Journal of Psycholinguistic Research, Vol. 23, No. 5, 1994

Processing Empty Categories: A Parallel Approach

Edward Gibson,[1,4] Gregory Hickok,[2] and Carson T. Schütze[3]

In many current theories of human sentence processing, the mechanisms and principles proposed to account for data related to processing (1) long-distance dependencies and (2) structural ambiguities, such as garden-path sentences, are independent, despite the fact that deciding whether or not to posit a gap is just a special case of ambiguity resolution. In this paper we demonstrate how the parallel parsing theory proposed by Gibson (1991, in press)—which was developed to account for nongap ambiguity resolution data—also explains a number of gap-positing facts, without additional strategies. In particular, we show how this theory correctly explains filled-object-gap effects and the lack of filled-subject-gap effects in English, as well as certain gap-processing effects in Dutch.

INTRODUCTION

There is a sizable literature on how empty categories, such as traces of long-distance movement, are processed (see Fodor, 1978; Clifton & Frazier, 1989;

We would like to thank the following for helpful comments and discussions on this material: Masha Babyonyshev, Julie Boland, Enriqueta Canseco-Gonzalez, Chuck Clifton, Matt Crocker, Fernanda Ferreira, Neal Pearlmutter, Colin Phillips, Orin Percus, Martin Pickering, Suzanne Stevenson, Laurie Stowe, Amy Weinberg, Ken Wexler, and members of the 1994 CUNY conference audience.

[1] Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

[2] Laboratory for Cognitive Neuroscience, The Salk Institute, La Jolla, California 92037.

[3] Department of Linguistics and Philosophy, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

[4] Address all correspondence to Edward Gibson, Department of Brain and Cognitive Sciences, E10-034C, MIT, Cambridge, Massachusetts 02139.

Frazier & Clifton, 1989; Boland & Tanenhaus, 1991; Swinney, 1991; and Frazier, 1993, for summary and review). This work has been aimed at understanding how and when dependencies between a filler and a gap are established. For example, in sentence 1, one issue is at what point in the parse the filler *what* is associated with the gap in direct object position:

1. What$_i$ did John paint t_i with the brush?

As pointed out by Fodor (1978), there are a number of possibilities: (i) The parser could follow a last-resort strategy, waiting for structural information that unambiguously identifies the true gap site (e.g., the end of the sentence); (ii) it could follow a first-resort strategy, positing a gap sooner, perhaps upon processing the verb, without waiting for structural confirmation; or (iii) gaps could be posited in a more probabilistic manner, according to lexical expectations, for example. Many variants of these possibilities have been proposed (see, e.g., Crocker, 1994; Gibson & Hickok, 1993; Pickering & Barry, 1991; Stevenson, 1994; and the references in each), but perhaps the best known of these is the active filler strategy [Clifton & Frazier, 1989; Frazier, 1987b; Frazier & Clifton, 1989; Frazier & Flores d'Arcais, 1989; cf. De Vicenzi's (1991) minimal chain principle], a variant of the first-resort strategy:

2. Active filler strategy (AFS): Assign an identified filler as soon as possible; i.e., rank the option of a gap above the option of a lexical noun phrase within the domain of an identified filler. (Frazier & Flores d'Arcais, 1989).

The AFS predicts that, given the option of positing a gap or waiting to see if a lexical NP complement appears, the parser will prefer to posit a gap. In the above example, then, the parser would posit a gap immediately upon encountering the verb, prior to the appearance of *with.* This gap-finding strategy thus predicts that if it turns out that there is a lexical NP in direct object position as in sentence 3 below, then some processing difficulty should arise:

3. What$_i$ did John paint the deck with t_i?

Such an effect has been demonstrated experimentally, and has come to be known as the "filled-gap" effect (Crain & Fodor, 1985; Stowe, 1986). Consider the following example from Stowe's experiment:

4. a. No-gap control sentence:
   My brother wanted to know if Ruth will bring us home to Mom at Christmas.
   b. Filled-object gap sentence:
   My brother wanted to know who Ruth will bring us home to at Christmas.

If, in the filled-object gap sentence 4b, the parser posits a direct-object gap prior to encountering the lexical NP in direct object position (*us*)—as the AFS predicts—then the reading times should increase relative to the non-gap control sentence 4a when a lexical NP appears in that position. This prediction is confirmed: The postverbal NP *us* takes significantly longer to read in filled-object gap sentences than in no-gap control sentences, thus providing support for the AFS.

However, another aspect of Stowe's (1986) experiment appears to raise problems for the AFS. Stowe also measured reading times for lexical NPs in subject position, such as *Ruth* in the filled-object gap sentence. The significance of this measurement is that such NPs represent a filled-subject gap; that is, the subject position is a potential gap site for the filler but that position is filled with a lexical NP. Since the AFS states that the parser will posit a gap rather than wait for a lexical NP, the AFS predicts that filled-subject gaps should also lead to processing difficulty and hence increased reading times. This prediction was not confirmed, however: Stowe reported that reading times for the NP *Ruth* in no-gap versus gap sentences did not differ. Thus, the lack of a filled-gap effect for filled-subject gaps is a surprising result from the perspective of the AFS.

Clifton & Frazier (1989, p. 313) recognized this difficulty for the AFS and speculated on two possible explanations for the lack of a filled-subject-gap effect. The first is that the lexical subject appears in the input before the parser can process the gap licensing structures. The second is that there is more ambiguity concerning "when the S will begin following the *who* than there is . . . when an NP will begin following a verb . . . discouraging the expectation of particular constituents in the former case." However, no independent evidence for either of these proposals is given. In light of this problem, we propose an account of gap-positing under which the contrast between subject and object position effects follows without stipulation.[5]

In addition to the literature on gap-filling strategies, there is also an extensive literature on how the human parser resolves structural ambiguities that do not involve gap-positing ambiguities; the two domains have often

[5] See Stevenson (1993, 1994) and Crocker (1994) for interesting alternative approaches to this issue.

been treated separately, but we argue that a unified approach is superior. Perhaps the best known of the proposed theories of structural ambiguity resolution is the principle of minimal attachment (Frazier, 1978; Frazier & Fodor, 1978; Frazier & Rayner, 1982):

5. Minimal Attachment: Attach incoming material into the phrase-marker being constructed using the fewest nodes consistent with the well-formedness rules of the language (Frazier & Rayner, 1982).

Under minimal attachment, the human parser initially follows a single structural analysis: the one that requires the fewest new phrase structure nodes to be constructed. Although this principle can account for a number of psycholinguistic effects (see Frazier, 1987a, for review), it cannot by itself account for gap effects.

In contrast, we with to propose an alternative theory that offers a uniform treatment of gap-filling ambiguities and other structural ambiguities in a parallel framework.[6] In particular, if we adopt the parallel parsing assumptions of Gibson (1991, in press), many gap effects are explained straightforwardly, with no extra strategies required. We show that the processing of gaps can be treated as just a special case of ambiguity resolution, wherein the decision as to whether to posit a gap at a particular position in a sentence is simply an instance of a local/temporary ambiguity. Thus, a more parsimonious theory results. This approach follows the current trend away from construction-specific parsing strategies, toward more general processing principles.

The balance of the paper is organized as follows. The second section provides an overview of the relevant components of the parsing framework proposed in Gibson (1991, in press), with discussion of how it explains a number of linguistic complexity facts. The third section shows how this parsing model can account for filled-gap effects in English, including the lack of a filled-subject-gap effects. The fourth section demonstrates how the parallel model can account for gap-positing effects in Dutch, a language from which the AFS has gained much of its support. Conclusions are given in the final section.

THE UNDERLYING PARSER

The parsing algorithm that we will use in explaining gap-filling is based on that proposed by Gibson (1991, in press), which we will now lay out.

[6] In a precursor to this work, this general hypothesis was also put forward by Gibson and Clark (1988), but without a satisfactory complexity theory, or explanations of empirical gap effects, such as filled-gap effects.

The intuition underlying the algorithm is that all locally possible readings of the input string can be retained by the parser in parallel, up to memory constraints on the relative costs of the structures. These costs, to be detailed below, are computed for each possible structure and compared every time a new input item is processed (e.g., words and empty categories). The result is a parser that allows limited parallelism by implementing a form of beam search,[7] maintaining only options that are within a certain cost of the structure that is currently the best (least expensive).

The limit on the parallelism—which is motivated by the fact that people have limited computational and memory resources—predicts the existence of two classes of ambiguities: those that can be analyzed within the beam width (easy ambiguities) and those for which it is necessary to go back to a structure that has fallen outside the beam width (hard ambiguities: garden-path effects). We propose that those ambiguities that require reanalysis, by virtue of lying outside the beam width, are those that give rise to conscious difficulty. The metric for determining the cost of a structure is composed of a number of factors related to the satisfaction of both linguistic and nonlinguistic constraints. [For other examples of constraint-satisfaction parsers in the same spirit, see Altmann & Steedman (1988), Crain & Steedman (1985), MacDonald (1993), MacDonald, Pearlmutter, & Seidenberg (1993), Trueswell, Tanenhaus, & Garnsey (1994), Trueswell, Tanenhaus, & Kello (1993).]

To date, the constraints whose violation has been proposed to incur cost for a partial sentence structure have been based on the θ-criterion of government binding theory (see Chomsky, 1981; see Pritchett, 1988, for a precursor to this work within a serial architecture) and a recency preference (cf. Frazier, 1978; Frazier & Fodor, 1978; Kimball, 1973). Only the former type will be relevant in this paper. Their intuitive basis is that local violations of the θ-criterion constitute unsatisfied constraints that should be minimized, so that each such violation in a possible partial structure for the input string incurs a cost. That is, the cost of a representation is a function of the number of elements that require θ-roles but cannot yet be assigned them and the number of elements that have θ-roles that need to be assigned but cannot yet assign them.

In order to define this notion formally, we first need to consider how the parser builds the tree-structure representations of a sentence. Following Kimball, (1973, 1975), Frazier (1978), and Frazier and Fodor (1978), it is assumed that the parser operates in a partly top-down, partly bottom-up fashion, using a head-corner parsing algorithm (Proudian & Pollard, 1985),

[7] For an overview of beam search and other search techniques, see an introductory artificial intelligence reference, e.g., Winston (1984).

which is closely related to a left-corner parsing algorithm (see, e.g., Aho & Ullman, 1972; see Gibson, in press, for complexity-based motivations for such an algorithm). Under the proposed head-corner parsing algorithm, complete X-bar representations (Chomsky, 1986; Jackendoff, 1977) for the incoming words are accessed from the mental lexicon (Flickinger, Pollard, & Wasow, 1985). These structures include all rightward predictions of X-bar structures that can potentially be in a sisterhood relationship with one of the X-bar projections of the current head, with further prediction possible if the first predicted node potentially lacks overt content (e.g., because complementizers can be null, an IP can be predicted from a predicted C^0 head). Nodes that are either headed by lexical material or grammatically licensed empty categories are designated "c-nodes," indicating that they have been confirmed. The predicted nodes, which are called "h-nodes" (hypothesized nodes), are determined by the lexical requirements of the words involved, together with the word-order parameter values and the category-specific modifier, specifier, and argument dependencies of the language in question. For example, an h-node IP is predicted above an English lexically headed c-node NP, because NP specifiers of IPs occur before the I^0 head of IP in English. In a head-final language like Dutch, an NP will also predict a VP and its V^0 head to the right, because verbs occur to the right of their arguments in Dutch. Furthermore, all potential posthead arguments and modifiers of the NP will also be predicted to the right. Using this approach to category prediction, structures are assembled by matching a projection of the current input item (either c-node or h-node) to a previously constructed h-node.

Given these concepts, we can set out the two relevant properties that are used in assessing the cost of a structure, measured on a scale of processing load units (PLUs); this cost is proposed to have some grounding in short-term memory requirements.

6. Property of thematic reception[8]: Associate a cost of x_{TR} PLUs to each chain containing a c-node that is in a position that can be associated with a θ-role in any of the structures currently under consideration and whose role-assigner is not unambiguously identifiable.
7. Property of lexical requirement: Associate a cost of x_{LR} PLUs to each unsatisfied lexical requirement position that is obligatory in any of the structures currently under consideration.

In the first definition, a position that can be associated with a θ-role is one that can form part of a chain whose tail is in an argument (i.e., θ-marked) position. For a role-assigner to be unambiguously identifiable, we must know the location of the head that assigns the θ-role. This can occur if the role-assigner has already been processed or if the structure is such that we can unambiguously predict the position of the head that will assign the role.

An unsatisfied lexical requirement position is a position to which a role must be assigned as a lexical requirement of some head, but that presently consists only of an h-node that dominates no thematic elements. A thematic element, in contrast to a functional element, is a noun, verb, adjective, or preposition.

It turns out, based on empirical processing facts, that x_{TR} and x_{LR} can be treated as the same constant. That is, every local θ-criterion violation, whether assessed by the property of thematic reception or the property of lexical requirement, is equally costly. Processing facts also show that the beam width that restricts parallelism in this system must lie between one and two local θ-criterion violations (Gibson, 1991, in press). That is, if a locally ambiguous sentence has two alternative parses of the input string that differ in cost by one violation or less, the parser can process either possible continuation easily. However, a difference of two or more violations causes the parser to discontinue operation on the more expensive structure, causing a significant (i.e., noticeable) garden-path effect if a return to that structure is forced by the remainder of the string.

These principles explain a large variety of easy and hard ambiguities. The sentences in example 8 are just a small sample of the garden-path effects that this theory successfully predicts:[9]

8. a. # The cotton clothing is made of grows in Mississippi. (Marcus, 1980)
   b. # I gave the boy a dog bit a dollar. (Pritchett, 1988)
   c. # The horse raced past the barn fell. (Bever, 1970)

For example, sentence 8a is an example of a prenominal modifier/relative clause ambiguity. In this sentence, the initial segment *the cotton clothing* can be interpreted either as a noun phrase with a prenominal modifier or as a noun phrase *the cotton* followed by a hypothesized relative clause whose subject, *clothing*, is present. It turns out that the simple NP reading is strongly preferred, giving rise to a strong garden-path effect. This strong preference is explained by the properties of thematic reception and lexical requirement. In particular, let us examine the point of local ambiguity in the

[8] This definition represents a refinement of the definition given in Gibson (1991), referring to chains rather than positions in chains.

[9] We will prefix sentences that are difficult to process with the symbol #.

parse of sentence 8a, in which the heads of h-node categories are notated *h* (irrelevant structural detail omitted, here and below):

9. a. [$_{IP}$ [$_{NP}$ the cotton clothing] [$_{I}$ *h*]]
   b. [$_{IP}$ [$_{NP}$ the [$_{N'}$ [$_{N'}$ cotton$_i$] [$_{CP}$ [$_{NP}$ O_i] [$_{IP}$ [$_{NP}$ clothing] [$_{I}$ *h*]]]]] [$_{I}$ *h*]]

Structure 9a is associated with one local θ-violation because the role-assigner for the NP *the cotton clothing* is not identifiable yet. Structure 9b, on the other hand, is associated with three local θ-violations because the role-assigners for the following NPs are not identifiable yet: (1) the NP *the cotton*, (2) the NP *clothing*, and (3) the nonlexical NP operator in the specifier position of the relative clause CP (whose lexical equivalent is something like the word *which*, as in *the cotton which clothing . . .*). Since the difference between these loads is therefore two local θ-violations, a garden-path effect results. The garden-path effects in sentences 8b and 8c are explained similarly, as are numerous other garden-path phenomena [see Gibson (1991, in press)] for further examples.[10]

An adequate theory of sentence processing must account not only for garden-path effects, but also for the lack of such effects in other instances of temporary ambiguity. The theory proposed in Gibson (1991, in press) explains a wide range of such easy ambiguities, a few of which are given in sentences 10:

10. a. The desert trains are tough on young people.
    b. The desert trains young people to be tough. (Frazier & Rayner, 1987; MacDonald, 1993)
    c. The linguist knew the solution.
    d. The linguist knew the solution would be difficult. (Frazier & Rayner, 1982)
    e. Is the crowd in the room?
    f. Is the crowd in the room happy? (Marcus, 1980)
    g. The report that the scientist examined was incorrect.
    h. The report that the scientist examined the spaceship was incorrect. (Gibson, 1991; adapted from Fodor, 1985)

The sentences 10a and 10b consist of a determiner followed by a word which is ambiguous between adjectival and nominal readings, followed by

[10] See the third section for a discussion of how the garden-path effect is predicted in main-clause/reduced-relative ambiguities as in sentence 8c.

a word which is ambiguous between nominal and verbal readings. In order to see how the model presented here explains the lack of difficulty associated with either completion of these partial sentences, consider the two structures for *the desert trains*:

11. a. [$_{IP}$ [$_{NP}$ the desert trains] [$_{I}$ *h*]]
    b. [$_{IP}$ [$_{NP}$ the desert] [$_{VP}$ trains [$_{NP}$ *h*]]]

Structure 11a is associated with one local θ-violation because the role-assigner for the NP headed by *trains* is not identifiable yet. Structure 11b is also associated with a single local θ-violation since the verb *trains* has a locally unsatisfied lexical requirement. There is therefore no difference between the costs of the two structures, and both are maintained as desired. The remaining sentences in example 10 are explained similarly (see Gibson, 1991; in press), along with numerous other easy ambiguities.

An additional natural prediction of this model for easy ambiguities is as follows. If one structure is ranked higher than another by virtue of being slightly less costly, but not so much so that the more expensive structure lies outside the beam width, it is natural to assume that the cheaper structure may be more easily accessed than the more expensive structure, resulting in ranked parallelism (cf. Gorrell, 1987, 1989; Kurtzman, 1985). Hence there may be an experimentally detectable preference for the slightly favored structure, even though both structures can be accessed without reanalysis. The processed of ambiguities like sentences 10c and 10d provides support for such a prediction. Consider the two possible structures for the initial input segment of these examples *the linguist knew the solution*:

12. a. [$_{IP}$ [$_{NP}$ The linguist] [$_{VP}$ knew [$_{NP}$ the solution]]]]
    b. [$_{IP}$ [$_{NP}$ The linguist] [$_{VP}$ knew [$_{CP}$ [$_{C}$ *e*] [$_{IP}$ [$_{NP}$ the solution] [$_{I}$ *h*]]]]]

The reading where the NP is the main clause object, example 12a, has no processing load associated with it, because all θ-requirements are satisfied at this point. The embedded subject attachment, example 12b, is associated with a load of one local θ-violation, because we cannot identify the θ-assigner for the embedded subject NP. Thus, the cost difference between the two attachments of the NP is one local θ-violation, which is within the beam width, so we correctly predict that the continuation of neither structure results in a garden-path effect. However, the structures do differ in cost, so we might expect that the less costly main-clause object reading is preferred, resulting in a slight slowdown in processing when subsequent words force the less preferred embedded-subject analysis 10d to be adopted. Such a slowdown for sentences like 10d is what some experimental reading-time studies have found (see Ferreira & Henderson, 1990; Frazier & Rayner, 1982; Rayner & Frazier, 1987; cf. Trueswell et al., 1993), so this additional prediction of the system seems to be borne out.

This type of preference plays a key role in our account of filled-gap effects and preferred readings for globally ambiguous sentences in the next two sections. Thus differences in processing load among representations are important at two levels: When they exceed the beam width, the more expensive structures cease being considered, but even when both structures are within the beam and maintained in parallel, the more expensive structure is slightly disfavored. In the latter case, if later words in the sentence force the parser to adopt the disfavored reading, some experimentally detectable (but not conscious) increase in processing load is predicted to occur, as the least expensive analysis can no longer be pursued.

The processing of examples like 10c and 10d also illustrates another important property of the proposed parser: Multiple segmentations of the input string can be retained in parallel (up to the proposed memory constraints), in the form of multiple stacks. A single-celled stack represents the input in a single segment wherein all input words have been attached to a single tree, but multicelled stacks represent input that has not been completely assembled yet. (Unassembled structures will generally be more expensive, especially if θ-roles can be assigned in single-celled representations, but cannot be so assigned in the multicelled representations). At the point of processing the determiner *the* following *knew* in sentences 10c and 10d, an h-node noun phrase is accessed, corresponding to the fact that a determiner predicts a head noun to follow. No categories above the h-node NP are predicted, however, because the head noun following the determiner *the* must be lexically realized. Only structures corresponding to sisters of some projection of the current c-node head in the input are predicted, in order to avoid redundancy and save computation and space.[11] But this NP is not predicted by the lexical entry for *knew* that subcategorizes for a sentential complement CP. Thus no attachments are possible between these structures. However, there is no reason to abandon the possibility that a noun phrase can grammatically continue the sentential complement reading, because of course it could: as specifier for the predicted IP. Hence the predicted NP is

[11] It is also impossible in left-recursive structures to predict what the highest level XP will be. For example, in a left-recursive NP like *her cousin's aunt's friend's dog's tail*, it is impossible to predict at the first word of the NP where the head noun of the matrix NP will occur. It is therefore pointless to attempt to make a similar such prediction from the determiner in sentences 10c and 10d, because the determiner could head a deeply embedded left-recursive NP.

attached in object position in one representation and saved on a stack in another, as depicted in the examples in 13:[12]

13. a. Stack 1:
 [$_{IP}$ [$_{NP}$ The linguist] [$_{VP}$ knew [$_{NP}$ [$_{DetP}$ the] *h*]]]
 b. Stack 2:
 [$_{IP}$ [$_{NP}$ The linguist] [$_{VP}$ knew [$_{CP}$ [$_{c}$*e*] [$_{IP}$ *h*]]]]
 [$_{NP}$ [$_{DetP}$ the] *h*]]]

In each of these structures, the lexical requirement of the verb *knew* is unsatisfied. Furthermore, in both structures the NP containing *the* is an h-node and hence incurs no thematic reception violation. Thus the cost of each structure is the same, one local θ-violation, so both are retained at the same rank. If the following word is the noun *solution*, then attachments are possible to both structures. First, it will attach as head of the NP object of *knew* in example 13a. Second, it will attach to the predicted NP on the stack in example 13b, and the resulting c-node NP, which will now have an IP predicted from the head noun *solution*, will attach as specifier of the IP that was predicted below *knew*, giving the structure in example 12b.

It is worth noting that the theory of processing costs based on the properties of thematic reception and lexical requirement receives strong independent support from the fact that these same two properties can also be used to explain processing overload effects, as in the sentences in 14, even when there is no ambiguity involved:

14. a. # The man who the woman who the dog bit met yesterday likes fish.
 b. # That for John to smoke would be annoying is obvious.

This is achieved by appealing to an absolute memory capacity threshold for a single representation, such that a structure requiring more cost than the threshold (in terms of local θ-criterion violations) cannot be fully maintained in memory. A sentence with such an expensive structure cannot be successfully parsed.[13]

[12] Because this is a parallel parser in which all grammatical possibilities are pursued (up to the proposed memory constraints), there are a large number of possible attachments to consider at each parser state. We will present only those attachments that have relevance to the issues under consideration here. Keep in mind that other attachments and stack pushes may be possible in addition to the ones that are being presented. Furthermore, to avoid the possible problems of using too much space, we assume that identical material in multiple structures in structure-shared (as in Earley, 1970, and Tomita, 1987), and hence multiple copies of it need not be stored.

[13] This absolute threshold may also apply to the total number of distinct violations that can be tolerated across all representations currently under consideration—see Gibson (1994) for an exploration of this possibility.

In summary, we have seen that the proposed parsing algorithm uses the measure of local θ-criterion violations to account for a wide variety of processing facts: the distribution of garden-path effects and more subtle processing slowdowns that result from temporary ambiguities, preferred readings of globally ambiguous sentences, and the difficulty of processing multiply embedded constructions that do not involve ambiguity.

ENGLISH GAP EFFECTS

Let us now examine what predictions the proposed model makes for ambiguities involving gap-positing in English. In particular, let us look again at the examples in 4 (repeated below), which illustrate the filled-gap effect:

4. a. My brother wanted to know if Ruth will bring us home to Mom at Christmas.
   b. My brother wanted to know who Ruth will bring us home to at Christmas.

At the point of processing the input string *my brother wanted to know*, the parser will have constructed a representation in which the verb *know* subcategorizes for a CP complement, along with other possible representations. When the word *who* is encountered next (as in sentence 4b), it is attached as the specifier of the CP, resulting in the structure in example 15:

15. My brother wanted to know [$_{CP}$ [$_{NP}$ who] [$_{C}$ *e*] [$_{IP}$ [$_{I}$ *h*]]]

Because the head of the complement CP is nonlexical, the category prediction algorithm predicts an h-node IP as complement of C. No further predictions are made because the head of IP must have lexical content (perhaps shared with a verbal head). Because there is a wh-NP in need of a thematic role, a gap can now (optionally) be attached to this structure in subject position of the hypothesized IP, formed the tail of a chain whose head is the wh-word, to give a second possible representation of the input:

16. My brother wanted to know [$_{CP}$ [$_{NP}$ who$_i$] [$_{C}$ *e*] [$_{IP}$ [$_{NP}$ t_i] [$_{I}$ *h*]]]

There are therefore two structures for the input string *my brother wanted to know who*, before the next word is encountered: 15 and 16. Interestingly, there is no difference in cost between these two structures according to the cost metric given above: Each structure contains a single local θ-criterion violation, which arises due to the fact that the theta-assigner for the wh-NP *who* is not unambiguously identifiable in either structure. It does not matter to the cost metric that the chain headed by *who* has a single element in one structure, and two elements in the other: The early chain formation counts neither for nor against either structure. Thus there is predicted to be no slowdown in processing for either structure. The parser can hence continue either structure with equal ease/difficulty, so that unlike the active filler strategy, this theory accounts straightforwardly for the lack of subject filled-gap effects in examples like sentences 4.

Let us now see whether object filled-gap effects are predicted. Consider the structure for the input *my brother wanted to know who Ruth will bring* at the point when the verb *bring* has just been attached:

17. My brother wanted to know [$_{CP}$ [$_{NP}$ who] [$_{IP}$ [$_{NP}$ Ruth] [$_{I}$ will] [$_{VP}$ [$_{V}$ bring [$_{NP}$ *h*]]]]]

As in the case of the subject gap, an object gap can now be (optionally) attached into this structure:

18. My brother wanted to know [$_{CP}$ [$_{NP}$ who$_i$] [$_{IP}$ [$_{NP}$ Ruth] [$_{I}$ will] [$_{VP}$ [$_{V}$ bring [$_{NP}$ t_i]]]]]

We can now compare costs between the two representations for the input string up to this point. The structure that lacks the chain, example 17, contains two local θ-criterion violations: one thematic reception violation, because the wh-NP *who* requires and lacks a role; and one lexical requirement violation, because the hypothesized object NP lexical requirement of the verb *bring* is currently unsatisfied. On the other hand, there is no cost whatsoever associated with the structure in which a chain is formed between the object position of the verb *bring* and the wh-NP *who*. In this structure, the wh-NP *who* receives a role by virtue of being linked to the object position of the verb *bring*, and the object lexical requirement of *bring* is satisfied by the gap which is linked to the wh-NP filler *who*. As a result, there is a cost difference of two local violations between the two structures, and consequently, a processing difference should emerge. The model as stated above therefore predicts a strong garden-path effect if a lexical object is attached later, because the cost difference between the two structures here is two local θ-violations. Thus the theory as presented predicts strong object filled-gap effects. Although the evidence is that there is a measurable object filled-gap effect (Stowe, 1986), it is not as strong an effect as in the garden-path effects observed in the sentences in 8. Hence, although the theory correctly predicts that object filled-gap effects will exist, it overpredicts the difficulty of these effects. This problem arises because in the dispreferred structure, both the filler and the potential role for this filler—the head of the chain and the potential tail of the same chain—are counted against the structure.

However, if the parser rates each representation as optimistically as possible, counting only one θ-violation for the set of positions within a structure which could potentially be part of the same chain, then this overprediction of difficulty in object filled-gaps may be avoided. That is, the parser computes costs according to the minimal number of potential chains in a structure. This idea can be implemented in a number of ways. The way that we propose here is to restate the properties of thematic reception and lexical requirement are stated such that costs apply only to unambiguous heads of chains. Under such a formulation, no more than one position in a potential chain counts against a structure. Let us therefore consider the following modified definitions:

19. Property of thematic reception: Associate a cost of x_{TR} PLUs to each confirmed node (1) that is in a position that can be associated with a θ-role in any of the structures currently under consideration, (1) that unambiguously heads a chain, and (3) whose role-assigner is not unambiguously identifiable.

20. Property of lexical requirement: Associate a cost of x_{LR} PLUs to each unsatisifed lexical requirement position (an h-node) that is obligatory in any of the structures currently under consideration and unambiguously heads a chain.

In general, an overt NP always heads a (possibly single-membered) chain unambiguously, and a trace never heads a chain, but an h-node may or may not represent the head of a chain if the structure contains a c-commanding filler of the same category that requires a thematic role. This added qualification to the definitions has the desired effect. Now, only the wh-NP *who* counts against the structure lacking a chain in 17, because this position unambiguously heads a chain. However, the hypothesized NP object lexical requirement position of the verb *bring* does not unambiguously head a chain—there is a c-commanding filler of the same category which lacks a

role—and hence it is no longer associated with cost via the property of lexical requirement.[14] As a result, the total cost difference between the two structures is only one local θ-criterion violation, so that a measurable object filled-gap effect is predicted, but only a relatively minor effect. This effect is predicted to be comparable to the effect of continuing with the sentential complement reading in an NP/sentential complement ambiguity such as in sentences 10c and 10d. Intuitively, this comparison seems appropriate, because neither effect causes conscious processing difficulty, but both are measurable with fine-grained on-line reading measurements.

It turns out that this small revision to the theory also provides a more elegant account of the garden-path effect that arises in a matrix clause/reduced relative clause ambiguity, as in sentence 8c (see Gibson, 1991, for the anaysis under previous assumptions). This example will also illustrate the way in which gap positing interacts with cost assessment. Consider the parse of sentence 8c at the point of parsing the word *raced*:

21. a. $[_{IP}$ $[_{NP}$ the horse$]$ $[_{VP}$ raced$]]$
    b. $[_{IP}$ $[_{NP}$ the $[_{N'}$ $[_{N'}$ horse$_i]$ $[_{CP}$ O_i $[_{IP}$ $[_{VP}$ raced $[_{NP}$ $h]]]]]]]$

Structure 21a has no load associated with it due to either the lexical requirement or thematic reception. Crucially note that the verb *raced* has an intransitive reading so that no load is required via the property of lexical

[14] Another way of formulating this proposal is to say that costs can be counted *after* traces are posited but *before* they are linked to another element via a chain. The intuition behind this is that traces, at least in government binding theory, are linguistic objects that have featural content and a referential index, just like overt elements; in particular, they have similar requirements for a c-commanding antecedent. Now, it is reasonable to suggest that an anaphor such as a reflexive pronoun can be attached to a partial sentence structure tree before we assess whether or not it has a valid antecedent or indeed any antecedent at all, because this assessment will depend on its position. We propose that a similar procedure applies to covert anaphors, i.e., traces. The only difference is that since traces do not appear in the input, they are inserted during the structure-prediction phase following the input of a word. That is, one option that is always available when predicting new structure is to predict a trace, if a valid site is available. Then, one reading of the sentence will contain a trace while the other(s) will not. At this point, no antecedent for the trace has been established, because prediction is a local operation that does not have access to potential antecedents. However, the grammar dictates that a trace in an argument position receives the θ-role assigned to that position, so this trace counts as satisfying a lexical requirement. Conversely, a trace by itself will *not* incur the cost of a thematic reception violation, because needing a θ-role assigner is a property of certain types of chains, and a trace is by definition only part of a chain. It is only after costs are counted that chain-indexation is attempted; if no antecedent for a trace can be found, the structure containing it is discarded. Costs are not assessed again until the next input word and the structures it predicts have been input.

requirement, which refers to obligatory arguments. In structure 21b, the reduced relative clause reading of *raced* is attached as a modifier of the NP *the horse*, but no gap associated with the operator has yet been attached. The load associated with structure 21b is two local θ-violations since the role-assigners for the NPs *the horse* and the nonlexical operator O_i have not been identified yet. By the new definition of the property of lexical requirement, there is no load associated with the complement position of *raced*, despite the face that *raced* does assign a role to that position, because this position is not the unambiguous head of a chain, since the operator c-commands it. Thus the difference between the processing loads associated with structures 21a and 21b is two local θ-violations. As has been seen previously, this load difference is enough for a strong local preference to take place, and a garden-path effect results.

Note, however, that once a trace is attached in the complement position of *raced* and linked to the operator, the load associated with the reduced relative reading will go down to a single θ-violation, because the operator chain will get a role from *raced*. Thus, if this gap were posited and linked immediately after processing the word *raced*—before comparing the costs of the two readings—the load difference would be only one local θ-violation and no garden-path would be predicted. We therefore require that the costs of structures be compared after the processing of each input word, before any gaps are posited. This procedure will in turn follow from treating traces themselves as input elements, as if they were part of the sentence string. That is, attempting to posit a trace involves putting a trace in the input buffer and then trying to attach it by the usual attachment algorithm. This approach not only yields a natural account of reduced relative garden-paths without any undesirable changes to other preference effects described in Gibson (1991), it also makes the right predictions for Dutch relative clauses and questions, as we will show in the next section.[15]

DUTCH GAP EFFECTS

Some of the strongest evidence for the active filler strategy has come from processing evidence in Dutch (Frazier, 1987b; Frazier & Flores

[15] To get reduced relative clause garden-path effects under the original cost definitions required positing the trace of passive movement in the lexicon, while wh-traces were not posited immediately. This stipulative component of the original formulation has thus been avoided here. We thank Suzanne Stevenson for her help and observations with respect to this issue.

Another problem with the previous account was that it incorrectly predicted that reduced relatives in object position should not be garden-paths, as noted by Weinberg (personal communication). See Gibson (1994) for a possible solution to this problem. Briefly, the proposal is that overload across representations is causing these preferences—see footnote 13.

d'Arcais, 1989). Because Dutch is a verb-final language in embedded clauses, the first NP following a relative pronoun can be either the subject or the object of the verb to follow: The trace of movement could come either before or after this NP. There is therefore additional local ambiguity in the processing of relative clauses in Dutch, compared to English. Consider the following examples containing locally ambiguous Dutch relative clauses, from Frazier (1987b):

22. Karl hielp de mijnwerkers die de boswachter vonden
    Karl helped the mineworkers who the forester found-PL
    "Karl helped the mineworkers who found the forester"
23. Karl hielp de mijnwerkers die de boswachter vond
    Karl helped the mineworkerse who the forester found-SG
    "Karl helped the mineworkers who the forester found"

The relative pronoun *die* is plural in number because it agrees with the plural NP *de mijnwerkers* ("the mineworkers"), and the NP *de boswachter* ("the forester") is singular. Thus if a singular verb is encountered next, such as *vond* (singular "found") in example 23, *de boswachter* is disambiguated as the subject, and the relative pronoun *die* must therefore be linked to the object position. Conversely, if a plural verb is encountered next, then the relative pronoun *die* must be linked to the subject position of this verb. In measurements of phrase-by-phrase reading times, Frazier (1987b) found that the disambiguation toward the subject relative clause reading in examples like 22 was read more quickly than the disambiguation toward the object relative clause reading, as in example 23. Furthermore, in examples that are ambiguous between subject- and object-relative clause readings (as in example 24 below), Frazier found that there is a preference for the subject-relative interpretation, as indicated by the results of an off-line question-answering task:

24. Karl hielp de mijnwerker die de boswachter vond
    Karl helped the mineworker who the forester found
    "Karl helped the mineworker who found the forster"
    "Karl helped the mineworker who the forester found"

Frazier (1987b) argued that the on-line reading time evidence, taken together with the off-line preferences in addition to evidence of similar preferences in the processing of ambiguous wh-questions (Frazier & Flores d'Arcais, 1989), offers converging support for the use of the active filler strategy in Dutch. As soon as the wh-filler *die* has been identified, the parser which adheres to the AFS attempts to locate a position for a gap to be linked to this filler. The subject of the S node within the relative clause is the first location that is found, and the trace is therefore posited there.[16] If it turns out that the wh-NP needs to be linked to the object position instead, then reanalysis is forced, and slower reading times result, as in example 23. If the relative clause ends up being globally ambiguous between subject- and object-relative clause readings, then the subject-gap analysis will be preferred, because it was obtained first.[17]

These general ideas are worked out in more detail in Crocker (1994), based on work in Crocker (1992). Crocker advocated the "active trace hypothesis" which is the active filler strategy applied in such a way that traces can be posited to the right of positions where lexical material can still potentially attach (Gibson & Hickok, 1993). In particular, Crocker proposed that a CP, an IP, and a VP are all constructed upon encountering the relative pronoun. The active trace hypothesis causes a trace to be posited in the first available site, specifier position of the IP, and the subject-gap preference is predicted.[18] As Crocker correctly pointed out, the subject-relative preference is hard to explain in a theory without empty categories (e.g., Pickering & Barry, 1991): If we wait until we see the subcategorizer to make the link

[16] Recall that one of the potential explanations that Clifton and Frazier (1989) offered for the lack of a subject filled-gap effect in English was that the potential attachment site for the trace takes some time to become available after the parser has encountered the wh-relative in CP, so that when the attachment site is available, a lexical NP has already been encountered in the input, and it attaches into this position instead of the trace. If this explanation of the lack of subject filled-gap effects in English is to be pursued, some explanation is necessary for why the same situation does not apply in Dutch, in order to rule out the possibility of attachment into subject position of the first lexical NP following the wh-relative, thus incorrectly predicting a preference for object-gap preferences in Dutch relative clauses.

[17] It should be noted that, while there is a subject-gap preference in Dutch extractions within a single caluse, there appears to be a reversal in long-distance extractions through an additional clause: an object-gap preference (Jordens, 1991; see Frazier, 1993, for some relevant theoretical discussion). This fact is problematic for the active filler strategy, and Frazier (1993) offered a potential solution, in terms of an additional constraint that the parser must locally satisfy, which prevents it from positing a subject-position trace in Dutch long-distance extractions. These facts turn out to be just as problematic for the approach offered here. However, because the experimental evidence is not yet clear in this area (especially in embedded contexts), we leave this interesting area to future work.

[18] Note that Crocker's algorithm implicitly assumes a topdown gap-positer, or one that builds the IP, then the gap, then the VP. Otherwise, no prediction is made by this system, because the gap might also be posited in the object of the proposed VP. If the VP were not predicted from the CP (as is proposed in the parallel framework here), then the subject-gap preference is clear. In fact, Konieczny, Hemforth, Scheepers, and Strube (1993) may have provided some independent evidence for this assumption in German, another head-final language (but cf. Frazier, 1987b, 1993).

with the filler, as proposed by Pickering & Barry (1991) (and by Gibson & Hickok, 1993, in a theory with empty categories), no differences are predicted, which is a potential problem, unless another explanation for these facts can be ascertained (cf. Frazier, 1987b, for related observations).[19]

Let us now see how the parallel parsing theory given above explains the Dutch subject-gap preferences. Consider the input string *Karl hielp de mijnwerker die de boswachter* which occurs as the initial segment of the ambiguous example 24. At the point of processing the NP *de mijnwerker,* there exists a representation in which this NP is object of the verb *hielp.* In this structure, possible modifiers of the NP *mijnwerker* are predicted to the right, including a relative clause CP:

25. Karl hielp [$_{NP}$ de mijnwerker [$_{CP}$ [$_{C}$ *h*]]]

A structure representing the relative pronoun *die* is now input. This structure includes a CP headed by a c-node non-lexical complementizer; an NP dominating *die* is its specifier. An h-node IP is predicted to the right of the head of C, indicating a prediction of the next category to come.

26. [$_{CP}$ [$_{NP}$ die] [$_{C}$ *e*] [$_{IP}$ [$_{I}$ *h*]]]

This structure matches the predicted CP to the right of *mijnwerker* in example 25, so the attachment is made, resulting in example 27:

27. Karl hielp [$_{NP}$ de mijnwerker [$_{CP}$ [$_{NP}$ die] [$_{C}$ *e*] [$_{IP}$ [$_{I}$ *h*]]]]

Because all attachments of the structures associated with *die* are completed at this point, the parser now computes the costs of the existing structures, and ranks and prunes structures accordingly. There is only a single structure, with a cost of one local θ-violation, corresponding to the fact that the θ-assigner for the wh-filler *die* is not identifiable yet. The presence of this wh-filler requiring a thematic role allows the parser to consider the possibility of attaching an empty category as a prospective tail for its chain. One attachment location is possible, in specifier position of the predicted IP, resulting in another structure for the same input string, example 28:

28. Karl hielp [$_{NP}$ de mijnwerker [$_{CP}$ [$_{NP}$ die$_i$] [$_{C}$ *e*] [$_{IP}$ [$_{NP}$ t_i] [$_{I}$ *h*]]]]

Note that, because the category prediction algorithm predicts only the next category that has some lexical realization, a VP is not predicted below

[19] To be fair, however, it should be noted that, unlike the AFS, the strategy that waits for the subcategorizer correctly predicts the lack of English filled subject-gap effects.

the IP, and hence an object-gap chain cannot be pursued at this point in the parsing process.

The costs associated with structures 27 and 28 are now computed and compared. The cost associated with structure 27 remains unchanged at one local θ-violation. The cost associated with structure 28 is also one local θ-violation, because the θ-assigner for the chain headed by the wh-NP *die* is not identifiable in this structure either. Note that the trace in the specifier position of the proposed IP is not associated with a local θ-violation cost because it is not unambiguously the head of a chain: In fact, it is unambiguously *not* the head of a chain. Thus the costs of the two structures are the same, and both structures are carried along at the same rank.

The NP *de boswachter* is input next, one word at a time. The determiner *de*, which predicts an NP, cannot attach to either representation, so its corresponding structure is placed on the stack. When the noun *boswachter* is then input, it matches the determiner's predicted NP. This noun *boswachter* also predicts an IP to its right (along with other categories such as a head VP and posthead modifiers), and this predicted IP matches the previously predicted IP in structure 27, where no gap has been posited. No attachments of the NP *de boswachter* are possible to structure 28, because there is no matching category to attach to. In particular, it cannot attach as the specifier of the IP in structure 28 because that position is already taken by a trace. A representation for the NP *de boswachter*, in which a VP is predicted (again, along with other predicted categories), is therefore placed onto a stacked representation of the input. The two possible representations of the input *Karl hielp de mijnwerker die de boswachter* are given in example 29:[20]

29. a. Stack 1:
Karl hielp [$_{NP}$ de mijnwerker [$_{CP}$ [$_{NP}$ die$_i$] [$_C$ *e*] [$_{IP}$ [$_{NP}$ t_i] [$_I$ *h*]]]]

[$_{VP}$ [$_{NP}$ de boswachter] [$_V$ *h*]]

b. Stack 2:
Karl hielp [$_{NP}$ de mijnwerker [$_{CP}$ [$_{NP}$ die] [$_C$ *e*] [$_{IP}$ [$_{NP}$ de boswachter] [$_I$ *h*]]]]

The parser now computes costs for the two structures. It turns out that each representation still has the same cost: two local θ-violations, corresponding to the chains headed by *die* and *de boswachter* that each need to identify their θ-assigners. The IP/VP complex *vond* is encountered next, and attachments to both structures 29a and 29b are possible. Consider first the stacked structure 29a. The VP component of the newly input item matches the predicted VP, and an attachment is made. The resulting IP then matches the predicted IP in the stacked structure, and a further attachment is made, resulting in the structure in 30:

30. Karl hielp [$_{NP}$ de mijnwerker [$_{CP}$ [$_{NP}$ die$_i$] [$_C$ *e*] [$_{IP}$ [$_{NP}$ t_i] [$_{VP}$ [$_{NP}$ de boswachter] [$_V$ vond]]]]]

Attachments of the structures representing *vond* to structure 29b are also attempted in parallel. In this case the new IP matches the predicted IP, and an attachment is made, giving 31:

31. Karl hielp [$_{NP}$ de mijnwerker [$_{CP}$ [$_{NP}$ die] [$_C$ *e*] [$_{IP}$ [$_{NP}$ de boswachter] [$_{VP}$ vond]]]]

Costs are calculated once again, and there is now a difference between the two structures. There is no cost associated with structure 30, because all NPs receive roles, and all roles are assigned. In particular, the chain headed by the wh-NP *die* now receives a role from the verb *vond*. Thus, this reading of the sentence is complete, since there are no more words in the input stream. In contrast, there is still a cost of one θ-violation associated with the wh-NP *die* in structure 31 because *die* has not identified its θ-role assigner, since the input item that will transmit its role to it (a trace) has not yet been processed. When a trace is processed next, it is attached in object position of the verb *vond*, thus completing the chain headed by *die* and nullifying its θ-violation cost.[21]

Given a forced-choice situation, as in Frazier's off-line preference determinations for sentences like 24, it is plausible that people will report the subject-gap interpretation more often, because its corresponding structure is available sooner and/or requires less processing to complete, since it does

[20] For exposition purposes, we have depicted the VP predicted by the NP *de boswachter*, but no other categories predicted by this NP. Keep in mind that other categories (e.g., IP, posthead modifiers) are also locally predicted by this NP, although these predictions end up being incorrect for the sentences under consideration.

[21] As noted by Amy Weinberg (personal communication), in order to arrive at the subject-/object-gap asymmetry in Dutch, we crucially assume that if a gap cannot attach to a structure, it is not saved on a stack, in contrast to structures representing lexical material, which are pushed onto stacks when they cannot be attached. This assumption is motivated by computational efficiency: to avoid producing redundant identical parses. In particular, if the parser could push gaps onto stacks in addition to continuing to posit and attach them after each word, it would be possible to arrive at numerous identical object-gap and subject-gap parses of the input, by a number of different routes. Because the processor can parse no additional structures by saving traces on stacks than what it can parse without doing so, we assume that the parser never makes these stack pushes.

not involve an extra operation of reading a trace from the input buffer after the final word of the sentence.[22]

The slight on-line reading time differences between subject and object-gap readings can be explained similarly. The times are faster for the subject-gap reading than for the object-gap reading because there is extra work required in the object-gap completion, namely, operations involved in attaching a gap and forming the chain, which have already been done for the subject-gap case at an earlier stage. Recall that under our view of parallel parsing, all possible computations related to a given input item are performed on all current structures before the next input item is processed. Thus, after processing the disambiguating verb, the subject relative structure will be complete, while the object relative reading cannot be completed until a trace is processed.[23] Thus, treating postulated traces as input items has the desired consequences here, just as in the English examples in the previous section.

The bottom line is that, under this analysis, subject-gap chains are slightly preferred over object-gap chains, because subject-gap chains can be formed early, at the wh-pronoun, but object-gap chains cannot be formed until the verb is processed; thus, at the verb there is more cost for the lack of object-gap chain, before the gap is posited.

Interestingly, although this theory makes the same predictions as the AFS regarding the preferred reading for example 24, it differs from the AFS in the predicted location of this effect. The AFS predicts that the preference occurs as soon as the relative pronoun *die* is processed, while the account proposed here predicts that the preference will not be present until the verb is processed. We leave it to future empirical studies to determine which of these predictions is correct.[24]

CONCLUSIONS

In this paper, we have shown how the effects of ambiguities involving filler-gap dependencies can be treated in just the same way as other ambiguity effects, without invoking gap-specific strategies. In particular, we have used the processing framework proposed in Gibson (1991, in press), which

[22] Alternatively, the preference could be attributed to the fact that, until the object gap was posited, the subject-gap structure was more highly ranked because it contained fewer violations. It is possible that the most recent transitory preferences among partial structures survive in some form through the completion of the sentence.

[23] Pursuing the alternative in footnote 22, another possibility is to attribute the reading-time difference to the fact that, over the course of processing the relative clause, the cost of the subject-gap structure is slightly lower on average.

[24] Thanks to Chuck Clifton and Martin Pickering for making this observation.

is motivated by a wide range of facts about ambiguity processing not involving gaps (as well as processing overload effects), to achieve the same empirical coverage as other proposed gap-positing strategies, but crucially without proposing any new mechanisms specific to the treatment of gaps. We have given an account of the distribution of filled-gap effects in English, as well as reading-time constrasts and global ambiguity preferences for Dutch relative clauses. Moreover, we have provided a more principled explanation of the contrast between object filled-gap effects and the lack of subject filled-gap effects in English than that provided by the active filler strategy, since we do not need to stipulate any special properties of subject position. The contrast falls out from independently motivated parsing principles that are used in the treatment of other types of ambiguity. By simply clarifying the definitions of the cost-counting metrics of Gibson (1991) and the points at which they are applied, we have extended that theory to provide a unified account of sentence processing, subsuming filler-gap dependencies under the more general class of local parsing ambiguities.

REFERENCES

Aho, A. V., & Ullman, J. D. (1972). *The theory of parsing, translation, and compiling: Vol. 1. Parsing.* Englewood Cliffs, NJ: Prentice-Hall.
Altmann, G., & Steedman, M. (1988). Interaction with context during human sentence processing. *Cognition, 30,* 191–238.
Boland, J. E., & Tanenhaus, M. K. (1991). The role of lexical representations in sentence processing. In G. B. Simpson (Ed.), *Understanding word and sentence.* Amsterdam: Elsevier Science Publishers B.V., North-Holland.
Chomsky, N. (1981). *Lectures on government and binding.* Dordrecht: Foris.
Chomsky, N. (1986). *Barriers.* Cambridge, MA: MIT Press.
Clifton, C., Jr., & Frazier, L. (1989). Comprehending sentences with long-distance dependencies. In G. Carlson and M. Tanenhaus (Eds.), *Linguistic structure in language processing* (pp. 273–317). Dordrecht, The Netherlands: Kluwer.
Crain, S., & Fodor, J. D. (1985). How can grammars help parsers? In D. R. Dowty, L. Karttunen, & A. M. Zwicky (Eds.), *Natural language parsing: Psychological, computational, and theoretical perspectives* (pp. 94–128). Cambridge, U.K.: Cambridge University Press.
Crain, S., & Steedman, M. (1985). On not being led up the garden path: The use of context by the psychological parser. In D. R. Dowty, L. Karttunen, & A. M. Zwicky (Eds.), *Natural language parsing: Psychological, computational and theoretical perspectives* (pp. 320–358). Cambridge, U.K.: Cambridge University Press.
Crocker, M. W. (1992). A logical model of competence and performance in the human sentence processor. Unpublished doctoral dissertation, University of Edinburgh, Edinburgh, U.K.
Crocker, M. W. (1994). On the nature of the principle-based sentence processor. In C. Clifton Jr., L. Frazier, and K. Rayner (Eds.), *Perspectives on sentence processing.* Hillsdale, N.J: Erlbaum.
De Vicenzi, M. (1991). *Syntactic parsing strategies in Italian.* Dordrecht: Kluwer.

Earley, J. (1970). An efficient context-free parsing algorithm. *CACM, 13,* 94–102.
Ferreira, F., & Henderson, J. M. (1990). The use of verb information in syntactic parsing: A comparison of evidence from eye movements and word-by-word self-paced reading. *Journal of Experimental Psychology: Learning, Memory and Cognition, 16,* 555–568.
Flickinger, D., Pollard, C., & Wasow, T. (1985). Structure-sharing in lexical representation. *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics.*
Fodor, J. D. (1978). Parsing strategies and constraints on transformations. *Linguistic Inquiry, 9,* 427–473.
Fodor, J. D. (1985). Deterministic parsing and subjacency. *Language and Cognitive Processes, 1,* 3–42.
Frazier, L. (1978). On comprehending sentences: Syntactic parsing strategies. Unpublished doctoral dissertation, University of Connecticut, Storrs.
Frazier, L. (1987a). Sentence processing: A tutorial review. In M. Coltheart (Ed.) *Attention and Performance XII.* Hillsdale, NJ: Erlbaum.
Frazier, L. (1987b). Syntactic processing evidence from Dutch. *Natural Language and Linguistic Theory, 5,* 519–559.
Frazier, L. (1993). Processing Dutch sentence structures. *Journal of Psycholinguistic Research, 22,* 85–108.
Frazier, L., & Clifton, C. Jr. (1989). Successive cyclicity in the grammar and parser. *Language and Cognitive Processes, 4,* 93–126.
Frazier, L., & Flores d'Arcais, G. (1989). Filler driven parsing: A study of gap filling in Dutch. *Journal of Memory and Language, 28,* 331–344.
Frazier, L., & Fodor, J. D. (1978). The sausage machine: A new two-stage parsing model. *Cognition, 6,* 291–325.
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.
Frazier, L., & Rayner, K. (1987). Resolution of syntactic category ambiguities: Eye movements in parsing lexically ambiguous sentences. *Journal of Memory and Language, 26,* 505–526.
Gibson, E. (1991). *A computational theory of linguistic processing: Memory limitations and processing breakdown.* Unpublished doctoral dissertation, Carnegie Mellon University, Pittsburgh.
Gibson, E. (1994). *Processing overload across multiple representations.* Unpublished, Department of Brain and Cognitive Sciences, MIT, Cambridge, MA.
Gibson, E. (in press). *Memory limitations and sentence processing breakdown.* Cambridge, MA: MIT Press.
Gibson, E., & Clark, R. (1988). Positing gaps in a parallel parser. In J. Blevins, and J. Carter (Eds.), *Proceedings of the Eighteenth North East Linguistic Society Conference* (pp. 141–155). Amherst, MA: GLSA Press.
Gibson, E., & Hickok, G. (1993). Sentence processing with empty categories. *Language and Cognitive Processes, 8,* 147–161.
Gorrell, P. G. (1987). *Studies of human syntactic processing: Ranked-parallel versus serial models.* Unpublished doctoral dissertation, University of Connecticut, Storrs.
Gorrell, P. (1989). Establishing the loci of serial and parallel effects in syntactic processing. *Journal of Psycholinguistic Research, 18,* 61–74.
Jackendoff, R. (1977). *X-bar Syntax: A Study of Phrase Structure.* Cambridge, MA: MIT Press.

Jordens, P. (1991). Linguistic knowledge in second language acquisition. In L. Eubank (Ed.), *Point Counterpoint: Universal grammar in the second language* (pp. 199–219). Amsterdam: John Benjamins.
Kimball, J. (1973). Seven principles of surface structure parsing in natural language. *Cognition, 2,* 15–47.
Kimball, J. (1975). Predictive analysis and over-the-top parsing. In J. Kimball (Ed.), *Syntax and semantics* (Vol. 4). New York: Academic Press.
Konieczny, L., Hemforth, B., Scheepers, C., & Strube, G. (1993). Semantikorientierte syntaxverabeitung. In C. Habel and G. Rickheit (Eds.), *Kognitive Linguisitik: Repraesentationen und Prozesse.* Opladen: Westdeutscher Verlag.
Kurtzman, H. S. (1985). *Studies in syntactic ambiguity resolution.* Unpublished doctoral dissertation, MIT, Cambridge, MA.
MacDonald, M. C. (1993). The interaction of lexical and syntactic ambiguity. *Journal of Memory and Language, 32,* 692–715.
MacDonald, M. C., Pearlmutter, N. J., & Seidenberg, M. S. (1993). *The lexical nature of syntactic ambiguity resolution* (Beckman Institute Cognitive Science Tech. Rep. UIUC-BI-CS-93-13). Urbana: University of Illinois.
Marcus, M. (1980). *A theory of syntactic recognition for natural language.* Cambridge, MA: MIT Press.
Pearlmutter, N. J., & MacDonald, M. C. *Individual differences and probabilistic constraints in syntactic ambiguity resolution.* Manuscript submitted for publication.
Pickering, M., & Barry, G. (1991). Sentence processing without empty categories. *Language and Cognitive Processes, 6,* 229–259.
Pritchett, B. (1988). Garden path phenoma and the grammatical basis of language processing. *Language, 64,* 539–576.
Proudian, D., & Pollard, C. (1985). Parsing head-driven phrase structure grammar. *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics.*
Rayner, K., & Frazier, L. (1987). Parsing temporarily ambiguous complements. *Quarterly Journal or Experimental Psychology, 39A,* 657–673.
Stevenson, S. (1993). Establishing long-distance dependencies in a hybrid network model of human parsing. *Proceedings of the 15th Annual Conference of the Cognitive Science Society.*
Stevenson, S. (1994). Competition and recency in a hybrid network model of syntactic disambiguation. *Journal of Psycholinguistic Research, 23,* 295–322.
Stowe, L. (1986). Evidence for on-line gap location. *Language and Cognitive Processes, 1,* 227–245.
Tomita, M. (1987). An efficient augmented context-free parsing algorithm. *Computational Linguistics, 13,* 31–46.
Swinney, D. A. (1991). The resolution of interdeterminacy during language comprehension: perspectives on modularity in lexical, structural and pragmatic process. In G. B. Simpson (Ed.), *Understanding word and sentence.* Amsterdam: Elsevier Science Publishers B. V., North-Holland.
Trueswell, J. C., Tanenhaus, M. K., & Garnsey, S. M. (1994). Semantic influences on parsing: Use of thematic role information in syntactic disambiguation. *Journal of Memory and Language, 33,* 285–318.
Trueswell, J. C., Tanenhaus, M. K., & Kello, C. (1993). Verb-specific constraints in sentence processing: Separating effects of lexical preference from garden-paths. *Journal of Experimental Psychology: Learning, Memory and Cognition, 19,* 528–553.
Winston, P. H. (1984). *Artificial intelligence.* Reading, MA: Addison-Wesley.