COGNITION

Cognition 68 (1998) 1–76

Linguistic complexity: locality of syntactic dependencies

Edward Gibson*

Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

Received 18 September 1997; accepted 24 April 1998

Abstract

This paper proposes a new theory of the relationship between the sentence processing mechanism and the available computational resources. This theory – the Syntactic Prediction Locality Theory (SPLT) – has two components: an integration cost component and a component for the memory cost associated with keeping track of obligatory syntactic requirements. Memory cost is hypothesized to be quantified in terms of the number of syntactic categories that are necessary to complete the current input string as a grammatical sentence. Furthermore, in accordance with results from the working memory literature both memory cost and integration cost are hypothesized to be heavily influenced by *locality* (1) the longer a predicted category must be kept in memory before the prediction is satisfied, the greater is the cost for maintaining that prediction; and (2) the greater the distance between an incoming word and the most local head or dependent to which it attaches, the greater the integration cost. The SPLT is shown to explain a wide range of processing complexity phenomena not previously accounted for under a single theory, including (1) the lower complexity of subject-extracted relative clauses compared to object-extracted relative clauses, (2) numerous processing overload effects across languages, including the unacceptability of multiply center-embedded structures, (3) the lower complexity of cross-serial dependencies relative to center-embedded dependencies, (4) heaviness effects, such that sentences are easier to understand when larger phrases are placed later and (5) numerous ambiguity effects, such as those which have been argued to be evidence for the Active Filler Hypothesis.

Keywords: Linguistic complexity; Syntactic dependency; Sentence processing; Computational resources

1. Introduction

The process of comprehending a sentence involves structuring a sequence of

* E-mail: gibson@psyche.mit.edu

PII S0010-0277(98)00034-1

words syntactically and semantically to arrive at a representation of the sentence's meaning. These processes consume computational resources. Computational resources in the form of memory resources are also required to maintain the current unintegrated syntactic and conceptual structures activated in memory during the processing of a sentence. Thus an important part of understanding how humans process language involves understanding the relationship between the sentence processing mechanism and the available computational resources.

One well-established complexity phenomenon to be explained by a theory of the relationship between the sentence processing mechanism and the available computational resources is the higher complexity of an object-extracted relative clause (RC) compared with a subject-extracted RC in a Subject-Verb-Object language like English:[1]

(1) (a) [$_S$ The reporter [$_{S'}$ who [$_S$ the senator attacked]] admitted the error].
(b) [$_S$ The reporter [$_{S'}$ who [$_S$ attacked the senator]] admitted the error].

In (1b), the relative pronoun 'who' is extracted from the subject position of the RC, whereas the same pronoun is extracted from the object position in (1a). The object extraction is more complex by a number of measures including phoneme-monitoring, on-line lexical decision, reading times, and response-accuracy to probe questions (Holmes, 1973; Hakes et al., 1976; Wanner and Maratsos, 1978; Holmes and O'Regan, 1981; Ford, 1983; Waters et al., 1987; King and Just, 1991). In addition, the volume of blood flow in the brain is greater in language areas for object-extractions than for subject-extractions (Just et al., 1996a,b; Stromswold et al., 1996), and aphasic stroke patients cannot reliably answer comprehension questions about object-extracted RCs, although they perform well on subject-extracted RCs (Caramazza and Zurif, 1976; Caplan and Futter, 1986; Grodzinsky, 1989; Hickok et al., 1993).

The source of the complexity difference is not related to lexical or plausibility differences because both structures involve the same lexical items in equally plausible relationships among one another. Furthermore, the complexity difference is not caused by a re-analysis difference due to a local ambiguity (a 'garden-path' effect). Although there is potentially a brief local ambiguity at the word 'who', there is no reanalysis effect at the disambiguating subject NP in the object-extraction (or at the disambiguating verb in the subject-extraction), compared with unambiguous control sentences (e.g. Stowe, 1986). Thus, theories of ambiguity resolution (e.g. Frazier, 1987a; MacDonald et al., 1994; Mitchell, 1994; Trueswell et al., 1994) make no predictions in these kinds of examples, nor

[1]For simplicity, I will assume a theoretically neutral phrase structure grammar, one which includes expansions of the form S $\rightarrow$ NP VP and S′ $\rightarrow$ Comp S. No complexity results hinge on this assumption. In particular, the processing theory to be described here is compatible with a wide range of phrase structure theories, including lexical functional grammar (Bresnan, 1982), dependency grammars (e.g. Hudson, 1984, 1990), categorial grammars (e.g. Ades and Steedman, 1982), Head-Driven Phrase Structure Grammar (Pollard and Sag, 1994), and the Minimalist program (Chomsky, 1995).

more generally in any construction which does not involve ambiguity. The only remaining plausible cause of the complexity difference is in the quantity of computational resources that the two constructions require to process.

A more general class of complexity effects to be accounted for by a theory of computational resources is the high complexity associated with *nested* (or *center-embedded*) structures, where a syntactic category *A* is said to be nested within another category *B* in the configuration in (2):

(2) [$_B$X A Y]

Increasing the number of nestings soon makes sentence structures unprocessable (Chomsky, 1957, 1965; Yngve, 1960; Chomsky and Miller, 1963; Miller and Chomsky, 1963; Miller and Isard, 1964). For example, consider the contrast between (3a), which contains a singly-nested relative clause (RC) structure and (3b), which contains a doubly-nested RC structure:[2]

(3) (a) [$_S$ The intern [$_{S'}$ who [$_S$ the nurse supervised]] had bothered the administrator [$_{S'}$ who [$_S$ lost the medical reports]]].
(b) # The administrator [$_{S'}$ who [$_S$ the intern [$_{S'}$ who [$_S$ the nurse supervised]] had bothered]] lost the medical reports.

In (3a), the RC 'who the nurse supervised' is nested within the matrix sentence subject-verb dependency 'the intern... had bothered'. In (3b) a second RC ('who the nurse supervised') interrupts the subject-verb dependency in the first embedded sentence ('the intern had bothered'), resulting in a structure that is so complex that it is unprocessable for most people. Note that the two sentences contain the same words and have the same meaning, so the complexity difference is not due to plausibility differences. Furthermore, there is no local ambiguity in (3b), so the processing difficulty associated with this sentence is not related to ambiguity confusions. This type of sentence processing breakdown effect is often referred to as a *processing overload* effect.[3]

Multiply nested structures are similarly complex cross-structurally and cross-linguistically. For example, the English clausal modifier structures are increasingly nested in (4) and are correspondingly increasingly difficult to understand. Similarly,

[2]Sentences that cause extreme processing difficulty are prefixed with the symbol '#'.

[3]This effect is distinct from processing breakdown caused by a preference in local ambiguity (a 'garden-path' effect), such as in (i):

(i) # The dog walked to the park was chasing the squirrel.

Although garden-path sentences like (i) are difficult to process, once the reader/listener realizes the correct interpretation, he/she can obtain this interpretation and process the structure without difficulty. In contrast, even when the reader/listener understands what the appropriate interpretation for a sentence structure causing a processing overload effect is, it is still not possible to arrive at this interpretation using the normal sentence processing mechanism.

the Japanese sentential complement structures in (5) have the same meaning, but (5b) is more nested than (5a), and is therefore harder to understand.

(4) (a) $[_S$ $[_{S'}$ If $[_S$ the mother gets upset]] $[_{S'}$ when $[_S$ the baby is crying]], $[_S$ the father will help], $[_{S'}$ so $[_S$ the grandmother can rest easily]]].
(b) $[_S$ $[_{S'}$ If $[_S$ $[_{;S'}$ when $[_S$ the baby is crying]], $[_S$ the mother gets upset]]], $[_S$ the father will help], $[_{S'}$ so $[_S$ the grandmother can rest easily]]].
(c) # $[_S$ $[_{S'}$ Because $[_S$ $[_{S'}$ if $[_S$ $[_{S'}$ when $[_S$ the baby is crying]], $[_S$ the mother gets upset]]], $[_S$ the father will help]]], $[_S$ the grandmother can rest easily]].

(5) (a) $[_S$ $[_{S'}$ $[_S$ Bebiisitaa-ga $[_{S'}$ $[_S$ ani-ga imooto-o ijimeta] to] itta] to] obasan-ga omotteiru]
babysitter-nom older-brother-nom younger-sister-acc bullied that said that aunt thinks
'My aunt thinks that the babysitter said that my older brother bullied my younger sister'
(b) # $[_S$ Obasan-ga $[_{S'}$ $[_S$ bebiisitaa-ga $[_{S'}$ $[_S$ ani-ga imooto-o ijimeta] to] itta] to] omotteiru]
aunt-nom babysitter-nom older-brother-nom younger-sister-acc bullied that said that thinks
'My aunt thinks that the babysitter said that my older brother bullied my younger sister'

Although the nesting complexity effects and the subject- versus object-extraction complexity effects are well-established phenomena in the literature, there is currently little agreement as to what properties of the complex constructions make them hard to understand. Some of the proposals in the literature include the following (see Gibson (1991) for a comprehensive review):[4]

[4]One factor which can contribute to the processing complexity of nested structures but is orthogonal to the factors to be investigated here is semantic similarity. Center-embedded RC structures like (3b) are easier to comprehend if the NPs come from distinct semantic classes and if the roles assigned by the following verbs also are compatible with distinct semantic classes, so that it is easy to guess who is doing what to whom (Stolz, 1967; Schlesinger, 1968; King and Just, 1991). For example, (ii) is easier to comprehend than (3b) (Stolz, 1967):

(ii) # The vase that the maid that the agency hired dropped on the floor broke into a hundred pieces

Although semantic-role disambiguation improves the acceptability of these kinds of structures, a complexity theory based on semantic role interference alone is insufficient to explain many complexity effects. For example, although (ii) is easier to comprehend than (3b), it is still very complex, and this complexity needs to be accounted for. Furthermore, including an additional pragmatically-distinguishable nested RC makes the structure virtually unprocessable, similar or more complex than (3b):

(ii) # The vase that the maid that the agency that the lawyer represented hired dropped on the floor broke into a hundred pieces. Hence factors other than semantic similarity or interference are responsible for the complexity of nested structures like (iii).

- Stacking incompletely parsed phrase-structure rules (Yngve, 1960; Chomsky and Miller, 1963; Miller and Chomsky, 1963; Miller and Isard, 1964; Abney and Johnson, 1991). This theory hypothesizes that complexity is indexed by the number of phrase structure rules that the parser has to hold in memory on a stack at a particular parse state. Assuming that the human parsing algorithm is partially top-down and partially bottom-up, using, e.g. a left-corner algorithm (Kimball, 1975; Johnson-Laird, 1983; see Abney and Johnson, 1991; Gibson, 1991; Stabler, 1994; for recent summaries), this theory accounts for the contrast between (3a) and (3b) as follows. The maximal stack depth in parsing (3a) is three, at the point of processing the determiner 'the' following 'who' in the first relative clause. There are three incompletely parsed phrase structure rules at this point: (1) the matrix S → NP VP rule, (2) the relative clause rule S′ → Comp S and (3) the NP rule NP → Det N. In contrast, the maximal stack depth in parsing (3b) is five, at the point of processing the determiner 'the' of the subject NP in the most embedded relative clause.
- Self-embedding (Miller and Chomsky, 1963; Miller and Isard, 1964; Gibson and Thomas, 1996). A category *A* is said to be *self-embedded* within another category *B* if it is nested as in the configuration in (2) and *A* and *B* are the same type of syntactic category, e.g. a sentence (S) node. The hypothesis is that the parser has additional parsing difficulty with self-embedded structures. The motivation for this claim is that a stack-based parser might confuse two stacked instances of the same category, and not know which one to go back to, creating additional difficulty. According to this theory, part of the difficulty associated with understanding each of (3b), (4c) and (5b) is due to the fact that the most embedded clause is self-embedded within the second clause, which is itself self-embedded within the matrix clause.
- Perspective shifts (MacWhinney, 1977, 1982; MacWhinney and Pleh, 1988; cf. Bever's (1970) double function hypothesis). According to the perspective-shift theory, processing resources are required to shift the perspective of a clause, where the perspective of a clause is taken from the subject of the clause. Under this theory, the reason that an object-extracted RC is harder to process than a subject-extracted RC is that processing an object-extracted RC requires two perspective shifts: (1) from the perspective of the matrix subject to the subject of the RC and (2) from the perspective of the subject of the RC back to the matrix subject, after the RC is processed. Processing the subject-extracted RC requires no perspective shifts, because the matrix subject is also the subject of the RC, so that both clauses come from the same perspective. Processing a doubly nested RC structure like (3b) requires multiple perspective shifts, and so is correspondingly more difficult.
- Incomplete syntactic/thematic dependencies (Kimball, 1973; Hakuta, 1981; MacWhinney, 1987; Gibson, 1991; Pickering and Barry, 1991; Lewis, 1993; Stabler, 1994). An extension of the stack-based idea which can account for the subject- versus object-extraction contrast associates memory cost with

incomplete syntactic *dependencies* rather than phrase structure rules. This theory allows an explanation of the contrast between subject- and object-extractions in (1), because object-extracted RCs involve one more nested dependency than do subject-extracted RCs: the embedding of the subject ‘the senator’ between the relative pronoun and its role-assigning verb ‘attacked’.

A slightly narrower version of this idea was proposed by Gibson (1991), who, building on similar ideas from Hakuta (1981) and MacWhinney (1987), hypothesized that there is a memory cost associated with each incomplete dependency involving a thematic-role assignment. Thus, an NP that requires a thematic role but has not yet received a role is associated with memory cost, as is a thematic role assigner which requires an argument but has not found one yet. Processing a subject-extraction like (1b) is associated with a maximal memory cost of two local thematic violations, at the point of processing the relative pronoun ‘who’: one for the matrix subject ‘the reporter’ and one for the relative pronoun ‘who’. In contrast, the maximal memory cost of processing (1a) is three local thematic violations: one for each of ‘the reporter’, ‘who’ and ‘the senator’.

Observations about the unacceptability of multiply nested structures like (3b) led Gibson to hypothesize that the maximal memory capacity is four local thematic violations, so that sentences with parse states requiring five local thematic violations are unprocessable. In particular, (3b) is unprocessable because its processing involves a state with five local thematic violations. At the point of processing the most embedded subject ‘the nurse’ there are five thematic violations: one corresponding to each of the initial NPs: ‘the administrator’, ‘who’, ‘the intern’, ‘who’ and ‘the nurse’.

A related account of the difficulty of processing doubly nested structures like (3b) is Kimball’s (1973) principle of two sentences, which states that the constituents of no more than two sentences (clauses) can be parsed at one time (cf. Cowper, 1976; Lewis, 1993; Stabler, 1994). According to this theory, the parser can maintain the predictions of one or two sentence nodes at one time, but there are not enough memory resources for the parser to predict three sentence nodes at the same time. This theory therefore correctly accounts for the unacceptability of sentence structures like (3b), because it requires three incomplete sentence nodes at its most complex point, at the point of processing the most embedded subject NP.

- The CC-READER (Capacity Constrained Reader) model (Just and Carpenter, 1992). This is an activation-based production-rule model of reading times for English singly-embedded relative clause extractions. This model assumes that there is a limited pool of working memory resources available for both integration operations and storage. This assumption allows the model to account for experimental observations involving different groups of participants with different quantities of working memory capacity. This model is not linguistically formalized well enough yet to see how it applies more generally, beyond the relative clause constructions it was designed to account for.

- Connectionist models. Another class of models of linguistic complexity are connectionist models (e.g. Kempen and Vosse, 1989; Elman, 1990, 1991; Weckerly and Elman, 1992; Miikkulainen, 1996; Christiansen, 1996, 1997; Christiansen and Chater, 1998). The goal for these models is to have the complexity phenomena fall out from the architecture of the processor. This kind of model, with a basis in neural architecture, may eventually provide an architectural explanation of the approach proposed here. However, because these types of models are still quite novel, they have not yet been applied to a wide range of phenomena across languages.

Although there have been many theories of the computational resources available for sentence processing, each of the theories accounts for only a limited range of the data currently available. Each of the theories is incompatible with a wide range of evidence. Furthermore, with the exception of the CC-READER model, the theories currently available do not account for comprehension times.

One complexity contrast which is particularly interesting, because no current theory accounts for it, is the contrast between embedding a relative clause within a sentential complement (SC) and the reverse embedding consisting of an SC within an RC (Cowper, 1976; Gibson, 1991; Gibson and Thomas, 1997a):[5]

(6) (a) Sentential complement, then relative clause (SC/RC):
The fact that the employee who the manager hired stole office supplies worried the executive.
(b) Relative clause, then sentential complement (RC/SC:
The executive who the fact that the employee stole office supplies worried hired the manager.

The SC/RC embedding is much easier to understand than the RC/SC embedding (Gibson and Thomas, 1997a). This contrast is not explained by any of the theories listed above. For example, the principle of two sentences predicts that both constructions are unprocessable, because each includes a parse state at which there are

[5]The difference between a relative clause and a sentential complement is that one of the NP positions inside a relative clause is empty, reflecting the presence of the wh-pronoun at the front of the clause. In contrast, there is no such empty argument position in a sentential complement. For example, the initial sequence of words ‘that the reporter discovered’ in (iv) and (v) is ambiguous between a relative clause modifier and a sentential complement of ‘information’:

(iv) [$_{NP}$ The information [$_{S'}$ that [$_{S}$ that the reporter discovered the tax documents]] worried the senator.

(iv) [$_{NP}$ The information [$_{S'}$ that [$_{S}$ that the reporter discovered]] worried the senator.

The presence of the overt NP as the object of ‘discovered’ in (iv) disambiguates this clause as a sentential complement. The lack of an overt object of ‘discovered’ in (v) disambiguates this clause as a relative clause.

three incomplete sentences: at the point of processing the most embedded subject ('the manager' in (6a) 'the employee' in (6b)).

This paper provides a new theory of the relationship between the language comprehension mechanism and the available computational resources: the Syntactic Prediction Locality Theory (SPLT). This theory contains two major components: (1) a memory cost component which dictates what quantity of computational resources are required to store a partial input sentence and (2) an integration cost component which dictates what quantity of computational resources need to be spent on integrating new words into the structures built thus far. The important idea in both of these components of the theory is *locality*: syntactic predictions held in memory over longer distances are more expensive (hence the name *syntactic prediction locality theory*), and longer distance head-dependent integrations are more expensive. The SPLT is notable for its simplicity, its quantitative predictions, and its ability to account for a range of data unexplained by other current theories, including the contrast between embedding orderings of sentential complements and relative clauses, as well as numerous other data from a number of languages. The paper is organized as follows. Section 2 proposes the SPLT and provides motivations and initial evidence for each of its components. Section 3 demonstrates how a variety of processing overload effects from the literature such the relative clause/sentential complement embedding effects are accounted for by the SPLT. Section 4 shows how the SPLT accounts for a range of heaviness or length effects in sentence processing. Section 5 shows how the SPLT can be applied to account for a range of ambiguity effects from the literature. A variant of the proposed memory cost theory is then presented in Section 6, and it is shown how this theory makes similar predictions to the SPLT. Concluding remarks, including some possible consequences of a theory like the SPLT in other aspects of the study of language, are presented in Section 7.

2. The Syntactic Prediction Locality Theory

This section has the following organization. First, some assumptions regarding the structure of the underlying sentence comprehension mechanism are presented in Section 2.1. The integration component of the SPLT is then presented in Section 2.2, followed by the memory component of the theory in Section 2.3. The relationship between the two components of the theory is then discussed in Section 2.4, leading to a theory of comprehension times. Section 2.5 presents a theory of intuitive complexity judgments within the SPLT framework. Evidence for various aspects of these proposals is provided in Section 2.6.

2.1. The underlying sentence comprehension mechanism

Recent results have suggested that constructing an interpretation for a sentence involves the moment-by-moment integration of a variety of different information sources, constrained by the available computational resources. The information sources include lexical constraints (Ford et al., 1982; MacDonald et al., 1994; Trueswell et al., 1994; Trueswell, 1996), plausibility constraints (Tyler and Marslen-Wilson, 1977; McClelland et al., 1989; Pearlmutter and MacDonald, 1992; Trueswell et al., 1994) and discourse context constraints (Crain and Steedman, 1985; Altmann and Steedman, 1988; Ni et al., 1996). Following Just and Carpenter (1992), MacDonald et al. (1994), Stevenson (1994) and Spivey-Knowlton and Tanenhaus (1996) among others, I will assume an activation-based approach to implement the interaction of these constraints. In particular, I will assume that each discourse structure for an input string is associated with an activation (e.g. a number between 0 and 1) which indicates how highly rated the representation is according to the combination of the constraints. Furthermore, I will assume that there is a target activation threshold T_1 such that the processor works to activate a discourse representation above this threshold. In reader-paced comprehension, I will assume that the processor does not move on to a new region of analysis until at least one representation for the input so far reaches this threshold.

I will also assume that there is a limited pool of computational resource units available to activate representations, so that if there are more resources available, then the activations occur more quickly. By the same token, the more resources that a particular component of a structural activation requires, the slower the activation will be. Furthermore, I assume that it takes different quantities of energy (or work) from this resource pool to perform different aspects of representation activation, reflecting the different constraints on sentence comprehension. For example, high frequency lexical items require fewer energy resources to become activated than low frequency lexical items do. As a result, high frequency lexical items can be activated by the resource pool more quickly. Relatedly, structures representing plausible meanings require fewer energy resources to become activated than structures representing implausible meanings. The two primary aspects of resource use to be discussed in this paper are structural integration and structural maintenance (or storage). Details of the hypotheses regarding these latter two aspects of resource use will be discussed in depth in the following sections.

In this framework, the structural integrations considered by the processor are limited by the syntactic constraints associated with each lexical item. Syntactic constraints are assumed to be implemented as lexically-based predictions about the categories that can follow the current lexical item *L*, which are in a head-dependent relationship with *L* (Gibson, 1991; cf. MacDonald et al., 1994). Some of these syntactic predictions are obligatory predictions (e.g. the prediction of a noun following a determiner) while others are optional (e.g. optional arguments of a verb, and all modifier relationships). Structure building within this framework consists of looking up the current word in the lexicon and then matching the categories of these lexical entries to the predictions in the structures built thus far.

If all the constraints favor one structure for the input, then that structure will quickly receive a high activation. If there are two competing analyses for the input which are similar in heuristic value, then it will take longer to bring one of the representations over the threshold T_1, because of the limited pool of resources available (MacDonald et al., 1994; Trueswell et al., 1994). If one or more constraints disfavor a representation (e.g. it is implausible, or it contains syntactic predictions that require a lot of resources to keep activated) then many energy resources will be needed to activate it above the threshold activation T_1. If there are no better competing structures available, then this structure will be followed, but it will be processed slowly because of the large quantity of resources that it needs to activate it. Furthermore, because of the limited size of the resource pool, a representation might require more resources to activate it over the threshold activation T_1 than are available in the pool.

Another important parsing issue concerns how many representations the processor can retain at one time during the parse of a sentence: one (the serial hypothesis) or more than one (the parallel hypothesis). Although this issue is orthogonal to the questions of processing complexity in unambiguous structures to be discussed in this section and the following two sections of this paper, the issue becomes more important when the resource complexity influences on ambiguity resolution are considered in Section 5. It is clear that the processor does not always follow all possible interpretations of a locally ambiguous input string, because of the existence of garden-path effects, as in (7) (see Gibson (1991) for a catalogue of ambiguities in English leading to processing difficulty):

(7) # The dog walked to the park was chasing the squirrel.

However, the existence of ambiguities which are difficult to resolve toward one of their possible resolutions does not decide between a serial processor and a ranked parallel processor. Following Gibson (1991) (cf. Kurtzman, 1985; Gorrell, 1987; Jurafsky, 1996), I will assume that the processor is ranked parallel, such that a lower-rated representation can be retained in parallel with the most highly rated representation from one parse state to the next as long as the heuristic value of the lower-ranked representation, as indicated by its activation, is close to that of the highest ranked interpretation at that point. In order to implement this assumption within the activation-based framework, I will assume the existence of a second threshold quantity of activation, T_2, where $T_2 < T_1$, such that a representation is retained from one parse state to the next as long as its activation is greater than or equal to T_2. Hence, if by the time that the activation on the most highly rated representation R_1 has reached the target threshold activation T_1 the activation on a second representation R_2 is greater than T_2, then R_2 is retained along with R_1 in the *active representation set*, the set of representations that the parser continues to work on. If the activation on another representation R_3 does not reach T_2 by the time that R_1's activation has reached T_1, then R_3 is not retained in the active representation set. The set of representations for the input considered by the processor is therefore divided into two sets: one – the active representation set – in which the representations are being considered as integration sites for incoming words; and a second – the inactive representation set – in which the representations are no longer being considered as integration sites for incoming words. Resources are spent to keep the representations in the active set activated, whereas no resources are spent to keep the representations in the inactive set activated, so that the activation of the inactive representations decays over time. Although the inactive representations are not being worked on, they remain in the representation space with a low activation, so that they may be reactivated later in a reanalysis stage if an incoming word cannot be integrated with any of the representations in the active representation set (see Gibson et al. (1998a) for discussion of how reanalysis takes place in this framework). Some important issues that are yet to be worked out in this general framework include: (1) how the different sources of information interact, (2) whether there is a hierarchy among the constraints such that some information is available earlier, (3) what the resource costs and activation levels are for the different constraints, (4) where the resource costs and activation levels originate and (5) what the thresholds T_1 and T_2 are. The answers to these questions are beyond the scope of the work presented here.

2.2. Integration cost

Sentence comprehension involves integrating new input words into the currently existing syntactic and discourse structure(s). Each integration has a syntactic component, responsible for attaching structures together, such as matching a syntactic category prediction or linking together elements in a dependency chain. Each integration also has a semantic and discourse component which assigns thematic roles and adds material to the discourse structure. It is assumed that each linguistic integration requires a fixed quantity of computational resources to perform the integration plus additional resources proportional to the distance between the elements being integrated (cf. Hudson (1995) for a related hypothesis regarding dependency distance and syntactic complexity). Thus longer-distance integrations require more resources, other factors being equal. The motivation for distance-based integration cost within the activation-based framework is as follows. It is assumed that each lexical item in a structure has an activation level independent of the activation level for the whole structure. The lexical activations decay as additional words are integrated. To perform an integration, it is necessary to first match the category of the current word with a syntactic prediction that is part of one of the candidate structures being pursued. This match then reactivates the lexical head/dependent associated with the syntactic prediction so that the plausibility of the head-dependent relationship can be evaluated within the discourse context. The lexical activation on a word *w* decays as more words are input and integrated into the current structure for the input, unless the new words are also involved in a head-dependent relationship with *w*. Thus it generally takes more resources to reactivate the lexical head for words further back in the input string. If some of the intermediate words are also involved in a head-dependent semantic relationship with *w*, then *w*'s lexical activation will have been recently re-activated, so that the integration cost will be less than if none of the intermediate words were involved in a head-dependent relationship with *w*.

The hypothesis that longer distance integrations are more expensive is supported by evidence that local integrations are easier to make than more distant integrations, leading to a locality or recency preference in instances of ambiguity (Kimball, 1973; Frazier and Fodor, 1978; Gibson, 1991; Stevenson, 1994; Gibson et al., 1996a). For example, consider the ambiguous attachment of the adverb ‘yesterday’ in (8):

(8) The bartender told the detective that the suspect left the country yesterday.

The adverb ‘yesterday’ can be linked to either the local verb ‘left’ or to the more distant verb ‘told’. The more local attachment is strongly preferred. In this activation-based framework, the local attachment is preferred because it takes less resources to reactivate the verb ‘left’ because it occurred more locally in the input string, so that its activation has decayed less than that of the verb ‘told’.

Two components of a distance-based integration cost function I(n) need to be specified: (1) what the function I(n) is and (2) what kind of linguistic elements cause processing increments, i.e. what characterizes n in I(n). For simplicity, it will initially be assumed that the function I(n) is a linear function whose slope is one and whose intercept is zero, i.e. I(n) = n. (However, see Section 2.6.7 for evidence that the integration and memory cost functions are not linear as n gets large.) Some possibilities to be considered for the kind of linguistic elements causing processing increments include: words, morphemes, components of syntactic structures (e.g. noun phrases, verb phrases), or components of discourse structures. Although processing all words probably causes some integration cost increment, it is hypothesized here that substantial integration cost increments are caused by processing words indicating new discourse structure (Kamp, 1981; Heim, 1982), particularly new discourse referents. A discourse referent is an entity that has a spatio-temporal location so that it can later be referred to with an anaphoric expression, such as a pronoun for NPs, or tense on a verb for events (Webber, 1988). The motivation for this hypothesis is that much computational effort is involved in building a structure for a new discourse referent (e.g. Haviland and Clark, 1974; Haliday and Hasan, 1976; Garrod and Sanford, 1977; see Garrod and Sanford (1994) for a recent summary). Expending this effort causes substantial decays in the activations associated with preceding lexical items. Thus processing an NP which refers to a new discourse object eventually leads to a substantial integration cost increment, as does processing a tensed verb, which indicates a discourse event.[6] The discourse-reference-based hypothesis provided here is a first approximation of the integration cost function:

(9) Linguistic integration cost
The integration cost associated with integrating a new input head h_2 with a

[6]The hypothesis that there is a substantial computational cost associated with constructing structures for new discourse referents is related to Crain and Steedman’s (1985) ambiguity resolution principle, the principle of parsimony, which prefers simpler discourse structures in cases of ambiguity (cf. Altmann and Steedman, 1988). According to the principle of parsimony, the sentence processing mechanism prefers not to assume the existence of additional unmentioned objects in the discourse, if it has the choice. See Section 3.5.3 for more discussion of the relevance of Crain et al.’s referential theory to the integration and memory cost functions.

head h_1 that is part of the current structure for the input consists of two parts: (1) a cost dependent on the complexity of the integration (e.g. constructing a new discourse referent); plus (2) a distance − based cost: a monotone increasing function I(n) energy units (EUs) of the number of new discourse referents that have been processed since h_1 was last highly activated. For simplicity, it is assumed that I(n) = n EUs.

It is likely that processing other aspects of discourse structures, such as new discourse predicates, also causes additional integration cost. See Section 2.6.4 where this possibility is elaborated. It is also likely that processing every intervening word, whether introducing a new discourse structure or not, causes some integration cost increment for the distance-based component. Furthermore, because the distance-based cost function is assumed to be determined by the amount of resources expended between the head and dependent that are being integrated, integration cost will also depend on the complexity of the integrations in the intermediate region. Although this is potentially an important source of integration cost, for simplicity of presentation we will also initially ignore this complexity source.[7]

2.3. Memory cost

The second component of the SPLT is a theory of linguistic memory cost. According to this component of the theory, there is a memory cost associated with remembering each category that is required to complete the current input string as a grammatical sentence. This claim requires a theory of phrase structure. For simplicity, I will assume a syntactic theory with a minimal number of functional categories, such as Head-driven Phrase Structure Grammar (Pollard and Sag, 1994) or Lexical Functional Grammar (Bresnan, 1982).[8] Under these theories, the minimal number of syntactic head categories in a sentence is two: a head noun for the subject, and a head verb for the predicate. If words are encountered that necessitate other syntactic heads to form a grammatical sentence, then these categories are also predicted, and additional memory load is incurred. For example, at the point of processing the second occurrence of the word ‘the’ in the object-extracted RC example (1a) ‘The reporter who the senator attacked admitted the error’, there are four obligatory syntactic predictions: (1) a verb for the matrix clause, (2) a verb for

[7]Another potential source of integration complexity is the interference of the intervening head-dependent relationships on the one being formed (cf. Lewis, 1996, 1998). To the extent that the intervening head-dependent relationships are similar to the one being formed, this may make the current head-dependent relationship more complex. This possible component of integration difficulty might account for the additional difficulty of processing nested structures like (vi) (Stolz, 1967; Schlesinger, 1968; example from Bever, 1970) (see footnote 4):

(iv) # The lion which the dog which the monkey chased bit died.

[8]The SPLT is also compatible with grammars assuming a range of functional categories such as Infl, Agr, Tense, etc. (e.g. Chomsky, 1995) under the assumption that memory cost indexes predicted *chains* rather than predicted categories, where a chain is a set of categories that are co-indexed through syntactic movement (Chomsky, 1981).

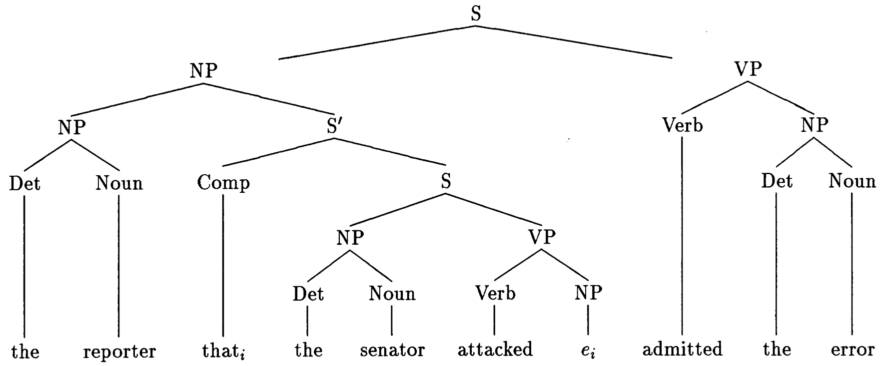

Fig. 1. The tree structure for the sentence ‘The reporter who the senator attacked admitted the error’.

the embedded clause, (3) a subject noun for the embedded clause and (4) an empty category NP for the wh-pronoun ‘who’ (see the tree diagram in Fig. 1).

It will initially be assumed that the memory cost function is the same as the integration cost function: a discourse-based locality function.[9] The conceptual motivation for this claim is as for the locality-based integration cost hypothesis: there is computational effort associated with building an intervening new discourse structure, which makes it harder to keep the current syntactic predictions in mind, so more effort must be spent in keeping the syntactic predictions activated. Hence, a category that was predicted earlier is associated with more cost than a category that was predicted more recently (cf. Kaplan, 1974; Wanner and Maratsos, 1978; Hawkins, 1990, 1994; Joshi, 1990; Rambow and Joshi, 1994). This assumption fits with what is known about short-term memory recall in non-linguistic domains: it is harder to retain items in short-term memory as more interfering items are processed (see, e.g. Waugh and Norman, 1965; see Baddeley, 1990; Anderson, 1994; and Lewis, 1996 for recent summaries).[10]

It is hypothesized that there is one exception to the locality-based memory cost proposal: The prediction of the matrix predicate is proposed to be cost-free. The motivation for this proposal is that the parser is always expecting a predicate (and possibly a subject for this predicate also), so that this expectation is built into the parsing mechanism, and is therefore cost-free.[11] All other category predictions are

[9]See Section 6 for a different approach, in which the memory cost for a predicted category remains at the same level over the processing of intervening material.

[10]It has been demonstrated that it is not just the passage of time that makes retaining items in working memory difficult. In particular, a recall task performed at two different rates of presentation within a natural range (i.e. not so fast as to make lexical access difficult: 1 s per item versus 4 s per item) resulted in the same pattern of memory recall (Waugh and Norman, 1965). Thus the initial hypothesis for the SPLT memory cost function is stated in terms of linguistic elements processed, not the quantity of time that has passed.

[11]This claim turns out to be a slight oversimplification, because of closure phenomena: the fact that opening a new clause causes shunting of the material in the old clause out of working memory. Closure phenomena are discussed in Section 2.6.6, and a more accurate hypothesis regarding which predicted predicates are cost-free is presented there also.

dependent on the lexical material that is encountered, and therefore cannot be predicted in advance, leading to a memory cost associated with each of these lexically-dependent categories. Memory cost will be quantified in terms of memory units (MUs). The SPLT memory cost hypothesis is summarized in (10):

(10) Syntactic prediction memory cost
(a) The prediction of the matrix predicate, V_0, is associated with no memory cost.
(b) For each required syntactic head C_i other than V_0, associate a memory cost of M(n) memory units MUs where M(n) is a monotone increasing function and n is the number of new discourse referents that have been processed since C_i was initially predicted.

As for the integration cost function, it turns out that almost any monotonically increasing function makes the correct predictions with respect to the contrasts to be accounted for here. For simplicity, it is initially assumed that the memory cost function M(n) is linear with a slope of one and an intercept of zero, i.e. M(n) = n. This assumption is consistent with much of the processing evidence to be reported here, with one class of exceptions: examples in which incomplete dependencies are greatly lengthened. These kinds of examples suggest that the memory cost function is not linear in its limiting behavior, but rather approaches a maximal cost (cf. Church, 1980; Gibson et al., 1996b). The relevant examples, along with a discussion of their implications with respect to the memory cost function, are presented in Section 2.6.7.

2.4. *The relationship between memory and integration cost*

Following Just and Carpenter (1992), it is assumed that linguistic integration processes and storage access the same pool of working memory resources.[12] The memory units in this pool of resources can therefore be used for either storage or computation (integration). The amount of energy or work that is necessary to perform an integration is quantified in terms of energy units:

(11) An energy unit (EU) = memory unit (M) $*$ time unit (TU)

For example, suppose there are ten MUs available for performing an integration which requires five EUs. If all ten MUs perform the integration, then the time required to complete the integration is 5 EUs/10 MUs = 0.5 TUs. If seven of the MUs are occupied with another task (e.g. storage) so that only three MUs are available to perform the integration, then the integration will take 5 EUs/3 MUs = 1.67 TUs.

[12] No claim is being made here with respect to whether the computational resources used in sentence processing are a general pool of memory resources, as argued by Just and Carpenter (1992), or whether they are a modular linguistic memory pool, as argued by Waters and Caplan (1996).

These assumptions result in the hypothesis that the time required to perform a linguistic integration, as indexed by reading times, is a function of the ratio of the integration cost required at that state to the space currently available for the computation. Thus the greater the memory cost, the smaller the resources available to perform linguistic integrations, and the longer the integration steps will take. These ideas are formalized in (12):[13]

(12) The timing of linguistic integration:
$t_{struct-integ} = C * I_{struct-integ}/(M_{capacity} - M_{current-memory-used})$;
$t_{struct-integ}$ is the time required to perform an integration;
C is a constant
$I_{struct-integ}$ is the quantity of energy resources (in EUs) required to perform the integration as determined by the function I(n) in (9)
$M_{capacity}$ is the linguistic working memory resource capacity of the listener/reader (in MUs)
$M_{current-memory-used}$ is the memory resources already in use (in MUs) as determined by the function M(n) in (10).

The evidence currently available which supports this hypothesis comes from experiments performed by King and Just (1991) who looked at the reading time behavior of subject pools with different memory capacities for language. This evidence is discussed in Section 2.6.3.

2.5. Memory cost and intuitive complexity judgments

Following earlier researchers, it is assumed that the relative intuitive acceptability of two unambiguous sentences which are controlled for plausibility is determined by the maximal quantity of memory resources that are required at any point during the parse of each sentence (see, e.g. Gibson (1991) and the references therein). If more memory resources are required than the available working memory capacity, then processing breakdown eventually results, as in the processing of (3b), repeated here as (13):

(13) # The administrator who the intern who the nurse supervised had bothered lost the medical reports.

According to this hypothesis, (13) is unprocessable because there is a state during its parse that requires more memory resources than are available. If sentence length were a major factor in determining intuitive sentence complexity, then (14) should be judged roughly as complex as (13), since the two sentences are of the same length and have the same meaning. However, (13) is much more complex than (14).

[13]The time for linguistic integration $T_{struct\text{-}integ}$ in (12) is not the only time involved in comprehending a word. In particular, lexical access, eye-movements in reading and button-presses in self-paced reading all require some time to compute, so that there is a base-line time on top of which linguistic integration time is added.

(14) The nurse supervised the intern who had bothered the administrator who lost the medical reports.

Furthermore, including additional right-branching clauses at either the beginning or end of (13) as in (15a) and (15b) results in sentence structures with the same intuitive complexity as (13):

(15) (a) # The doctor thought that the medical student had complained that the administrator who the intern who the nurse supervised had bothered lost the medical reports.
(b) # The administrator who the intern who the nurse supervised had bothered lost the medical reports which the doctor thought that the medical student had complained about.

Thus intuitive complexity is not determined by the average memory complexity over a sentence. We therefore assume that the intuitive complexity of a sentence is determined by the maximal memory complexity reached during the processing of the sentence.

2.6. Empirical evidence for the components of the SPLT

This section provides empirical evidence for a number of components of the SPLT. Section 2.6.1 provides evidence for the discourse-based definition of locality within the memory cost function. Section 2.6.2 provides evidence for the locality-based integration hypothesis. Section 2.6.3 provides evidence for the single resource pool hypothesis. In Section 2.6.4 evidence is described which supports an extended version of the memory and integration cost functions according to which there is additional cost for new discourse predicates. Section 2.6.5 provides evidence for the claim that there is no memory cost for predicting the matrix predicate in a sentence. In Section 2.6.6, evidence is provided for a clause-based closure principle. Section 2.6.7 provides evidence that the integration and memory cost functions are non-linear in the limit, approaching a maximal cost.

2.6.1. A discourse-based memory cost function

According to the discourse-based memory cost hypothesis, intervening elements which cause substantial integration and memory cost increments are words introducing new discourse referents: NPs (object referents) and the main verbs of VPs (event referents). Evidence for this claim is provided by Gibson and Warren (1998a) who showed that doubly nested RC structures are easier to process when a first- or second-person pronoun (an *indexical* pronoun) is in the subject position of the most embedded clause, as compared with similar structures in which a proper name, a full NP or a pronoun with no referent is in the subject position of the most embedded:

(16) (a) Indexical pronoun: The student who the professor who I collaborated with had advised copied the article.

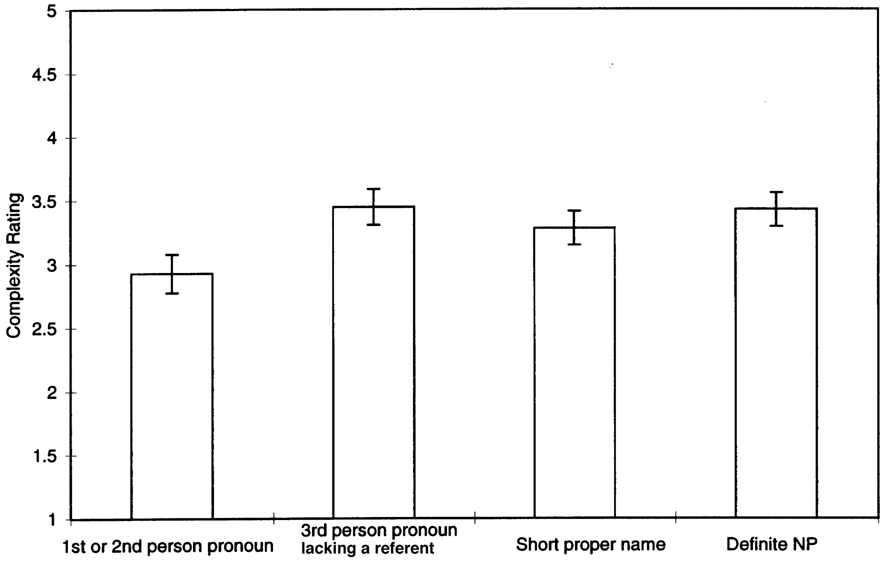

Fig. 2. Complexity ratings for nested structures containing different kinds of NPs in the most embedded subject position (from Gibson and Warren, 1998a). The scale that participants used went from 1 (not complex: easy to understand) to 5 (very complex: hard to understand).

(b) Short name: The student who the professor who Jen collaborated with had advised copied the article.
(c) Full NP: The student who the professor who the scientist collaborated with had advised copied the article.
(d) No referent pronoun: The student who the professor who they collaborated with had advised copied the article.

In an acceptability questionnaire, participants rated the items with the indexical pronouns significantly easier to process than any of the other three conditions (see Fig. 2): (1) items with short names (e.g. 'Jen' in (16b)), (2) items with full definite NPs (e.g. 'the scientist' in (16c)) and (3) items with a pronoun lacking a discourse referent (e.g. 'they' in (16d)).[14] These observations can be accounted for in the SPLT framework if the memory cost for a predicted category is increased when a new discourse referent is processed. At the point of processing the most embedded subject in these structures, there are four predicted categories associated with increasing memory cost (in addition to the prediction of the matrix verb): two embedded verbs and two embedded NP-empty-categories. According to the proposed SPLT memory cost function, the memory cost for each of these predicted categories increases when an NP which is new to the discourse is processed at this point (e.g. 'they' in (16d); 'Jen' in (16b); or 'the scientist' in (16c)). If the most

[14]The higher complexity ratings of the condition containing a pronoun lacking a discourse referent may reflect the infelicity of using a third person pronoun without an antecedent.

embedded subject is an indexical pronoun (e.g. the first-person pronoun 'I' in (16a), then its referent is in the current discourse (which is assumed to always include a speaker/writer and a hearer/reader) and the memory costs for the predicted categories do not increase. As a result, the maximal memory cost required to process a doubly nested RC structure with a new referent in its most embedded subject position is greater than that required to process a similar structure with an old referent in its most embedded subject position. See Section 3.2 for details of memory cost computations on doubly nested RC structures like these.

The discourse-based memory cost hypothesis therefore provides an account for the observation that a doubly nested RC structure in a null context is processable only when the most embedded NP is an indexical pronoun (Bever, 1970, 1974; Kac, 1981; Gibson, 1991; Gibson and Warren, 1998a; Kluender, 1998):

(17) (a) The reporter [everyone [I met] trusts] said the president won't resign yet. (Bever, 1974)
(b) A book [that some Italian [I've never heard of] wrote] will be published soon by MIT Press. (Frank, 1992)
(c) Isn't it true that example sentences [that people [that you know] produce] are more likely to be accepted. (De Roeck et al., 1982)

Bever (1970, 1974) was the first to note that examples like these were acceptable. He attributed their acceptability to the syntactic non-similarity of the three kinds of subject NPs in the structure. However, this account does not explain why a pronoun and not some other dissimilar NP, such as a proper name or an indefinite NP, must occur in the most embedded subject position in order to make the structure acceptable. Kac (1981) was the first to notice the generalization that these structures were acceptable with pronouns in the most embedded position, except that he hypothesized that having *any* pronoun in the subject position of the most embedded RC makes a doubly nested RC structure easier to process than corresponding sentences with full NPs. This claim is too strong, because structures whose most embedded subject is a pronoun without a referent in the discourse, as in (16d), are as complex as those whose most embedded subject is a name or a full name, as in (16b) and (16c).

Other syntactic complexity theories are also not compatible with the discourse-dependent complexity difference discussed here. For example, Kimball's (1973) theory indexes complexity with the number of incomplete clauses that need to be kept in memory at a particular parse location, and Gibson's (1991) theory indexes complexity with the number of incomplete thematic role assignments. These metrics do not predict the observed discourse-based contrast, nor are they extendible to account for the difference.

2.6.2. Evidence for the distance-based integration hypothesis

Under the single resource pool hypothesis in (12) above, it is necessary to determine the quantity of linguistic memory resources that are used at each processing state in order to predict exact SPLT integration times. However, reasonable first approximations of comprehension times can be obtained from the integration costs

(18) Object – extracted relative clause

	Input word								
	The	reporter	who	the	senator	attacked	admitted	the	error
Integration cost (in EUs)	–	I(0)	I(0)	I(0)	I(0)	I(1) + I(2)	I(3)	I(0)	I(0) + I(1)

alone, as long as the linguistic memory storage used is not excessive at these integration points. When memory costs are also taken into consideration, integration times at points of higher memory cost will be increased relative to integration times at points of lower memory cost. For simplicity, examples of the SPLT integration costs will first be compared with reading times, before the memory component of the theory is presented. Additionally, we will consider only the distance-based component of the integration cost function. The SPLT distance-based integration cost profiles for the object-extraction (1a) and the subject-extraction (1b) are given in (18) and (19) respectively.

The first point at which an integration takes place is at the second word ‘reporter’, which is integrated with the preceding word ‘the’. No new discourse referents have been processed since the determiner ‘the’ was processed, so that the distance-based integration cost is I(0) EUs at this point. Note that the two words ‘the’ and ‘reporter’ together form a new discourse referent, so constructing this new referent also consumes some integration resources. We are initially ignoring this cost, because it is part of the distance-based component of integration cost. We will also ignore this cost for the other words in these sentences which head new discourse referents: the nouns ‘reporter’, ‘senator’, and ‘error’; and the tensed verbs ‘attacked’ and ‘admitted’.

The next word ‘who’ is now integrated into the structure, attaching to the most recent word ‘reporter’. No new discourse referents have been processed since the attachment site ‘reporter’ was processed (note that ‘who’ is not a new discourse referent: it is a pronominal element referring to the established discourse referent ‘the reporter’), resulting in a cost of I(0) EUs. The word ‘the’ is integrated next, again with no intervening new discourse referents, for a cost of I(0) EUs. The word ‘senator’ is integrated next, again at a cost of I(0) EUs.

Processing the next word, ‘attacked,’ involves two integration steps. First, the verb ‘attacked,’ is attached as the verb for the NP ‘the senator’. This attachment involves assigning the agent thematic role from the verb ‘attacked’ to the NP ‘the senator’. This integration consumes I(1) EUs, because one new discourse referent (‘attacked’) has been processed since the subject NP ‘the senator’ was processed. The second integration consists of attaching an empty-category as object of the verb ‘attacked’ and co-indexing it with the relative pronoun ‘who’. Two new discourse referents have been processed since ‘who’ was input – the object referent ‘the senator’ and the event referent ‘attacked’ – so that this integration consumes an additional I(2) EUs, for a total cost of I(1) + I(2) EUs.

The verb ‘admitted’ is then integrated as the main verb for the subject NP ‘the reporter’. This integration consumes I(3) EUs because three new discourse referents have been processed since ‘the reporter’ was input: the object referent ‘the senator' and the two event referents 'attacked' and 'admitted'. Next, the word 'the' is integrated into the structure, at a cost of I(0) EUs. Finally, the noun 'error' is integrated into the structure. This step involves two integrations: the noun 'error' integrating with the determiner 'the' at a cost of I(0) EUs, and the NP 'the error' integrating as the object of the verb 'admitted' at a cost of I(1) EUs, because one new discourse referent – 'the error' – has been processed since the verb 'admitted' was input.

The distance-based integration cost profile for the subject-extraction, (1b), is presented in (19). The integration costs for this construction are the same as for the object-extraction, except in the embedded clause. In the subject-extraction, the cost of integrating 'attacked' is I(0) + I(1) EUs, corresponding to (1) the attachment of a gap in subject-position of the embedded clause, crossing zero new discourse referents and (2) the attachment of the embedded verb 'attacked' to its preceding subject, an integration which crosses one new discourse referent ('attacked').[15] The determiner 'the' is integrated next, consuming a cost of I(0) EUs. The noun 'senator' is then integrated at a cost of I(0) + I(1) EUs.

(19) Subject – extracted relative clause

	Input word								
	The	reporter	who	attacked	the	senator	admitted	the	error
Integration cost (in EUs)	–	I(0)	I(0)	I(0) + I(1)	I(0)	I(0) + I(1)	I(3)	I(0)	I(0) + I(1)

Thus the SPLT integration theory predicts reading times to be fast throughout both relative clause constructions, with the exception of (1) the matrix verb in each, where a long distance integration occurs and (2) the embedded verb in the object-extraction construction, where two substantial integrations take place. Furthermore, if integration cost is also dependent on the complexity of the intervening integrations, as the activation-based account predicts, then the integration cost on the matrix verb 'admitted' should be larger in the object-extraction than in the subject-extraction, because more complex integrations take place between the subject and the matrix verb in the object-extraction than in the subject-extraction. In particular, integrating the embedded verb is more complex in the object-extraction structure than in the subject-extraction structure. (Note that the integration cost profiles presented here have been simplified, so that the predicted difference between the integration costs of the matrix verb in the two extraction structures is not represented in (18) and (19).)

A comparison between SPLT predicted integration costs and reading times for the two relative clause constructions is presented in Fig. 3, based on data gathered by

[15] If one assumes a syntactic theory according to which there is no empty category mediating the subject-extraction in an example like (19), then the integration cost is simply I(1) EUs, corresponding to the cost of attaching the verb 'attacked' to its subject 'who'. The integration cost is virtually the same under either set of assumptions, because the cost of attaching the mediating empty category is very low, at I(0) EUs. Thus theories with and without subject empty categories make virtually the same predictions here.

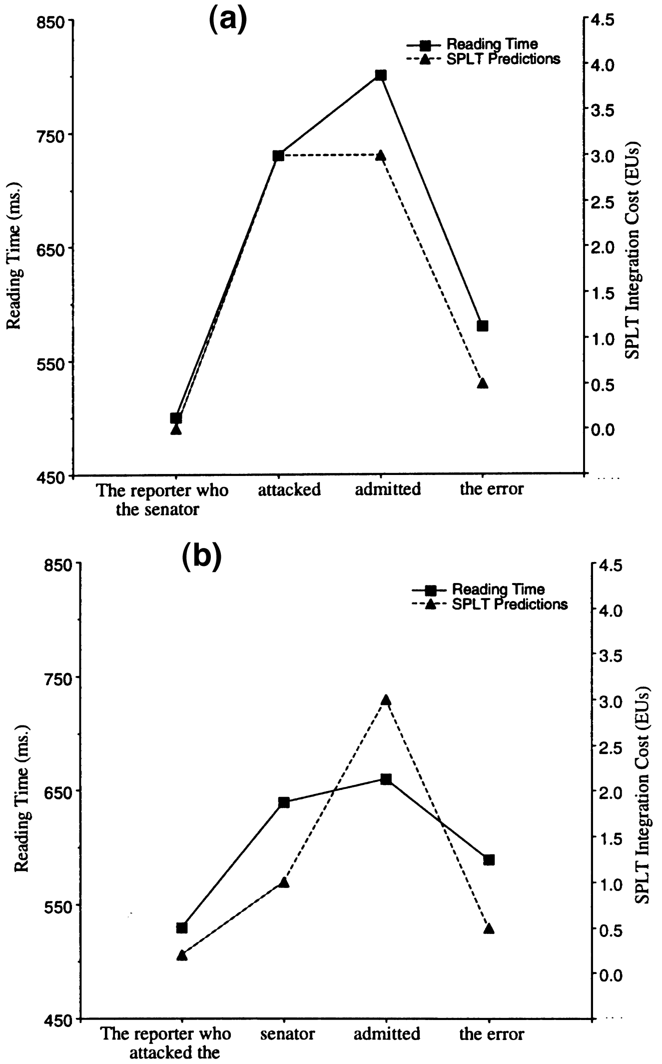

Fig. 3. A comparison between reading times recorded by King and Just (1991) and the integration costs predicted by the SPLT assuming I(n) = n. The object-extracted relative clauses are presented in panel (a), and the subject-extracted relative clauses are presented in panel (b).

King and Just (1991) in a self-paced word-by-word reading experiment, divided into the regions which King and Just analyzed. The task in King and Just's experiment involved reading sentences as well as remembering the final word for each sentence in a series of three sentences, where the target relative clause structures were always in the second or third sentence. This additional component to the task makes the reading times slow, almost double normal reading times.

As can be seen from the graphs, the correlation between the predicted times and the actual reading times in King and Just's experiments is high for both types of RCs. However, because the reading task in these experiments was difficult for subjects, it is worth comparing the predicted costs to reading times from a task in which there is no additional memory component, so that the task is as much like normal language comprehension as possible. Gibson and Ko (1998) are in the process of conducting such an experiment using the self-paced word-by-word paradigm using 32 different relative clause items (much like the ones King and Just used)

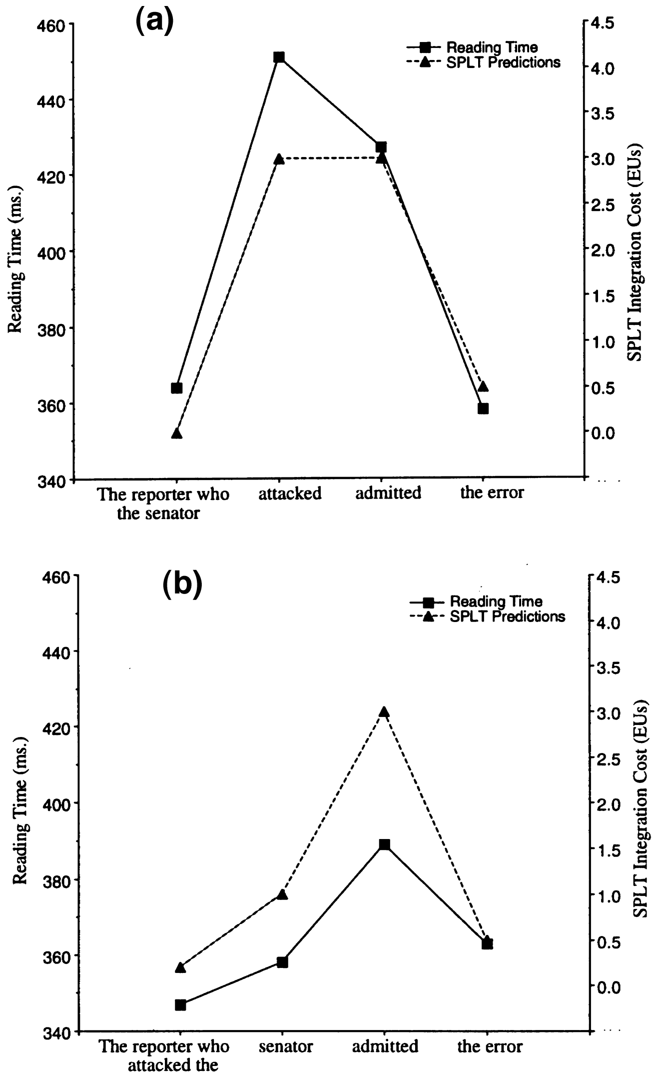

Fig. 4. A comparison between reading times recorded in Gibson and Ko's experiment and the integration costs predicted by the SPLT assuming I(n) = n. The object-extracted relative clauses are presented in panel (a), and the subject-extracted relative clauses are presented in panel (b).

and 75 filler sentences of similar length and complexity. Six conditions in addition to the two discussed here were also run in this experiment, testing other predictions of the SPLT not discussed here. Each sentence in the experiment was followed by a question which tested the reader's comprehension. Participants generally understood the sentences in the experiment, as indicated by a 91% accuracy rate over the two conditions discussed here, after 48 participants had taken part. A comparison between SPLT predicted integration costs and reading times from this experiment on items whose comprehension questions were answered correctly is presented in Fig. 4, divided into the same regions as in Fig. 3. As can be seen from these graphs, the correlation between the predicted times and the actual reading times is also high in this experiment. As predicted by the SPLT, reading times in the object extraction are fast before the two verbs, slow at the two verbs, and then fast again afterward. In contrast, reading times do not increase much during the processing of a subject-extracted relative clause, but then increase at the matrix verb.

In addition, note that reading times are slower at the matrix verb in the object-extraction than in the subject-extraction in both experiments. Although not taken into consideration by the oversimplified integration function I(n), this difference also follows from the SPLT locality hypothesis, because integration cost is dependent on the complexity of the intervening integrations. Thus, reading times at the matrix verb in the object-extracted RC are slower than in the subject-extracted RC because the integrations intervening between the subject NP and the matrix verb are more complex in the object-extraction. Taken as a whole, the reading time data provide evidence for the locality-based SPLT integration theory.

2.6.3. Memory span effects: evidence for the single resource pool hypothesis

In a study of individual differences in working memory capacity and reading, King and Just (1991) presented results suggesting (1) that participants with lower memory capacity – as measured by the reading span task developed by Daneman and Carpenter (1980) – read more slowly than participants with higher memory capacity and (2) that participants with low memory capacity read even more slowly at points of high linguistic computation and memory cost. In particular, the low memory span participants read even more slowly at the embedded verb of the object-extracted relative clause construction (a point of relatively high integration complexity), as compared to at the embedded verb of the subject-extracted relative clause construction (a point of relatively low integration complexity). These two results are explained by Just and Carpenter's hypothesis that linguistic storage and integration access the same pool of resources (Just and Carpenter, 1992). In particular, if there is less storage space (a smaller working memory capacity), then integrations take longer, resulting in the first observation of a main effect of working memory capacity. Furthermore, the relationship between storage space used and integration time required is not linear: the more storage space that the processor uses, the slower that the processor works. This non-linear relationship predicts the observed interaction between memory capacity and integration time. Since the SPLT incorporates the same hypothesis of a single pool of resources for storage and computation, these results follow in the same way.

2.6.4. An extended version of the memory and integration cost functions: additional cost for new discourse predicates

There are many possible memory and integration cost functions other than the one that has been assumed here. For example, within the realm of discourse-based proposals, it is possible that processing each distinct discourse referent, whether new or old, causes an increase in memory cost for a predicted syntactic category. When processing an old discourse referent there may be additional processing cost associated with searching for the appropriate referent, e.g. in pronoun resolution. This processing cost may cause an increase in the memory cost for the predicted syntactic categories at that parse state. It is also possible that different types of intervening discourse referents cause different increments in memory cost for predicted syntactic categories. For example, processing an event referent might consume more resources and hence cause a larger memory increment for a predicted syntactic category than processing an object referent.

Another natural extension of the memory cost function in (10) increments memory and integration cost not only when new discourse referents are processed, but also when intervening discourse *predicates* in a discourse representation structure – those linguistic elements representing predications – are processed, because predicates also involve additional discourse structure. Thus it may be that not only nouns and verbs cause memory cost increments for predicted syntactic categories. Adjectives and contentful prepositions may also cause memory cost increments, because they indicate predications. Although there is currently no relevant experimental evidence from the processing of unambiguous structures either for or against this hypothesis, there exists some evidence for the hypothesis from the processing of ambiguous structures. In particular, if a preposition separating two NP sites in a relative clause attachment ambiguity is predicative (e.g. 'con' in Spanish, 'with' in English) then there is a stronger preference to attach the relative clause to the more local NP site than if the NP sites provide all of the thematic information and the intervening preposition is thematically empty such as 'de' in Spanish or 'of' in English (Gilboy et al., 1995; cf. De Vincenzi and Job, 1993). This results in a preference for local attachment of the relative clause in both the Spanish and English versions of (20) whereas the local attachment attraction is much weaker in (21), so much so that there is a non-local attachment preference in the Spanish version:[16]

(20) El conde pidó [$_{NP_1}$ el bistec con [$_{NP_2}$ la salsa]] [que el cocinero preparaba especialmente bien].
'The count ordered [$_{NP_1}$ the steak with [$_{NP_2}$ the sauce]] [that the chef prepared especially well].'

[16] Cuetos and Mitchell's data suggest that there is a slight local attachment in English examples like (21) (Cuetos and Mitchell, 1988). Gibson et al. (1996a) hypothesize that the cross-linguistic difference between English and Spanish attachment preferences in examples like (21) is caused by a difference between the weight associated with the high attachment preference factor (predicate proximity in Gibson et al.'s system) across languages. See Gibson et al. (1996a, 1998b) and Hemforth et al. (1998) for further discussion of this issue, which is beyond the scope of this paper.

(21) Un alumno insultó a [$_{NP_1}$ los profesores de [$_{NP_2}$ las clases]] [que no gustaron a los estudiantes].
'The student insulted [$_{NP_1}$ the professors of [$_{NP_2}$ the courses]] [that were disliked by the students].'

Because prepositions like ‘of’ (‘de’) do not involve predication beyond what is specified by their head nouns, they do not cause the addition of a new predicate to the discourse representation, and hence there is only one intervening predicative element – NP_2 ‘las clases’ (‘the courses’) – between the relative clause and the non-local attachment site in structures like (21). In contrast, there are two intervening predicative elements between the relative clause and the non-local attachment site in structures like (20): NP_2 ‘la salsa’ (‘the sauce’) and the preposition ‘con’ (‘with’). The attachment cost to the non-local site is therefore greater in structures like (20), leading to a stronger preference for local attachment.

Further evidence for the extended version of the memory cost function is provided by relative clause attachment preferences involving two NP sites with adjectival modifiers. The predicate-based memory cost function predicts that the presence of a prenominal adjective modifying the second NP site should increase the preference for local attachment, but that the presence of an adjective modifying the upper should not change the attachment preferences. Although the experimental data is limited, this is exactly the pattern of data observed by Gilboy et al. (1995) in their studies of Spanish and English relative clause attachments. For example, consider the English examples in (22):

(22) (a) I was surprised by [$_{NP_1}$ the etching of [$_{NP_2}$ the sculpture]] [that was in the town hall].
(b) I was surprised by [$_{NP_1}$ the etching of [$_{NP_2}$ the odd sculpture]] [that was in the town hall].
(c) I was surprised by [$_{NP_1}$ the odd etching of [$_{NP_2}$ the sculpture]] [that was in the town hall].

If an adjective modifies the second NP site as in (22b), then the distance to the first site in terms of the number of predicates crossed is larger than if there is no adjective as in (22a). However, if an adjective modifies the first NP site as in (22a), then the distance to the first site is no different than if there is no adjective. Hence the local attachment preference is predicted to be strongest in examples like (22b), with no difference predicted between examples like (22a) and (22c). This is the pattern of results that Gilboy et al. (1995) observed for both English and Spanish.

2.6.5. Zero memory cost for predicting the matrix predicate

According to the SPLT memory cost hypothesis in (10), there is no memory cost associated with predicting the matrix predicate, because this prediction is built into the parser. Perhaps a simpler hypothesis is that the memory function is the same for all predicted categories. However, there is evidence that this is not the case. In particular, Hakes et al. (1976) demonstrated using a phoneme-monitoring task that a relative clause is no more difficult to process when it modifies the subject of the matrix verb, as in (23a), than when it modifies the object of the matrix verb, as in (23b) (see also Holmes (1973) and Baird and Koslick (1974) for related evidence):

(23) (a) Matrix subject modifier: The reporter [that the senator attacked] ignored the president.
(b) Matrix object modifier: The president ignored the reporter [that the senator attacked].

Although the task was sensitive enough to detect differences between subject- and object-extractions (demonstrating that object-extractions are more complex), no statistical difference was observed between matrix-subject- and matrix-object-RCs during the processing of the RCs.[17] This result is not consistent with a memory cost function which increases over intervening linguistic material for a predicted matrix verb. In particular, such a function predicts that modifying the subject should cause an increase in the memory cost for predicting the matrix verb, whereas modifying the object should not cause such an increment.

On the other hand, if there is no memory cost associated with predicting the matrix predicate then the lack of a complexity difference is accounted for. Corroborating evidence for this proposal is provided by Gibson and Thomas (1997a) who observed that there is no complexity difference between pre-verbal and post-verbal conditions of three different constructions involving more complex embeddings.

2.6.6. Clause-based syntactic closure

It turns out that the zero-cost matrix-predicate hypothesis in (10a) is an oversimplification. In particular, this hypothesis predicts a contrast in memory load between processing a construction in the matrix clause, where there is no memory cost associated with predicting the matrix verb, and processing the same construction in an embedded clause, where there is an increasing memory cost associated with predicting the embedded verb. For example, a contrast between (24a) and (24b) is predicted:

(24) (a) The reporter that the senator attacked ignored the president.
(b) The editor said that the reporter that the senator attacked ignored the president.

(24a) contains a relative clause modifying the subject NP of the matrix clause, and (24b) consists of (24a) in an embedded context. As the SPLT memory cost hypothesis is currently stated, there is no memory cost associated with the prediction of the matrix verb 'ignored' during the processing of (24a), but there is a memory cost associated with the prediction of the embedded verbal complement of the matrix clause (also the verb 'ignored') during the processing of (24b). Thus, the maximal memory resources

[17] Although not significant, the numerical effect actually favored the subject-modifier over the object-modifier, the reverse direction from that would be expected if memory cost increased for the matrix subject-verb dependency.

required to process (24b) (at the point of processing the noun ‘senator’) include an additional M(2) MUs relative to the maximal memory resources required to process the matrix version (24a). A substantial complexity difference between the two is therefore predicted by this memory cost hypothesis. However, this prediction is incorrect: there is no complexity contrast between (24a) and (24b).

There is a simple solution to this quandary. As originally observed by Kimball (1973) and Church (1980), the parser cannot keep an unbounded number of sites open for potential attachments because of the bounded size of working memory. The parser must therefore implement some closure principle whereby some potential attachment sites are shunted away, no longer to be considered as potential attachment sites. There currently exists a large amount of evidence for a clause-based closure principle (Jarvella, 1971, 1979; Caplan, 1972; Frazier and Fodor, 1978; Marslen-Wilson et al., 1978; Bever and Townsend, 1979; Milsark, 1983; Roberts et al., 1997). For example, all languages have a strong locality preference in attachment ambiguities involving two VP attachment sites, as in (25) (Kimball, 1973; Milsark, 1983; Carreiras and Clifton, 1993; Mitchell, 1994; Gibson et al., 1996b):

(25) (a) Spanish: Juan dijo que Bill se murió (# morirá) ayer.
(b) English: John said Bill died (# will die) yesterday.

In both the English and Spanish versions of (25) there is a strong preference to attach the adverbial ‘yesterday’ (‘ayer’) to the more recently occurring VP. This strong preference is present even when the tense information on the local attachment site – future tense – is incompatible with the adverbial – ‘yesterday’. This situation suggests that the higher clause is not being considered as a potential attachment site.The clause-based closure principle proposed here is presented in (26):[18]

[18] A possible alternative to (26) is (vii):

(vii) Alternative clausal closure principle: The initiation of a new clause causes closure of an old clause.

However, such a principle incorrectly predicts that sentences like (vii) and (ix) are difficult to parse (Church, 1980; Gibson, 1991):

(viii) i called the guy who smashed up my brand new car a rotten driver.

(ix) Thre policeman persuaded the man who admitted his guilt to turn himself in.

Consider (viii). If the clause headed by the verb ‘called’ is closed upon initiation of the relative clause ‘who smashed up my brand new car’, then there should be severe processing difficulty when the NP ‘a rotten driver’ is encountered, because this NP needs to be integrated into the earlier clause, which has been shunted out of working memory, by hypothesis. But no such difficulty occurs. Similarly, (26) predicts severe difficulty when processing ‘to turn himself in’ in (ix), but there is no such difficulty. As a result of observations like these, the proposed closure principle does not apply when a preferred dependent of an element in a current clause has not yet appeared (e.g. the second NP argument of ‘called’ in (viii); the clausal argument of ‘convinced’ in (ix)).

(26) Clause – based closure: The initiation of a new clause causes closure of an earlier clause whose preferred dependents have been satisfied.

Given a clause-based closure principle like (26), the lack of complexity contrast between (24a) and (24b) can be accounted for if one assumes that the top-level predicted predicate in the current memory buffer is cost-free, rather than only the predicted matrix predicate. After processing the word ‘that’ in (24b), the matrix verb ‘said’ is chunked out of working memory, so that the highest predicted predicate at this point is the predicted embedded clause. Under the assumption that the top-level predicted clause is cost-free, the prediction of this predicate is associated with zero memory cost, and the lack of a memory complexity difference follows.

In order to formalize the refined zero-cost predicted category hypothesis, the term *matrix argument clause* is defined in (27):

(27) A matrix argument clause is either the matrix clause or an argument clause of a matrix argument clause.

The verb ‘said’ in (24b) heads a matrix argument clause, because it is the head verb of the matrix clause. The verb ‘ignored’ in (24b) also heads a matrix argument clause, because it is the head verb of an argument of the matrix clause. In contrast, the verb ‘attacked’ does not head a matrix argument clause, because the clause of which it is the head is a modifier of an NP in the matrix clause.

The zero-cost top-level predicate hypothesis in (28) now replaces the zero-cost matrix predicate assumption from (10a):

(28) The prediction of the top-level matrix argument predicate, V_0, is associated with no memory cost.

Given this hypothesis together with clause-based closure, there is no cost associated with predicting the embedded verb ‘ignored’ in (24b) after processing the complementizer ‘that’, because this prediction is for the top-level matrix argument predicate. Hence there is no complexity difference between (24a) and (24b).[19]

Finally, consider an example from a head-final language like Japanese, in which the clause structure is left-branching:

[19]The definition of a matrix argument clause is necessary in order to continue to account for the lack of a difference between subject-modified and object-modified relative clause structures as described in the previous section. If the top-level predicted predicate were associated with zero cost, whether or not it was a matrix argument predicate, then subject-modified RCs would be predicted to be more complex than object-modified RCs. In particular, a new clause has been initiated and all preferred dependents of the matrix clause have been satisfied at the point when the object-modified RC is initiated by the word ‘that’ in (23b). Hence this prediction would be associated with zero cost, and this structure would require less memory resources than a corresponding subject-modification, contrary to the available results. Accordingly, only argument clauses (and not modifier clauses) can be associated with zero memory cost.

(29) Inu-ga kanda hito-ga soojo-o mita
dog-nom bite man-nom girl-acc see
’The man who the dog bit sees the girl.’

As in English, as soon as the sentence is initiated, the matrix predicate is predicted. This prediction is satisfied when the verb *kanda* (‘bite’) is encountered. When the next word *hito-ga* (‘man-nom’) is processed, the initial clause is attached as a relative clause modifying this noun, and a new predicate is predicted, which becomes the current top-level matrix prediction. The prediction of the new predicate is associated with no memory cost, because this is the top-most matrix argument predicate at the point that it is predicted. (It also happens to be the prediction of the matrix predicate.) When this prediction is satisfied, this predicate could turn out to be the matrix predicate (as in (29)) or it might be another clausal dependent of a further predicate yet to occur.

2.6.7. Evidence that the integration and memory cost functions are non-linear in the limit

As initial hypotheses, it has been assumed that the integration and memory cost functions are linear with a zero intercept: $I(n) = K_{Int}*n$ and $M(n) = K_{Mem}*n$. These assumptions are probably oversimplifications. In particular, there is evidence that these cost functions are not linear in their limiting behavior. The best evidence for the non-linearity is with respect to the memory cost function. One prediction of a linear memory cost function is that it is steadily increasing, even in the limit. Thus, given a single syntactic prediction whose cost is determined by the function, there should be some distance such that the total cost associated with maintaining this prediction exceeds the processor’s available computational resources. For example, the prediction of the second object of a ditransitive verb can be delayed indefinitely by adding modifiers to the first object. If the memory cost function M(n) is linear, then there should be some number of modifiers (e.g. prepositional phrases or relative clauses) such that the addition of one further modifier causes processing breakdown. Intuitive judgments suggest that this prediction is not borne out:

(30) (a) John gave the beautiful pendant that he had seen in the jewelry store window to his mother.
(b) John gave the beautiful pendant that he had seen in the jewelry store window next to a watch with a diamond wristband to his mother.

(31) (a) Mary sent the friend who recommended the real estate agent who found the great apartment a present.
(b) Mary sent the friend who recommended the real estate agent who found the great apartment which was only $500 per month a present.

Once past a certain distance, there is no noticeable complexity difference between different distance predicted category realizations, and this point occurs well before memory overload. Thus there does not seem to be a minimum distance after which processing breaks down. In particular, (30b) and (31b) are more difficult to process than (30a) and (31b), but all of the examples are much more processable than a multiply center-embedded example like (3b), repeated here as (32):

(32) # The administrator who the intern who the nurse supervised had bothered lost the medical reports.

This evidence suggests that the memory cost function heads asymptotically toward a maximal complexity. One type of function which is monotonically increasing and asymptotically approaches a maximal complexity and which has been used to model recall effects in the working memory literature is the sigmoid function

$$M(n) = \frac{1}{1+e^{-An+B}} + C,$$

where $M(n)$ is the memory cost of an item, and n is the number of intervening elements. A sigmoid function can be thought of as a damped linear function, such that the slope heads toward zero as the number of intervening elements increases towards infinity. In order to make specific memory load predictions, the parameters A, B, and C in the general sigmoid memory cost function require values. Default values of $A = 1$ and $C = 0$ for the slope and intercept constants will be assumed. The positive value of the slope ($A = 1$) results in an increasing function, interpreted as the memory cost associated with a predicted category. The constant B is the n-coordinate at which the function is centered, where the cost increment stops increasing. A positive value is necessary, so that the initial cost is near 0 MUs. Evidence presented in Section 3.5 suggests that the complexity of an individual prediction does not increase much after processing two or three intervening new discourse referents. Hence the value for the parameter B will be initially assumed to be 1, which has the result that there is a small complexity difference for extractions over three or more new discourse referents. The graph for the sigmoidal version of the memory cost function is presented in Fig. 5.

For simplicity, the linear version of the function will be used when working through early examples, despite the fact that the sigmoidal version of the memory cost function is more accurate. All of the results reported in this paper are also consistent with the sigmoid function introduced here. Like the memory cost func-

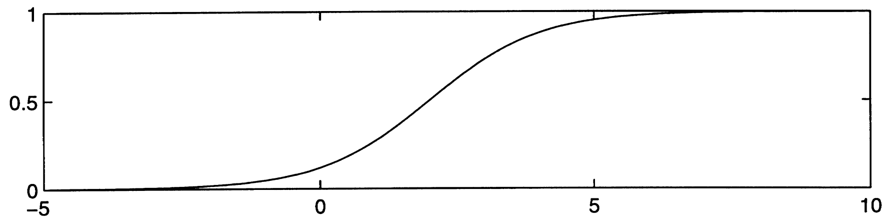

Fig. 5. The sigmoidal version of the memory cost function, $M(n) = \frac{1}{1+e^{-n+2}}$.

tion, the integration cost function will also be assumed to be nonlinear in the limit (cf. Church, 1980; Gibson et al., 1996a; who make similar claims).

3. Processing overload effects

In this section it is demonstrated that the SPLT accounts for a wide range of processing overload phenomena across numerous constructions in a number of languages. Before discussing more complex structures, the memory cost notation to be used is established using the simple subject- and object-extracted RC examples in (1).

3.1. Example memory complexity profiles: subject- and object-extracted relative clauses

The assumption of increased memory cost for longer incomplete syntactic predictions provides a simple explanation for many syntactic complexity contrasts. Before more complicated comparisons are discussed, it is first shown that the SPLT makes the appropriate prediction for the complexity difference between object- and subject-extracted RCs, repeated here in (33):

(33) (a) Object-extraction: The reporter who the senator attacked admitted the error.
(b) Subject-extraction: The reporter who attacked the senator admitted the error.

The memory complexity profiles for (33a) and (33b) within the SPLT are presented below in Tables 1 and 2, respectively. The left-hand column in each SPLT complexity profile contains all the syntactic predictions which are required during the parse of the sentence. The words in the sentence are given at the top of each of the other columns. An entry in a SPLT complexity matrix consists of the memory load associated with the prediction (the row) at the word position (the column) in terms of the function M(n), where n is the number of new discourse referents which have been processed since the prediction was first made. An asterisk is placed in a cell coordinate corresponding to the position at which a syntactic prediction is satisfied. These asterisks therefore correspond to some of the positions where structural integrations take place.[20] Dashes are placed in coordinates for syntactic predictions which are not active at the indexed word positions. The total memory cost at each word position is presented below each of the individual memory costs in terms of the function M(n). In reading the tables, recall that the prediction of the top-level verb is assumed to be associated with no memory cost, so no function M(n) is

[20]The locations at which predictions are being matched are not the only locations where structural integrations are taking place. Integrations also take place when an unpredicted element (e.g. a modifier) is attached into the structure.

Table 1
The word-by-word SPLT memory cost profile for the processing of the object-extracted relative clause sentence ‘The reporter who the senator attacked admitted the error’

Syntactic prediction	Input word								
	The	reporter	who	the	senator	attacked	admitted	the	error
Matrix verb	0	0	0	0	0	0	0	0	0
Matrix subject	M(0)	*	–	–	–	–	–	–	–
Matrix object NP	–	–	–	–	–	–	M(0)	M(0)	*
Embedded subject NP	–	–	–	M(0)	*	–	–	–	–
Embedded verb	–	–	M(0)	M(0)	M(1)	*	–	–	–
Wh-pronoun gap	–	–	M(0)	M(0)	M(1)	*	–	–	–
Total cost (MUs)	M(0)	0	2M(0)	3M(0)	2M(1)	0	M(0)	M(0)	0

associated with this prediction. First, consider the complexity profile for the object-extraction, presented in Table 1. The memory cost upon processing the first word of the sentence ‘the’ is M(0) MUs, corresponding to the prediction of the matrix subject (M(0) MUs) and verb (0 MUs). The memory cost associated with the next parse state, at the word ‘reporter’, is 0 MUs, corresponding to the prediction of the matrix verb. The memory cost associated with the next parse state, at ‘who’, is 2M(0) MUs, corresponding to:

1. 0 MUs for the prediction of the matrix verb
2. M(0) MUs for the prediction of the embedded verb
3. M(0) MUs for the prediction of a NP-gap position co-indexed with the wh-pronoun ‘who’.

The memory cost associated with the next parse state, at ‘the’, is 3M(0) MUs, corresponding to the same predictions as at the previous state, with one additional prediction: that of the embedded subject noun. At the following word ‘senator’, the memory cost for each of the two relative clause predictions (the embedded verb and the pronoun empty-category site) increases to M(1) MUs, because a new discourse referent corresponding to the NP ‘the senator’ now intervenes between the point at which the predictions were first made and their eventual realization. The total memory cost at this state is therefore 2M(1) MUs. After processing the verb ‘attacked’, the embedded verb and NP-gap predictions are satisfied, resulting in a 0 MU memory cost at this point. Next, the verb ‘admitted’ is attached into the current structure, satisfying the matrix verb prediction. An object NP is predicted by the verb ‘admitted’ at this point, causing a memory cost of M(0) MUs. Under the assumption that $M(n) = n$, the maximal memory complexity for processing the object-extracted RC structure is 2M(1) MUs, which occurs at the point of processing the embedded NP subject ‘senator’.

The memory cost profile for the subject-extraction, (33b), is presented in Table 2. The cost computations are similar to those for the object-extraction. The maximal memory complexity of the subject-extraction is less than that of the object-extraction, at 2M(0) MUs, which occurs at the point of processing the relative pronoun

Table 2
The word-by-word SPLT memory cost profile for the processing of the subject-extracted relative clause sentence ‘The reporter who attacked the senator admitted the error’

Syntactic prediction	Input word								
	The	reporter	who	attacked	the	senator	admitted	the	error
Matrix verb	0	0	0	0	0	0	0	0	0
Matrix subject	M(0)	*	–	–	–	–	–	–	–
Matrix object NP	–	–	–	–	–	–	M(0)	M(0)	*
Embedded verb	–	–	M(0)	*	–	–	–	–	–
Wh-pronoun gap	–	–	M(0)	*	–	–	–	–	–
Embedded object NP	–	–	–	M(0)	M(0)	*	–	–	–
Total cost (MUs)	M(0)	0	2M(0)	M(0)	M(0)	0	M(0)	M(0)	0

‘who’. Thus processing a subject-extracted relative clause is predicted to require less memory resources than processing an object-extracted relative clause.

3.2. The severe processing complexity of doubly center-embedded relative clauses

Let us now consider how the SPLT explains the unacceptability of more complex embeddings. First, let us see how the SPLT applies to doubly center-embedded relative clause constructions in English, as in (3b), repeated here in (34). The SPLT complexity profile for (34) is given in Table 3.

(34) # The administrator who the intern who the nurse supervised had bothered lost the medical reports.

The maximal memory complexity of processing this structure occurs at the most embedded noun ‘nurse’, at which point the memory complexity is 2M(2) + 2M(1) MUs corresponding to:

1. The prediction of the outer relative clause verb, which has been retained over the processing of two new discourse referents, resulting in a cost of M(2) MUs.
2. The prediction of an empty category position for the wh-pronoun of the outer relative clause, which has been retained over the processing of two new discourse referents, resulting in a cost of M(2) MUs.
3. The prediction of the inner relative clause verb, which has been retained over the processing of one new discourse referent, resulting in a cost of M(1) MUs.
4. The prediction of an empty category position for the wh-pronoun of the inner relative clause, which has been retained over the processing of one new discourse referent, resulting in a cost of M(1) MUs.

The memory cost also remains high over the processing of the next two words – 2M(3) MUs – before categories are encountered that reduce the memory load substantially. The point of maximal integration complexity (the point where reading times are predicted to be longest) is at the second main verb ‘bothered’, at which

Table 3
The word-by-word SPLT memory cost profile for the processing of the doubly-nested relative clause sentence 'The administrator who the intern who the nurse supervised had bothered lost the medical reports'

Syntactic prediction	Input word						
	The	adminis-trator	who	the	intern	who	the
Matrix verb	0	0	0	0	0	0	0
Matrix subject	M(0)	*	–	–	–	–	–
RC_1 verb	–	–	M(0)	M(0)	M(1)	M(1)	M(1)
RC_1 wh-pronoun gap	–	–	M(0)	M(0)	M(1)	M(1)	M(1)
RC_1 subject	–	–	–	M(0)	*	–	–
RC_2 verb	–	–	–	–	–	M(0)	M(0)
RC_2 wh-pronoun gap	–	–	–	–	–	M(0)	M(0)
RC_2 subject	–	–	–	–	–	–	M(0)
Total cost	M(0)	0	2M(0)	3M(0)	2M(1)	2M(1) + 2M(0)	2M(1) + 3M(0)

	Input word							
	nurse	super-vised	had	bothered	lost	the	medical	reports
Matrix verb	0	0	0	0	0	0	0	0
Matrix NP object	–	–	–	–	M(0)	M(0)	M(0)	*
RC_1 verb	M(2)	M(3)	M(3)	*	–	–	–	–
RC_1 wh-pronoun gap	M(2)	M(3)	M(3)	*	–	–	–	–
RC_2 verb	M(1)	*	–	–	–	–	–	–
RC_2 wh-pronoun gap	M(1)	*	–	–	–	–	–	–
Total cost	2M(2) + 2M(1)	2M(3)	2M(3)	0	M(0)	M(0)	M(0)	0

point there are two long integrations to be performed: assigning a thematic role from 'bothered' to 'the intern', and linking the empty category patient of 'bothered' to the first instance of 'who'. This prediction fits the available on-line experimental data for the processing of such structures (Kurtzman, pers. commun., reported in Gibson, 1991).

If the most embedded subject NP of a doubly-nested relative clause structure is an indexical pronoun as in (35a), so that it is in the current discourse, then the memory cost at the embedded subject would be only 2M(1) + 2M(0) MUs, corresponding to the same predicted categories as discussed above, but with one fewer new discourse referent crossed for each of the predictions:

(35) (a) Indexical pronoun: The student who the professor who I collaborated with had advised copied the article.
(b) Full NP: The student who the professor who the scientist collaborated with had advised copied the article.

The memory cost for the indexical pronoun version of the doubly-nested relative clause structure reaches a maximal complexity at the most embedded verb, at which point the complexity is 2M(2) MUs, but this is still less than the memory cost for any state in processing the non-indexical pronoun nested structures, so that the contrast between the two is accounted for.

Since sentences like (34) are unacceptably difficult to process for most English speakers, it is hypothesized that the memory load at the most embedded noun is more than the capacity of most people's linguistic working memory. Interestingly, a subject's realization that such a sentence structure is unacceptable does not occur at the point of maximal memory complexity: It seems to come at the second verb, when the structural integration cost would be highest. In fact, if the second verb is omitted as in (36a) and (36c), the resulting structures are more acceptable, even though they are actually ungrammatical (Frazier, 1985; Christiansen, 1997; Gibson and Thomas, 1997b):

(36) (a) * The administrator who the intern who the nurse supervised lost the medical reports.
(b) # The patient who the nurse who the clinic had hired admitted met Jack.
(c) * The patient who the nurse who the clinic had hired met Jack.

The relative acceptability of ungrammatical sentence structures like (36a) and (36c) indicates that an overload of working memory does not cause an error signal to be sent to the human parser. Rather, the parser does what it can with the memory resources that it has, either forgetting something or attempting to collapse together some structures requiring memory resources. If the sentence is grammatical, then the error signal appears later downstream when too many role assigners are present for

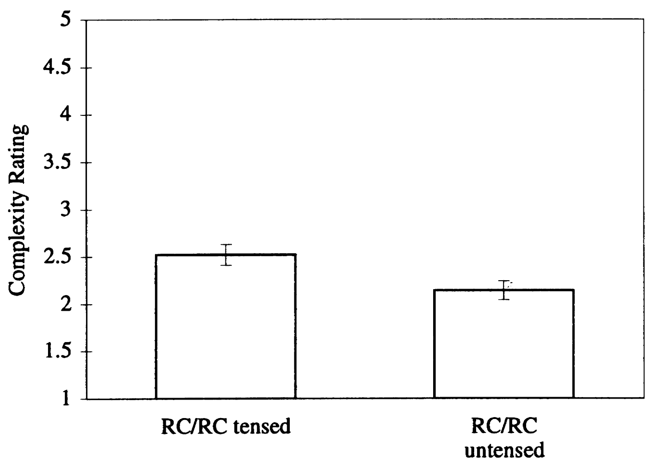

Fig. 6. Complexity ratings for doubly nested clauses whose most embedded clause is tensed or untensed (from Gibson and Thomas, 1997a). The scale that participants used went from 1 (not complex: easy to understand) to 5 (very complex: hard to understand).

the material in working memory. If the sentence has the form of (36a) (which is not grammatical) then the parser never receives such an error message, and the sentence is processed as if it were grammatical. See Gibson and Thomas (1997b) for more on this issue.

3.3. The complexity of embeddings involving tensed clauses

In a complexity rating questionnaire experiment, Gibson and Thomas (1997a) found that participants had much less difficulty with nested clausal structures when the most embedded clausal modifier was untensed, as in (37b), than when it was tensed, as in (37a) (see Fig. 6):

(37) (a) RC/RC, tensed: The school board which the teachers who were neglecting the students had angered troubled the superintendent.
(b) RC/RC, untensed: The school board which the teachers neglecting the students had angered troubled the superintendent.

This result follows from the SPLT under the assumption that tensed verbs incur a larger memory cost than untensed verbs, because of the additional discourse requirements of tense. In particular, the maximal memory load of a structure like (37a) is 2M(3) MUs, which occurs at the point of processing the NP ‘the students’. There are three predicted categories at this point: the matrix verb (no cost), the outer relative clause verb (M(3) MUs, corresponding to following three new discourse referents that have been processed since the prediction was made: ‘the teachers’, ‘were neglecting’ and ‘the students’) and an empty category to be associated with the relative clause pronoun ‘which’ (also M(3) MUs).

The maximal memory cost in processing a doubly nested structure whose most embedded clause is untensed, as in (37b), occurs at the same place as in (37a), but the cost here is lower – only 2M(2) MUs – because only two new discourse referents have been processed since the two memory-cost inducing syntactic predictions were made. Crucially, the most embedded VP ‘neglecting’ is untensed so that it does not indicate a new discourse referent, and so processing this VP does not cause as large a memory cost increment as in the structure in which the most embedded clause is tensed.

3.4. The syntactic complexity of embeddings of sentential complements and relative clauses

Now let us consider the complexity difference between the two embeddings of a sentential complement and a relative clause:

(38) (a) Sentential complement, then relative clause (SC/RC):
The fact that the employee who the manager hired stole office supplies worried the executive.
(b) Relative clause, then sentential complement (RC/SC) :

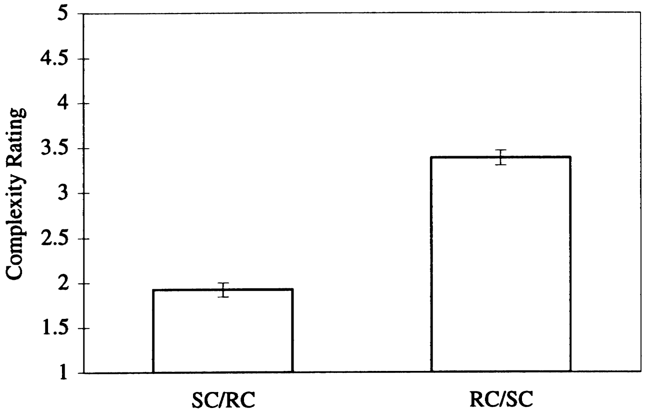

Fig. 7. Complexity ratings for two different nesting orderings of relative clauses and sentential complements (from Gibson and Thomas, 1997a). The scale that participants used went from 1 (not complex: easy to understand) to 5 (very complex: hard to understand).

The executive who the fact that the employee stole office supplies worried hired the manager.

In a complexity rating experiment, Gibson and Thomas (1997a) observed that participants found the embedding consisting of a relative clause within a sentential complement, as in (38a), much less complex than the reverse embedding, as in (38b) (see Fig. 7). According to the SPLT, the difference between the two constructions rests on the fact that a predicted sentential complement involves only one long incomplete category prediction – that of a verb – whereas a predicted relative clause involves two long incomplete category predictions – that of a verb and an empty category position for the wh-pronoun. For example, upon processing the complementizer 'that' in (38a) two categories are required to complete the sentential complement: a noun and a verb. The noun requirement is satisfied very soon, when the noun 'employee' is processed. The requirement for the verb is not met until much later downstream when the verb 'stole' is encountered, after the embedded relative clause has been processed. Thus there is only one long unresolved category prediction when the sentential complement is the outer of the two embedded clauses. In contrast, when the relative clause is the outer of the two embedded clauses, as in (38b), the predictions of both the verb and the empty category position for the wh-pronoun must be kept in memory during the processing of the embedded sentential complement. The two predictions are resolved far downstream, upon processing the verb 'worried'.

To see the details of how the theory applies to these kinds of examples, the SPLT complexity profiles for the SC/RC and RC/SC constructions are given in Tables 4 and 5 respectively.

Given the assumption that M(n) = n, the maximal memory complexity of the SC/

Table 4
The word-by-word SPLT memory cost profile for the processing of the SC/RC sentence structure 'The fact that the employee who the manager hired stole office supplies worried the executive'

Syntactic prediction	Input word						
	The	fact	that	the	employee	who	the
Matrix verb	0	0	0	0	0	0	0
Matrix subject	M(0)	*	–	–	–	–	–
Verb complement of 'that'	–	–	M(0)	M(0)	M(1)	M(1)	M(1)
Embedded subject	–	–	M(0)	M(0)	*	–	–
RC verb	–	–	–	–	–	M(0)	M(0)
Wh-pronoun gap	–	–	–	–	–	M(0)	M(0)
RC subject	–	–	–	–	–	–	M(0)
Total cost	M(0)	0	2M(0)	2M(0)	M(1)	M(1) + 2M(0)	M(1) + 3M(0)

Syntactic prediction	Input word							
	manager	hired	stole	office	supplies	worried	the	executive
Matrix verb	0	0	0	0	0	0	0	0
Verb complement of 'that'	M(2)	M(3)	*	–	–	–	–	–
NP object of 'stole'	–	–	M(0)	M(0)	*	–	–	–
RC verb	M(1)	*	–	–	–	–	–	–
Wh-pronoun gap	M(1)	*	–	–	–	–	–	–
RC subject	*	–	–	–	–	–	–	–
Matrix NP object	–	–	–	–	–	–	M(0)	*
Total cost	M(2) + 2M(1)	M(3)	M(0)	M(0)	0	M(0)	M(0)	0

RC construction occurs upon processing the word 'manager', at which point the memory cost is M(2) + 2M(1) MUs = 4 MUs corresponding to:

1. 0 MUs for the prediction of the matrix verb
2. M(2) MUs for the prediction of the verb in sentential complement of 'the fact that...'
3. M(1) MUs for the prediction of the verb in the relative clause, which was predicted after processing 'the employee who...'
4. M(1) MUs for the prediction of a position co-indexed with the wh-pronoun 'who' in the relative clause.

In contrast, the maximal memory complexity of the RC/SC construction is 2M(4) MUs = 8 MUs, which occurs upon processing the word 'supplies'. At this point, the following unresolved predictions are present in the structure for the input:

1. 0 MUs for the prediction of the matrix verb
2. M(4) MUs for the prediction of the verb in the relative clause, which was predicted after processing 'the executive who...'
3. M(4) MUs for the prediction of a position co-indexed with the wh-pronoun 'who' in the relative clause.

Table 5
The word-by-word SPLT memory cost profile for the processing of the RC/SC sentence structure ‘The executive who the fact that the employee stole office supplies worried hired the manager’

Syntactic prediction	Input word						
	The	executive	who	the	fact	that	the
Matrix verb	0	0	0	0	0	0	0
Matrix subject	M(0)	*	–	–	–	–	–
RC verb	–	–	M(0)	M(0)	M(1)	M(1)	M(1)
Wh-pronoun gap	–	–	M(0)	M(0)	M(1)	M(1)	M(1)
RC subject	–	–	–	M(0)	*	–	–
Verb complement of ‘that’	–	–	–	–	–	M(0)	M(0)
SC subject	–	–	–	–	–	M(0)	M(0)
Total cost	M(0)	0	2M(0)	3M(0)	2M(1)	2M(1) + 2M(0)	2M(1) + 2M(0)

	Input word							
	employee	stole	office	supplies	worried	hired	the	manager
Matrix verb	0	0	0	0	0	0	0	0
Matrix NP object	–	–	–	–	–	M(0)	M(0)	*
RC verb	M(2)	M(3)	M(3)	M(4)	*	–	–	–
Wh-pronoun gap	M(2)	M(3)	M(3)	M(4)	*	–	–	–
Verb complement of ‘that’	M(1)	*	–	–	–	–	–	–
SC subject	*	–	–	–	–	–	–	–
NP object of ‘stole’	–	M(0)	M(0)	*	–	–	–	–
Total cost	2M(2) + M(1)	2M(3) + M(0)	2M(3) + M(0)	2M(4)	0	M(0)	M(0)	0

Since $2M(4) > M(2) + 2M(1)$ for many monotonically increasing functions M(n) (including linear functions) the SPLT correctly predicts a memory complexity difference between the two constructions, as desired. Thus the complexity difference between the SC/RC and RC/SC structures under the SPLT is not dependent on the specific assumptions regarding the memory cost function, other than the assumption that costs increase over distance. The function that has been assumed here increments cost in terms of new discourse referents, but a function which increments cost in terms of words or morphemes also gives the same results for the examples that have been discussed here.

The difference between SC/RC and RC/SC structures is also present in the integration complexities. In particular, the maximum integration complexity for the SC/RC structure is I(6) EUs at ‘worried’, whereas the maximum integration complexity for the RC/SC structure is I(4) + I(5) EUs, also at ‘worried’. See Section 6 for more discussion.

3.4.1. Predictions of the discourse-based memory cost function as applied to RC/SC constructions

Just as doubly nested relative clause structures are easier to process when there is an indexical pronoun in the most embedded subject position, RC/SC structures should also be much improved if the NPs in the embedded clause are pronouns in the discourse. Introspective judgments suggest that this prediction is correct:

(39) (a) # The intern who the information that the doctor hated the patient had bothered supervised the nurse.
(b) # The intern who the information that Joe hated Sue had bothered supervised the nurse.
(c) The intern who the information that you hated me had bothered supervised the nurse.
(d) The intern who the information that you hated him had bothered supervised the nurse.
(e) The race organizer who the possibility that it would rain had worried was relieved to see the beautiful day.

(39a) and (39b), which include NPs new to the discourse in the embedded sentential complement clause, are intuitively more difficult to process than (39c) and (39d), which include pronouns whose referents are already present in the current discourse. Similarly, (39e) is also processable, because it contains only one new referent in its most embedded clause: the event ‘it would rain’.

3.5. Processing overload generalization: two long unresolved predicted categories

A generalization that presents itself from the SPLT analysis is that structures which include a parse state with at least two long unresolved predicted categories in addition to the top-level verb are unacceptable, and those without such a state are usually acceptable. Under the memory cost function assumed here, a ‘long’ unresolved prediction is one spanning at least three intervening new discourse referents. Thus, sentences whose parses include parse states whose memory cost is 2M(3) MUs or greater are generally not acceptable, while sentences whose parses do not include such a costly parse state are generally acceptable. A reasonable conclusion from this analysis is that linguistic working memory capacity is somewhere around 2M(3) MUs or just below. This generalization applies to the classes of examples that have been explored thus far as follows:

1. Doubly center-embedded relative clause structures whose most embedded relative clause is an object extraction, such as (34), include parse states whose memory load is 2M(3) MUs, and an additional state whose memory load is 2M(2) + 2M(1) MUs. These structures are correspondingly unacceptable.
2. A construction consisting of a sentential complement embedded within a relative clause, such as (38b), includes a parse state whose memory load is 2M(4) MUs. This construction is therefore unacceptable.
3. The maximal complexity of a construction consisting of a relative clause embedded within a sentential complement, such as (38a), is only M(2) +

2M(1) MUs, which is less than 2M(3) MUs. Hence this construction is acceptable.

In the remainder of this section, it is demonstrated that this generalization holds across a wide range of constructions in English. See Cowper (1976) and Gibson (1991) for variations of some of the constructions discussed thus far that are also handled by the SPLT.

3.5.1. A long-distance dependency across a sentential complement/relative clause construction

The SPLT predicts that constructions with one long unresolved category prediction will become much harder to process if a second long unresolved category is also required. For example, a construction with one long unresolved category prediction should become much more complex if one of its later constituents is fronted, or if the whole construction is embedded within a pre-verbal clausal modifier. As was demonstrated in Section 3.4 there is only at most one long incomplete predicted category in the processing of a sentential complement/relative clause construction: that of the embedded verb. Hence adding a single long predicted category to a sentential complement/relative clause construction should result in an unprocessable construction. This prediction is verified by the unacceptability of (40b) (cf. a similar observation in a different framework in Gibson, 1991):

(40) (a) Sentential complement, then relative clause:
The fact that the employee who the manager hired stole office supplies worried the executive.
(b) Wh-question, sentential complement, then relative clause:
Who did the fact that the employee who the manager hired stole office supplies worry?

At the point of processing ‘hired’ in (40a), the memory complexity is M(3) MUs, corresponding to the predictions of the embedded verb following ‘fact’ and the matrix verb. The memory complexity at the same point in (40b) is M(4) + M(3) MUs, corresponding to the same predictions as in (40a) with one addition: the prediction of an NP empty category corresponding to the sentence-initial wh-phrase ‘who’. This memory complexity is more than the proposed memory capacity, thus accounting for the unacceptability of this structure.

3.5.2. Complex center-embeddings consisting of only two clauses

Unlike other theories, the SPLT predicts that sentences with only two clauses can be highly complex. For example, the SPLT predicts that lengthening the subject of a relative clause in a wh-question will lead to processing overload. This prediction appears to be correct, as demonstrated by the complexity increase over the examples in (41):

(41) (a) Who did the artist who the woman met at the party talk to?

(b) ?# Who did the artist who the woman from the college met at the party talk to?
(c) # Who did the artist who the woman from the college in the suburbs met at the party talk to?

The maximal memory complexity of (41a) is M(2) + 2M(1) MUs, which occurs upon processing the most embedded subject noun 'woman'. Hence this structure should be roughly comparable in perceived complexity to an SC/RC structure. Adding the PP 'from the college' following this subject as in (41b) pushes the maximal complexity up to M(3) + 2M(2) MUs at 'college'. Adding an additional PP pushes the maximal complexity up to M(4) + 2M(3) MUs following the third PP, as in (41c), causing a clear case of unacceptability.

As a second kind of example demonstrating that sentences with only two clauses can be unprocessable due to memory overload, consider once again the RC/RC and RC/SC structures, as in (42).

(42) (a) RC/RC: # The administrator who the intern who was supervising the nurse had bothered lost the medical reports.
(b) RC/SC: # The intern who the information that the administrator lost the medical reports had bothered supervised the nurse.

According to the SPLT, the complexity of these structures does not depend on the fact that they contain three clauses, so variations of these structures in which the most embedded clause is nominalized should also be highly complex. Intuitive evidence suggests that the nominalization examples are also very difficult to understand:

(43) (a) RC/PP: # The administrator who the intern responsible for the supervision of the nurse had bothered lost the medical reports.
(b) RC/PP: # The intern who the information about the administrator's loss of the medical reports had bothered supervised the nurse.

The maximal memory complexities of (43a) and (43b) are 2M(3) MUs and 2M(4) MUs respectively, so both should be quite difficult to process. Although these sentences are complex, intuitions suggest that they may not be as difficult as other structures examined thus far whose maximal complexity are 2M(3) or 2M(4) MUs. This complexity difference may be due to the possibility that traversing intervening tensed clauses may be more costly than traversing new NP object referents. In these examples, none of the intervening discourse referents is a tensed clause, unlike other complex structures thus far.

3.5.3. Extractions across conjoined NPs: evidence for a more complex memory and integration cost function

An interesting question which is raised by the formulation of the theory in terms of new discourse referents is whether the conjuncts in a conjoined NP count as independent discourse referents for memory and integration cost-counting purposes.[21] For example, consider the comparison between (44a) and (44b):

(44) (a) # Who did the artist who the photographer with the bodyguard for the actor met at the party talk to?
(b) Who did the artist who the photographer and the bodyguard and the actor met at the party talk to?

(44a) seems intuitively more complex than (44b). This complexity difference might follow from the fact that (44a) includes predicative relationships marked by prepositions among the intervening NPs, which add cost to the memory cost predictions (see Section 2.6.4). Furthermore, the discourse structure for ‘the photographer with the bodyguard for the actor’ is quite complicated, especially because the use of the modifying PPs might presuppose the existence of other contrastive elements in the discourse (cf. Crain and Steedman, 1985; Altmann and Steedman, 1988; Ni et al., 1996). In contrast, the discourse structure for ‘the photographer and the bodyguard and the actor’ is simple, because it does not presuppose the existence of other objects in the discourse.

It is also possible that the conjuncts in a conjoined NP do not count independently towards memory and integration cost complexity. It might be possible to motivate this in discourse theory if it is difficult to refer to a conjunct in a conjoined NP using a referential expression such as a pronoun.

3.5.4. Extractions from prepositional phrases

Pickering and Barry (1991) observed that sentences involving multiply nested stranded prepositions are highly complex:

(45) # This is the saucer which Mary put the cup which she poured the milk into on.

The unacceptability of this structure is predicted by the SPLT, because its maximal memory complexity reaches M(5) + M(3) + M(2) MUs at the point of processing the NP ‘the milk’. Three heads are required to finish the sentence grammatically at this point: (1) an NP-gap corresponding to the first instance of ‘which’ (M(5) MUs), (2) a destination PP/adverbial argument for the verb ‘put’ (M(3) MUs) and (3) an NP-gap corresponding to the second instance of ‘which’ (M(2) MUs).

Pickering and Barry (1991) also observed that structurally similar sentences like (46) are not complex, despite the fact that the relative clauses are nested within the PP-fillers and their corresponding empty-category positions:

(46) This is the saucer [$_{PP_i}$ on which] Mary put the cup [$_{PP_j}$ into which] she poured the milk t_i t_j.

If the parser had to wait until after the direct objects of the verbs to posit the empty

[21] Thanks to Gerard Kempen for raising this question.

categories associated with the PP fillers, then the maximal memory cost of (46) would be M(4) + M(2) + M(1) MUs at ‘poured’, a similar memory complexity to that of (45). But (46) is much less complex than (45), contrary to prediction. Based on this observation, Pickering and Barry argued that the parser makes direct associations between fillers and their role-assigning verbal heads, without waiting for their default positions (see Traxler and Pickering (1996) for reading time evidence that this is the case). In a syntactic framework that includes empty-categories mediating long-distance dependencies, this assumption is equivalent to assuming that empty categories can be posited at their licensing heads without waiting for their default positions after the direct objects (Gibson and Hickok, 1993; Crocker, 1994). Under such a framework, the empty-category associated with ‘on which’ is attached to the VP headed by the verb ‘put’ at the point of processing ‘put’, and the empty-category associated with ‘on which’ is attached to the VP headed by the verb ‘poured’ at the point of processing ‘poured’. As a result, the maximal complexity of (46) is only M(1) MUs, and the contrast between (45) and (46) is accounted for.

Other preposition-stranding examples discussed by Pickering and Barry (1991) also receive straight-forward analyses under the SPLT. For example, consider the lack of a large complexity contrast between (47a) and (47b):

(47) (a) Which box did you put the wedding cake in?
(b) In which box did you put the wedding cake?

Pickering and Barry noted that a large contrast in complexity appears when the NP object of ‘put’ is lengthened:

(48) (a) # Which box did you put the very large and beautifully decorated wedding cake bought from the expensive bakery in?
(b) In which box did you put the very large and beautifully decorated wedding cake bought from the expensive bakery?

Pickering and Barry suggested that the complexity difference between (48a) and (48b) is due to a difference in the distances between fillers and role-assigners in the two examples. The SPLT provides a formalization of this idea. In particular, the maximal memory complexity of (48a) is M(4) + M(3) MUs, occurring at the word ‘bakery’, corresponding to:

1. The prediction of an NP position for ‘which box’, a prediction that has been held in memory across the processing of ‘put’, ‘the...wedding cake’, ‘bought’ and ‘the expensive bakery’, leading to a cost of M(4) MUs.
2. The prediction of a PP argument of ‘put’, a prediction that has been held in memory across the processing of ‘the... wedding cake’, ‘bought’ and ‘the expensive bakery’, leading to a cost of M(3) MUs.

In contrast, the maximal memory complexity of (48b) is much less, at M(1) MUs at the point of processing ‘box’, and the difference is explained.

The SPLT makes other interesting predictions regarding preposition stranding.

One such prediction is that lengthening the subject NP in (47a) should result in less memory complexity than lengthening the object NP, as in (48a). This prediction is made because part of the severe complexity of (48a) is due to the long distance between the verb ‘put’ and its required PP argument. This distance is greatly shortened if the subject NP rather than the object NP is lengthened. The prediction of lower complexity is borne out:

(49) Which box did the very tall and beautifully dressed bride who was married in the church down the hill put the cake in?

The maximal complexity of (49) occurs at the point of processing ‘cake’ at which point the memory cost is M(6) + M(1) MUs, which is less than M(4) + M(3) MUs under the non-linear memory cost function. A direct test of this claim was carried out by Gibson and Warren (1998b) using a questionnaire acceptability rating task on sentence structures like those in (50):

(50) (a) Short subject NP, no intervening object:
The usher closed the theater into which the girl sneaked the boy who ate the popcorn last Wednesday before they went to the dance.
(b) Long subject NP, short object NP:
The usher closed the theater which the boy who ate the popcorn sneaked the girl into last Wednesday before they went to the dance.
(c) Short subject NP, long object NP:
The usher closed the theater which the girl sneaked the boy who ate the popcorn into last Wednesday before they went to the dance.

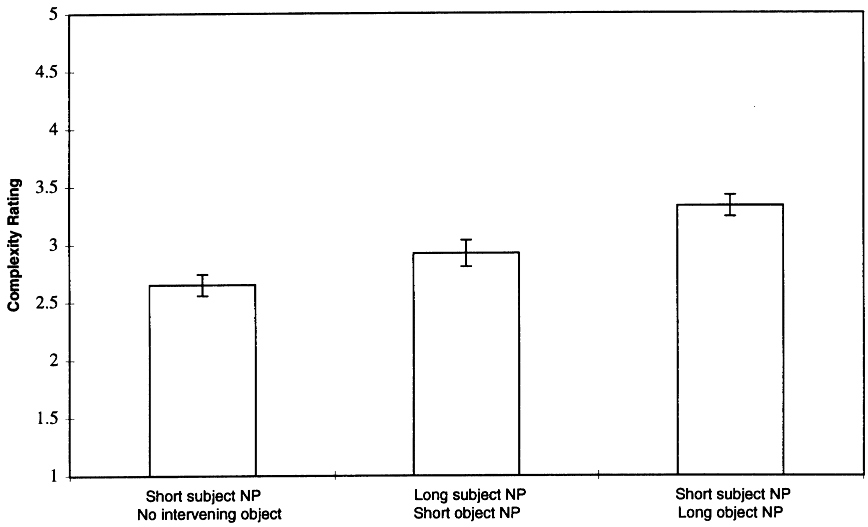

Fig. 8. Complexity ratings for relative clause structures whose filler is a prepositional phrase or is extracted out of a prepositional phrase (from Gibson and Warren, 1998a). The scale that participants used went from 1 (not complex: easy to understand) to 5 (very complex: hard to understand).

The structures were designed using double object verbs like ‘sneak’ which could plausibly take a human first object, so that the same lexical items could be used in all three conditions. The short distance extraction items were rated as least complex of the three, followed by the long-distance extracted long-subject items, with the long-distance extracted long-object items the most complex of the three, as predicted by the SPLT (see Fig. 8).

3.6. Cross-linguistic support: Japanese linguistic complexity

A number of syntactic complexity results from the processing of head-final languages like Japanese are explained within the SPLT. One interesting complexity effect that the SPLT accounts for is that it is possible to stack a large number of sentence initial NPs without causing severe processing difficulty, as in (51) (Lewis, 1993):

(51) (a) [Taroo-ga Hajime-ni [Akira-ga Hanako-ni Sigeru-o syookai sita] to itta]
Taroo-nom Hajime-dat Akira-nom Hanako-dat Shigeru-acc introduced that said
’Taroo said to Hajime that Akira introduced Shigeru to Hanako.’

(51) is processable in spite of the fact that it contains five initial NPs. The lack of severe complexity for this kind of example is predicted by the SPLT. In particular, the maximal memory complexity is only 2M(2) MUs, which occurs at the point of processing the fifth NP ‘Sigeru-o’, corresponding to:

1. 0 MUs for the prediction of the matrix verb
2. M(2) MUs for the prediction of the embedded verb
3. M(2) MUs for the prediction of the complementizer ‘to’.

This maximal complexity is less than 2M(3) MUs, and the structure is therefore processable.

In contrast, sentence structures like (52) contain only three sentence-initial NPs but are perceived as more complex than (51):

(52) ? # Taroo-wa [Hajime-ga [Akira-ga naita to] omotteiru to] itta
Taroo-topic Hajime-nom Akira-nom cried that thinks that said
‘Taroo said that Hajime thinks that Akira cried.’

The maximal complexity of (52) according to the SPLT is 2M(2) + M(1) MUs, which occurs upon processing the verb ‘naita’ (‘cried’). At this point, the following category predictions are associated with memory cost:

1. 0 MUs for the prediction of the matrix verb
2. M(2) MUs for the prediction of the first embedded verb
3. M(2) MUs for the prediction of the outer complementizer

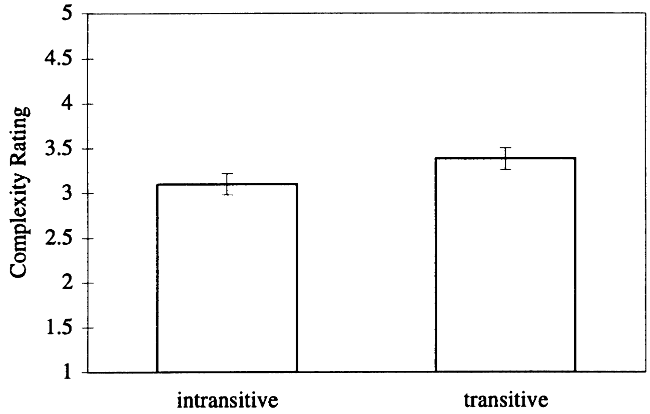

Fig. 9. Complexity ratings for Japanese nested structures containing either one NP in the most embedded clause (intransitive) or 2 NPs in the most embedded clause (transitive) (from Babyonyshev and Gibson, 1997). The scale that participants used went from 1 (not complex: easy to understand) to 5 (very complex: hard to understand).

4. M(1) MUs for the prediction of the most embedded complementizer.

The maximal cost is therefore more than that associated with processing a structure like (51), and the contrast is explained.

The contrast between examples like (51) and (52) argues for a memory complexity theory based on predicted categories rather than one based on predicted dependencies. A dependency-based complexity metric predicts that (52) should be easier to process than (51), because there are maximally only three incomplete dependencies in (52) (one for each initial NP), but there are maximally five such incomplete dependencies in (51). Furthermore the dependencies are incomplete for longer in (51). On the other hand, under the predicted category theory, the lower complexity of (51) relative to (52) is explained, because there are only at most three predicted syntactic heads in the processing of (51) (two verbs and one complementizer), whereas there are five predicted syntactic heads in the processing of (52) (three verbs and two complementizers). Hence a complexity theory based on predicted categories is preferred.

Babyonyshev and Gibson (1997) provide experimental results from acceptability judgement questionnaires that demonstrate that adding an extra nominal argument (an object NP) to the most embedded clause in a structure like (52) and (53a) makes the structures more complex, as shown in Fig. 9 (cf. Babyonyshev and Gibson, 1995). The examples that Babyonyshev and Gibson investigated are of the form in (53):

(53) (a) Obasan-wa [$_{S'}$ [$_{S}$ bebiisitaa-ga [$_{S'}$ [$_{S}$ imooto-ga naita] to] itta] to] omotteiru
aunt-top babysitter-nom younger-sister-nom cried that said that thinks
'My aunt thinks that the babysitter said that my younger sister cried'

(b) Obasan-wa [$_{S'}$ [$_S$ bebiisitaa-ga [$_{S'}$ [$_S$ ani-ga imooto-o ijimeta] to] itta] to] omotteiru
aunt-top babysitter-nom older-brother-nom younger-sister-acc bullied that said that thinks
'My aunt thinks that the babysitter said that my older brother bullied my younger sister.'

The maximal complexity for (53a) is 2M(2) + M(1) MUs, as described above for the similar structure in (52). On the other hand, the maximal complexity for (53b) is 2M(3) + M(2) MUs, which occurs at the point of processing the verb 'ijimeta' ('bullied'), and corresponds to the same category predictions as for (53a), but carried for one additional NP, the embedded direct object of the transitive verb. The complexity breaks down into the following components:

1. 0 MUs for the prediction of the matrix verb
2. M(3) MUs for the prediction of the first embedded verb
3. M(3) MUs for the prediction of the outer complementizer
4. M(2) MUs for the prediction of the most embedded complementizer.

The maximal cost is therefore more than that associated with processing a structure like (53a), and the contrast is explained. Furthermore, the lack of acceptability of (53b) is potentially explained, since results from English have shown that a memory cost of 2M(3) + M(2) MUs causes a processing overload effect.

Babyonyshev and Gibson (1997) also demonstrate that the SPLT explains the contrast between the two embeddings of a relative clause and a sentential complement, the analysis of which is similar to the analysis of the parallel English constructions, as described in Section 3.4. See Babyonyshev and Gibson (1997) for more details about how this works out in Japanese.

3.7. Center-embedded versus cross-serial dependencies

In a seminal study, Bach et al. (1986) demonstrated that structures involving cross-serial dependencies (as in (54a)) are easier to process than structures involving center-embedded dependencies (as in (54b)):

(54) (a) Cross − serial (Dutch): ... NP_1 NP_2 NP_3 NP_4 V_1 V_2 V_3
Jeanine heeft de mannen Hans de paarden helpen leren voeren.
Joanna has the men Hans the horses helped teach feed.
'Joanna helped the men teach Hans to feed the horses.'
(b) Center − embedded (German): ... NP_1 NP_2 NP_3 NP_4 V_3 V_2 V_1
Johanna hat den Männern Hans die Pferde füttern lehren helfen.
Joanna has the men Hans the horses feed teach helped.
'Joanna helped the men teach Hans to feed the horses.'

The construction involved in making this comparison contains a verb like 'help' or 'teach', which subcategorizes for an infinitival sentence. In both German and

Dutch, the verb occurs in the final position of embedded clauses. The two languages vary with respect to the order of multiply-embedded verbs in the help/teach class: In German the verbs are nested as in (54b), whereas in Dutch the verbs are arranged cross-serially as in (54a). Using acceptability ratings, Bach et al. demonstrated that, on controlled sets of examples across the two languages, the Dutch cross-serial dependency construction is easier to process than the German center-embedded construction.[22] From their data, Bach et al. argued against stack-based parsing algorithms, but they did not provide a theory of linguistic complexity that explains the complexity difference (cf. Joshi, 1990; Rambow and Joshi; 1994; who provide part of such a theory; see also Christiansen and Chater (1998) for a connectionist account of these phenomena).

The SPLT explains the complexity difference directly: the categories that are predicted first are associated with the most memory cost, so satisfying these first results in lower complexity for cross-serial dependencies than for nested dependencies. To see this in detail, one should first observe that the syntactic complexity at the point of processing the most embedded object NP ‘the horses’ (‘de paarden’ in Dutch, ‘die Pferde’ in German) is M(3) + M(2) + M(1) MUs in both languages, corresponding to:[23]

1. 0 MUs for the prediction of the matrix verb
2. M(3) MUs for the prediction of an NP position for the fronted NP ‘Joanna’ (‘Jeanine’, ‘Johanna’)[24]
3. M(2) MUs for the prediction of the first embedded verb
4. M(1) MUs for the prediction of the most embedded verb.

The next element to occur in the Dutch structure is the matrix verb, which licenses a position for the fronted NP, thus satisfying that prediction. The memory costs for the other two predictions rise to M(3) and M(2) MUs at this point for a complexity of M(3) + M(2) MUs. The memory cost decreases from this point on in the Dutch structure, so that the maximal memory complexity for processing the Dutch cross-serial structure is M(3) + M(2) + M(1) MUs, which occurs at the most embedded NP.

Now consider the German structure, whose memory cost is M(3) + M(2) + M(1) MUs after processing the NP ‘die Pferde’ (‘the horse’). The next element to occur is the most embedded verb, satisfying the least expensive current prediction. The memory costs for the other two rising-cost predictions increase to M(4) and M(3) MUs, resulting in a total memory cost of M(4) + M(3) MUs, the maximal memory

[22] Hoeksema (1981) made a similar observation based on intuitive judgments with respect to a comparison between Dutch and Frisian, which has center-embedded dependencies like German.

[23] The memory complexity presented here is for the target multiply embedded structure. The Dutch sentence is locally ambiguous at this point among a number of possible interpretations, including the target multiply embedded structure and a structure with one less embedding whose most embedded verb is ditransitive, taking three of the NPs as its arguments. The less embedded structure is preferred (Dickey and Vonk, 1997), arguably because of its lower SPLT memory complexity. Thus reanalysis may be required to arrive at the target structure in Dutch.

[24] Dutch and German are verb-second languages, which allow fronting of an XP, as long as the second position is filled with the tensed verb for the matrix clause.

complexity of the parse of this structure, and more than the maximal complexity for processing the Dutch structure. Thus the contrast between the Dutch and German structures is explained by the SPLT. Furthermore, the unacceptability of the German structure is also explained, because its maximal complexity is M(4) + M(3) MUs, which has been demonstrated in English to be too much for the limited capacity of working memory. The borderline acceptability/unacceptability of the Dutch structure is also explained, because its maximal complexity is high, as M(3) + M(2) + M(1) MUs.

4. Length/heaviness effects

It has long been observed that linguistic constructions are easier to comprehend if the longer or 'heavier' items appear later in a sentence, given an option (Behaghel, 1910; Bever, 1970; Quirk et al., 1972; Hawkins, 1990, 1994; Wasow, 1997). For example, if a verb is followed by two arguments or modifiers, and if one of the two phrases is much longer than the other, then the structure is easier to understand if the heavy phrase is ordered second, even if there is a grammatical reason to put it first. Thus the structures in (56) are easier to process than the structures in (55):

(55) (a) The young boy gave [$_{NP}$ the beautiful green pendant that had been in the jewelry store window for weeks] [$_{PP}$ to the girl].
(b) The landlord talked [$_{PP}$ to the visitor who had just arrived and was staying for the summer] [$_{PP}$ about the problem].

(56) (a) The young boy gave [$_{PP}$ to the girl] [$_{PP}$ the beautiful green pendant that had been in the jewelry store window for weeks].
(b) The landlord talked [$_{PP}$ about the problem] [$_{PP}$ to the visitor who had just arrived and was staying for the summer].

This preference has been argued to be explained by a complexity theory according to which it is more taxing to hold items in memory longer, leading to a preference to 'save the hardest for last' (Bever, 1970; cf. the Early Immediate Constituents proposal of Hawkins (1990, 1994), as in the SPLT). Indeed, the SPLT directly accounts for the ordering preference. In particular, the ordering complexity effects follow from the integration cost component of the SPLT which dictates that longer distance attachments are more expensive to make than shorter distance ones. In particular, the reason that the structures in (55) are harder to process than the structures in (56) is that the maximal integration cost is higher in the examples in (55) than it is in the examples (56), due to a long-distance integration that is required in the processing of (55), but not in the processing of (56). For example, the maximal integration cost of parsing (55b) occurs at the point of processing the preposition 'about', at which point there is an integration cost of I(4) EUs associated with attaching the preposition 'about' to the verb phrase headed by 'talked'. The integration cost here is I(4) EUs because the attachment crosses four new discourse referents, corresponding to two NPs and two verbs. In (56b), on the other hand, the cost associated with attaching each of the prepositional phrase dependents of ‘talked’ is considerably less, I(0) and I(1) EUs. A similar analysis applies to the comparison between (55a) and (56a).

The difference between (55a) and (56a) is also accounted for under the SPLT by a difference in the maximal memory costs in each, because both of the verbal dependents (an NP and a PP) are required. In (55a), the maximal memory complexity occurs at the end of processing the heavy-NP, at which point the predicted PP argument has been retained over four new discourse referents, leading to a cost of M(4) MUs. In contrast, the prediction of the second argument in (56a) – what ends up being the heavy NP – needs to be retained over only the short PP, for a maximal memory cost of M(1) MUs. Hence less memory cost is needed in processing structures with obligatory heavy-shifted phrases.

This memory account extends to a potential explanation for the difference in terms of production complexity, assuming that the SPLT applies in production as well as comprehension (cf. Bock and Levelt, 1995; Wasow, 1997). The idea in the production account is the same as in the comprehension analyses: that it is more costly for the producer to keep a category in mind across the production of other interfering elements. Structures which are less complex to produce will involve keeping fewer categories in mind over fewer intervening elements. For the production account to work, it must be the case that the production mechanism applies partially top-down but does not plan much structure modifying the heads of constituents to the right of the constituent currently being produced, otherwise all constituent orderings would be equally complex according to a production-oriented version of the SPLT.[25]

Let us examine a specific example to see how this framework accounts for heaviness effects. In particular, suppose that the speaker intends to utter a sentence indicating the meaning underlying (55b) (or equivalently (56b)). At the point of generating the matrix verb ‘talked’, there are two potential orderings of the verb’s dependents: the PP headed by ‘about’ first, or the PP headed by ‘to’ first. If the heavier item is produced first, then there is a relatively large memory cost in producing the construction, because the second head has to be kept in mind across the production of the intervening long phrase. On the other hand, if the lighter item is produced first, then there is a relatively small memory cost in producing the construction, because the second head is kept in mind across the production of only a short phrase.

[25]To see this, suppose, contrary to hypothesis, that the production mechanism plans out the full content of all constituents in a sentence before the sentence is produced. Suppose also that the sentence to be generated could be generated using one of two potential orderings of two arguments of a verb, as in (55) and (56). If the heavier argument is generated first, then the remaining light argument needs to be kept in mind for a relatively long time, during the production of the heavy argument. On the other hand, if the lighter argument is generated first, then the remaining heavy argument needs to be kept in mind for a relatively short time, but the item being kept in mind is heavy. There is no reason why the complexity of one of these orderings should be larger than the other.

4.1. Other heaviness contrasts: sentential subjects

The SPLT similarly explains numerous other heavy-shift complexities, including the greater complexity of a sentential subject construction as compared with a construction in which the sentential subject is shifted to the right and replaced by an expletive (Frazier, 1985; Frazier and Rayner, 1988). This complexity difference is made particularly salient when the sentential subject is somewhat complex, as in (57):

(57) (a) That the administrator who the nurse supervised had lost the medical reports bothered the intern.
(b) It bothered the intern that the administrator who the nurse supervised had lost the medical reports.

The reason for the complexity difference between (57a) and (57b) according to the SPLT is that the cost of integrating the clausal argument headed by 'that' to its role-assigning verb 'bothered' is much greater in (57a) than is the corresponding integration in (57b). In particular, the integration cost at 'bothered' is I(5) EUs in (57a), whereas the cost of the similar integration in (57b) is only I(1) EUs, at the word 'that'. Other integrations are of similar complexity in the two sentence structures.

A greater memory load is also required to process (57a) than (57b). The maximal memory complexity of processing (57a) is M(2) + 2M(1) MUs at the point of processing the noun 'nurse', corresponding to (1) M(2) MUs for the prediction of the verb in the sentential subject, (2) 2M(1) MUs for the prediction of the RC verb and empty category and (3) 0 MUs for the matrix verb. The maximal complexity of processing the expletive version is only 2M(1) MUs, which also occurs at the point of processing 'nurse', corresponding to 2M(1) MUs for the prediction of the RC verb and empty category and 0 MUs for prediction of the embedded verb. Note that the prediction of the embedded verb in (57a) is associated with no memory load after the complementizer 'that' has been processed, because the matrix verb has been closed off by clausal closure at this point. Thus the maximal memory cost in processing the sentential subject versions is greater.

5. Ambiguity effects

The evidence put forward thus far in support of the SPLT has come from the processing of unambiguous structures. This section considers some predictions that the SPLT makes when it is used as part of a metric in ambiguity resolution, as was discussed in Section 2.1. The SPLT ambiguity resolution claim is summarized in (58) (cf. the general idea of applying a theory of the processing complexity of unambiguous structures to ambiguity resolution, as put forward in Gibson (1991) and Gibson et al. (1994) using a different memory complexity metric):

(58) Ambiguity resolution hypothesis:
In choosing among ambiguous structures, two of the factors that the processor uses to evaluate its choices are SPLT memory and integration cost (in addition to lexical frequencies, plausibility, and context).

If SPLT memory and integration costs are controlled, then lexical frequency and plausibility information will determine the preferred interpretation. Furthermore, if the SPLT memory cost is the only differentiating factor between two choices, then the activation of the structure with the smaller memory cost will reach the target threshold T_1 more quickly, so that it will be the preferred interpretation. Similarly, if other factors are controlled then SPLT integration cost will determine the preferred interpretation, accounting for locality preference effects, as discussed in Section 2.2.

In conflicts between minimizing memory and integration cost, it is assumed that, other factors being equal, the processor will prefer a structure with a small integration cost over a structure with a large memory cost, and it will prefer a structure with a small memory cost relative to one with a large integration cost. In conflicts between choices where the costs are more closely balanced, e.g. a structure with a memory cost of M(n) MUs as compared with one with an integration cost of I(n) EUs, it will initially be assumed that the processor prefers to minimize memory cost. The motivation for this assumption is that memory cost is effectively potential integration cost – integration cost that will be expended later – and processing additional material cannot lower this cost, only increase it, leading to a larger integration cost downstream. So given a choice between (1) performing a Y EU integration and ending up with a structure with an M(n) + X MU memory cost and (2) performing an I(n) + Y EU integration and ending up with a structure with an X MU memory cost, the processor prefers to make the complex integration (I(n) + Y EUs), thereby locally minimizing memory cost. Although this choice does not locally minimize integration cost, it does so over the course of the sentence.

5.1. Processing empty categories

5.1.1. The Active Filler Strategy

There is a sizable literature on how empty categories, such as traces of long-distance movement, are processed (see e.g. Fodor, 1978; Clifton and Frazier, 1989; Boland and Tanenhaus, 1991; Kaan, 1997; for summaries). Thus far, the most empirically well-established hypothesis is the Active Filler Strategy (Frazier, 1987b; Clifton and Frazier, 1989; Frazier and Flores d'Arcais, 1989; Pickering and Barry, 1991; Gibson and Hickok, 1993; Stevenson, 1993; Crocker, 1994; Gorrell, 1996), which was generalized by De Vincenzi (1991) to the Minimal Chain Principle:

(59) 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 and Flores d'Arcais, 1989)

(60) The Minimal Chain Principle (MCP):
Postulate required chain members at the earliest point grammatically possible but postulate no potentially unnecessary chain members. (De Vincenzi, 1991)

For simplicity, the discussion will be restricted to the AFS, but it should be kept in mind that the observations apply equally well to the MCP. The AFS provides explanations for numerous experimental results with respect to the processing of empty categories cross-linguistically. For example, people read a lexical object of a transitive verb more slowly when there is an uninterpreted filler in the syntactic context than when there is no such filler in the context, resulting in a 'filled-gap' effect. Thus, in (61) the NP object 'us' of the verb 'bring' is read more slowly in (61b) than in (61a) (Stowe, 1986; see also Crain and Fodor, 1985):

(61) (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.

According to the AFS, this result occurs because there is a preference to associate the filler 'who' with an object position empty category as soon as the verb 'bring' is encountered in (61b). This leads to the need to reanalyze as soon as the word 'us' is encountered, which must be analyzed as the lexical direct object of 'bring'. No such reanalysis is required in the processing of (61a), and so this sentence structure is processed more quickly.

5.1.2. The SPLT: an extension of the Active Filler Strategy

Many effects which have been argued to be evidence for the Active Filler Strategy follow directly from the SPLT as applied to ambiguity as proposed in (58), so that a variant of these proposals follows as a corollary of the SPLT. Given (58), the preference for positing an object-gap following a verb in the presence of a filler follows directly. In particular, if a gap is posited, then there is no longer a memory cost associated with predicting this position from the filler. If a gap is not posited, then the memory cost associated with the predicted category remains. Thus there is less memory cost if the gap is posited, and the filled-gap effect results.

More concretely, consider the memory cost associated with the structure for (61b) at the point of processing the auxiliary verb 'will'. The memory cost here is M(1) MUs corresponding to: (1) the prediction of an NP-gap position for the wh-pronoun 'who' (M(1) MUs) and (2) the prediction of the embedded verb (0 MUs, because the higher clauses are closed at this point). After the verb 'bring' has been processed, there are at least two potential structures: one in which an NP-gap co-indexed with 'who' is posited in the object position of the verb 'bring', and one in which no such gap is posited. The memory cost associated with the structure in which the gap is posited is 0 MUs, because all obligatory syntactic predictions have been satisfied at this point in this structure. In contrast, the memory cost associated with the structure in which no gap is posited is M(2) + M(0) MUs corresponding to: (1) the prediction of an NP-gap for 'who' (now increased to M(2) MUs) and (2) the prediction of an object NP following 'bring' (M(0) MUs). The memory cost associated with the structure in which the gap is posited is less, and this structure therefore is preferred (see Gibson et al. (1994) for a similar analysis using a different syntactic complexity metric). It should be noted that forming the relationship between the gap position and the wh-filler is associated with an integration cost of I(2) EUs, but this cost is outweighed by the high memory cost (M(2) + M(0) MUs) of not performing the integration.

Numerous other processing effects normally attributed to the AFS follow from the SPLT in a similar fashion. One processing effect that the SPLT captures in an interestingly different way from the AFS is the preference for a subject-gap interpretation in ambiguous subject- or object-extracted structures in head-final languages like Dutch and German (Frazier, 1987b; Frazier and Flores d'Arcais, 1989; Mecklinger et al., 1995; Meng, 1995; Schriefers et al., 1995; Schlesewsky et al., 1996; Kaan, 1997). For example, the relative clause in each of (62) and (63) is initiated by the case-ambiguous wh-pronoun *die* ('who') and the NP *de boswachter* ('the forester').

(62) Subject-Object-Verb RC:
Karl hielp [de mijnwerkers die de boswachter vonden].
Karl helped the mine workers who the forester found-plural
'Karl helped the mine workers who found the forester.'

(63) Object-Subject-Verb RC:
Karl hielp [de mijnwerkers die de boswachter vond].
Karl helped the mine workers who the forester found-singular
'Karl helped the mine workers who the forester found.'

Since Dutch is verb-final in embedded clauses, the wh-pronoun 'die' can either be the subject or the object of the verb to follow, resulting in either a Subject-Object-Verb (SOV) word order or an Object-Subject-Verb (OSV) word order. The SOV interpretation is preferred, as evidenced by longer reading times on the RCs containing a disambiguating verb forcing the OSV interpretation, as in (63). This preference follows from the AFS under the additional assumption that the parser builds structure top-down, and the first viable location for a trace that the parser reaches is in the subject position of the clause (Frazier, 1987b; Frazier and Flores d'Arcais, 1989).

The preference for the subject-gap interpretation follows straightforwardly from the SPLT. In particular, at the point of processing the ambiguous pronoun 'die', there are two potential analyses: 'die' as the subject, or 'die' as the object. If 'die' is analyzed as the subject, then only a verb and a subject NP-trace head are needed to complete the input as a grammatical sentence. For example, an intransitive verb would satisfy both of these predictions. On the other hand, if 'die' is analyzed as the object, then three heads are needed to complete the input as a grammatical sentence:

a verb, an object NP-trace and a subject noun for the verb. The parser therefore chooses the subject interpretation because it requires fewer predicted categories, resulting in a lower memory load, and the preference is explained (see Schlesewsky et al., 1996, 1997; for alternative accounts of the subject-gap preference in terms of memory load, using a different syntactic complexity metric).[26]

5.1.3. Distinguishing the Syntactic Prediction Locality Theory from the Active Filler Strategy

In addition to accounting for data which has been claimed as evidence for the AFS, the SPLT also makes a number of predictions that distinguish it from the AFS. Since the SPLT is a syntactic complexity metric and not simply an ambiguity resolution strategy like the AFS, it predicts that processing a clause-initial object NP should induce more of a memory load than processing a clause-initial subject NP, independent of ambiguity. This prediction is made because a clause-initial object NP requires at least two heads to complete a grammatical sentence – a verb, and a subject noun – whereas a clause-initial subject NP requires only one head to complete a grammatical sentence – a verb.

This prediction can be tested in verb-final languages whose NPs are case-marked, thus providing a reliable cue to an NP’s syntactic position. It is hypothesized that the frequency information about case-marking is encoded in the lexicon such that a lexical entry for a morpheme *M* includes frequency information about the categories which follow *M* and are potential heads or dependents of *M*. Preliminary evidence for this hypothesis about the structure of the lexicon comes from English reading experiments where it has been demonstrated that people are sensitive to the frequencies of the syntactic categories following verbs (Trueswell et al., 1993; Trueswell 1996). Thus the lexical entry for an English verb includes frequency information about the possible dependent categories to follow. Applying the category-prediction lexical-frequency hypothesis to a head-final language like German, the lexical entry for a case-marking will include frequency information about the syntactic positions that the case-marking can reflect. For example, although some German predicates take non-nominative subjects (e.g. ‘Mir ist kalt’, ‘1st-person-sing-dat is cold’, ‘I am cold’), most German dative-cased NPs occur as the indirect object of a verb, and most German accusative-cased NPs occur in direct object position.

Given these assumptions about the structure of lexical entries, the SPLT can be tested in languages like German and Finnish. In relevant German experiments, Hemforth (1993) compared structures with a clause-initial nominative NP, as in (64a), to structures with a clause-initial accusative NP, as in (64b), using a self-paced word-by-word reading paradigm:

(64) (a) der rote Bär küsste die kleine blonde Frau.
the-nom red-nom bear-nom kissed the small fair woman
‘The red bear kissed the small fair woman.’

[26]The SPLT analysis does not depend on the existence of empty categories in these structures or any of the structures to follow: Even without empty categories, more categories are required to complete an accusative-initial clause than a nominative-initial clause.

(b) den roten Bären küsste die kleine blonde Frau.
the-acc red-acc bear-acc kissed the small fair woman
'The small fair woman kissed the red bear.'

For the sentences with clause-initial accusative-marked NPs, reading times were longer over the clause-initial NP and at the verb, as expected under the SPLT analysis.[27] In particular, because nominatively cased NPs occur most frequently in subject position, processing a sentence-initial nominatively cased NP causes the prediction of only a verb to form a grammatical sentence. On the other hand, because accusatively cased NPs occur most frequently in object position, processing a sentence-initial accusatively cased NP causes the prediction of a subject noun in addition to a verb (and one further head under some syntactic analyses: an NP-trace corresponding to the fronted accusative object NP). Note that, because there is no ambiguity in this complexity comparison, the Active Filler Strategy makes no predictions, in spite of their similarity with the kinds of ambiguities that the AFS is intended to explain.

In relevant Finnish experiments, Hyönä and Hujanen (1998) demonstrated that processing a clause-initial genitive object NP or a locative adverbial NP induces more of a processing load than processing a clause-initial nominative subject NP, in spite of the grammaticality of each word order. An example set of the compared structures is given in (65).

(65) (a) Subject:
Lopulta **politiikka** tuhoaa joustavuuden päätöksenteossa.
finally politics (nom). destroy flexibility (gen.) decision-making (loc.)
'Finally the politics destroy the flexibility in decision-making.'
(b) Object:
Lopulta **politiikan** tuhoaa jatkuvasti kasvava nukkuvien puolue.
finally politics (gen.) destroy continuously growing body of non-voters (nom.).
'Finally the politics are destroyed by the continuously growing body of non-voters.'
(c) Adverbial:
Lopulta **politiikassa** tuhoaa moni poliitikko kansansuosionsa.
finally politics (iness.) destroys many politician popularity-his
'Finally in politics many politicians destroy their popularity.'
Note: inessive case is a locative case corresponding to the English preposition ''in''.

Hyönä and Hujanen observed that reading times were faster in the Subject condition than in the Object and Adverbial conditions during the processing of the

[27]For the self-paced reading study, the reading times were numerically but not significantly different within the initial NP, but the difference reached significance when eye-movements were measured in later studies (B. Hemforth, pers. commun. to Schlesewsky et al., 1997).

clause-initial NP. Hyönä and Hujanen also observed that the frequency of the Subject-Verb-Object word order is much larger than either the Object-Verb-Subject or Adverbial-Verb-Subject word orders. Based on these observations, they argued that the cause of the relative ease of processing the Subject-Verb-Object word order is its high frequency in the input: high frequency word order patterns are easy to comprehend (cf. Bates and MacWhinney, 1989; Mitchell and Cuetos, 1991; Mitchell, 1994).

The SPLT offers a different account of these findings. The relative ease of processing the SVO word order occurs because the nominative subject appears initially, so that there is no memory load associated with keeping in mind the expectation of a subject noun after this noun has been encountered. The OVS and the Adv-V-S word orders are more complex at the initial nouns because it is necessary to retain the prediction of a subject noun at this location. In addition to accounting for the observed reading data, the SPLT may also provide an account of the relative frequencies of SVO, OVS and Adv-V-S word orders. Under the assumption that less complex structures will be produced more frequently than more complex structures, other factors being equal (Gibson and Pearlmutter, 1994), SVO sentences are expected to be more frequent than OVS and Adv-V-S word orders, because they require less memory to produce and comprehend according to the SPLT.

As well as predicting that nominative NPs should be easier to process than other kinds of NPs clause-initially, the SPLT also makes predictions about the relative complexity of other clause-initial NPs, to the extent that processing other NPs clause-initially may require additional heads to be present. In particular, the SPLT predicts that a dative NP should be even more complex than an accusative NP clause-initially, because most dative NPs are dependent on verbs which have both a subject (typically marked with nominative case) and a direct object (typically marked with accusative case). Thus, processing a dative NP clause-initially causes the parser to build a structure in which the dative NP is the second object of a verb to come, so that a subject noun and a direct object noun are also needed under this lexically preferred analysis. This prediction of the SPLT has been verified in the processing of German in experiments performed by Bader et al. (1996) (see also Schlesewsky et al., 1996, 1997) on example sentences like those in (66):

(66) (a) Accusative ambiguous:
Dirigenten, die ein schwieriges Werk einstudiert haben, kann ein Kritiker ruhig umjubeln.
conductors who a difficult work rehearsed have can a critic safely cheer
‘A critic can safely cheer conductors who have rehearsed a difficult work.’
(b) Dative ambiguous:
Dirigenten, die ein schwieriges Werk einstudiert haben, kann ein Kritiker ruhig applaudieren.
conductors who a difficult work rehearsed have can a critic safely applaud
‘A critic can safely applaud conductors who have rehearsed a difficult work.’

(c) Dative unambiguous:
Musikern, die ein schwieriges Werk einstudiert haben, kann ein Kritiker ruhig applaudieren.
musicians who a difficult work rehearsed have can a critic safely applaud
'A critic can safely applaud musicians who have rehearsed a difficult work.'

The German noun 'Dirigenten' ('conductors') in (66a) and (66b) is ambiguous between accusative and dative case. The results from Bader et al.'s off-line and on-line studies demonstrated that people initially follow the accusative reading, and are forced to reanalyze if the dative reading is the one that was required. Furthermore, Bader et al. showed that the unambiguous dative condition (66c) is intermediate in complexity between the ambiguous accusative and the ambiguous dative conditions.

From these results, Bader et al. argued for two assumptions for the processing of NPs (in addition to a strategy for positing empty categories, like the Active Filler Strategy or the Minimal Chain Principle):

(67) (a) If possible, prefer structural Case over lexical Case.
(b) If possible, prefer nominative Case over accusative Case.

(67a) describes the preference to treat a clause-initial NP as a nominative or an accusative (structural Case) rather than as a dative (lexical Case). (67b) describes the preference to treat a clause-initial NP as a nominative rather than as an accusative, as discussed earlier (for which Bader et al. give additional evidence).

Under the SPLT, there is no need to add the assumptions in (67): they follow straightforwardly for the processing of clause-initial NPs. The preference for treating the case-ambiguous clause-initial NP as accusative rather than dative is predicted because there are fewer required categories if the NP is taken to be an accusative rather than a dative. Because of the additional memory cost in processing a dative clause-initially, the difference between the unambiguous dative and the ambiguous accusative is accounted for. The additional difference between the ambiguous dative/accusative and the unambiguous dative conditions is predicted because the parser follows the accusative reading of the case-ambiguous word, leading to reanalysis difficulty later.

5.2. SPLT syntactic complexity is independent of frequency and plausibility information

Many ambiguities other than those involving filler-gap dependencies have been explored extensively over the past twenty years. Although syntactically-based theories of ambiguity resolution were prevalent ten years ago (e.g. Frazier, 1987a), more recent studies have led researchers to downplay the role of syntactic complexity in determining the human parser's choice when faced with syntactic ambiguity (e.g. MacDonald et al., 1994; Trueswell et al., 1994). For example, it has been demonstrated that lexical frequency and plausibility factors determine the initial preference in the main-verb (MV)/reduced-relative (RR) ambiguity, an example of which is provided in (68) (Trueswell et al., 1994; cf. Ferreira and Clifton, 1986):

(68) (a) The defendant examined by the lawyer turned out to be unreliable.
(b) The evidence examined by the lawyer turned out to be unreliable.

The word 'examined' is ambiguous between a simple past tense reading, leading to the main verb interpretation, and a past participle, leading to a reduced relative clause reading. The reduced relative reading is required in both of the examples in (68), but people process them differently because of different lexical frequencies and plausibilities associated with the two possible readings in each. In particular, 'defendant' is highly plausible as the agent of 'examined', supporting the main-verb reading, whereas 'evidence' is highly implausible as the agent of 'examined'. Both are plausible patients for the verb 'examined', supporting the reduced-relative reading. Furthermore, there is an overall frequency bias in favor of treating 'examined' as a simple past tense rather than as a past participle. The combination of these lexical factors leads people to favor the MV reading initially in (68a), leading to a need for reanalysis during the agentive prepositional phrase 'by the lawyer'. This reanalysis is measurable relative to an unambiguous control sentence structure, as in (69a):

(69) (a) The defendant that was examined by the lawyer turned out to be unreliable.
(b) The evidence that was examined by the lawyer turned out to be unreliable.

In contrast, the plausibility factors favor the RR reading so strongly in (68b) that there is no evidence of reanalysis in the disambiguating prepositional phrase, compared with the unambiguous control in (69b). Based upon evidence like this, researchers like Trueswell et al. and MacDonald et al. hypothesized that the parser uses lexical frequency and plausibility information to perform disambiguation, and that there is no independent syntactic complexity metric. For example, MacDonald et al. make the following claim: 'Reinterpreting syntactic ambiguity resolution as a form of lexical ambiguity resolution obviates the need for special parsing principles to account for syntactic interpretation preferences...' (p. 676).

However, when examined in light of the SPLT, the evidence put forward in favor of this hypothesis is relevant only to certain specific syntactic complexity metrics (e.g. Frazier's principle of Minimal Attachment), and not to syntactic complexity in general. It turns out that the ambiguities which have been shown to be highly susceptible to differences in lexical frequency and plausibility information are ambiguities for which there is little memory and integration complexity difference between the two readings according to the SPLT. For example, consider the MV/RR ambiguity with respect to the SPLT. After processing 'examined', the syntactic memory cost associated with the MV reading is M(0) MUs, corresponding to the prediction of a direct object NP. The syntactic memory cost associated with the RR reading is at most M(0) MUs, corresponding to (1) the prediction of an adverbial modifier of the verb 'examined' (which may be optional, and hence not relevant to the SPLT) and (2) the prediction of the matrix verb.[28] Thus the memory cost difference between the two structures is either M(0) or 0 MUs. The processing of unambiguous structures has demonstrated that M(0) MUs is a small quantity of memory resources, so that there is at most a small syntactic complexity difference (if any) between the MV and RR structures according to the SPLT. Furthermore, both possible integrations are very local, so that there is little if any difference in this aspect of computational resource use between the two structures. If one assumes the multiple-constraint ambiguity resolution hypothesis in (58) (repeated here), then the results from the studies on the MV/RR ambiguity follow.

(58) Ambiguity resolution hypothesis:
In choosing between ambiguous structures, two of the factors that the processor uses to evaluate its choices are SPLT memory and integration cost (in addition to lexical frequencies, plausibility, and context). In conflicts between minimizing memory cost and integration cost of the same locality cost, it will initially be assumed that the processor prefers to minimize memory cost.

In particular, there is a small memory cost and integration cost difference between the MV and RR structures in the ambiguities in (68) at the point of processing 'examined', so that frequency and plausibility dominate the preference at this point. The MV structure is therefore initially preferred in (68a) and the RR structure is initially preferred in (68b).

So far, only a narrow range of syntactic ambiguities have been investigated with respect to the claim that syntactic complexity is derivable from lexical frequency and plausibility differences. The syntactic ambiguities that have been studied most carefully are the following:

1. V NP PP (Taraban and McClelland, 1988; Spivey-Knowlton and Sedivy, 1995; cf. Rayner et al., 1983; Clifton et al., 1991);
2. V NP/S (Trueswell et al., 1993; Garnsey et al., 1997; cf. Frazier and Rayner, 1982; Ferreira and Henderson, 1990);
3. Det N N/V (Frazier and Rayner, 1987; MacDonald, 1993).

For all of these ambiguities, there are small memory and integration cost difference between the two ambiguities according to the SPLT, so the SPLT ambiguity resolution hypothesis in (58) is consistent with the observation that frequency and plausibility information guides the initial preferences in these ambiguities.

In order to test the SPLT hypothesis, it is necessary to either control plausibility and frequency information or to examine ambiguities whose memory cost or integration cost differs more substantially according to the SPLT. We have already seen cases in which integration cost differs substantially in two different interpretations of an ambiguity, resulting in a locality preference (see Section 2.2). Gibson et al.

[28]During the processing of the reduced relative, the memory cost actually reaches 2M(0) MUs.

(1997) recently explored ambiguities in which the SPLT memory costs varied substantially across the two ambiguous structures. In one study, the MV/RR ambiguity was investigated within the syntactic context of an embedded relative clause, as in (70):

(70) The witness who the evidence examined by the lawyer implicated seemed to be very nervous.

The items were constructed using Trueswell et al.'s items as a base, so that plausibility factors highly favored the RR interpretation of the ambiguous verb 'examined'. However, syntactic complexity as measured by the SPLT memory cost strongly favors the MV reading. In particular, the memory cost for the structure containing the implausible MV reading of 'examined' is 0 MUs, corresponding to the prediction of the matrix verb, while the memory cost for the plausible RR reading is 2M(2) MUs corresponding to:

1. The prediction of the matrix verb: 0 Mus
2. The prediction of the embedded verb: M(2) MUs (an NP and a verb have been processed since this prediction was first made)
3. The prediction of the NP-gap for the wh-pronoun: M(2) MUs (an NP and a verb have been processed since this prediction was first made).

The memory cost difference between the two readings is therefore 2M(2) MUs, which is much larger than in any of the ambiguities that have been explored before this.[29] If the SPLT memory cost is being used in ambiguity resolution on-line, then people will have more difficulty following the RR reading of the verb 'examined' in (70) than they will in a control case like (68b). To test this hypothesis, Gibson et al. measured the reading times of 60 participants taking part in a self-paced word-by-word moving-window paradigm to compare the processing of sentences like (70) along with unambiguous controls like (71) to closely related examples with a small memory complexity difference between the MV and RR readings and their unambiguous controls, as in (72).

(71) Large memory cost difference, unambiguous control:
The witness who the evidence that was examined by the lawyer implicated seemed to be very nervous.

(72) (a) Small memory cost difference, ambiguous:
The witness thought that the evidence examined by the lawyer implicated his next – door neighbor.

[29]The integration cost is correspondingly larger in the MV structure, reflecting the positing of a trace in the object position of 'examined'. According to the ambiguity resolution hypothesis in (58), the parser favors minimizing the memory cost in situations like this.

[30]Residual reading times are computed by subtracting from raw reading times each participant's predicted time to read words of the same length, calculated by a linear regression equation across all sentences in the experiment (Ferreira and Clifton, 1986).

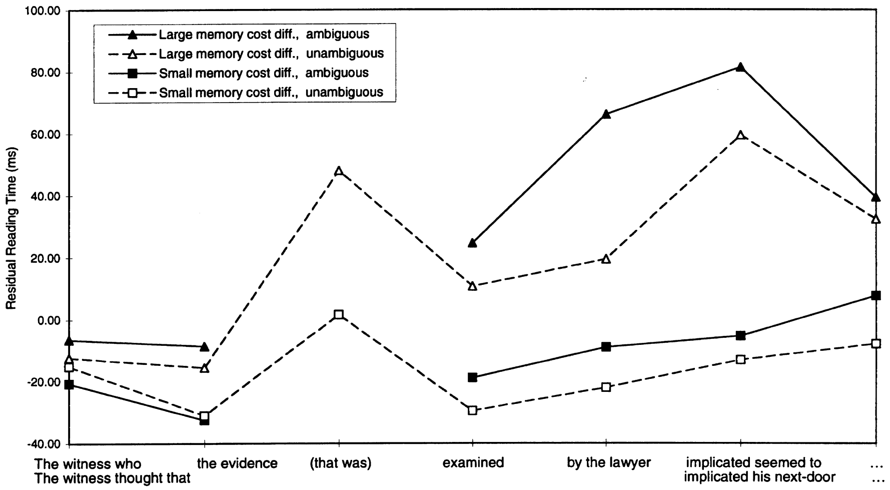

Fig. 10. Residual reading times for 60 subjects taking part in a self-paced word-by-word self-paced moving-window reading experiment involving four conditions which crossed SPLT memory complexity difference (high memory complexity, low memory complexity difference) with ambiguity (ambiguous, unambiguous).

(b) Small memory cost difference, unambiguous:
The witness thought that the evidence that was examined by the lawyer implicated his next – door neighbor.

As in the simple MV/RR ambiguities in (68), the memory complexity at the point of processing ‘examined’ in (72a) is M(0) MUs for each of the MV and RR interpretations of the ambiguity, so that there is a small memory complexity difference between the ambiguities for this structure.

The participants’ residual reading times for each of these conditions are plotted in Fig. 10.[30] The first result of interest to the SPLT is that reading times for the region ‘the evidence examined by the lawyer’ were significantly faster in the small memory cost difference conditions than in the large memory cost difference conditions. Although the same integrations are being performed in all of these conditions, there is a larger syntactic memory load in the large memory cost difference conditions than in the small memory cost difference conditions, leading to longer reading times in these conditions.

Second, the experimental participants read the disambiguating prepositional phrase more slowly in the ambiguous large memory cost condition than in the disambiguated large memory cost condition, as predicted by the SPLT applied to ambiguity resolution. Crucially, there was also a significant interaction of memory cost difference (large, small) and ambiguity (ambiguous, unambiguous) in the disambiguating PP, such that the reanalysis effect was significantly larger in the large memory cost conditions than in the small memory complexity conditions. This is consistent with the hypothesis that people initially follow the main-verb reading in (70), because of its much lower syntactic complexity and in spite of its implausibility. On the other hand, people use the plausibility information to perform the disambiguation in (72a), because the syntactic complexity difference between the readings is much smaller. Thus syntactic memory cost complexity as measured by the SPLT appears to be an independent factor involved in ambiguity resolution which is not reducible to frequency and plausibility. Gibson et al. also report results from a set of experiments on a different ambiguity with similar conclusions.

It should be noted that the relative clause MV/RR ambiguity occurs frequently, in every object-extracted relative clause whose main verb is ambiguous between the simple past and the past participle. This is not a rare occurrence: there are hundreds in the first 10% of the Brown corpus (Kucera and Francis, 1967) from the Penn treebank corpus (Marcus et al., 1993), a 1 000 000 word corpus of hand-parsed sentences. It happens that the overwhelming majority of instances of this ambiguity are resolved in favor of the MV reading. If anything, ambiguities like this one may be more interesting than ambiguities which are resolved more evenly, because the biases suggest that they are not accidental. The SPLT provides a potential explanation, because generating a reduced relative clause modifying the subject of a relative clause requires a much larger quantity of resources than generating the main verb of the relative clause at the same point.[31]

6. A variant of the memory cost theory which is not based on locality

In the computational resource theory presented here, both integration cost and memory cost are hypothesized to increase over distance. However, it turns out that the conceptual motivation for the hypothesis that *both* of these costs increase over the processing of intermediate structure is somewhat weak within the activation-based processing theory. In particular, if the processor is spending resources to keep earlier predictions highly activated, then the integrations should be low cost. On the other hand, if the processor is not spending resources to keep earlier predictions highly activated, then the integrations should be high cost because the early predictions have to be reactivated. In short, if one of memory and integration costs is distance-based, then the other should not be distance-based within an activation-based processing theory.

Direct evidence for the distance-based integration cost hypothesis was provided in the form of reading time profiles for simple relative clause structures, but the evi-

[31]Recently, a number of researchers have proposed that *structural* or *contingent* frequencies are involved in ambiguity resolution (e.g. Mitchell, 1994; Spivey-Knowlton and Tanenhaus, 1996; McRae et al., 1997; Tabor et al., 1997; cf. Jurafsky, 1996). According to these proposals, the human sentence processor keeps track of lexical frequencies depending on their structural contexts. These theories might therefore be able to account for differential ambiguity-resolution performance on the MV/RR ambiguity, as long as the corpus frequencies match the performance data. However, there are currently no general versions of the contingent-frequency hypothesis, so that it cannot be tested with respect to any ambiguities other than the few for which there have been specific proposals.

dence for the distance-based memory cost hypothesis was more indirect, based on a theory of complexity ratings. Thus, it is worth exploring a variant of the SPLT according to which the integration cost increases over the processing of intermediate structure (as before), but the memory cost for a predicted category remains fixed over the processing of intermediate structure.[32] The question is whether this variant of the SPLT can be empirically adequate to account for the broad range of complexity phenomena discussed here. In order to work out this idea in more detail, it is necessary to propose a theory of complexity judgments under such a framework. Recall that under the increasing memory-cost theory it was proposed that complexity judgments correspond to the maximal quantity of memory resources used in the course of processing a sentence. This hypothesis is no longer viable if memory costs do not increase over the processing of intervening structure.

One possible candidate for a theory of intuitive complexity is that complexity judgments parallel comprehension times, so that the on-line intuitive complexity of processing a sentence is determined by the ratio of integration cost to computational resource space available at that point. A complexity rating for a sentence as a whole might therefore be determined by the maximal value for this ratio during the processing of a sentence:

(73) Complexity rating hypothesis:
The intuitive complexity of a sentence is determined by the maximal ratio of integration cost to computational resources available, over the course of parsing the sentence.

Furthermore, it is hypothesized that a minimal quantity of computational resources is required for an integration to be successful, so that a processing overload effect occurs if the ratio of integration cost to resources available is greater than a fixed constant. It turns out that this variant of the SPLT can account for most of the processing overload effects presented here straight-forwardly. For example, the complexity of nested relative clause structures such as (74) is accounted for:

(74) The student who the professor who the scientist collaborated with had advised copied the article.

For simplicity, the memory cost component of the theory will be ignored.[33] The point of maximal integration cost in (74) occurs at the second verb 'advised', at which point there are two long distance integrations: (1) the subject-verb integration between 'the professor' and 'advised', whose cost is I(3) EUs and (2) the integration between the wh-filler 'who' and the object position of the verb 'advised', whose cost

[32] I owe special thanks to Colin Phillips who was the first to propose this idea to me. Thanks also to Dick Hudson and Theo Vosse for making similar observations.

[33] Memory cost still plays an important role in this variant of the SPLT. For example, it is needed to account for the observation that reading times are slower for the same integrations at points of higher memory cost (e.g. Gibson et al., 1997), and to account for the ambiguity effects in Section 5.

is I(4) EUs. The combination of these integration costs at this parse state, where there is also a memory cost, may be too much for the processor to handle.

The contrast between examples like (74) whose most embedded subject is a new discourse referent ('the scientist') and similar structures whose most embedded subject is a pronoun in the discourse (e.g. 'I') is accounted for in much the same way as under the memory-based theory. Since one less new discourse referent is crossed by each of the integrations at the point of maximal integration cost, the maximal integration cost in such a structure is only I(2) + I(3) EUs, so that this structure is more processable. This integration-based theory of intuitive complexity accounts for many of the complexity observations described in Section 3, including the contrast between the different embedding orderings of relative clauses and sentential complements, among many other complexity effects.

There are additional conceptual advantages to the integration-based variant of SPLT. One major benefit of this theory is that it is no longer necessary to stipulate that the prediction of the top-level clause is cost-free: All syntactic predictions can be associated with the same non-increasing memory cost. Recall that the empirical motivation for stipulating that the prediction of the top-level clause is cost-free is that there is no measurable complexity difference between pre-matrix-verb nested RCs as in (74) and post-matrix-verb nested RCs as in (75) (Hakes et al., 1976; Gibson and Thomas, 1997a):

(75) # The article was copied by the student who the professor who the scientist collaborated with had advised.

The maximal integration cost of (75) is I(3) + I(4) EUs, occurring at the point of processing the verb 'advised'. This is the same maximal integration cost as in processing (74), so that the lack of a complexity contrast is directly accounted for under this variant of the SPLT.[34]

Since it is not necessary to stipulate that the prediction of the top-level verb is cost-free under this variant of the SPLT, the theory of clausal closure developed under the SPLT is also unnecessary, because its presence was forced by the zero-cost top-level verbal prediction stipulation. Thus the integration-based variant of the SPLT potentially offers considerable conceptual advantages over the memory-based SPLT.

[34]As stated thus far, the integration-based variant of the SPLT still predicts that (75) should be less complex than (74), because there is a syntactic prediction memory cost associated with the prediction of the matrix verb at the point of maximal integration complexity in (74) ('advised') which is not present at the point of maximal integration complexity in (75). However, the memory cost component of sentence understanding has been oversimplified in the SPLT. There is memory cost in language comprehension not only in keeping track of what categories need to appear, but also in keeping the structure for the input so far in memory, possibly restricted to the current clause. The memory cost for storing the input so far is larger for the RC/RC structure in post-verbal position than for the RC/RC structure in pre-verbal position, counterbalancing the syntactic prediction memory cost which weighs more heavily on the RC/RC structure in pre-verbal position, so that no difference between the two is necessarily expected.

7. Summary and conclusions

A theory of linguistic complexity has been proposed here which associates increasing memory cost with predicted syntactic categories and increasing integration cost with distance of attachment. This theory – the Syntactic Prediction Locality Theory – provides a unified theory of a large array of disparate processing phenomena, including the following:

1. On-line reading times of subject- and object-extracted relative clauses
2. The complexity of doubly-nested relative clause constructions
3. The greater complexity of embedding a sentential complement within a relative clause than the reverse embedding
4. The lack of complexity of multiply embedded structures with pronouns in the most embedded subject position
5. The high complexity of certain two-clause constructions
6. The greater complexity of nesting clauses with more arguments in Japanese
7. The lack of complexity of two-clause sentences with five initial NPs in Japanese
8. Heaviness effects
9. The greater complexity of center-embedded constructions as compared with cross-serial constructions
10. Ambiguity effects: (a) Gap-positing preference effects; (b) Syntactic complexity effects independent of plausibility and frequency.

In addition to accounting for these kinds of data, understanding the relationship between language processing and the available computational resources may have many other as yet unexplored consequences. For example, the frequencies of different constructions within a language may be partially explained by a such a theory (Gibson and Pearlmutter, 1994; Gibson et al., 1996b; cf. MacDonald, 1997). A computational resource theory may also help to explain stages of language acquisition in children acquiring a first language. Since children have a more limited working memory capacity, some constructions will be beyond their processing capacities, and they will not produce or comprehend them correctly until the capacity grows large enough (cf. Bloom, 1970, 1990; Valian, 1990). Thirdly, understanding the computational resources involved in sentence comprehension may also help to provide an account of syntactic comprehension disorders in aphasia. Recent work has suggested that some aphasics' comprehension problems arise due to severe limitations in the their processing capacity or processing efficiency (Frazier and Friederici, 1991; Haarman and Kolk, 1991; Miyake et al., 1994). A theory like the SPLT provides an independently motivated theory of linguistic complexity that may help account for some of the aphasic data. Finally, understanding the relationship between language processing and computational resources may eventually have computational applications in which the comprehensibility of language is important, such as grammatical-style checking and the automatic generation of human language (e.g. in machine translation).

Acknowledgements

A great many people provided invaluable feedback in the writing of this paper. I would first like to extend special thanks to the following people who have been collaborators for many of the experiments discussed here, which led to the proposed theory: Maria Babyonyshev, Dan Grodner, James Thomas, Susanne Tunstall and Tessa Warren. I would also like to extend special thanks to those who have engaged with me in numerous lengthy discussions about this work: Ani Patel, Neal Pearlmutter, Colin Phillips, and Carson Schütze. Finally, I would like to thank the following people for their comments and discussions of this work: two anonymous reviewers, Chris Bader, Bob Berwick, Tom Bever, David Caplan, Bob Carpenter, Patricia Carpenter, Morten Christiansen, Stephen Crain, Marcel Just, Edith Kaan, Ron Kaplan, Gerard Kempen, Robert Kluender, Alec Marantz, Jacques Mehler, Risto Miikkulainen, Edson Miyamoto, Ken Nakatani, David Pesetsky, Steven Pinker, Molly Potter, Rose Roberts, Philippe Schlenker, Gregg Solomon, Laurie Stowe, Whitney Tabor, Mike Tanenhaus, Theo Vosse, Nigel Ward, Tom Wasow, Gloria Waters, Ken Wexler, audiences at MIT, the University of Pittsburgh, Yale University, the 1996 Maryland Mayfest, the June, 1996 computational psycholinguistics conference in Wassanaar the Netherlands, the 1997 CUNY sentence processing conference, and the August, 1997 computational psycholinguistics conference in Berkeley, California. I take full responsibility for the remaining errors, even though some of them are probably Neal's fault.

References

Abney, S.P., Johnson, M., 1991. Memory requirements and local ambiguities of parsing strategies. Journal of Psycholinguistic Research 20, 233–250.

Ades, A., Steedman, M.J., 1982. On the order of words. Linguistics and Philosophy 4, 517–558.

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

Anderson, J.R., 1994. Learning and Memory. Wiley, New York.

Babyonyshev, M., Gibson, E., 1995. Processing overload in Japanese. In: Schütze, C.T., Ganger, J.B., Broihier, K. (Eds.), Papers on Language Processing and Acquisition, MIT Working Papers in Linguistics 26, MIT, Cambridge, MA, pp. 1–35.

Babyonyshev, M., Gibson, E., 1997. Japanese Linguistic Complexity. Manuscript, MIT, Cambridge, MA.

Bach, E., Brown, C., Marslen-Wilson, W., 1986. Crossed and nested dependencies in German and Dutch: a psycholinguistic study. Language and Cognitive Processes 1, 249–262.

Baddeley, A., 1990. Human Memory. Allyn and Bacon, Needham Heights, MA.

Bader, M., Bayer, J., Hopf, J.-M., Meng, M., 1996. Case-assignment in processing German verb-final clauses. In: Schütze, C. (Ed.), Proceedings of the NELS 26 Sentence Processing Workshop, MIT Occasional Papers in Linguistics 9, Cambridge, MA, pp. 1–25.

Baird, R., Koslick, J.D., 1974. Recall of grammatical relations within clause-containing sentences. Journal of Psycholinguistic Research 3, 165–171.

Bates, E., MacWhinney, B., 1989. Functionalism and the competition model. In: MacWhinney, B., Bates, E. The Crosslinguistic Study of Sentence Processing. Cambridge University Press, Cambridge, UK, pp. 3–73.

Behaghel, O., 1910. Beziehungen zwischen Umfang und Reihenfolge von Satzgliedern. Indogermanische Forschungen 25, 110–142.

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

Bever, T.G., 1974. The ascent of the specious, or there's a lot we don't know about mirrors. In: Cohen, D. (Ed.), Explaining Linguistic Phenomena, Hemisphere, Washington, pp. 173–200.

Bever, T.G., Townsend, D., 1979. Perceptual mechanisms and formal properties of main and subordinate clauses. In: Cooper, W.E., Walker, E.C.T. (Eds.), Sentence Processing: Psycholinguistic Studies Presented to Merrill Garrett. Erlbaum, Hillsdale, NJ, pp. 159–226.

Bloom, L., 1970. Language Development: Form and Function in Emerging Grammars. MIT Press, Cambridge, MA.

Bloom, P., 1990. Subjectless sentences in child language. Linguistic Inquiry 21, 491–504.

Bock, K., Levelt, W., 1995. Language production: grammatical encoding. In: Gernsbacher, M. (Ed.), Handbook of Psycholinguistics, Academic Press, New York, pp. 945–984.

Boland, J., Tanenhaus, M., 1991. The role of lexical representations in sentence processing. In: Simpson, G.E. (Ed.), Understanding Word and Sentence. Elsevier, North-Holland, pp. 331–365.

Bresnan, J., 1982. The Mental Representation of Grammatical Relations. MIT Press, Cambridge, MA.

Caplan, D., 1972. Clause boundaries and recognition latencies for words in sentences. Perception and Psychophysics 12, 73–76.

Caplan, D., Futter, C., 1986. Assignment of thematic roles by an agrammatic aphasic patient. Brain and Language 27, 117–135.

Caramazza, A., Zurif, E., 1976. Dissociation of algorithmic and heuristic processes in sentence comprehension: evidence from aphasia. Brain and Language 3, 572–582.

Carreiras, M., Clifton, C. Jr., 1993. Relative clause interpretation preferences in Spanish and English. Language and Speech 36, 353–372.

Chomsky, N., 1957. Syntactic Structures. Mouton, The Hague.

Chomsky, N., 1965. Aspects of the Theory of Syntax. MIT Press, Cambridge, MA.

Chomsky, N., 1981. Lectures on Government and Binding. Foris, Dordrecht, The Netherlands.

Chomsky, N., 1995. The Minimalist Program. MIT Press, Cambridge, MA.

Chomsky, N., Miller, G.A., 1963. Introduction to the formal analysis of natural languages. In: Luce, R.D., Bush, R.R., Galanter, E. (Eds.), Handbook of Mathematical Psychology, vol. 2. Wiley, New York, pp. 269–321.

Christiansen, M.H., 1996. Recursive sentence structure in connectionist networks. Paper presented at the ninth annual CUNY Conference on Human Sentence Processing, CUNY, March, 1996.

Christiansen, M.H., 1997. The (un)grammaticality of doubly center-embedded sentences: a connectionist perspective. Poster presented at the 10th CUNY Sentence Processing Conference, Santa Monica, CA, March, 1997.

Christiansen, M.H., Chater, N., 1998. Toward a connectionist model of recursion in human linguistic performance. Cognitive Science, in press.

Church, K.W., 1980. On Memory Limitations in Natural Language Processing. MA Thesis, MIT, Cambridge, MA.

Clifton, C., Jr., Frazier, L., 1989. Comprehending sentences with long-distance dependencies. In: Carlson, G., Tanenhaus, M. (Eds.), Linguistic Structure in Language Processing. Kluwer, Dordrecht, The Netherlands, pp. 273–317.

Clifton, C. Jr., Speer, S., Abney, S.P., 1991. Parsing arguments: phrase structure and argument structure as determinants of initial parsing decisions. Journal of Memory and Language 30, 251–271.

Cowper, E.A., 1976. Constraints on Sentence Complexity: A Model for Syntactic Processing. PhD Thesis, Brown University, Providence, RI.

Crain, S., Fodor, J.D., 1985. How can grammars help parsers? In: Dowty, D.R., Karttunen, L., Zwicky, A.M. (Eds.), Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives. Cambridge University Press, Cambridge, UK, pp. 94–128.

Crain, S., Steedman, M., 1985. On not being led up the garden path: the use of context by the psychological parser. In: Dowty, D., Karttunen, L., Zwicky, A. (Eds.), Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives. Cambridge University Press, Cambridge, UK, pp. 320–358.

Crocker, M.W., 1994. On the nature of the principle-based sentence processor. In: Clifton, C., Jr., Frazier, L., Rayner, K. (Eds.), Perspectives on Sentence Processing. Lawrence Erlbaum, Hillsdale, NJ, pp. 245–266.

Cuetos, F., Mitchell, D.C., 1988. Cross-linguistic differences in parsing: restrictions on the use of the late closure strategy in Spanish. Cognition 30, 73–105.

Daneman, M., Carpenter, P.A., 1980. Individual differences in working memory and reading. Journal of Verbal Learning and Verbal Behavior 19, 450–466.

De Roeck, A., Johnson, R., King, M., Rosner, M., Sampson, G., Varile, N., 1982. A myth about center-embedding. Lingua 58, 327–340.

De Vincenzi, M., 1991. Syntactic Parsing Strategies in Italian. Kluwer, Dordrecht, The Netherlands.

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

Dickey, M., Vonk, W., 1997. Center-embedded structures in Dutch: an on-line study. Poster presented at the 10th CUNY Sentence Processing Conference, Santa Monica, CA, March, 1997.

Elman, J.L., 1990. Finding structure in time. Cognitive Science 14, 179–211.

Elman, J.L., 1991. Distributed representations, simple recurrent networks and grammatical structure. Machine Learning 7, 195–225.

Ferreira, F., Clifton, C. Jr., 1986. The independence of syntactic processing. Journal of Memory and Language 25, 348–368.

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.

Fodor, J.D., 1978. Parsing strategies and constraints on transformations. Linguistic Inquiry 9, 427–473.

Ford, M., 1983. A method for obtaining measures of local parsing complexity throughout sentences. Journal of Verbal Learning and Verbal Behavior 22, 203–218.

Ford, M., Bresnan, J., Kaplan, R., 1982. A competence-based theory of syntactic closure. In: Bresnan, J. (Ed.), The Mental Representation of Grammatical Relations. MIT Press, Cambridge, MA, pp. 727–796.

Frank, R., 1992. Syntactic Locality and Tree-Adjoining Grammar: Grammatical, Acquisition and Processing Perspectives. PhD Thesis, University of Pennsylvania, Philadelphia, PA.

Frazier, L., 1985. Syntactic complexity. In: Dowty, D., Karttunen, L., Zwicky, A. (Eds.), Natural Language Processing: Psychological, Computational and Theoretical Perspectives. Cambridge University Press, Cambridge, UK, pp. 129–189.

Frazier, L.,1987a. Sentence processing: a tutorial review. In: Coltheart, M. (Ed.), Attention and Performance XII. Erlbaum, Hillsdale, NJ, pp. 559–585.

Frazier, L., 1987b. Syntactic processing evidence from Dutch. Natural Language and Linguistic Theory 5, 519–559.

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., Friederici, A., 1991. On deriving the properties of agrammatic comprehension. Brain and Language 40, 51–66.

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.

Frazier, L., Rayner, K., 1988. Parameterizing the language-processing system: left- vs. right-branching within and across languages. In: Hawkins, J.A. (Ed.), Explaining Language Universals. Blackwell, Oxford, UK, pp. 247–279.

Garnsey, S.M., Pearlmutter, N.J., Myers, E., Lotocky, B., 1997. The relative contributions of verb bias and plausibility to the comprehension of temporarily ambiguous sentences. Journal of Memory and Language 37, 58–93.

Garrod, S., Sanford, A.J., 1977. Interpreting anaphoric relations: the integration of semantic information while reading. Journal of Verbal Learning and Verbal Behavior 16, 77–90.
Garrod, S.C., Sanford, A.J., 1994. Resolving sentences in a discourse context: how discourse representation affects language understanding. In: Gernsbacher, M.A. (Ed.), Handbook of Psycholinguistics. Academic Press, New York, pp. 675–698.
Gibson, E., 1991. A Computational Theory of Human Linguistic Processing: Memory Limitations and Processing Breakdown. PhD Thesis, Carnegie Mellon University, Pittsburgh, PA.
Gibson, E., Hickok, G., 1993. Sentence processing with empty categories. Language and Cognitive Processes 8, 147–161.
Gibson, E., Pearlmutter, N., 1994. A corpus-based analysis of psycholinguistic constraints on PP attachment. In: Clifton, C., Jr., Frazier, L., Rayner, K., (Eds.), Perspectives on Sentence Processing. Erlbaum, Hillsdale, NJ, pp. 181–198.
Gibson, E., Thomas, J., 1996. The processing complexity of English center-embedded and self-embedded structures. In: Schütze, C. (Ed.), Proceedings of the NELS 26 Sentence Processing Workshop, MIT Occasional Papers in Linguistics 9, Cambridge, MA, pp. 45–71.
Gibson, E., Thomas, J., 1997a. Processing Load Judgements in English: Evidence for the Syntactic Prediction Locality Theory of syntactic complexity. Manuscript, MIT, Cambridge, MA.
Gibson, E., Thomas, J., 1997b. Memory limitations and structural forgetting: the perception of complex ungrammatical sentences as grammatical. Manuscript, MIT, Cambridge, MA.
Gibson, E., Ko, K., 1998. Processing main and embedded clauses, in preparation.
Gibson, E., Warren, T., 1998a. Discourse reference and syntactic complexity. Manuscript, MIT, Cambridge, MA.
Gibson, E., Warren, T., 1998b. The syntactic complexity of preposition stranding. Manuscript, MIT, Cambridge, MA.
Gibson, E., Hickok, G., Schütze, C., 1994. Processing empty categories: a parallel approach. Journal of Psycholinguistic Research 23, 381–406.
Gibson, E., Pearlmutter, N., Canseco-Gonzales, E., Hickok, G., 1996a. Recency preference in the human sentence processing mechanism. Cognition 59, 23–59.
Gibson, E., Schütze, C., Salomon, A., 1996b. The relationship between the frequency and the processing complexity of linguistic structure. Journal of Psycholinguistic Research 25, 59–92.
Gibson, E., Grodner, D., Tunstall, S., 1997. Evidence for a syntactic complexity metric in ambiguity resolution independent of lexical effects. Poster presented at the 10th CUNY Sentence Processing Conference, Santa Monica, CA, March, 1997.
Gibson, E. Babyonyshev, M., Kaan, E., 1998a. Sentence reanalysis: generalized incremental processing within a parallel processing framework. Poster presented at the 11th CUNY Sentence Processing Conference, Rutgers, NJ, March, 1998.
Gibson, E., Pearlmutter, N., Torrens, V., 1998b. Recency and lexical preferences in Spanish. Memory and Cognition, in press.
Gilboy, E., Sopena, J.M., Clifton, C. Jr., Frazier, L., 1995. Argument structure and association preferences in Spanish and English complex NPs. Cognition 54, 131–167.
Gorrell, P., 1987. Studies of Human Syntactic Processing: Ranked-Parallel versus Serial Models. PhD Thesis, University of Connecticut, Storrs, CT.
Gorrell, P., 1996. Parsing theory and phrase-order variation in German V2 clauses. Journal of Psycholinguistic Research 25, 135–156.
Grodzinsky, Y., 1989. Agrammatic comprehension of relative clauses. Brain and Language 31, 480–499.
Haarman, H.J., Kolk, H.H.J., 1991. A computer model of the temporal course of agrammatic sentence understanding: the effects of variation in severity and sentence complexity. Cognitive Science 15, 49–87.
Hakes, B., Evans, J., Brannon, L., 1976. Understanding sentences with relative clauses. Memory and Cognition 4, 283–296.
Hakuta, K., 1981. Grammatical description versus configurational arrangement in language acquisition: the case of relative clauses in Japanese. Cognition 9, 197–236.

Haliday, M.A.K., Hasan, R., 1976. Cohesion in English. Longman, London.
Haviland, S.E., Clark, H.H., 1974. What's new? Acquiring new information as a process in comprehension. Journal of Verbal Learning and Verbal Behavior 13, 512–521.
Hawkins, J.A., 1990. A parsing theory of word order universals. Linguistic Inquiry 21, 223–262.
Hawkins, J.A., 1994. A Performance Theory of Order and Constituency. Cambridge University Press, Cambridge, UK.
Heim, I., 1982. The Semantics of Definite and Indefinite Noun Phrases. PhD Thesis, University of Massachussets, Amherst, MA.
Hemforth, B., 1993. Kognitives Parsing: Repräsentation und Verarbeitung sprachlichen Wissens. Infix, Sankt Augustin.
Hemforth, B., Konieczny, L., Scheepers, C., 1998. Syntactic attachment and anaphor resolution: two sides of relative clause attachment. In: Crocker, M., Pickering, M., Clifton, C., Jr. (Eds.), Architectures and Mechanisms for Language Processing. Cambridge University Press, Cambridge, UK, in press.
Hickok, G., Zurif, E., Canseco-Gonzalez, E., 1993. Structural description of agrammatic comprehension. Brain and Language 45, 371–395.
Hoeksema, J., 1981. Verbale verstrengeling ontstrengeld. Spektator 10, 221–249.
Holmes, V., 1973. Order of main and subordinate clauses in sentence perception. Journal of Verbal Learning and Verbal Behavior 12, 285–293.
Holmes, V.M., O'Regan, J.K., 1981. Eye fixation patterns during the reading of relative clause sentences. Journal of Verbal Learning and Verbal Behavior 20, 417–430.
Hudson, R.A., 1984. Word Grammar. Blackwell, Oxford, UK.
Hudson, R.A., 1990. English Word Grammar. Blackwell, Oxford, UK.
Hudson, R.A., 1995. Measuring syntactic difficulty. Manuscript, University College, London.
Hyönä, J., Hujanen, H., 1998. Effects of case marking and word order on sentence parsing in Finnish: an eye fixation analysis. Quarterly Journal of Experimental Psychology, in press.
Jarvella, R.J., 1971. Syntactic processing of connected speech. Journal of Verbal Learning and Verbal Behavior 10, 409–416.
Jarvella, R.J., 1979. Immediate memory and discourse processing. In: Bower, G.H. (Ed.), The Psychology of Learning and Motivation, vol. 13. Academic Press, New York, pp. 379–421.
Johnson-Laird, P., 1983. Mental Models. Cambridge University Press, Cambridge, UK.
Joshi, A.K., 1990. Processing crossed and nested dependencies: an automaton perspective on the psycholinguistic results. Language and Cognitive Processes 5, 1–27.
Jurafsky, D., 1996. A probabilistic model of lexical and syntactic access and disambiguation. Cognitive Science 20, 137–194.
Just, M.A., Carpenter, P.A., 1992. A capacity theory of comprehension: individual differences in working memory. Psychological Review 99, 122–149.
Just, M.A., Carpenter, P.A., Keller, T.A., 1996. The capacity theory of comprehension: new frontiers of evidence and arguments. Psychological Review 103, 773–780.
Just, M.A., Carpenter, P.A., Keller, T.A., Eddy, W.F., Thulborn, K.R., 1996. Brain activation modulated by sentence comprehension. Science 274, 114–116.
Kaan, E., 1997. Processing Subject–Object Ambiguities in Dutch. PhD, thesis, University of Groningen.
Kac, M.B., 1981. Center-embedding revisited. In Proceedings of the Third Annual Conference of the Cognitive Science Society. Erlbaum, Hillsdale, NJ, pp. 123–124.
Kamp, H., 1981. A theory of truth and semantic representations. In: Groenendijk, J., Janssen, T.M.V., Stokhof, M. (Eds.), Formal Methods in the Study of Language. Mathematisch Centrum, Amsterdam.
Kaplan, R.N., 1974. Transient Processing Load in Relative Clauses. PhD Thesis, Harvard University, Cambridge, MA.
Kempen, G., Vosse, T., 1989. Incremental syntactic tree formation in human sentence processing: a cognitive architecture based on activation decay and simulated annealing. Connection Science 1, 273–290.
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: Kimball, J. (Ed.), Syntax and Semantics, Vol. 4. Academic Press, New York, pp. 155–179.

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

Kluender, R., 1998. On the distinction between strong and weak islands: a processing perspective. In: Culicover, P., McNally, L. (Eds.), Syntax and Semantics: The Limits of Syntax. Academic Press, New York, in press.

Kucera, H., Francis, W.N., 1967. Computational Analysis of Present Day American English, Brown University Press, Providence, RI.

Kurtzman, H., 1985. Studies in Syntactic Ambigiuty Resolution. PhD Thesis, MIT, Cambridge, MA.

Lewis, R., 1993. An Architecturally-based Theory of Human Sentence Processing. PhD Thesis, Carnegie Mellon University, Pittsburgh, PA.

Lewis, R., 1996. A theory of grammatical but unacceptable embeddings. Journal of Psycholinguistic Research 25, 93–116.

Lewis, R., 1998. Interference in working memory: retroactive and proactive interference in parsing. Paper presented at the eleventh annual CUNY Sentence Processing Conference, Rutgers University, New Brunswick, NJ, March, 1998.

MacDonald, M.C., 1993. The interaction of lexical and syntactic ambiguity. Journal of Memory and Language 32, 692–715.

MacDonald, M.C., 1997. Lexical representations and sentence processing: An introduction. Language and Cognitive Processes 12, 121–136.

MacDonald, M., Pearlmutter, N., Seidenberg, M., 1994. The lexical nature of syntactic ambiguity resolution. Psychological Review 101, 676–703.

MacWhinney, B., 1977. Starting points. Language 53, 152–168.

MacWhinney, B., 1982. Basic syntactic processes. In: Kuczaj, S. (Ed.), Language Acquisition: Volume 1, Syntax and Semantics. Erlbaum, Hillsdale, NJ.

MacWhinney, B., 1987. The competition model. In: MacWhinney, B. (Ed.), Mechanisms of Language Acquisition. Erlbaum, Hillsdale, NJ, pp. 249–308.

MacWhinney, B., Pleh, C., 1988. The processing of restrictive relative clauses in Hungarian. Cognition 29, 95–141.

Marcus, M.P., Santorini, B., Marcinkiewicz, M.A., 1993. Building a large annotated corpus of English: the Penn treebank. Computational Linguistics 19, 313–330.

Marslen-Wilson, W., Tyler, L.K., Seidenberg, M., 1978. Sentence processing and the clause boundary. In: Levelt, W.J.M., Flores d'Arcais, G.B. (Eds.), Studies in the Perception of Language. Wiley, New York, pp. 219–246.

McClelland, J.L., St. John, M., Taraban, R., 1989. Sentence comprehension: a parallel distributed processing approach. Language and Cognitive Processes 4, SI287–335.

McRae, K., Spivey-Knowlton, M.J., Tanenhaus, M.K., 1997. Modeling the influence of thematic fit (and other constraints) in on-line sentence comprehension. Paper presented at the 10th CUNY Sentence Processing Conference, Santa Monica, CA, March, 1997.

Mecklinger, K., Schriefers, H., Steinhauer, K., Friederici, A., 1995. Processing relative clauses varying on syntactic and semantic dimensions. Memory and Cognition 23, 477–494.

Meng, M., 1995. Processing wh-questions in German and Dutch: differential effects of disambiguation and their interpretation. Poster presented at the Architectures and Models for Language Processing Conference, Edinburgh University, Edinburgh, UK.

Miikkulainen, R., 1996. Subsymbolic case-role analysis of sentences with embedded clauses. Cognitive Science 20, 47–73.

Miller, G.A., Chomsky, N., 1963. Finitary models of language users. In: Luce, R.D., Bush, R.R., Galanter, E. (Eds.), Handbook of Mathematical Psychology, volume 2. Wiley, New York, pp. 419–491.

Miller, G.A., Isard, S., 1964. Free recall of self-embedded English sentences. Information and Control 7, 292–303.

Milsark, G.L., 1983. On length and structure in sentence parsing. Cognition 13, 129–134.

Mitchell, D.C., 1994. Sentence parsing. In: Gernsbacher, M.A. (Ed.), Handbook of Psycholinguistics. Academic Press, New York, pp. 375–409.

Mitchell, D.C., Cuetos, F., 1991. The origins of parsing strategies. Conference Proceedings: Current Issues in Natural Language Processing. University of Texas at Austin, Austin, TX, pp. 1–12.
Miyake, A., Carpenter, P.A., Just, M.A., 1994. A capacity approach to syntactic comprehension disorders: making normal adults perform like aphasic patients. Cognitive Neuropsychology 11, 671–717.
Ni, W., Crain, S., Shankweiler, D., 1996. Sidestepping garden-paths: assessing the contributions of syntax, semantics and plausibility in resolving ambiguities. Language and Cognitive Processes 11, 283–334.
Pearlmutter, N.J., MacDonald, M.C., 1992. Plausibility and syntactic ambiguity resolution. Proceedings of the 14th Conference of the Cognitive Science Society. Hillsdale, NJ: Erlbaum, pp. 498–503.
Pickering, M., Barry, G., 1991. Sentence processing without empty categories. Language and Cognitive Processes 6, 229–259.
Pollard, C., Sag, I.A., 1994. Head-Driven Phrase Structure Grammar. University of Chicago Press, Chicago.
Quirk, R. Greenbaum, S., Leech, G., Svartvik, J., 1972. A grammar of contemporary English. Longman, London.
Rambow, O., Joshi, A., 1994. A processing model for free word-order languages. In: Clifton, C., Jr., Frazier, L., Rayner, K. (Eds.), Perspectives on Sentence Processing. Erlbaum, Hillsdale, NJ, pp. 267–301.
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.
Roberts, R., Hooker, C., Gibson, E., 1997. Sentence memory capacity and working memory. Poster presented at the 10th CUNY Sentence Processing Conference, Santa Monica, CA, March, 1997.
Schlesewsky, M., Fanselow, G., Kliegl, R., 1997. The costs of wh-movement in German. Manuscript, University of Potsdam, Potsdam, Germany.
Schlesewsky, M., Fanselow, G., Kliegl, R., Krems, J., 1996. Preferences for grammatical functions in the processing of locally ambiguous wh-questions in German. Manuscript, University of Potsdam, Potsdam, Germany.
Schlesinger, I.M., 1968. Sentence Structure and the Reading Process. Mouton, The Hague.
Schriefers, H., Friederici, A., Kühn, K., 1995. The processing of locally ambiguous relative clauses in German. Journal of Memory and Language 34, 499–520.
Spivey-Knowlton, M., Sedivy, J., 1995. Resolving attachment ambiguities with multiple constraints. Cognition 55, 227–267.
Spivey-Knowlton, M.J., Tanenhaus, M.K., 1996. Syntactic ambiguity resolution in discourse: modeling the effects of referential context and lexical frequency within an integration-competition framework. Paper presented at the 9th CUNY Sentence Processing Conference, March, 1996.
Stabler, E.P., 1994. The finite connectivity of linguistic structures. In: Clifton, C., Jr., Frazier, L., Rayner, K. (Eds.), Perspectives on Sentence Processing. Erlbaum, Hillsdale, NJ, pp. 303–336.
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. Hillsdale, NJ: Erlbaum.
Stevenson, S., 1994. Competition and recency in a hybrid network model of syntactic disambiguation. Journal of Psycholinguistic Research 23, 295–322.
Stolz, W.S., 1967. A study of the ability to decode grammatically novel sentences. Journal of Verbal Learning and Verbal Behavior 6, 867–873.
Stowe, L., 1986. Evidence for on-line gap location. Language and Cognitive Processes 1, 227–245.
Stromswold, K., Caplan, D., Alpert, N., Rauch, S., 1996. Localization of syntactic comprehension by Positron Emission Tomography. Brain and Language 52, 452–473.
Tabor, W., Juliano, C., Tanenhaus, M.K., 1997. Parsing in a dynamical system: an attractor-based account of the interaction of lexical and structural constraints in sentence processing. Language and Cognitive Processes 12, 211–272.
Taraban, R., McClelland, J.R., 1988. Constituent attachment and thematic role assignment in sentence

processing: influences of content-based expectations. Journal of Memory and Language 27, 597–632.

Traxler, M.J., Pickering, M.J., 1996. Plausibility and the processing of unbounded dependencies: an eye-tracking study. Journal of Memory and Language 35, 454–475.

Trueswell, J.C., 1996. The role of lexical frequency in syntactic ambiguity resolution. Journal of Memory and Language 35, 566–585.

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.

Tyler, L.K., Marslen-Wilson, W.D., 1977. The on-line effects of semantic context on syntactic processing. Journal of Verbal Learning and Verbal Behavior 16, 683–692.

Valian, V., 1990. Null subjects: a problem for parameter-setting models of language acquisition. Cognition 35, 105–122.

Wanner, E., Maratsos, M., 1978. An ATN approach in comprehension. In: Halle, M., Bresnan, J., Miller, G. (Eds.), Linguistic Theory and Psychological Reality. MIT Press, Cambridge, MA, pp. 119–161.

Wasow, T., 1997. Remarks on grammatical weight. Language Variation and Change 9, 81–105.

Waters, G.S., Caplan, D., 1996. The capacity theory of sentence comprehension: a critique of Just and Carpenter (1992). Psychological Review 103, 761–772.

Waters, G.S., Caplan, D., Hildebrandt, N., 1987. Working memory and written sentence comprehension. In: Coltheart, M. (Ed.), Attention and Performance: Volume 12, the Psychology of Reading. Erlbaum, Hillsdale, NJ, pp. 531–555.

Waugh, N.C., Norman, D.A., 1965. Primary memory. Psychological Review 72, 89–104.

Webber, B.L., 1988. Tense as discourse anaphor. Computational Linguistics 14, 61–73.

Weckerly, J., Elman, J.L., 1992. A PDP approach to processing center-embedded sentences. In: Proceedings of the 14th annual Conference of the Cognitive Science Society. Erlbaum, Hillsdale, NJ, pp. 414–419.

Yngve, V.H., 1960. A model and an hypothesis for language structure. Proceedings of the American Philosophical Society 104, 444–466.