena. Snip is a proposal for a simple, automatic, on-line repair process. The chapter concludes by briefly describing how snip can be embedded in a more comprehensive sentence processing architecture that maintains the structural sensitivity of purely syntactic theories like Pritchett's (1992), yet still accounts for the flexibility of parsing as revealed by interactive studies.

1 Introduction

Two central questions that a theory of human parsing must answer are: (1) How do people interpret locally ambiguous material? (2) How do people

* Many thanks to Jill Lehman and the late Allen Newell, who provided invaluable contributions to this work. Thanks also to Julie Boland and two anonymous reviewers, who provided useful comments on this chapter.

revise their interpretations of ambiguous material based on later disambiguating information? This chapter is an attempt to make progress on the second question by developing a theory of reanalysis called *limited repair* parsing (Lewis, 1992). Limited repair explains how the human sentence processor can sometimes rapidly leap off the garden path and find its way back onto the path to a successful parse. The theory has been implemented in a computational model of sentence comprehension, NL-Soar (Lehman *et al.*, 1991; Lewis, 1993a; Lewis, 1993b).

Repair parsers deal with the problem of local ambiguity by modifying previously built structure when the chosen structure proves to be inconsistent with the disambiguating material. The explicit modification of existing structure distinguishes repair parsing from parallel or multi-path parsing, minimal commitment parsing, and backtracking/reparsing strategies. Although the idea of repair has been implicit in some psycholinguistic work (and has recently emerged quite explicitly in the diagnosis model of Fodor & Inoue, 1995), there has been no clear formulation of the general class of repair parsers. This chapter makes a first step toward such a formulation, shows how repair parsing may offer significant computational advantages over other alternatives for reanalysis, and proposes a particular repair mechanism which explains a wide range of cross-linguistic reanalysis phenomena.

Why is understanding reanalysis just as important as understanding ambiguity resolution? Frazier & Rayner (1982) put it most strongly: "… it is emphasized that an understanding of the parser's revision procedures is essential to an explanation of why certain linguistic structures cannot be successfully parsed by humans" (p. 178). Indeed, one important source of data that can be used to reveal the nature of reanalysis processes concerns the contrast between ambiguous structures that can give rise to noticeable garden path effects, such as the subject/object ambiguity in (1) (Frazier & Rayner, 1982), and those ambiguous structures that cause little or no perceptible difficulty, such as the subject/object ambiguity in (2) (Juliano & Tanenhaus, 1994; Mitchell, 1987; Pritchett, 1988):

(1) Although Mary forgot her husband didn't seem very upset yesterday.

(2) a. Mary forgot her husband at the airport yesterday.
b. Mary forgot her husband needed a ride yesterday.

The structure in (1) can give rise to an impression of ungrammaticality, indicating a failure of reanalysis. The structure in (2) may produce longer reading times in the disambiguating region (e.g., an extra 50 ms on *needed* compared to unambiguous structures with overt complementizers; Ferreira &

Henderson, 1990), but does not usually cause noticeable difficulty, regardless of which interpretation is required. A theory of reanalysis is required to explain the contrasts. Why does reanalysis fail in structures such as (1), and succeed in structures such as (2)? Apparently, reanalysis processes are not all-powerful, but are constrained in some way. The particular model proposed below limits repair by a locality assumption that increases computational efficiency and leads to interesting predictions about difficult and easy reanalyses. Furthermore, it provides a concrete hypothesis about exactly what is going on in that extra 50 ms or so in the disambiguating regions of the easier sentences.

The remainder of the chapter is organized as follows. Section 2 describes how reanalysis can be understood as a functional requirement on human parsing, independently of how that function is realized. This will permit us to see how four rather different classes of reanalysis are all responsive to the same functional requirement. In section 3, each of the four major parsing models will be described in terms of the spaces they search and the strategies they use for searching them. This leads us to an interesting theoretical result: repair parsing is an effective parsing method because it *reformulates* the parsing search space so that it better supports efficient depth-first search. Section 4 describes a specific kind of repair mechanism, limited cue-based repair, and shows how it accounts for a wide range of cross-linguistic reanalysis phenomena. Section 5 sketches how the repair mechanism is embedded in a more comprehensive model of sentence processing.

2 Reanalysis as a Functional Requirement

Reanalysis is a function that must be performed by any system for parsing natural language. The following gives a functional definition of reanalysis:

(3) Reanalysis is the function of revising the interpretation of previously perceived linguistic material based on information that follows later.

Local ambiguity is the phenomenon that gives rise to the requirement for reanalysis: the intended interpretation of linguistic input is often determined by information that comes later in the sentence. For example, consider the familiar subject/object ambiguity in (4):

(4) I know the little boy hiding under the table…

The comprehender might adopt one of the following options for the initial interpretation:

(5) a. Interpret the string as meaning *I know who the boy is*.

b. Interpret the string as meaning I know something (to be made explicit immediately) about the boy.

c. Interpret the string as meaning either *I know who the boy is*, or *I know something about the boy*.

d. Delay interpretation.

If the input now continues:

(6) …will bump his head

the comprehender must revise the initial interpretation if it happened to select some option other than (5b). It is irrelevant what kind of ambiguity resolution process is used, no process can always make the correct decision. Nothing in this analysis hinges on the difficulty or ease of making the revision. The only point is a functional one: in some cases the revision must be made if the material is to be understood. (Parallel models and minimal commitment models may seem to provide counterexamples to the requirement for reanalysis, but we shall see in the next section that this is not the case.)

One might argue that the pure delay strategy (5d) is in fact a way of avoiding reanalysis, since there is no interpretation to be revised. However, such a strategy cannot be systematically sustained if the comprehender is constrained to emit some interpretation incrementally (within a finite time bound). The problem is that local ambiguities in natural language can stretch on indefinitely. For example, in long distance filler-gap ambiguities such as (7) below there is never a point at which one can be sure that it is safe to commit to a decision about the location of the gap.

(7) I saw the landlord that Amparo believes Ted knows Janet…

3 Four Theories of Reanalysis

The functional definition in (3) abstracts away from the process that is used to realize reanalysis. In this section we consider the computational properties of four major classes of possible reanalysis processes: reanalysis by backtracking, reanalysis by selection from parallel alternatives, reanalysis by refining commitments, and reanalysis by repair.

3.1 Reanalysis by Backtracking (Reparsing)

Frazier & Rayner's (1982) classification of reanalysis processes in reading provides a good starting point for understanding the space of possible processes. They distinguished between forward, backward, and selective reanalysis. *Forward* reanalysis occurs when the reader returns to the beginning of the sentence and recomprehends the sentence, presumably looking for choice points at which to make alternative decisions. In *backward* reanalysis, the reader works backward through the sentence, trying out different alternatives. In *selective* reanalysis, the reader attempts to focus only on the portion of the sentence that caused the problem. Frazier & Rayner looked for evidence for these different reanalysis types from explicit eye movements in reading. Most of the behavior was consistent with selective reanalysis, although subjects sometimes did start at the beginning and reread. The data did not support backward reanalysis at all.

All three kinds of reanalysis can be classified as *backtracking* schemes (Winograd, 1983), because they involve the comprehension system returning to previous states in the parse and pursuing alternative analyses by reparsing or recomprehending the ambiguous material. Reanalysis by backtracking to prior states is possible if the parser has a memory of previous choice points and states. If the pursued interpretation is incompatible with disambiguating material, the parser can revisit earlier choice points and try alternative paths.

Which state the parser returns to is determined by the particular control strategy used. *Chronological backtracking* systematically revisits previous choice points in reverse order. For example, a push-down stack can be used in ATN parsing to implement chronological backtracking. Unfortunately, the result is exponential search (Winograd, 1983). *Dependency-directed backtracking* uses information about the error itself in an attempt to return directly to the choice point that was responsible for the misanalysis. The ranked flagged serial model of Inoue & Fodor (1995) is an example of a selective backtracking model. These various control strategies for backtracking yield the forward, backward, and selective reanalysis distinction of Frazier & Rayner (1982).

Figure 1 shows how different kinds of backtracking can be viewed as variations on depth-first search in a space of legal parsing operations. Each node in the search space represents a partial, well-formed parse tree. Each operator or transition in the space represents a parsing operation (for example, the application of a phrase-structure rule) that extends the parse tree and pushes the system into a new state in the space. Some of the state transitions correspond to consuming input. Local ambiguity corresponds to choice points in the parsing space: multiple legal parsing operations are applicable at a given point. The goal state is a well-formed interpretation of the input. If the parser takes a wrong turn at an ambiguity, a dead-end is reached on a garden path and the parser must backtrack by revisiting prior states and pursuing alternative paths.

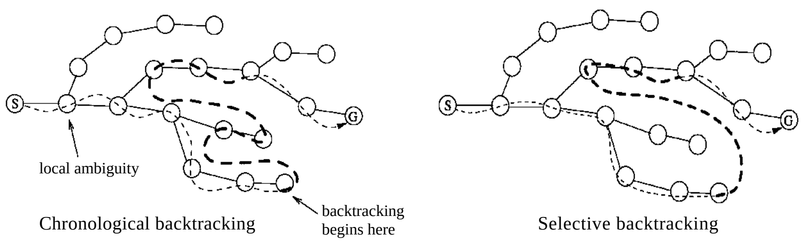


FIGURE 1: Two kinds of reanalysis by depth-first backtracking. The dashed lines indicate the path the parser takes in the search space; the heavy dashed lines indicate the reanalysis. *S* indicates the starting state, *G* indicates the goal state (a well-formed syntactic representation of the input).

There must be sufficient memory of previous states and choice points to support the control strategy used. Exhaustive chronological backtracking (Figure 1, left), requires a complete memory of parser states at all the previous choice points. Selective backtracking (Figure 1, right) can achieve some savings by only storing a limited number of prior parser states. This was the method used by the finite-state parser of Blank (1989). Blank's parser is an example of a *limited-memory selective backtracking* system. The parser was sometimes garden-pathed because the necessary parse states were not always available to backtrack to. Blank fixed, in advance, a small set of *boundary registers* that saved states at specific kinds of phrasal boundaries. Blank noted that the type and number of such registers is unconstrained—he simply chose a set that seemed reasonable and predicted certain kinds of strong garden path effects.

We have implicitly assumed so far that backtracking in reading results in overt regressions and reparsing.[1] However, as Frazier & Rayner (1982) point out, and as subsequent eye-tracking studies have demonstrated, reanalysis need not involve regressive eye movements. Often reanalysis is detected by

---

[1] Backtracking can occur in speech processing as well, when the listener repeats something to herself that she did not understand, or recomprehends as the speaker repeats himself. A kind of covert backtracking occurs when the listener covertly rehearses and recomprehends something, or reinterprets material in the short-term phonological trace without rehearsal (Gathercole & Baddeley, 1993).

longer times spent on the disambiguating region as the eye fixations continue forward through the sentence. In fact, all backtracking reanalysis strategies can be realized covertly, without explicit eye movements and reparsing. The cost for doing so is additional short-term memory storage. Overt backtracking reparses the actual input, while covert backtracking reparses a memory of the input. This distinction was never an issue for natural language parsing systems in artificial intelligence or computational linguistics, but it is clearly an important one for psycholinguistics.

Thus, the backtracking strategy that places the least demands on memory is forward reanalysis by overt recomprehension from the beginning of the sentence. This strategy requires no memory for input or prior parsing states—it needs just enough memory for the parser to remember not to continue going down the same path.[2] The drawback of this method is time, but it is a reliable strategy when all else fails.

As a theory of *covert*, rapid reanalysis—i.e., as a theory of what is happening in the longer fixations of disambiguating material when the eye continues to move forward, or as speech continues to flow in—standard backtracking suffers from two drawbacks. First, it may require an implausible amount of wasted reparsing. For example, reanalyzing the object NP to subject position in (8) below would require reparsing the entire NP over again (Winograd, 1983):

(8) Mary knew the bright red Honda parked next to hers belonged to the landlord.

Variations on standard backtracking avoid this wasted reparsing by keeping track of, and reusing, well-formed subphrases in the parse (e.g., Abney, 1989; Holbrook *et al.*, 1992; Konieczny, 1996). The second drawback is unavoidable in principle: in addition to the current parse state, the parser must keep track of prior states and input, and have extra mechanisms available for reinstantiating and managing the prior states (e.g., the stack used in chronological backtracking).

3.2 Reanalysis by Selection from Parallel Alternatives

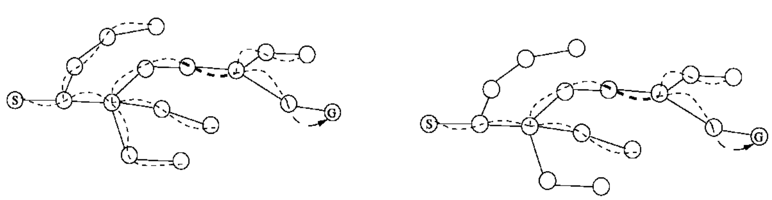
While backtracking searches depth-first, Figure 2 shows how parallel parsing is a breadth-first search through the parse space. Parallel parsing may seem to present an obvious counterexample to the claim that analysis is a functional requirement, because it does not appear to engage in any revision

[2] O(N) bits for N choice points for exhaustive search of the space.

processes. This objection confuses the kind of parsing process with the function that it realizes. It is true that parallel models do not need to make changes to their structural representations when the disambiguating information arrives, because the required structure can be selected and carried forward, and the incorrect structure dropped from further consideration. Nevertheless, the process of ruling out the incorrect structure (e.g., explicit pruning operations (Gibson, 1991), or redistribution of activation (Just & Carpenter, 1992)) is precisely what carries out the reanalysis function. This also holds of models in the constraint-based lexical framework (MacDonald *et al.*, 1994; Trueswell *et al.*, 1994) that adopt a kind of weighted-parallel scheme.

All-paths parsing (Figure 2, left) carries forward all possible interpretations in parallel. This places unrealistic memory demands on the human parser, and quickly runs afoul of empirical data on garden path effects (it predicts there should never be noticeable garden paths). Computationally, breadth-first search is more memory-intensive than depth-first search, and should be avoided unless depth-first search runs the risk of getting lost in deep or infinite paths (Winston, 1992).

Thus, the critical theoretical and empirical issue is how to limit the parallelism (Figure 2, right) so that the parser can still handle easy local ambiguities. Gibson (1991), following Gorrell (1987), proposed a limited parallel model motivated by short-term memory limitations. Gibson's theory took the form of a syntactic metric which measured memory load. The metric was used to make precise predictions about which alternatives will be pursued and which will be abandoned, turning the breadth-first search into a limited beam search (see e.g., Winston, 1992).



All-paths parallel parsing

Limited path parallel parsing

FIGURE 2: Reanalysis by selection from parallel alternatives. The dashed lines indicate the path the parser pursues in the search space; the heavy dashed lines indicate the reanalysis that takes place when some paths are dropped from consideration. *S* indicates the starting state, *G* indicates the goal state (a well-formed syntactic representation of the input).

The impressive empirical coverage of Gibson's metric would seem to provide additional support for the plausibility of a memory-limited parallel model. However, the way the metric is actually applied points away from such an account. Gibson's model works in the following way: it computes multiple parse trees in parallel, and assigns a memory cost to each partial tree based on the metric (the metric counts the number of unsatisfied syntactic relations of various types). Certain structures are then pruned based on their memory metrics. However, the pruning rule does not eliminate structures when they jointly or individually exceed some posited memory bound. Instead, the rule computes the *difference* in memory load between two structures, and if the difference is greater than some constant *P*, it prunes the higher cost structure. Multiple high-cost structures may be maintained in parallel as long as they do not differ too much in cost. Furthermore, this way of using the metric, and the pruning constant *P*, has no relation to the use of the metric to predict processing overload effects on center-embedded structures (Gibson, 1991). There, the metric is used more straightforwardly as a measure of memory load: structures are predicted to be unprocessable (and pruned) if they exceed some posited bound *K* (different from *P*).

One interpretation of the Gibson metric, therefore, is that it actually reflects the operation of a depth-first single-path parser, which has some limited capacity to repair or transform one structure into another.[3] Under this view, the metric implicitly serves as a metric of structural closeness or similarity. If the two structures are not too different, the repair process of transforming one to the other is easy, but if they are structurally too far apart, the process is difficult. Another interpretation of the metric is that it reflects a parallel system that is not maintaining completely independent structures, but using some structure sharing technique to conserve space. The more two structures share substructure, the closer their metrics will be, and the less memory they will require. Hence, it is less likely that one of them will be pruned.

Structure sharing addresses, in part, a difficult problem that any parallel parser must solve. When a parallel parser reaches an ambiguous point with two or more possible paths, it must create new copies of the original structure in order to pursue all the attachments. This kind of powerful copy-on-demand mechanism is an implausible candidate for a subcomponent of the human sentence processor.[4]

---

[3] Gibson (1991) acknowledges such an interpretation, but does not pursue it.

[4] This is the process used in Gibson's computational implementation of the theory, though he made no psychological claims about this aspect of the model.

The alternative is that the parser engage in some kind of structure sharing or sophisticated lazy evaluation (delaying copying until absolutely necessary; Godden, 1990; Lytinen & Tomuro, 1996). The choice of mechanism depends in part on the nature of the grammatical representation. Simple context free grammars admit straightforward structure sharing, but sharing can be more complicated with other kinds of grammars (Winograd, 1983). For example, unification grammars do not readily admit sharing at all, but demand either copying of structure, or lazy (delayed) copying. A dependency grammar representation that shares structure requires careful bookkeeping of which link goes with which interpretation (Lombardo, this volume).

Finally, note that none of these arguments against parallel, breadth-first parsing are arguments against parallel *generation* of syntactic alternatives at ambiguous points (Altmann & Steedman, 1988). These are orthogonal issues. The arguments are against *maintaining and pursuing* distinct interpretations in parallel. The distinction can be seen clearly in the Soar-based comprehension model of Lewis (1993a). This architecture separates the parallel generation and selection of potential parsing operators from their serial application. At each ambiguous point, multiple operators are generated that correspond to the intention to produce different syntactic structures. Of these multiple possibilities, one operator is selected,[5] and the execution of that operator produces the single selected structure.

3.3 Reanalysis by Monotonic Refinement of Commitments

Like parallel parsers, minimal commitment parsers seem to be obvious counterexamples to the functional requirement for reanalysis. Minimal commitment parsers only make monotonic additions to the representations of the parse tree they are incrementally building (Gorrell, 1993; Weinberg, 1993). But note that although the representation itself evolves monotonically over time, *what* the representation represents (the parse tree) is revised non-monotonically. In other words, minimal commitment parsers do revise their structural interpretations in the face of disambiguating information. In our earlier example, a minimal commitment parser would change its interpretation from (5a) to (5b). The fact that the same process (adding dominance and precedence relations) is used for both initial interpretation and reanalysis does not mean that there are not two distinct functions being accomplished. It simply means this particular approach has achieved some economy of mechanism (though we will question this achievement below). Both parallel models and minimal commitment

[5] Possibly on the basis of semantic or contextual factors (Lewis, 1993a; Lewis, 1996a).

models can therefore be seen as theories that are designed specifically to respond to the requirement for on-line reanalysis.

The way minimal commitment parsers effect reanalysis has several desirable properties that distinguish them from backtracking and parallel parsers:

1. There is no need to keep track of prior states or choices points, as in backtracking.

2. There is no need to reparse ambiguous input as in standard backtracking.

3. There is no need for arbitrary copy mechanisms or structure sharing strategies, as in parallel parsing.

4. The memory requirements are less than in parallel parsing.

Minimal commitment models therefore seem to handle the problem of reanalysis with considerable computational simplicity and economy. Yet all of the advantages listed above have little to do with monotonicity or underspecification. They instead derive from a restructuring of the parse space so that the search can always continue forward, effecting successful reanalysis (most of the time) without explicit backtracking. The next section shows how repair parsers can be formulated that have all these desirable properties of minimal commitment parsers, while abandoning monotonicity and underspecification.

Why abandon monotonicity and underspecification? Minimal commitment representations lead to some paradoxes when considering their role in a functionally complete system engaging in semantic interpretation (Lewis, 1993a). The most plausible assumption is that the syntactic representations are immediately and incrementally interpreted (Weinberg, 1993). This presumably requires determining which syntactic relations actually hold. Because the dominance assertions do not explicitly specify which relations hold, additional computation must be performed to make the *immediate* dominance relations explicit. These computations are purely syntactic in nature, since they process purely syntactic representations. Whether this process is considered part of the parser itself, or assigned to the semantic interpreter (an odd partitioning of function), monotonicity is violated since the immediate dominance relations change nonmonotonically throughout the parse. In short, it is not clear how to reconcile minimal commitment parsers with immediacy of interpretation, without abandoning monotonicity. Since, as argued above, the desired computational features do not depend on monotonicity, this is not such a bad move.

3.4 Reanalysis by Limited Repair

Limited repair is a way of reformulating the parsing problem space so that it efficiently supports depth-first search without backtracking. The starting point is to note that we can cast the parsing process as a search for a well-formed syntactic representation (e.g., Fong & Berwick, 1990), rather than a search through legal applications of grammatical rules. The parsing search space can therefore be formulated in several ways. In addition to legal attachment operators, we are free to include any kind of structure building operators, including non-standard attachment operations that may momentarily violate some well-formedness constraint. We can also include operators that undo or destructively modify existing structures, such as operators that detach phrases and operators that move phrases.

These new modification operators are at the heart of repair parsing. The basic idea is to augment the standard parsing space used in depth-first backtracking parsing with repair operators that enable the system to continue moving through the space when it reaches a dead-end (or, more accurately, what once was a dead-end). Figure 3 shows what the structure of a new repair space might look like. In short, the parser can sometimes leap off the garden path, rather than explicitly backtracking to prior decision points. In making the repair to the correct state, it is possible that the parser may visit states that have no counterpart in the space explored by a standard backtracking parser. The repair process may also revisit a prior state, but only through a process of reconstruction (or de-construction), not by retrieving it from some short-term store.

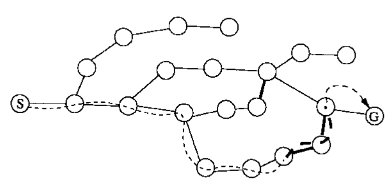


FIGURE 3: Reanalysis by repair of the existing interpretation. The dashed lines indicate the path the parser pursues; the heavy dashed lines indicate the reanalysis that takes place during repair. The heavy state transitions denote new repair operators that augment the original space. *S* indicates the starting state, *G* indicates the goal state (a well-formed syntactic representation of the input).

In no way does this problem space reformulation change the *task* of parsing. The grammatical constraints on the parser output are not relaxed in any way.

The only thing that changes is the structure of the space in which the parser searches for a well-formed representation.

What kind of repair operators should be added to the space? It might be possible to augment the parsing space with some highly general repair operators that can be applied to any state (perhaps a kind of move-α for repair). This would ensure that the full search space is reachable from any given state, yielding reanalysis as powerful as general backtracking. However, that could also yield a combinatoric search that is as at least as bad as chronological backtracking, and possibly worse.

The generation and application of the repair (modification) operators must therefore be constrained in some way. Figure 3 depicts a constrained version of repair. Two features of limited repair spaces are apparent in this little example. First, the additional repair operators are only available from a small subset of the total set of states. Second, there are no guarantees of completeness: it is still possible to go irreparably down the garden path. For example, if the parser takes the wrong branch at the first choice point in Figure 3, there is no way to repair from the garden path back to the correct path.

Even before specifying the details fully, we can see that limited repair would have several desirable computational and psycholinguistic properties as a theory of covert reanalysis:

- It uses memory-efficient depth-first search, rather than breadth-first search.
- It avoids the additional memory overhead of backtracking.
- It avoids the need for copy mechanisms or structure sharing techniques.
- It incrementally produces interpretable, fully specified syntactic structures.
- It potentially distinguishes between difficult-to-repair and easily repaired structures.

It is worth noting that reformulating problem spaces to make problem solving easier for some task is an old and important idea in artificial intelligence (Newell & Simon, 1972). In fact, *repair* in particular has been used in some domains to great effect (Minton *et al.*, 1992). A dramatic demonstration was recently made using the *n*-queens problem, a classic search puzzle that requires finding a way to position *n*-queens on an $n \times n$ chessboard so that no queen can attack any other queen. The standard algorithms, which performed backtracking search in a space of (locally) legal queen placements, were able to solve *n*-queens problems with *n* up to about 100 (Nadel, 1990). The problem space was then reformulated as a repair space, which started with a board full of queens in illegal positions, and used a repair operator ('swap') for making minor modifications to queen positions. The result was a system that could solve the *n*-queens problem in polynomial time (Sosic & Gu, 1990). The authors were able to solve problems with $n = 500{,}000$ on 25MHz workstations.

3.5 Summary of the Four Major Theoretical Classes

We can summarize the four kinds of reanalysis models and their computational properties as follows:

- *Depth-first backtracking* parsers accomplish reanalysis by revisiting prior choice points or states in the parse, and pursuing alternative paths by reparsing the ambiguous material. Different control strategies (e.g., chronological backtracking, dependency-directed backtracking) yield the forward, backward, and selective reanalysis distinction of Frazier & Rayner (1982). The different strategies place different demands on memory, with forward reanalysis making the least demands. Reanalysis by backtracking can be overt (corresponding to regressive eye fixations in reading) or covert (corresponding to a reparsing of an internal memory of the input). The primary disadvantage of backtracking, particularly as a theory of covert reanalysis, is the memory overhead required to maintain prior states and input (relative to minimal commitment and repair parsers) and the additional mechanism needed to manage this memory.

- *Parallel* and *limited parallel* parsers accomplish reanalysis by pursuing multiple structural interpretations in parallel, and disposing of those interpretations that are not consistent with later input. Parallel parsers effect a breadth-first search in the space of partial interpretations. The primary disadvantage of any breadth-first search is the increased space requirement (relative to depth-first alternatives). Standard techniques for reducing the memory requirements and the need for structure copying (structure sharing or lazy evaluation) require complicated mechanisms for grammars more complex than context-free grammars. Finally, the most detailed attempt yet to work out a memory-limited parallel account of difficult and easy local ambiguities (Gibson, 1991) yielded a theory that is at least as consistent with single-path repair parsing as parallel parsing—perhaps more so.

- *Minimal commitment* parsers accomplish reanalysis by making monotonic additions to underspecified parse tree representations. They require less memory than either backtracking or parallel models, and require no structure sharing or structure copying. Furthermore, no additional mechanism is required to effect the reanalysis. The primary disadvantage relative to all the other models is that additional computation is necessary to make explicit the syntactic relations necessary for immediate, incremental semantic interpretation. These relations change nonmonotonically, therefore some other component of the system must be effecting what is essentially a destructive syntactic revision.

- *Limited repair* parsers accomplish reanalysis by destructively modifying the current parse tree. They augment the standard parsing space used in depth-first backtracking parsing with repair operators that enable the system to continue moving forward when it reaches a dead-end. In short, limited repair parsers reformulate the search space to support *single-state depth-first search.* The primary advantage over backtracking and parallel parsers is reduced memory load—only a single state must be maintained at any time. There is no need for structure sharing or structure copying mechanisms, and unlike minimal commitment parsers, repair parsers produce fully specified, interpretable syntactic structures. The disadvantage, relative to all the other models, is that explicit modification operators must be added to the search space.

Table 1 lists some of the explicit reanalysis theories proposed, classified into variations on the four major classes discussed above.[6] Although repair parsers are clearly a viable class on purely theoretic grounds, they have been little explored relative to the other models. Nevertheless, repair parsing was used in one of the earliest and most successful natural language processing efforts in artificial intelligence: the PROGRAMMAR parser of Winograd's SHRDLU system (Winograd, 1972, 1983). Winograd's system used a set of specific rules mapping problem situations onto structure modifying actions. Lombardo's (this volume) parser is one modern theory pursuing a similar approach.

One important theory of reanalysis that defies this classification is Pritchett's (1988, 1992) model. Pritchett's *On-line Locality Constraint* (OLLC) provides an abstract syntactic characterization of easy reanalysis. The OLLC is computed over the preferred and correct structures. If the preferred interpretation is structurally too distant (in some precisely defined

[6] Hybrid approaches are also possible. For example, limited repair and backtracking show up together in the models of Lewis (1993a) and Konieczny (1996).

sense) from the correct structure, Pritchett's theory predicts a strong garden path effect. This model is interpreted naturally in terms of a depth-first parser with some rapid repair or revision process. Like the Gibson (1991) metric, the OLLC can be construed as a characterization of when the correct interpretation is out of reach of on-line repair. Pritchett in fact used this kind of language himself in describing the theory, but made no explicit commitment to single-path or parallel parsing.

The theoretical analysis presented in this section is insufficient to rule out backtracking, parallel parsing, or minimal commitment parsing as candidates for theories of reanalysis, but it raises a set of computational concerns that may be addressed by limited repair parsers. The putative theoretical advantage of repair parsers depends in large part on finding simple candidate repair operations. The empirical success of repair parsers depends in large part on providing good accounts of the reanalysis phenomena foreshadowed in section 1 above. The next section presents a repair parser intended to meet these theoretical and empirical goals.

TABLE 1: Some theories of reanalysis

Backtracking		
	Overt backtracking	
		Forward (Frazier & Rayner, 1982; Lewis, 1993a)
		Backward (Frazier & Rayner, 1982)
		Selective (Carpenter & Daneman, 1981; Frazier & Rayner, 1982; Ferreira & Henderson, 1991; Inoue & Fodor, 1995)
	Covert backtracking	
		Forward
		Backward (Kaplan, 1972; Wanner & Maratsos, 1978)
		Backward, reuse of well-formed subphrases (Holbrook *et al.*, 1992)
		Selective, limited memory (Blank, 1989)
		Selective, reuse of well-formed subphrases (Abney, 1989; Konieczny, 1996)
Parallel Parsing		
	All-paths parsing	
	Ranked limited path parsing (Kurtzman, 1985; Gorrell, 1987)	
	Multi-path parsing with limited memory (King & Just, 1991; Gibson, 1991; Just & Carpenter, 1992)	
Minimal Commitment Parsing		
	Lookahead/delay (Marcus, 1980)	
	Under-specified representations (Weinberg, 1993; Gorrell, 1993)	
	Under-specified representations, language-specific control knowledge (Sturt & Crocker, 1995)	
Repair Parsing		
	Limited mechanism (Konieczny, 1996; Lehman *et al.*, 1991; Lewis, 1993a; Stevenson, 1994)	
	Limited control knowledge (Fodor & Inoue, 1995; Lombardo, this volume; Winograd, 1972)	

4 Limited Cue-Driven Repair

The particular repair mechanism described in this section was developed in the context of a detailed computational model of human sentence comprehension, NL-Soar (Lewis, 1993a). NL-Soar is based on the Soar cognitive architecture (Newell, 1990; Laird *et al.*, 1987). We will only mention here briefly three points about Soar and NL-Soar that are relevant to understanding the repair theory. First, Soar is a problem space architecture: all tasks, including language comprehension, are formulated as a series of applications of cognitive operators to states in a short-term working memory. The operators move through some problem space until a desired or goal state is found. Second, Soar adopts the *single-state assumption*: only one state per problem space is accessible in working memory at any given time. There is no architectural support for maintaining prior states in service of backtracking (Newell, 1990). Third, Soar is a theory of both deliberate and automatic behavior, so it is suitable not only for modeling long stretches (tens of seconds to minutes) of problem solving, but immediate cognitive behavior in the hundreds of milliseconds range (Newell, 1990). Embedding the comprehension model in Soar has deep implications for many aspects of sentence processing, including ambiguity resolution, but this chapter focuses only on the repair mechanism itself.

4.1 The Snip Operator

NL-Soar posits a set of rapid cognitive operators for parsing and interpretation. The most primitive parsing operations are *link* operators, which perform syntactic attachments (Figure 4), incrementally building up a parse tree.[7] Since Soar maintains only a single state, reanalysis must be achieved by either deliberate recomprehension (rereading) of the input or repair of the existing state.

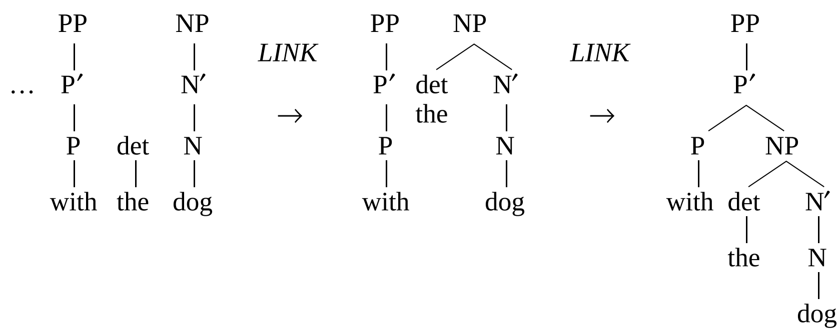


FIGURE 4: Building syntactic structure in NL-Soar. At the left is the partial structure after projecting *with* to a PP and projecting *dog* to an NP (via a series of link operators not shown). Next, a link operator attaches the determiner *the* to the NP, forming [$_{NP}$ the dog]. Finally, a link operator attaches [$_{NP}$ the dog] as the complement of the PP.

The minimal amount of new mechanism required to effect on-line repair is an operator that breaks an existing relation in the parse tree. We call this operator *snip* (Lewis, 1992). In fact, snip does not complete the repair; it just destroys a bit of structure and then lets the other link operators take over to finish the job.

[7] The tree is built in a head-driven, bottom-up fashion, though that is not a commitment that is crucial to the repair theory.

As an example of how snip works, consider what happens with the subject/object ambiguity in (9) below:

(9) Thad knows Shaq is tall.

Shaq is initially attached in the complement position of *knows*. When *is* arrives, it is projected to the sentential complement node CP.[8] Next, a link is proposed to attach the CP in the complement position of *knows*. This proposal is made because *knows* can take sentential complements as well as nominals.

Figure 5 shows what happens next. The result is a momentary superposition of two separate interpretations, with two nodes competing for the same structural position ([COMP,V′] of *knows*). This inconsistency triggers a local snip operator, which breaks the link between [$_{V'}$ knows] and [$_{NP}$ Shaq]. Next, [$_{NP}$ Shaq] is attached in subject position ([SPEC,IP]), and the repair is complete.

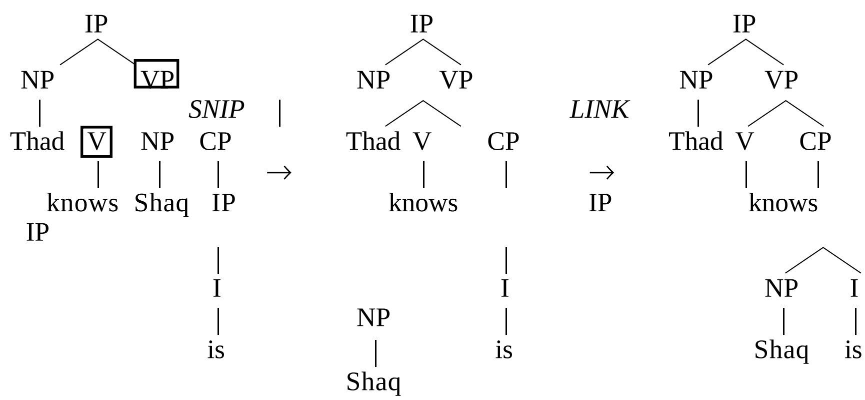


FIGURE 5: An example of simple cue-driven, destructive repair with the snip operator. The incoming CP competes for the same complement position as *Shaq*, triggering a snip to detach *Shaq*. Next, *Shaq* is attached as the subject of the incoming clause. The boxed nodes identify the locality of the inconsistency, which delimits the generation of snip operators.

The initial attachment of the sentential complement was attempted despite the fact that another constituent was already occupying the complement position. This is the only way in which the attachment operators are nonstandard: they relax the constraint on uniqueness of complements and specifiers. Fodor & Inoue (1995) adopted a similar strategy, which they

8 CPs are projected in the absence of overt complementizers, following Pritchett (1992).

dubbed ‘attach anyway’. The critical thing to note is that the constraints on the final output have *not* been relaxed. Only the set of processes and intermediate states that finally lead to the well-formed representation have changed.

4.2 Constraining the Generation of Snip

It was argued above that repair operators must be constrained in some way, else they risk opening the system up to combinatoric search as costly as arbitrary backtracking. The generation of snip is highly constrained in the following way. Snip is proposed only for structural relations that are *local* to a detected inconsistency (where *local* is defined precisely as within the maximal projection containing the inconsistency). The boxed nodes in Figure 5 identify the locality of the inconsistency. These nodes delimit the scope of consideration for generating snip operators.

The mechanism relies on simple, local structural cues to detect that something has gone wrong with the parse. This way of triggering snip serves as a way of focusing the repair directly on the problem. A free generation of snip for every link in the utterance model is possible, of course, but would radically increase the search space even for moderately-sized syntactic structures.

As an example of how the theory predicts a difficult garden path, consider the following subject/object ambiguity (Frazier & Rayner, 1982):

(10) Since Jay always jogs a mile seems like a short distance to him.

Here, *a mile* is taken initially as the complement of *jogs*, just as in (9) *Shaq* is initially attached as the complement of *knows*. Because *jogs* does not take sentential complements, the initial phrase *Since Jay jogs* is adjoined to the incoming *seems*. In fact, this is its correct final position. However, [$_{IP}$ seems] is still missing its subject. But in this case a snip operator is not generated for the complement relation between [$_{V'}$ jogs] and [$_{NP}$ a mile], because the relation is not local to the detected inconsistency (the missing obligatory subject). This situation is shown in Figure 6, with the boxed nodes again representing the locality of the inconsistency. As a result, [$_{NP}$ a mile] is not reanalyzed as the subject of [$_{IP}$ seems], and the necessary repair fails.

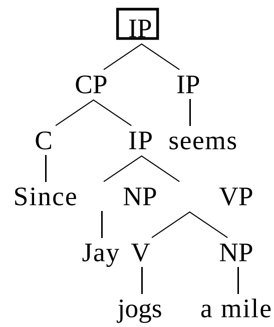


FIGURE 6: Failure to repair a subject/object ambiguity.

In addition to being a repair theory, this theory of reanalysis has a number of important features:

- The theory is formulated *independently of what guides the initial choice* at the ambiguity point. Thus, the theory classifies structure types as *potential* garden paths, not definite garden paths. The repair theory determines whether or not a given inconsistent structure can be repaired. Whether a local ambiguity gives rise to a strong garden path effect in any particular context is a function *both* of the ambiguity resolution process itself, and of the efficacy of the repair.

- The theory is a *cue-driven* theory of repair, because it depends on a few simple local cues to trigger the repair process.

- The theory is a *functional* theory, in that it posits mechanisms to efficiently carry out the functions of comprehension. NL-Soar is not a linguistic metric that distinguishes easy and difficult ambiguities. It is an implemented computational theory that classifies certain sentences as potential strong garden paths because it may fail to parse those sentences.

- The theory posits that *on-line repair is not costly*; in fact, the prediction is that these small local repairs are a frequently occurring part of normal sentence comprehension. Furthermore, the realization of repair in the Soar architecture yields the prediction that this repair may take as little as 50-100 ms more (the cost of the additional snip and reattachment processes).

- The theory posits a *minimal amount of additional new mechanisms and control* to carry out the repair. The only new operator is snip; repairs are carried out by snip and a sequence of the existing parsing operators. The only significant new piece of control knowledge is a rule that says roughly: *don't relink what you just snipped*.

4.3 Some Cross-Linguistic Predictions

This section explores predictions that the snip-based repair mechanism makes across a wide range of structural ambiguities, including some that derive from lexical categorial ambiguities. The structures presented here are a subset of the roughly 70 structures analyzed in Lewis (1993a) and Lewis & Lehman (1994).

4.3.1 Object/Subject Ambiguities

We have already seen how the theory handles two cases involving object/subject ambiguities ((9) and (10) above). The unproblematic ambiguity in (9) arises in Mandarin as well (Pritchett, 1992):

(11) a. Wo wang le Zhangsan.
I forget PERF Zhangsan
'I forgot Zhangsan.'

b. Wo wang le Zhangsan yao qu.
I forget PERF Zhangsan will go
'I forgot Zhangsan would go.'

NL-Soar repairs the Mandarin structure just as it does the English counterpart—the incoming clause attaches as the complement of [$_{V'}$ forgot], triggering the local snip of [$_{NP}$ Zhangsan]. NL-Soar also correctly predicts the following difficult subject/object garden paths in Mandarin (Gorrell, 1991) and Hebrew (Pritchett, 1992):

(12) Zhangsan yi du shu jiu diao le.
Zhangsan as-soon-as read book then fall PERF
'As soon as Zhangsan read the book fell.'

(13) Axrey she-shatiti maim hitgalu be-b'er.
After COMP-drank-1st water were-found in the well
'After I drank water was found in the well.'

The explanation is the same as in the English example (9): the local snip is not generated to remove the object NP from complement position.

4.3.2 Object/Object Ambiguities

Pritchett (1991) presents a Japanese object/object ambiguity that leads to a strong garden path effect.

(14) Frank-ni Tom-ga Guy-o syookai suru to John-wa iwaseta.
Frank-DAT Tom-NOM Guy-ACC introduce COMP John-TOP said CAUSE
'John made Frank say Tom introduced Guy.'

The initial sequence through *to* (the complementizer) is taken as a complete complementized clause:

(15) [$_{CP}$ [$_{IP}$ Frank-ni Tom-ga Guy-o syookai suru] to]
'Tom introduced Guy to Frank.'

Next, [$_{NP}$ John] is encountered and left unattached, waiting for the final verb. The final verb, *iwaseta*, is a causative verb requiring three arguments, including an obligatory *ni*-marked causee. Only two arguments are available: [$_{NP}$ John] and the initial CP. [$_{NP}$ Frank-ni] must be reanalysed as an argument of [$_{V'}$ iwaseta]. However, the required snip within the CP is not local to the VP which is missing the argument, so the repair fails, resulting in a garden path effect. The same effect shows up in Korean as well (Pritchett, 1992).

4.3.3 Complement/Adjunct Ambiguities

Incoming phrases can often be interpreted as either complements or adjuncts. Consider the unproblematic sentences in (16):

(16) a. Is the block on the table?
b. Is the block on the table red?

The prepositional phrase *on the table* may be interpreted as a modifier of *block* or the complement of *is*. Assume that the complement attachment is pursued first. When *red* arrives, it is projected to an AP and attached in complement position, which triggers the snip operator to detach [$_{PP}$ in the box] in the same manner that we have seen above. Once detached, [$_{PP}$ on the table] is simply adjoined to [$_{N'}$ block]. A similar explanation predicts the acceptability of complement clause/subject-relative ambiguities (Gibson, 1991):

(17) a. John told the man that Mary kissed Bill.
b. John told the man that kissed Mary that Bill saw Phil.

However, other complement/adjunct ambiguities do cause difficulty. Crain & Steedman (1985) found that complement clause/object-relative ambiguities produce strong garden path effects:

(18) The psychologist told the wife that he was having trouble with to leave.

Prepositional phrase argument/adjunct ambiguities may also produce noticeable garden path effects (in contrast to (16) above) (Gibson, 1991):

(19) I sent the letters to Ron to Teresa.

Unfortunately, NL-Soar cannot account for these effects, because the repair succeeds in both cases, as in (16). It might be possible to find some independent source of difficulty, but as currently formulated, the theory is making an incorrect prediction.

Crain & Steedman (1985) found that a garden path arises when a clause is interpreted as a relative clause but the complement reading is required. This is the opposite of the standard complement/adjunct garden path above in (18), in which the ambiguous clause is initially taken as a complement, but the relative reading is required (Crain & Steedman induced the relative reading through a contextual manipulation, not reproduced here):

(20) The psychologist told the wife that he was having trouble with her husband.

NL-Soar can account for this effect: even if [$_{NP}$ her husband] is attached as the complement of [$_{P'}$ with], the critical snip operator required to detach the clause from [$_{N'}$ wife] will not be generated.

4.3.4 Main Verb/Relative Ambiguities

We now explore variations on the classic relative clause garden path. Consider the canonical example (Bever, 1970):

(21) The horse raced past the barn fell.

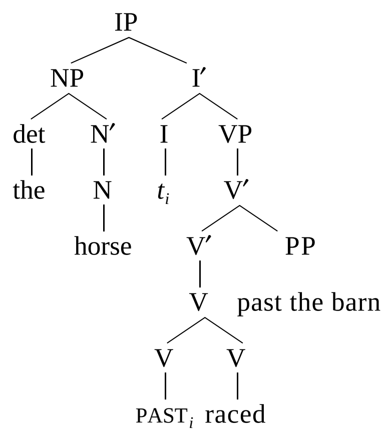
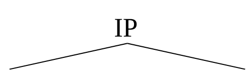


FIGURE 7: The main verb reading of *The horse raced past the barn*. The inflectional features that head the IP phrase adjoin to the verb, leaving a trace in the head of IP.

Figure 7 shows the complete structure for the main verb interpretation of (21). The inflectional features that head the IP phrase adjoin to the verb, leaving a trace in head of IP. (This joining of inflectional features to the verb is assumed in some form by many syntactic theories; e.g., McCawley, 1988, calls it *Tense-hopping*, and assumes an adjunction structure like the one presented here.)[9] On the other hand, passive forms like *driven* and *raced* are untensed. In the reduced relative reading of *the horse raced past the barn*, which uses the passive interpretation of *raced*, the inflection is not present.

Consider the repair required to successfully parse (21). The structure in Figure 7 must be repaired into the reduced relative structure. This involves removing [$_{NP}$ the horse] from [SPEC,IP] position, and snipping the adjoined inflectional features. When *fell* arrives and is projected to VP, the only place it may attach is in complement position of I′. This produces an inconsistency local to the IP, as shown in Figure 8. However, this fails to trigger all the required snips; in particular, the crucial inflection adjunction is left undisturbed, so the passive reading cannot be recovered.



[9] This particular adjunction analysis is not critical for the empirical predictions; what is important is that the tense information is local to the verb.

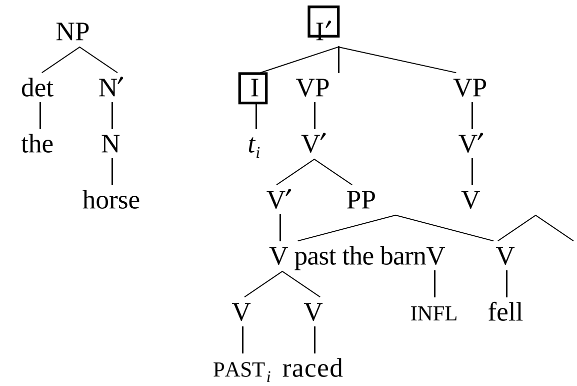


FIGURE 8: The main verb/reduced relative garden path effect. When *fell* arrives and is projected to VP, the only place it may attach is in the complement position of I′. This produces an inconsistency local to IP (denoted by the boxed nodes). However, this fails to trigger all the required snips; in particular, the crucial inflection adjunction is left undisturbed, so the required passive reading cannot be recovered.

The intervening modifier [$_{PP}$ past the barn] is irrelevant to this explanation. Thus, the theory correctly predicts the existence of very short reduced relative garden paths (Kurtzman, 1985; Abney, 1989):

(22) The boat floated sank.

Such examples are one demonstration of the independence of garden path effects from length effects (Pritchett, 1992; Lewis, 1993a). The explanation also extends to ditransitive verbs (23) (Rayner *et al.*, 1983) and embedded relatives (24) (Gibson, 1991):

(23) My friend sent the flowers smiled broadly.

(24) The dog that was fed next to the cat walked to the park chewed the bone.

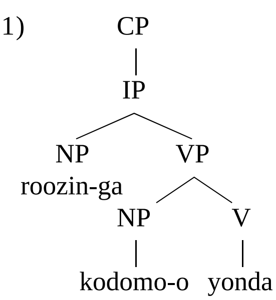
Surprisingly, not all main verb/relative clause ambiguities produce garden path effects. Mazuka *et al.* (1989) present an interesting unproblematic Japanese construction involving a main verb/relative ambiguity:

(25) a. Roozin-ga kodomo-o yonda.
old man-NOM child-ACC called
(The old man called the child.)

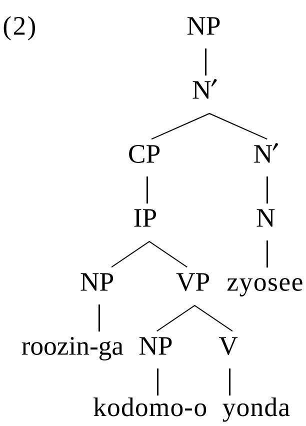
b. Roozin-ga kodomo-o yonda zyosee to hanasi-o sita.
old man-NOM child-ACC called woman with talk-ACC did
(The old man talked with the woman who called the child.)

In (25a), the NP NP V sequence *roozin-ga kodomo-o yonda* is interpreted as the main clause *The old man called the child*. In (26b), the relative clause reading is required, disambiguated by the appearance of $[_{NP}$ zyosee]. Unlike the familiar English main verb/reduced relative ambiguity, NL-Soar can repair the structure in (25b). Figure 9 shows how. The main clause interpretation is pursued initially, with $[_{NP}$ roozin] in subject [SPEC,IP] position and $[_{NP}$ kodomo-o] in complement position of $[_{VP}$ yonda]. Next, $[_{NP}$ zyosee] arrives and the CP adjoins to $[_{N'}$ zyosee] as a modifying clause (unlike the English version, the relative clause is active, not passive, and therefore the clause remains tensed). The appropriate traces are generated in [SPEC,CP] and [SPEC,IP] position, in the same manner as English relative clauses. The [SPEC,IP] trace creates a local inconsistency at the IP node, triggering a snip of $[_{NP}$ roozin]. $[_{NP}$ roozin] is now available to attach as the subject of the incoming $[_{IP}$ to hanasi-o sita], and the repair succeeds.

(1)



(2)



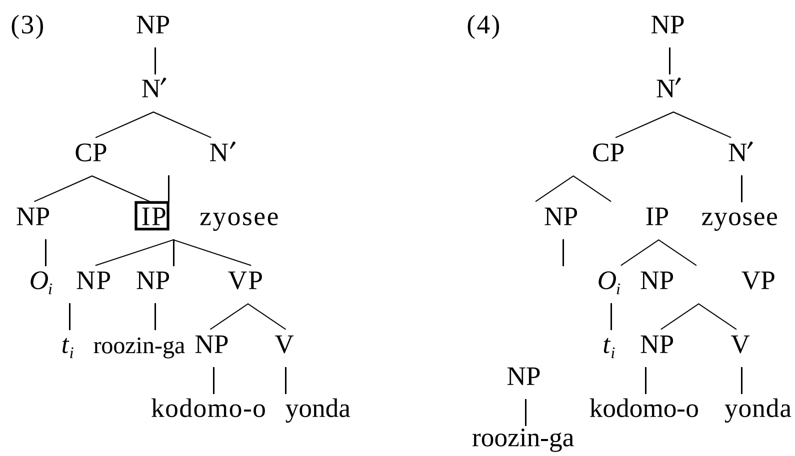


FIGURE 9: Repairing an unproblematic Japanese main verb/relative clause ambiguity. The initial CP is attached as a relative modifier of the incoming NP, leading to the creation of a trace in [SPEC,IP]. This triggers the snip of the misanalyzed NP, making it available to serve as subject of the main clause.

4.3.5 Lexical Categorial Ambiguities

Lexical ambiguity often gives rise to structural ambiguity. Consider the basic noun/adjective ambiguity in (26):

(26) a. The square is red.
b. The square table is red.

When *square* arrives, both categories are retrieved in parallel from the lexicon, and NP and AP nodes are projected. Next, the determiner *the* is attached in [SPEC,NP] position, forming [$_{NP}$ the square]. Then *table* arrives and is projected to NP. Next, the adjective phrase [$_{AP}$ square] is adjoined to [$_{N'}$ table]. Each syntactic link is well-formed, but two mutually incompatible bits of structure have been produced, since the single token *square* cannot simultaneously be an adjective and a noun.

A snip operator is immediately triggered by the inconsistency of syntactic structure attached to competing senses of the same lexical token. A locally generated snip breaks the [SPEC,NP] link between the determiner and [$_{NP}$ square]. Next, a link operator attaches the determiner in specifier position of [$_{NP}$ table], forming [$_{NP}$ the square table]. This completes the repair. In this way, the snip operator extends naturally to handle repairs involving lexical ambiguity.

We have now seen all three kinds of structural cues that trigger the snip operator:

1. multiple phrases competing for the same structural position;

2. missing obligatory phrases (e.g., missing subjects); and

3. attachments to competing categorial interpretations of the same lexical item.

NL-Soar also handles multiple lexical ambiguities. For example, noun/verb ambiguities may be preceded by adjective/noun ambiguities without causing difficulty (Milne, 1982; Pritchett, 1992):

(27) a. The square blocks the triangle.
b. The square blocks are red.

NL-Soar effects the repairs in (27) in the same manner as (26); see Lewis (1993a) for details. Yet some noun/verb ambiguities do cause difficulty. If the unproblematic ambiguity in (27) is followed by a reduced relative, the result is a strong garden path (Milne, 1982; Pritchett, 1992):

(28) The building blocks the sun faded are red.

Suppose that *blocks* is initially taken as the main verb, and *sun* as the complement. When *faded* arrives, it can be attached as a reduced relative modifying *sun*. Once *are* is projected to an IP, no additional attachments are possible. Furthermore, there are no local inconsistencies to generate a snip, so the repair is never initiated.

4.3.6 Filler-Gap Ambiguities

Filler-gap sentences provide another interesting test of the repair mechanism, because the location of the gap is often not known with certainty. Consider (29):

(29) a. John found the ball$_i$ that the boy hit t_i.
b. John found the ball$_i$ that the boy hit the window with t_i.

In (29a), the trace or gap appears in the complement position of the verb (*the boy hit the ball*), while in (29b), the sentence continues so that the trace appears as the object of a preposition (*the boy hit the window with the ball*). Neither sentence causes difficulty. Figure 10 shows how the repair of the object trace is triggered. First, [$_{NP}$ the window] arrives and attaches as the complement of [$_{V'}$ hit]. This creates the familiar local inconsistency: two nodes (the trace, and the NP) occupying the same structural position. A snip is generated to remove the trace. When [$_{PP}$ with] adjoins to [$_{V'}$ hit], a new trace is generated as the object of the preposition.

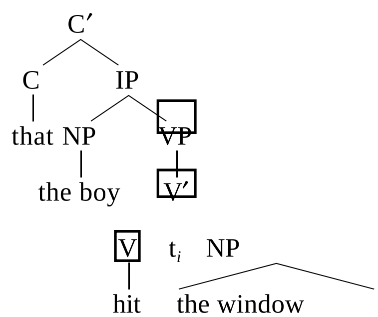


FIGURE 10: Triggering the repair of a filler-gap ambiguity.

A striking aspect of these filler-gap ambiguities, noted in the introduction, is that they may be propagated over long distances:

(30) a. Who do you believe?
b. Who do you believe Jim suspects Bob knows Pat hates?

An object trace is posited after each verb in (30b). As an NP arrives to fill the object slot, the trace is snipped, and a new one generated at the next verb. Thus, (30b) involves at least three separate repair operations. These repair operations provide an explanation for the filled-gap effect—the increased duration of eye fixations on the overt NPs that occupy the misanalyzed gaps (e.g., Stowe, 1986).

4.3.7 Other Examples

The snip-based repair mechanism is quite simple and limited, but surprisingly effective. The sentences below give some additional examples of local ambiguities that are easily repaired by snip:

(31) a. Without her we failed. (Pritchett, 1988)
b. Without her contributions we failed.

(32) a. The cop gave her earrings. (Gibson, 1991)
b. The cop gave her earrings to the dog.

(33) a. The defendant examined the evidence. (Pritchett, 1988)
b. The defendant examined by the lawyer shocked the jury.

(34) a. I like green. (Pritchett, 1988)
b. I like green dragons.

(35) a. I know that. (Gibson, 1991)
b. I know that boy.
c. I know that dogs should play.

(36) a. I saw her duck fly away. (Pritchett, 1992)
b. I saw her duck into an alleyway.

(37) a. I went to the store.
b. I went to the store and the barber.

Other examples of impossible-to-repair structures include:

(38) Without her contributions failed to come in. (Frazier, 1978)

(39) Sue gave the man who was racing the car. (Pritchett, 1992)

(40) The cotton clothing is made of grows in Mississippi.(Marcus, 1980)

(41) Before she knew that she went to the store.

(42) That coffee tastes terrible surprised John. (Gibson, 1991)

The structures in (39) and (41) are particularly interesting because they illustrate how *overt* cues are required to trigger the repair operation (Fodor & Inoue, 1995); if the parser assigns the full string a partial structure that is well-formed, there are no local inconsistencies to get the repair started, and the parser simply halts. The contrasts between the easy and difficult *that*-ambiguities are also interesting, because they clearly demonstrate the importance of local structural context in guiding the parser to the right categorial sense of ambiguous words. Switching syntactic categories *per se* is not necessarily difficult.

5 Toward a Complete Theory of Garden Path Effects

The on-line repair model presented here is actually just part of NL-Soar's account of garden path effects. The other pieces of the puzzle are NL-Soar's theory of ambiguity resolution, and its model of limited syntactic short-term memory. The following is a just a brief sketch of the theory; for further details, see Lewis (1993a, 1996b). The main point of this section is to provide a glimpse of how the simple repair mechanism described here can be embedded in a complete architecture for comprehension, and how the resulting theory can provide explanations for a wider variety of reanalysis-related phenomena.

The major components of NL-Soar are a short-term memory for syntactic structure subject to similarity-based interference (Lewis, 1996b), an open control structure for ambiguity resolution, and the limited repair mechanism outlined in this chapter. The structure of the working memory is theoretically motivated by computational concerns, and empirically motivated by difficulty with center-embedded structures. NL-Soar's ambiguity resolution is open to potentially any knowledge source (semantic and contextual) because Soar's control structure permits any long-term knowledge to be brought to bear at each decision point.[10] The combination of the snip-based repair mechanism and the open control structure means that NL-Soar can account for both the structural regularities captured by the syntactic metrics of garden path difficulty (e.g, Pritchett, 1992; Gibson, 1991), and the flexible, context-sensitive ambiguity resolution of interactive theories (e.g, Altmann & Steedman, 1988).

The open control of parsing also means that once NL-Soar's on-line repair process has failed, it can engage in a *deliberate* recovery process by carefully rereading the problematic sentence. 'Carefully' is given a precise computational definition in Soar: in effect, it means that Soar is forcing itself to reconsider (albeit more slowly) decisions that it previously made automatically (Laird, 1988). This has three implications. First, NL-Soar predicts that, given time, people can recover from difficult garden paths. Second, because of Soar's continuous learning, the model actually acquires new ambiguity resolution strategies as a result of this more careful

[10] However, for a semantic or contextual parsing strategy to be brought to bear on-line, it must be in a highly compiled form so that it can be retrieved in real-time at the point of the ambiguity. There are no guarantees that any given semantic or contextual strategy will be available in this form at just the right time. As a result, the system may yield modular behavior (ignoring semantic and contextual constraints) either because some other ambiguity resolution strategy is more highly automatized and causes the system to charge down the wrong path blindly, or because there is not enough time to deliberate about the decision.

recomprehension. In future similar contexts, the system may not necessarily be led down the garden path. Third, this deliberate recovery process may provides a locus for developing an account of the length-based differences in reanalysis of difficult garden path structures demonstrated by Ferreira & Henderson (1991).

The limited short-term memory affects both ambiguity resolution and repair. Although NL-Soar was described here as if it kept complete parse trees in working memory, in fact that is not the case. Constraints on working memory mean that there is a severe limit on the number of possible attachment sites that can be kept open at any given time; in short, some structural ambiguities are not even detected. Working memory constraints also affect the repair processes. If there is too much structure intervening between the ambiguity and the disambiguating material, the structural relations that must be repaired may no longer be in working memory. This explains how Warner & Glass (1987) were able to turn otherwise easy subject/object ambiguities (of the *I forgot Pam needed…* variety) into severe garden path structures:

(43) The girls believe the man who believes the very strong ugly boys struck the dog killed the cats.

6 Conclusion

This chapter developed a theory of reanalysis based on limited repair. The theory has both computational and empirical motivation, as evidenced by the computational comparison to other theories of reanalysis, and by the account of a wide range of cross-linguistic reanalysis effects. To summarize the major points:

1. Reanalysis is a functional requirement for any natural language parser: all parsers must sometimes revise the interpretation of locally ambiguous material based on later information.

2. The four major kinds of parsing models that handle reanalysis are backtracking, parallel parsing, minimal commitment parsing, and repair parsing. As theories of covert reanalysis, the first three suffer from several computational or theoretical drawbacks which repair parsers do not share. These include increased memory demands and requirements for additional mechanisms to perform structure copying or sharing. Although backtracking does not seem suitable as a theory of covert, on-line reanalysis, selective backtracking provides a suitable theory of deliberate and overt recomprehension.

3. Repair parsers are an effective parsing technique because they reformulate the parsing search space to support efficient depth-first parsing without backtracking. Repair parsers use operations that modify or repair existing syntactic structure. These operations permit the parser to 'leap off' the garden path in certain cases back onto the path to a successful parse.

4. Repair operations must be limited in some way, otherwise repair will give way to combinatoric search as costly as general backtracking.

5. Limited cue-based repair, based on a new destructive parsing operation (snip), provides one promising mechanism for repair. This theory suggests that the repair itself is not difficult. On the contrary, limited repair is an on-line process that is frequently invoked to handle reanalysis. Difficulty arises when the appropriate local cues are not present to trigger the repair. Those structures that do not provide the right cues seem to map fairly well onto the class of structures that cause noticeable garden path effects, cross-linguistically.

6. Limited repair mechanisms can be embedded in complete architectures for comprehension, yielding more comprehensive theories of reanalysis effects which explain how short-term memory, experience, and context affect both on-line and deliberate reanalysis.

It is encouraging to see several different theoretical approaches—in particular, Stevenson's (1994) competition model, Fodor & Inoue's (1995) diagnosis model, Lombardo's dependency parser (this volume), and NL-Soar—all converge on limited repair as a fundamental component of human sentence processing. If the general analysis of repair parsing provided in this chapter is correct, this should not be surprising, because we have the beginnings of a computational basis for understanding why repair is a simple and effective way of handling local ambiguity.

References

Abney, S.P. 1989. A computational model of human parsing. *Journal of Psycholinguistic Research*, 18, 1, 129-144.

Altmann, G. & Steedman, M. 1988. Interaction with context during human sentence processing. *Cognition*, 30, 191-238.

Bever, T.G. 1970. The cognitive basis for linguistic structures. In J.R. Hayes (ed.), *Cognition and the Development of Language*. New York, NY: Wiley, 279-362.

Blank, G.D. 1989. A finite and real-time processor for natural language. *Communications of the ACM*, 32, 10, 1174-1189.

Carpenter, P.A. & Daneman, M. 1981. Lexical retrieval and error recovery in reading: A model based on eye fixations. *Journal of Verbal Learning and Verbal Behavior*, 20, 137-160.

Crain, S. & Steedman, M. 1985. On not being led up the garden path: The use of context by the psychological syntax processor. In D.R. Dowty, L. Karttunen, & A.M. Zwicky (eds.), *Natural Language Parsing*. Cambridge, U.K.: Cambridge University Press, 320-358.

Ferreira, F. & Henderson, J.M. 1990. Use of verb information in syntactic parsing: Evidence from eye movements and word-by-word self-paced reading. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 16, 555-568.

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 paths. *Journal of Psycholinguistic Research,* 23, 407-434.

Fong, S. & Berwick, R. 1990. The computational implementation of principle-based parsers. In M. Tomita (ed.), *Proceedings of the First International Workshop on Parsing Technologies*. Pittsburgh, PA: Carnegie Mellon University, 75-84.

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

Gathercole, S.E. & Baddeley, A.D. 1993. *Working Memory and Language*. Hove, UK: Lawrence Erlbaum Associates.

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.

Godden, K. 1990. Lazy unification. *Association for Computational Linguistics*, 28, 180-187.

Gorrell, P 1993. Incremental structure building and the determinism hypothesis. Paper presented at *Sixth Annual CUNY Conference on Human Sentence Processing,* Amherst, MA.

Gorrell, P. 1991. Subcategorization and sentence processing. In R.C. Berwick, S.P. Abney, & C. Tenny (eds.), *Principle Based Parsing: Computation and Psycholinguistics*. Dordrecht: Kluwer Academic Publishers, 279-300.

Gorrell, P. 1987. *Studies of Human Syntactic Processing: Ranked-Parallel versus Serial Models*. Unpublished doctoral dissertation, University of Connecticut, Storrs, CT.

Holbrook, J.K., Eiselt, K.P., & Mahesh, K. 1992. A unified process model of syntactic and semantic error recovery in serntence understanding. In *Proceedings of the 14th Annual Conference of the Cognitive Science Society*, 195-200.

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.

Juliano, C. & Tanenhaus, M.K. 1994. A constraint-based lexicalist account of the subject/object attachment preference. *Journal of Psycholinguistic Research*, 23, 6, 459-471.

Just, M.A. & Carpenter, P.A. 1992. A capacity theory of comprehension: Individual differences in working memory. *Psychological Review*, 99, 1, 122-149.

Kaplan, R. 1972. Augmented transition networks as psychological models of sentence comprehension. *Artificial Intelligence*, 3, 77-100.

King, J. & Just, M.A. 1991. Individual differences in syntactic processing: The role of working memory. *Journal of Memory and Language*, 30, 580-602.

Konieczny, L. 1996. *Human Sentence Processing: A Semantics-Oriented Parsing Approach*. Unpublished doctoral dissertation, University of Freiburg.

Kurtzman, H.S. 1985. *Studies in Syntactic Ambiguity Resolution*. Unpublished doctoral dissertation, MIT, Cambridge, MA.

Laird, J.E. 1988. Recovery from incorrect knowledge in Soar. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, The American Association for Artificial Intelligence, 618-623.

Laird, J.E., Newell, A., & Rosenbloom, P.S. 1987. Soar: An architecture for general intelligence. *Artificial Intelligence*, 33, 1-64.

Lehman, J.F., Lewis, R.L., & Newell, A. 1991. Integrating knowledge sources in language comprehension. In *Proceedings of the 13th Annual Conference of the Cognitive Science Society*, 461-466. Also in P.S. Rosenbloom, J.E. Laird, & A. Newell (eds.), *The Soar Papers: Research on Integrated Intelligence*. Cambridge, MA: MIT Press, 1993, 1309-1314.

Lewis, R.L. 1992. Recent developments in the NL-Soar garden path theory. Technical Report CMU-CS-92-141, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA.

Lewis, R.L. 1993a. *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.

Lewis, R.L. 1993b. An architecturally-based theory of sentence comprehension. In *Proceedings of the 15th Annual Conference of the Cognitive Science Society*, 108-113.

Lewis, R.L. 1996a. Architecture matters: What Soar has to say about modularity. In D. Steier & T. Mitchell (eds.), *Mind Matters: Contributions to Cognitive and Computer Science in Honor of Allen Newell*. Hillsdale, NJ: Lawrence Erlbaum, 75-84.

Lewis, R.L. 1996b. Interference in short-term memory: The magical number two (or three) in sentence processing. *Journal of Psycholinguistic Research*, 25, 1, 93-115.

Lewis, R.L. & Lehman, J.F. 1994. A theory of the computational architecture of sentence comprehension. Unpublished manuscript, Cognitive Science Laboratory, Princeton University, Princeton, NJ.

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

Lytinen, S.L. & Tomuro, N. 1996. Left-corner unification-based natural language processing. In *Proceedings of the 13th National Conference on Artificial Intelligence*. The American Association for Artificial Intelligence, 1037-1043.

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

Marcus, M.P. 1980. *A Theory of Syntactic Recognition for Natural Language*. Cambridge, MA: MIT Press.

Mazuka, R., Itoh, K., Kiritani, S., Niwa, S., Ikejiru, K., & Naitoh, K. 1989. Processing of Japanese garden path, center-embedded, and multiply left-embedded sentences. *Annual Bulletin of the Research Institute of Logopedics and Phoniatrics, Vol. 23*. University of Tokyo, 187-212.

McCawley, J.D. 1988. *The Syntactic Phenomena of English, Vol. 1*. Chicago, IL: University of Chicago Press.

Milne, R.W. 1982. Predicting garden path sentences. *Cognitive Science*, 6, 349-373.

Minton, S., Johnston, M.D., Philips, A.B., & Laird, P. 1992. Minimizing conflicts: A heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence*, 58, 161-205.

Mitchell, D. 1987. Lexical guidance in human parsing: Locus and processing characteristics. In M. Coltheart (ed.), *The Psychology of Reading, Attention and Performance, Vol. 12*. Hillsdale, NJ: Lawrence Erlbaum, 601-681.

Nadel, B.A. 1990. Representation selection for constraint satisfaction: A case study using *n*-queens. *IEEE Expert*, 16-23.

Newell, A. 1990. *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.

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

Pritchett, B.L. 1991. Head position and parsing ambiguity. *Journal of Psycholinguistic Research*, 20, 251-270.

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.

Sosic, R. & Gu, J. 1990. A polynomial time algorithm for the *n*-queens problem. *SIGART Bulletin*, 1, 3, 7-11.

Stevenson, S. 1994. Competition and recency in a hybrid network model of syntactic disambiguation. *Journal of Psycholinguistic Research*, 23, 4, 295-322.

Stowe, L. 1986. Evidence for on-line gap location. *Language and Cognitive Processes*, 1, 277-245.

Sturt, P. & Crocker, M. 1995. Monotonic parsing and reanalysis. Paper presented at *Eighth Annual CUNY Sentence Processing Conference*, Tucson, AZ.

Trueswell, J.C., Tanenhaus, M.K., & Garnsey, S.M. 1994. Semantic influences in parsing: Use of thematic role information in syntactic ambiguity resolution. *Journal of Memory and Language*, 33, 285-318.

Wanner, E. & Maratsos, M. 1978. An ATN approach to comprehension. In M. Halle, J. Bresnan, & G.A. Miller (eds.), *Linguistic Theory and Psychological Reality*. Cambridge, MA: MIT Press, 119-161.

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

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

Winograd, T. 1972. Understanding natural language. *Cognitive Psychology*, 3, 1-191.

Winograd, T. 1983. *Language as a Cognitive Process, Vol. 1: Syntax*. Reading, MA: Addison Wesley.

Winston, P.H. 1992. *Artificial Intelligence, Third Edition*. Reading, MA: Addison Wesley.