This is a preprint of an article whose final and definitive form has been published as
Boston, Marisa Ferrara , Hale, John T. , Vasishth, Shravan and Kliegl, Reinhold(2011)
'Parallel processing and sentence comprehension difficulty', Language and Cognitive Processes, 26: 3, 301 — 349,
First published on: 17 January 2011 (iFirst)
To link to this Article: DOI: 10.1080/01690965.2010.492228
URL: http://dx.doi.org/10.1080/01690965.2010.492228

Parallel processing and sentence comprehension difficulty

Marisa Ferrara Boston
Cornell University

John T. Hale
Cornell University

Shravan Vasishth
University of Potsdam

Reinhold Kliegl
University of Potsdam

John T. Hale
Department of Linguistics
Cornell University
Morrill Hall room 217
Ithaca, New York 14853-4701
E-mail: jthale@cornell.edu
Telephone: (814) 880-4173
Fax: 607-255-2044

Abstract

Eye fixation durations during normal reading correlate with processing difficulty but the specific cognitive mechanisms reflected in these measures are not well understood. This study finds support in German readers' eye-fixations for two distinct difficulty metrics: surprisal, which reflects the change in probabilities across syntactic analyses as new words are integrated, and retrieval, which quantifies comprehension difficulty in terms of working memory constraints. We examine the predictions of both metrics using a family of dependency parsers indexed by an upper limit on the number of candidate syntactic analyses they retain at successive words. Surprisal models all fixation measures and regression probability. By contrast, retrieval does not model any measure in serial processing. As more candidate analyses are considered in parallel at each word, retrieval can account for the same measures as surprisal. This pattern suggests an important role for ranked parallelism in theories of sentence comprehension.

The authors thank Sabrina Gerth, Roger Levy, Richard Lewis, Joakim Nivre, and Mats Rooth for comments and suggestions on this work. This research was supported by the National Science Foundation Career Award 0741666 to John Hale, and the Deutsche Forschungsgemeinschaft (German Science Foundation) project "Computational models of human sentence processing: A model comparison approach" awarded to Shravan Vasishth and Reinhold Kliegl (2008-2010).

Introduction

What cognitive mechanisms are reflected in sentence comprehension difficulty? This has been a central question in psycholinguistic research, and the literature acknowledges two broad categories of answer. One kind of answer is predicated on resource limitations in the human sentence processing mechanism (Miller & Chomsky, 1963; Clifton & Frazier, 1989; Gibson, 1991; Lewis & Vasishth, 2005). The other kind of answer appeals to misplaced expectations or predictions as an analysis is built (Elman, 1990; Mitchell, 1995; Jurafsky, 1996; Hale, 2001). Some recent work suggests that these two different kinds of answer may in fact explain distinct aspects of human sentence processing (Demberg & Keller, 2008; Levy, 2008). If correct, any such two-factor explanation immediately leads to the question: what are the relative contributions of the two factors, and how do they interact with common resources like memory? In this paper, we furnish an answer to this question.

Standardizing on one probabilistic parsing method, we work out the predictions of both surprisal (Hale, 2001) and cue-based retrieval (Lewis & Vasishth, 2005). We use both metrics to quantify comprehension difficulty across a family of psycholinguistic models that differ only in the number of syntactic analyses they explore in parallel (Lewis, 2000; Gibson & Pearlmutter, 2000). These theoretical predictions are evaluated at a range of parallel processing levels against fixation durations collected in a German eyetracking database, the Potsdam Sentence Corpus (PSC) (Kliegl, Grabner, Rolfs, & Engbert, 2004). This corpus consists of 144 sentences[1] with fixation duration data from 222 readers.

The surprisal predictions that we derive account for eye fixation measures at all levels of parallel processing. The retrieval predictions that we derive account for all measures as well, but only when multiple syntactic analyses are considered in parallel: retrieval in a serial parser does not predict the data. This pattern of results draws out the point that a fully-specified model of human parsing assumes not only a grammar and an attachment strategy, but also some number of ranks available for parallel processing

Before describing the parsing model itself, we first discuss the overall methodology in the context of prior work. Subsequent sections sketch out our particular implementation of the surprisal and retrieval difficulty metrics within one shared parsing mechanism. A fuller, more technical presentation is provided in the Appendix. The main text goes on to present the results of the evaluation of the theories' predictions as regards the Potsdam Sentence Corpus. The paper concludes with some remarks on the implications of our findings.

Methodology & Background

Towards a more accurate picture of the cognitive mechanisms operative in sentence comprehension, this work poses the following research question: how helpful are notions of syntactic surprise and memory retrieval latency in accounting for German readers' eye fixations? To address this question we pursue a corpus study methodology. While the corpus remains constant, we examine several different candidate explanations for the observations collected in it. The candidates formalize certain leading ideas about sentence comprehension into a systematic collection of parsing-difficulty theories. For each theory, we use regression to estimate the best-fitting relationship between each theory's derived predictions and the actual PSC observations. From the overall pattern of good and bad fits, we infer a relationship between dimensions along which the candidate theories vary. Table 1 lays this methodology out graphically. This table highlights the research objective: to compare theories that use the same parser but vary either the complexity metric (horizontal dimension) or the degree of parallelism (vertical dimension).

[1] The PSC sentences exhibit a range of syntactic phenomena reflecting everyday language rather than tricky cases such as garden path sentences.

		German parser		
		with surprisal	with retrieval	with surprisal and retrieval
Amount of Parallel Processing	serial (1)	?	?	?
	ranked parallel (5)	?	?	?
	ranked parallel (10)	?	?	?
	ranked parallel (15)	?	?	?
	ranked parallel (20)	?	?	?
	ranked parallel (25)	?	?	?
	ranked parallel (100)	?	?	?

Table 1: The methodology adopted in the present research: holding the parser and empirical dataset constant, we systematically vary either the complexity metric (horizontal dimension) or the degree of parallelism (vertical dimension). The numbers in parentheses mark the degree of parallelism. We compute the relative quality of fit for each model against the empirical dataset to determine how well the model accounts for eye fixation measures while reading.

The question-marks are quantities that we calculate in the course of this study. They quantify the degree-of-fit for multiple linear regression models that include several other factors known to influence eye-movements. These degree-of-fit measures, according to a particular candidate theory, can reveal telling patterns. They index how well a multiple linear regression model accounts for the same collection of observed eye-movement measures (e.g. single-fixation durations) when paired with specific predictor variables (e.g. theorized surprisal) at a given level of parallelism in parsing. The sources of these surprisal and retrieval predictions are defined in sufficient computational detail that the origins of any differences are clear. Our agenda is to juxtapose a broad range of PSC eye-movement measures against each cell's corresponding theory in an effort to discern which predictors are truly consequential in an explanation of German reading difficulty that involves syntactic parsing.

Unlike studies that consider just one parsing mechanism or just one complexity metric, the methodology we apply in this paper offers the chance to uncover relationships between controversial alternative accounts of the same data. It has the potential to find asymmetries that would not come out in a meta-analysis of surprisal or retrieval studies that themselves employ incomparable parsing mechanisms. This methodology also has the potential to find interactions between different aspects of the same comprehension theory. One of these aspects is memory capacity for parallel processing. Because of the centrality of this background concept, the next two subsections review parallel processing and some of its implications for comprehension models. Remaining subsections take up the grammar and the parsing strategy, respectively.

The idea of parallel processing in human sentence comprehension

In this study, we examine the relative adequacy of alternative theories of human sentence processing that each assume different levels of parallel processing. But what do we mean by parallel processing? The basic idea — that people can, on some level, do more than one cognitive operation at a time while understanding sentences — is an old one. Fodor, Bever, and Garrett (1974) lay out two poles of opposition.

> There are, patently, two broad theoretical options. On the one hand, one might imagine that the perceptual system is a parallel processor in the sense that given a portion of a sentence which has n possible linguistic structures,

> each of the n structures is computed and "carried" in short-term storage. If a disambiguating item is encountered, all but one of the n analyses are rejected, with the residual analysis being the one which is stored. If no disambiguating material is encountered, all n analyses are retained, and the sentence is represented as ambiguous in n ways.
>
> Alternatively, one might suppose that the system is a serial processor in the sense that given a portion of a sentence which has n possible linguistic structures, only one of the n structures is computed. This structure is accepted as the correct analysis unless disambiguating material incompatible with it is encountered. If such material *is* encountered, then the processor must go back to the ambiguous material and compute a different analysis.
>
> There are obviously a variety of modifications and blends of these two proposals that one might consider.
>
> page 362

At the time that Fodor and colleagues were writing, parallel processing was under study. Lackner and Garrett (1973) found support for parallelism in a dichotic listening task, and Cowper's (1976) model availed itself of up to three "tracks" in accounting for performance phenomena across several languages.

But it was the serial parsing idea that was prominently realized in computational cognitive models during the 1970s. For instance, Kaplan (1972) used Augmented Transition Networks (ATNs) to deduce a variety of detailed predictions about relative clauses which were upheld in experimental work by himself and others (Wanner & Maratsos, 1978). The ATNs Kaplan considered are serial processors: they pursue one linguistic structure at a time and backtrack when they reach an impasse. By the 1980s researchers like Kurtzman (1984) and Gorrell (1989) began to reconsider parsing models with parallel processing as an alternative to other serial-processing proposals like the Sausage Machine (Frazier & Fodor, 1978) and Parsifal (Marcus, 1980). Gibson (1991) developed the idea of "ranked parallel" approaches. It became de rigueur during the 1990s to interpret new experimental results from the vantage point of ranked parallel as well as serial processing.

Around this same time, parallel processing also surfaced as an essential element in many connectionist models of language processing (McClelland & Kawamoto, 1986; McClelland, St. John, & Taraban, 1989; MacDonald, Pearlmutter, & Seidenberg, 1994). This is natural, because parallel processing is a fundamental assumption of connectionist modeling. Spivey and Tanenhaus (1998), for instance, embrace parallel processing in a passage where they characterize the processing models for which they intend to offer an explicit connectionist implementation.

> Specific models differ in their details, but as a class they share two common features: (a) multiple constraints are combined to compute alternative interpretations in parallel and (b) the alternatives compete with one another during processing.
>
> page 1522

By the end of the 1990s, members of the scientific community could be divided according to their stance on the degree to which they believed parallel processing operates in human sentence processing. This was widely acknowledged as a fundamental issue in the cognitive science of language. A variety of experimental studies grappled with it, but none proved decisive. Lack of progress on this question was perhaps unsurprising, given that

the sentence comprehension theories widely known in psycholinguistics at the time were not well-formalized.

The idea of parallelism in broad-coverage parsing

With the advent of statistical natural language processing (NLP) in the 1990s, another way of addressing the serial/parallel question started to emerge. Following Jurafsky's (1996, §2.1) declaration that "the underlying architecture of the human language interpretation mechanism is parallel" researchers like Roark and Johnson (1999) began to adapt probabilistic parser designs from natural language engineering to serve as cognitive models of human comprehension. These designs typically used parallel processing for efficiency. But T. Brants and Crocker (2000) found that, with the right ranking factors, parallel processing was nearly unnecessary — the same level of parsing performance could be maintained at a degree of parallelism so low as to be almost serial. Indeed, Lewis (1993) demonstrated that a fundamentally serial cognitive architecture like SOAR (Rosenbloom, Laird, & Newell, 1993) was in fact compatible with a large subset of the available comprehension question. The theoretical innovations of the 1990s allowed for a re-consideration of the same parallelism question on a larger scale.

Broad-coverage approaches borrowed from NLP the imperative to cover a wide range of language structures. Researchers such as Crocker and Brants (2000) enjoined psycholinguists to consider garden-variety language, not just garden-path sentences. This new generation of sentence processing models abandoned hand-crafted grammars in favor of parsing methods that could be relied upon to work with large corpora. Under these conditions Fodor, Bever and Garrett's (1974) notion of syntactically "incompatible" material becomes useless: because probabilistic parsers must be robust, nothing can be ruled out for certain. Instead, the relative ranking of their "n possible linguistic structures" takes center stage.

Broad-coverage models of human sentence comprehension typically impose fine-grained preferences on the analyses that they consider in parallel, preferences that go beyod the pursue-vs-abandon distinction. It is standard to codify these preferences using probability. Despite widespread availability of these techniques, the implications of parallel processing for sentence comprehension theories have so far not been studied in a way that is simultaneously empirical and computationally explicit.We strive to do exactly that in this research, building on assumptions about grammar and parsing that are layed out in the next two subsections.

Dependency grammar

Even older than the question "how many alternative readings does the parser entertain?" is the question "what is the right theory of sentence structure?" The profusion of syntactic theories is a familiar problem for sentence-processing researchers. It often seems that when a grammar-based hypothesis fails to find support in the results of a behavioral experiment, its proponents quickly offer a modified version. These proponents suggest that their new proposal is the same in spirit, yet compatible in detail with whatever results cast doubt on the old theory. The really central claims of a particular syntactic theory, as regards sentence comprehension, can be difficult to pin down.

To cope with this problem, in this research we step back and work with a simplified grammar formalism called dependency grammar that stands in for a consensus view of syntax. Dependency grammar as a distinct linguistic tradition traces its intellectual lineage back to Tesnière (1959) and Hays (1964), and continues to develop in more recent work such as Mel'čuk (1988) and Hudson (2007). Its foundational concept — that words

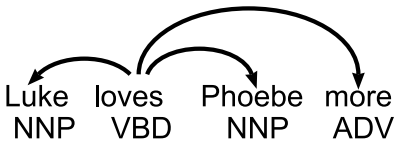


Figure 1. A dependency graph identifies heads and dependents in a sentence.

depend on other words — is adopted, explicitly or implicitly, in virtually every modern syntactic theory. An example structural description in dependency grammar is given in Figure 1.

The arcs in Figure 1 emanate from words in the role of *head* to other words that are said to be their *dependents*. The symbols underneath the words are part-of-speech tags: NNP stands for proper noun, VBD for verb and ADV for adverb. Heads are said to "govern" their dependents. In typical dependency grammars, a head may have multiple dependents but not the other way around.

This kind of asymmetric, word-to-word relationship figures in many well-known approaches to grammar. For instance, in Head-Driven Phrase Structure Grammar a word's ARG-ST list constrains its possible dependents (Pollard & Sag, 1994). In Minimalist Grammars based on the notion of Bare Phrase Structure (Chomsky, 1995), words bear selection features that cause them to enter dependency relationships in the course of a derivation. In Tree Adjoining Grammars, words figure in elementary trees whose substitution nodes set up an asymmetric relationship with other words in other elementary trees (Kroch & Joshi, 1985; R. Frank, 2002).

Each of these approaches brings its own inventory of additional concepts and notation to the structural analysis of sentences. For instance, Figure 1 could be enriched by noting that the **Phoebe**'s dependency on the word **loves** is a case of direct-objecthood, whereas the dependency with **more** is a kind of modification. In the hope that our results might speak to a common core of attachment decisions implicated by a wide variety of syntactic theories, we avoid decorating our dependency arcs with additional labels in this work. Such an enrichment would be a natural follow-up project, however, for those seeking to tease apart the perceptual implications of different syntactic theories.

Incremental parsing

The final swath of background against which the research reported in subsequent sections is set has to do with the design space of incremental parsers. While the task of a parser is simply to recover structural descriptions from sequences of words, an *incremental* parser is additionally subject to the requirement that it work through its input words from left to right, just as a human would. In this work we adopt the straightforward view of incremental parsing as a process of repeatedly adding[2] to partial structural descriptions of sentence-initial fragments (Marcus, Hindle, & Fleck, 1983; Barton & Berwick, 1985; Weinberg, 1993).

The essential point is that ambiguity sets up difficult decisions about what to add. We adopt the perspective illustrated in Figure 2 according to which particular ways of extending a dependency graph over a sequence of words are bona fide *operators* in the sense of Newell and Simon (1972). In Figure 2 the operator names are written inside

[2]This monotonic view fits a large class of parsing algorithms, but excludes "repair" or reanalysis operations like snip or tree lowering that change sentence structures in ways other than adding to them (Lewis, 1993; Sturt & Crocker, 1996; Buch-Kromann, 2001).

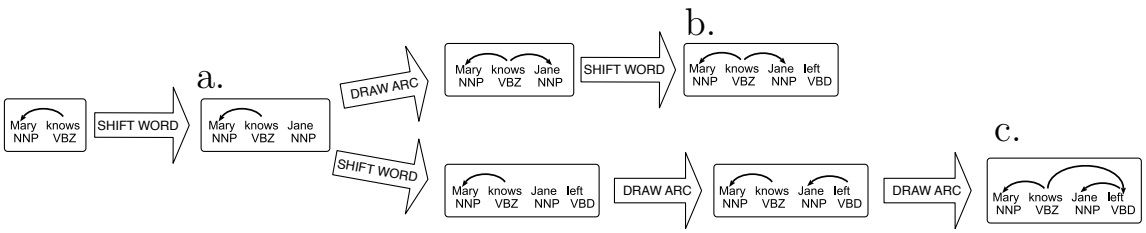


Figure 2. Partial syntactic analyses occupy states in the problem space.

the large rightward-pointing arrows. They are actions in a problem space that take an incremental parser into a new state. In this overview of conceptual background, we leave the notion of state intentionally vague. We note that, at a minimum, states must include information about the dependency arcs that have been drawn by earlier operators.

As in other cognitive problem spaces, an incremental parser may not be able to determine locally which operator leads to a successful analysis. This is the familiar notion of garden-pathing, where local decisions lead to errors in the global parse. Parallel processing represents a way to hedge bets about which pathway will ultimately work out. If enough memory is available, all states can be retained, and exhaustive search can be relied upon to find the best path. On the other hand, if only a limited amount of memory is available, difficult choices need to be made about which states are kept[3]. A parallel parser exploring the space depicted in Figure 2 might be said to be in the *disjunction* of states (b) or (c) at the word **left**. With broad-coverage grammars, these disjunctions can get very long very quickly. Long sentences can lead to states with many possible successors if a parser has to consider many words as potential attachment sites. This happens because, absent significant constraint by the grammar, the problem space's *branching factor* can be very high. Figure 2's miniature problem space has a branching factor of two: from the state marked (a), two different successor states can be reached depending on whether the DRAW-ARC operator or the SHIFT-WORD operator is chosen.

A major research goal within computational psycholinguistics is to find a human-like parse ranking scheme that would push the scores of reasonable and ridiculous analyses far apart. In this work we adopt a simple regime where the score of a state is the product of the conditional probabilities of all the *operator-applications*, or transitions between states, that it took to arrive at that state. These probabilities are estimated based on simulated "experiences" reading German newspaper text (S. Brants et al., 2004). The probability model itself reflects the truism that people's sentence comprehension is largely determined by words they have already heard, as opposed to those they have yet to hear.

The simple model that we use in this paper is neither complete in the sense of being guaranteed to visit all states, nor is it a language model in the technical sense of defining a distribution on an infinite class of word-sequences. We impose these limitations in an effort to define a more psychologically-realistic hypothesis. Curtailing the idealizations inherent in earlier work (Hale, 2001) we view the computational limits of this parser as bounds on rationality (Simon, 1955; Gigerenzer et. al, 1999). By focusing on the parsing process as opposed to the definition of the language, and by introducing a parametic memory limit, we attempt to move towards a model that corresponds to the way human parsing does work — rather than the way it ought to work.

[3] This picture is, in practice, complicated by the fact that significant economies can be gained by sharing the representation of parser states that overlap in some way. In this paper we do not address the cognitive implications of structure-sharing.

Having outlined the methodology and a few points of background, the next section sketches out this particular study's implementation of the surprisal and retrieval complexity metrics in a common parsing mechanism.

A systematic collection of comprehension-difficulty theories

As indicated in Table 1 on page 3, our goal is to compare the predictions made by alternative theories of sentence comprehension difficulty against the fixed set of observations collected in the PSC. To ensure that these comparisons are interpretable we consider just one basic parsing mechanism. We vary both the amount of memory available for parallel processing as well as the way in which we interpret the parser's internal states as making psychological difficulty predictions. This interpretaton is known as the *complexity metric*. A complexity metric is not itself a theory of sentence processing, but rather an auxiliary hypothesis that, in combination with a parsing mechanism, can be used to draw difficulty predictions for given sentences. The parsing mechanism outlines how the sentence is understood; the complexity metric identifies which parts of this process are cognitively taxing. In this section, we describe a parsing mechanism and two complexity metrics. We refer to the same German example throughout to show how the difficulty predictions derive from each metric.

A parallel-processing variant of Nivre's dependency parser

The incremental parsing mechanism applied in this paper uses operators defined by Nivre (2004). Rather than just one DRAW-ARC action, Nivre distinguishes between operators that draw left-pointing arcs versus operators that draw right-pointing arcs. Building on the work of Covington (2001), Nivre also includes an operator called REDUCE. This operator allows the parser to rule out words as potential attachment sites in the sentence prefix that has already been seen. REDUCE can lessen the branching factor in problem sub-spaces by eliminating attachment possibilities. The dual of REDUCE is SHIFT, which brings new words under consideration for eventual casting in the role of head or dependent. These four operators are summarized informally in Table 2. Appendix A provides additional details.

LEFT	the next word becomes the governor of the closest attachment site
RIGHT	the next word becomes a dependent of the closest attachment site and becomes itself a potential attachment site
SHIFT	the next word becomes a potential attachment site; no arcs are drawn
REDUCE	rule out closest attachment site

Table 2: Informal description of Nivre parser actions

Our implementation deploys these operators in a probabilistic, incremental dependency parser for part-of-speech tag sequences[4]. We impose no categorical matching requirements on the system. By contrast, in an explicitly grammar-based parser, if a dependency arc fails to be licensed by a rule, then states incorporating that arc are not considered. In our probabilistic Nivre implementation, all conceivable attachments are possible and we leave it to the system of ranking preferences to determine which is best.

[4]S. Frank (2009) and Demberg and Keller (2008) call their work at this level of analysis "unlexicalized surprisal.".

To make the collection of Nivre operators into a parallel parser, we apply a standard technique from artificial intelligence (AI) called local beam search (Russell & Norvig, 2003, 115). The "beam" in local beam search names the collection of states that are still in play at any given word. Local beam search subsumes the idea of ranked parallel parsing in the sense that it is defined on *states* rather than analyses. States incorporate more information than just a partial dependency graph. For instance, states may include, as a result of REDUCE, information disqualifying certain words from ever serving as attachment points.

The maximum number of states that a local beam search procedure can handle is denoted k. Since each state contains just one (partial) dependency analysis, this number is the same as what Fodor et al. (1974) refer to as n. In that follows, we stick to the AI notation to avoid confusion with the standard psychological notion of number-of-participants. For local beam search to have beam-width k means that, out of all the successor states reachable by any operator from any parent state in the previous beam, only the top k best states will survive in the next iteration of search. This use of a fixed parameter to control parallelism contrasts with that of Roark (2004,?). Roark uses a self-adjusting beam defined in terms of the probability model. Under this arrangement, the degree of parallel processing can go up or down depending on how focused the ranking preferences are in a particular state. Our method permits the sentence processing theorist to vary one without disturbing the other.

Perhaps the clearest way to think about parallel processing in this context is to recognize the beam of alternative states as a disjunction. This disjunction is a kind of macrostate; from word to word, the incremental parser occupies a particular macrostate composed of up to k microstates. Each new iteration of local beam search considers all possible successor states attainable via the Nivre operators in Table 2. But only the highest-scoring k states out of these candidate successors are included in the next macrostate. The theoretical challenge is to somehow aggregate aspects of the microstates into difficulty predictions that faithfully index the computational work going on in the transition from macrostate to macrostate. The following subsections describe two different ways of doing this.

Surprisal

Attneave (1959, 6) introduced the term "surprisal" to cognitive science as part of a wave of interest in information-processing and information theory during the late 1950s and early 1960s. A surprisal is the logarithm of the reciprocal of a probability. By an elementary law of logarithms this is equivalent to the negative-log of the probability itself. This mathematical definition codifies the commonsense idea that low-probability events are surprising. The logarithmic aspect of the formulation follows Hartley (1928). Hale (2001) revived surprisal as part of a way to predict sentence processing difficulty at a word. He used Stolcke's Earley algorithm to work out the total probability of all parser states reachable before a word, denoted α_{i-1} as well as after the word, denoted α_i (Earley, 1970; Stolcke, 1995). The ratio of these two values is the transition probability at the i^{th} word of a sentence. The negative logarithm — the surprisal — of this transition probability indexes how "surprising" the transition itself is, compared to other transitions a parser might have been forced to go through. This method makes it possible, at least with small grammars, to calculate the total probability of all reachable parser states at intermediate points within a sentence[5]. We shall refer to this as the "ideal" prefix probability because it reflects all possible ways of parsing the first i words or the *prefix* of length i. The summation implied by the phrase "total probability" is explicitly written-out below in Definition 1.

[5]The feasibility of calculating ideal prefix probabilites is underwritten by the fact that the Earley parser is a chart parser that aggressively shares the substructure of derivations. See footnote 3.

$$\alpha_i \quad = \quad \sum_{\substack{t \text{ is a sequence of parser operations} \\ \text{that successfully analyzes up to position } i}} \text{Prob}(t) \tag{1}$$

The notation Prob denotes the value assigned to a parser state. In this paper, Prob(t) is the product of the probabilities of all the operators in the sequence t. Given a definition of α, the surprisal associated with a transition between positions $i-1$ and i would then be as in Definition 2.

$$\text{surprisal}(i) \quad = \quad -\log_2 \left(\frac{\alpha_i}{\alpha_{i-1}} \right) \tag{2}$$

Any modeling based on Definition 2 proceeds from the idealization that all possible syntactic analyses of the prefix string may influence difficulty at the next word. The definition expresses this idealization by the variable t being universally quantified: the summation is over all sequences. This assumption only seems reasonable as long as one is looking at a handful of natural language sentences. From the perspective of broad-coverage grammars for everyday language, the number of possible analyses becomes truly daunting. Most of these are linguistically implausible readings that no human would ever consider as part of a natural interpretation of an initial sentence fragment. To our taste, detailed consideration of all of these improbable analyses suggests a kind of omnipotence upon which modern psycholinguistics has, at the very least, cast some doubt. In the spirit of bounded rationality, we instead use Definition 3 in this paper as a more realistic substitute.

$$\alpha_i^k \quad = \quad \sum_{\substack{t \text{ is a sequence of parser operations} \\ \text{that successfully analyzes up to position } i \\ \text{and arrives at one of the top } k \text{ states}}} \text{Prob}(t) \tag{3}$$

The superscripted k in α_i^k is exactly as described above in the explanation of local beam search. Surprisal, when calculated using α^k as defined in 3 for some parallelism level k reflects just those parser states that are actually visited by the local beam search procedure. The variable t ranges over just those sequences of operations that managed to remain in the top k the entire way out to position i. Surprisal at a word is still defined on macrostates as in Hale (2001) but in this bounded-rationality formulation, there is a limit (k) on the number of microstates that a macrostate may contain. One might identify the quantity $-\log\left(\alpha_i^k/\alpha_{i-1}^k\right)$ as "realistic surprisal." It is a surprisal, but it is a surprisal derived from a less-idealized parsing model that incorporates local beam search.

Surprisal: a worked example.

To see how surprisal works, consider the words **goss** and **Kapitaen** in Figure 3. As indicated in the thermometers below the words, surprisal is lower at the noun **Kapitaen** than it is at the verb **goss**.

On a gross level, this correctly reflects a statistical property[6] of a German corpus where the probability of a noun following an adjective exceeds the probability of a verb following a noun. However, the surprisal values that we compute in this paper are consequences of the parser operations used to build the relevant syntactic analysis. Table 3 shows these operations.

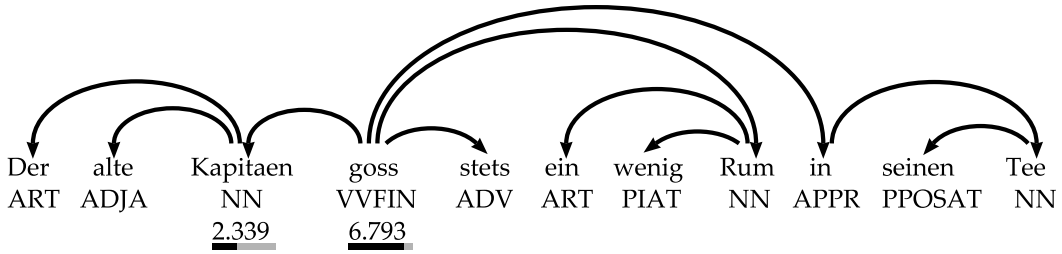


Figure 3. Surprisal is a word-by-word complexity metric.

1. `Shift` ART
2. `Shift` ADJA
3. `Left` NN ← ADJA
4. `Left` NN ← ART
5. `Shift` NN
6. `Left` VVFIN ← NN
7. `Shift` VVFIN
8. `Right` VVFIN → ADV
9. `Shift` ART
10. `Shift` PIAT
11. `Left` NN ← PIAT
12. `Left` NN ← ART
13. `Reduce` ADV
14. `Right` VVFIN → NN
15. `Reduce` NN
16. `Right` VVFIN → APPR
17. `Shift` PPOSAT
18. `Left` NN ← PPOSAT
19. `Right` APPR → NN
20. `Reduce` NN
21. `Reduce` APPR
22. `Reduce` VVFIN

Table 3: Parser state trajectory for the sentence depicted in Figure 3.

Figure 4 depicts the numerical ingredients of this "realistic" surprisal complexity metric in a $k = 3$ parser operating on our running example sentence from Figure 3. The boxes indicate the states the parser visits, with states vertically ordered by their probability. The input sentence is laid out horizontally across the page, with the grey lines signifying transitions between words.

The numbers inside the state boxes indicate the precise height of the box; they are sometimes called *forward* probabilities. These forward probabilities do no more than record the product of operator-application probabilities on paths that lead to this state. These operator-application probabilities are given as numerical annotations on the lines connecting the state boxes. The heavy line is the path corresponding to the operator sequence in Table 3. Relatively low surprisal at **Kapitaen** restates the fact that the negative log ratio

[6] Appendix A details the estimation method.

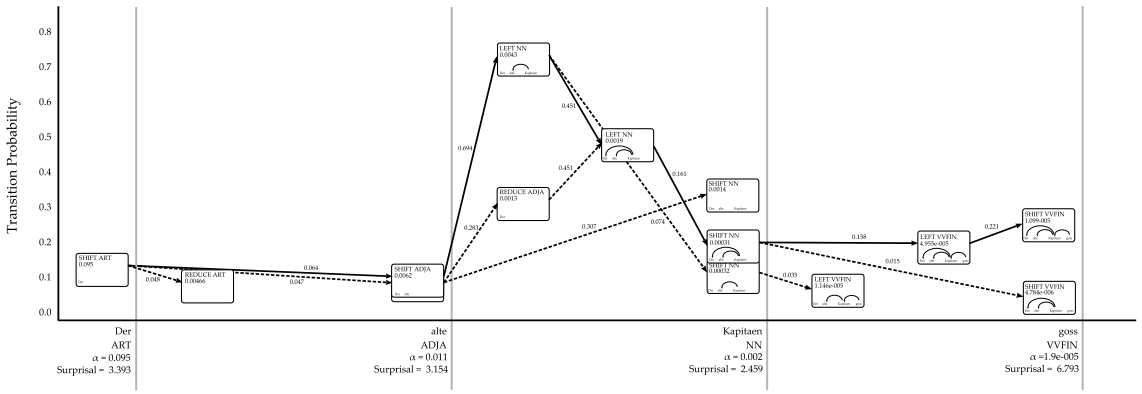


Figure 4. Sketch of surprisal calculation for a $k = 3$ parser.

of $\alpha^3_{\text{Kapitaen}}$ and α^3_{alte} on this probability model is small. We say that the surprisal of the word in that position is 2.339 bits. Surprisal is greater at the verb **goss** because the total probability of the parser actions required to extend the previous best-states to cover that word is comparatively low (recall that surprisal is a negative logarithm of a probability). This compulsion to transit low-probability states is indexed by greater surprisal at **goss** compared to **Kapitaen**.

A bigram or trigram model could also capture the fact that a noun is likely to follow an adjective. However, surprisal in a dependency parser can reflect very long-distance dependencies like the seven word span in Figure 5. This ability stands in contrast with Markov models, which are subject to a finite length limit (Park & Brew, 2006).

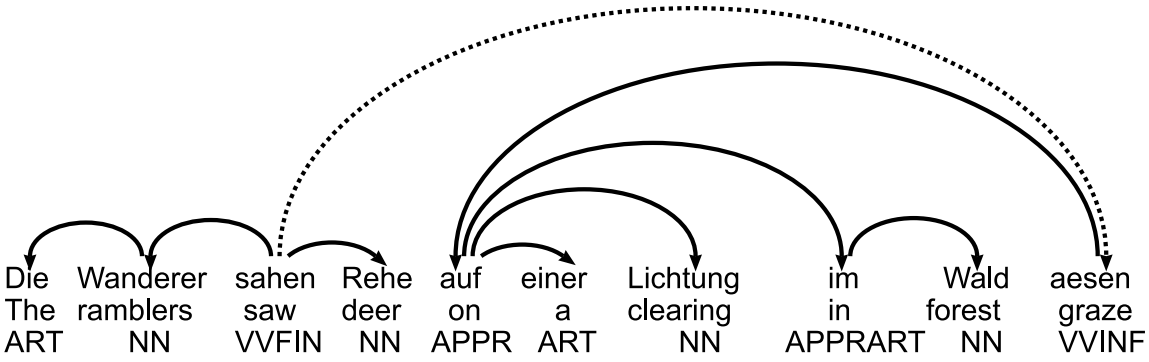


Figure 5. The longest dependency in the PSC.

Surprisal imposes comparatively few requirements at the implementation level. It has been combined with a variety of language-analysis devices including the Simple Recurrent Network (S. Frank, 2009). Surprisal is also agnostic about the degree of parallelism. This differs from the Tuning Hypothesis, which is serial (Mitchell, 1994, 1995). In what follows we leverage this generality to examine different theories of comprehension difficulty that all share the surprisal complexity metric but make different commitments as regards parallel parsing.

Retrieval

The temporal nature of speech suggests that sentence understanding requires a comprehending person to remember properties of earlier words. As each new word is heard, linguistic structures associated with earlier words need to be *retrieved* to perceive the meaningful content of the sentence. Does such "remembering" employ the same working memory as the rest of cognition? The cue-based retrieval theory of Lewis and Vasishth (2005) (henceforth, LV05) holds that it does, integrating an account of sentence processing difficulty into a general theory of cognitive architecture, ACT-R (Anderson & Lebiere, 1998; Anderson, 2005). In this work we re-formalize a subset of key theoretical ideas from LV05 in the dependency parsing system described above.

Absent some degree of familiarity with ACT-R itself, these theoretical choices may appear arbitrary. But they constitute a zero-parameter model. By zero-parameter model, we mean that the same ACT-R assumptions that have been repeatedly applied to different domains of cognition over the years also seem to apply well in accounting for sentence comprehension difficulty. Success in the domain of sentence comprehension strengthens the overall case for ACT-R as a general theory of cognition.

ACT-R.

ACT-R is an acronym that stands for Adaptive Control of Thought — Rational. It represents the latest in a series of cognitive architectures developed over the past thirty years by John R. Anderson (1976, 1983, 1990). Each of these theories is different, but there are identifiable themes that have persisted throughout Anderson's work. The most fundamental theme is that computational modeling of human cognition is not premature. Rather than being simply an equivalent calculation or an abstract description of what cognitive agents do, Anderson's cognitive architectures have all been intended as detailed, practical theories of what is actually going on in the minds of real people engaged in tasks that require intelligence. Although pursuit of this goal has led Anderson and his colleagues to develop computer simulation programs that are consistent with the theories, the programs themselves are not the theories. They do, however, make it much easier to work out the joint consequence of complex combinations of theoretical proposals.

Another theme running through Anderson's work is the distinction between procedural and declarative knowledge. This distinction is reified in ACT-R, which has both a declarative memory for "facts" and a procedural memory for things that the model "knows how to do." Each sort of knowledge is particular to a domain. In a cognitive model of arithmetic, a fact might represent the knowledge that twelve times seven is eighty-four, whereas in a sentence-processing model, a fact might encode the belief that a particular noun-adjective combination forms a phrase. Individual declarative memory elements are known as *chunks*. Each one has an activation level that determines the latency and accuracy with which it can be retrieved. Chunks are to be contrasted with pieces of procedural knowledge. Anderson's work adopts the idea of a production system from Newell and Simon (1972). A production is an association between two states of mind[7]. If a production *applies*, a thinker transits from the old state-of-mind to the new state of mind defined in the production. ACT-R countenances exactly one state of mind at any given instant. To put it another way, "the production system comprises the central bottleneck" (Anderson, 2005, 315).

Sentence processing as memory retrieval.

[7] A production's ability to refer to intermediate states of mind differentiates it from the classic notion of an association between overt stimulus and overt response. This aspect renders production systems "cognitive" as opposed to merely behavioral models.

Earlier work, LV05, argues not only that ACT-R is an appropriate medium in which to express parsing models, but indeed that its specific theoretical commitments together can make sense of several outstanding puzzles in the field of human sentence comprehension. As a step towards a complete cognitive model of sentence comprehension, LV05 define a production system that incrementally builds syntactic structure in declarative memory. These structures respect X-bar theory, an approach that incorporates ideas from dependency grammar quite directly (Kornai & Pullum, 1990). Because these syntactic structures are built using procedural knowledge in ACT-R, the model is a serial parser. To postulate, as LV05 do, that the developing syntactic structures are held in declarative memory, is to hypothesize that the remembered pieces of syntactic structure are subject to the same activation dynamics as in other cognitive domains. Individual productions in LV05's production system access declarative memory to attach new pieces of sentence structure. LV05 show that the pattern of retrieval latencies derived from these memory accesses, under standard ACT-R assumptions, derives human reading time patterns across a range of English constructions that includes center-embedding, garden-path sentences, and relative clauses.

The ACT-R chunk-activation dynamics thus play a key role in LV05's activation-based sentence processing model. These dynamics give rise to two effects, decay and similarity-based interference (SBI). Decay and SBI also appear in memory tasks with non-linguistic stimuli (Anderson et al., 2004). The idea of decay is that words heard farther back are more difficult to retrieve for attachments; this idea is realized in work by M. Just and Carpenter (1992, 133) and in abstract form by Chomsky (1965, 13-14) and Gibson and colleagues (?, ?, 11), (Gibson, 2000; Warren & Gibson, 2002; Grodner & Gibson, 2005)[8]. The idea of SBI, spelled out explicitly in the context of sentence processing by ? (?); Lewis (1996), is that earlier words can act as distractors if they happen to also match along certain *cues*, like plural number, accusative case, or animacy. Such distractors make sentence comprehension harder at points where an earlier word must be retrieved (Van Dyke & McElree, 2006). Both of these effects are widespread (Lewis & Nakayama, 2001; Van Dyke & Lewis, 2003; Van Dyke & McElree, 2006; Van Dyke, 2007; Vasishth & Lewis, 2006; Hofmeister, 2007; Logačev & Vasishth, 2009).

We adapt the ACT-R theory of memory-element activation dynamics applied in LV05 to the broad-coverage dependency parsing system discussed above. In this broad-coverage setting, we leave hand-crafting of the grammar behind (cf. Patil, Vasishth, and Kliegl (2009)). But we retain ACT-R's power-law of activation decay. We also retain LV05's interpretation of parser actions as productions that cause retrievals. Specifically, we interpret the LEFT and RIGHT operators as causing a retrieval of the word at the left end of the newly-drawn dependency arc. The latency of these retrievals, as a function of the ACT-R declarative memory chunk dynamics, is the key determinant of the difficulty prediction that we deduce on this complexity metric.

The fan effect.

The ACT-R declarative memory dynamics that model interference are designed to derive something called the fan or "min" effect (Anderson, 1976, 276ff). Anderson (1974) finds that, in a memory test, subjects' verification responses about a remembered entity grow slower as more propositions come to be associated with that entity. For instance, in

[8]It may be worth noting some differences between the two most recent variants of the decay idea. Both the DLT and LV05's activation-based theory predict that recency decreases difficulty. However, the DLT does not include any notion of interference. Moreover, LV05's activation theory, but not the DLT, acknowledges the possibility of re-activation. Such re-activation can account for cases where increased head-dependent distance facilitates comprehension (Vasishth & Lewis, 2006; Shaher, Engelmann, Logacev, Vasishth, & Srinivasan, 2009; Hofmeister, 2009).

Figure 6, a hippie would be remembered in a memory chunk to which three properties are linked. It has a fan of 3.

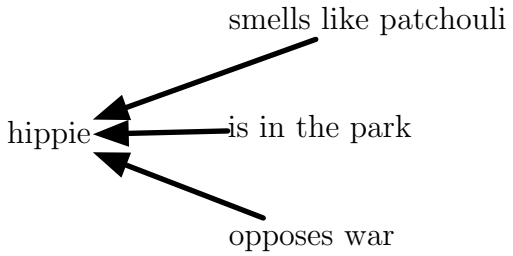


Figure 6. A fact in memory to which other facts are connected.

As a subject links more properties to the same entity, pushing the fan value up, it takes longer and longer to verify probe propositions in a set that was previously learned to criterion. The standard ACT-R memory dynamics in the LV05 model derive this behavior. In LV05, "fan" reflects a panoply of linguistic factors including morphological features like tense and number, but also tree-geometric connections to other nodes in a developing phrase structure.

This paper's adaptation of LV05 to broad coverage and parallel parsing is much simpler. It derives the fan effect from just one cue: a word's part of speech. We theorize that retrievals are delayed by a factor reflecting the presence of other words sharing the same grammatical category that have come earlier in the input sentence. This principle applies to all words, regardless of their attachment status, thus implementing a simplified version of similarity-based interference. The motivation for reducing the cues to part-of-speech is merely a matter of convenience: if more detailed information were available for each word to be processed (e.g., case and animacy information), these cues could be deployed in the calculation of interference costs. Our adaptation should therefore be seen as a simplifying assumption which could in principle be extended.

Retrieval: a worked example.

To see how retrieval works, consider the first few words, **Der**, **alte** and **Kapitaen** of our running example. The problem states explored by a $k = 3$ parser are arranged by predicted difficulty in Figure 7. This figure is analogous to the earlier Figure 3, but note that the vertical axis is predicted duration in milliseconds, rather than predicted surprisal in bits. Times accumulate for parser actions associated with a single word, but are reset between words.

Table 4 shows the time course of events postulated in the model. All times are cumulative except for Action Time, which is specific to each word's parsing time. The first two words are handled with the SHIFT operator, and each takes a constant amount of time. At **Kapitaen**, the LEFT operator applies twice to attach the article **Der** and the adjective **alte** as dependents; these are steps 3 and 4 respectively from Table 3. The retrieval of **Der** takes longer than the retrieval of **alte** because its memory chunk's activation has decayed ever so slightly.

The durations in Table 4 reflect serial parsing where the beam-width is set to $k = 1$. For larger beam-widths, we take the maximum retrieval time for any analysis in the beam. Selecting max as the mode of combination means that the predictions reflect difficulty associated with the worst-case analysis in the beam[9]. This assumption is analogous to

[9]An anonymous reviewer brings up the wide variety of alternatives to this"max" mode of combina-

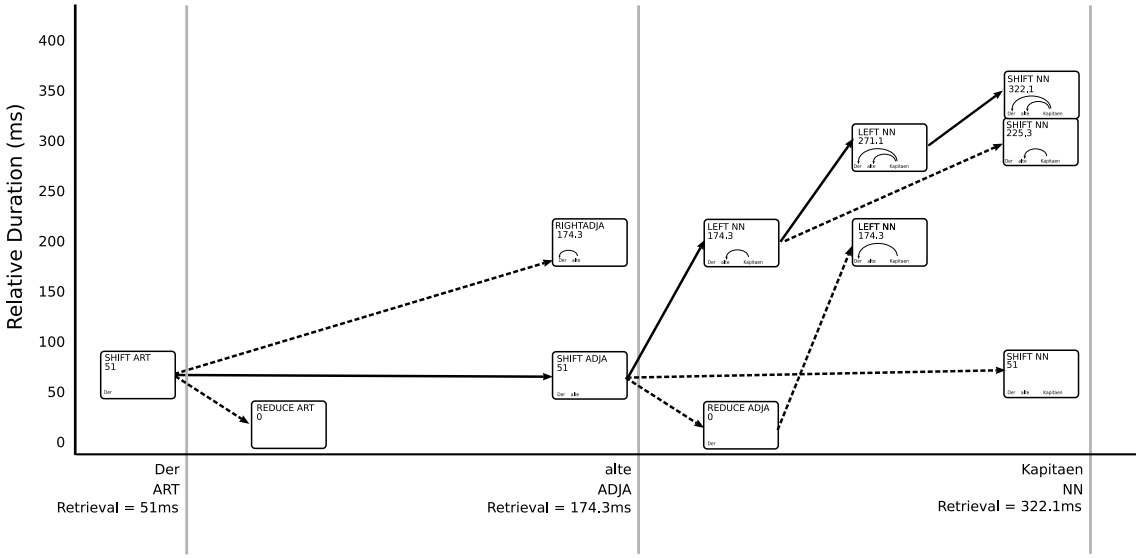


Figure 7. Sketch of retrieval times for a $k = 3$ parser.

```
READING: ‘‘Der’’ at 0.
0.051:  SHIFT ‘‘Der’’.
*****************************************
Action Time for ‘‘Der’’:  0.051
*****************************************
READING: ‘‘alte’’ at 0.051
0.102:  SHIFT ‘‘alte’’.
*****************************************
Action Time for ‘‘alte’’:  0.051
*****************************************
READING: ‘‘Kapitaen’’ at 0.102
0.152:  Decide to attach ‘‘alte’’ as dependent.
0.196617105931:  Retrieve ‘‘alte’’.
0.246617105931:  LEFT-ARC with ‘‘alte’’.
0.296617105931:  Decide to attach ‘‘Der’’ as dependent.
0.394929128334:  Retrieve ‘‘Der’’.
0.444929128334:  LEFT-ARC with ‘‘Der’’.
0.494929128334:  SHIFT ‘‘Kapitaen’’.
*****************************************
Action Time for ‘‘Kapitaen’’:  0.39292913
*****************************************
READING: ‘‘goss’’ at 0.494929128334...
```

Table 4: A partial trace of the retrieval latency for a PSC sentence, in seconds.

that made in other models such as the Dependency Locality Theory, where maximum integration complexity in a sentence is used to index processing difficulty (Gibson, 1998, 40).

The equations defining the ACT-R memory chunk dynamics are provided in Appendix B. Aside from arbitrary decisions about the scale of constant durations of Nivre operators, the particular formulation of retrieval given above follows directly from principles of ACT-R. The numerical constants in Equations 8-11 are kept at the default ACT-R values used in previous research on modeling sentence comprehension (Lewis & Vasishth, 2005). This implementation of retrieval is thus a zero-parameter model in the sense that no ACT-R parameters are estimated.

Examining both surprisal and retrieval in the same incremental parser states facilitates a fine-grained analysis of the differences between the two complexity metrics. The next sections compare the predictions of the two metrics to fixation durations recorded in the Potsdam Sentence Corpus.

Fixation durations in the Potsdam Sentence Corpus

This section examines how surprisal and retrieval predict eye fixations for the PSC. The first section describes the PSC data, followed by a discussion of how our approach relates to eye-movement control models. We then detail the methods, results, and implications of this study.

Data

Reading involves an alternation of fixations and saccades that move words into the center of the visual field. In the PSC, fixations between saccades last mostly between 150 and 300 ms, with a mean single-fixation duration of 206 ms (Kliegl, Nuthmann, & Engbert, 2006). The eyes do not solely move forward from word to word in a left-to-right fashion with single fixations (40%), but they also skip (21%) or refixate (13%) words, or regress back to a previous word of the sentence (8%). These eye movements are correlated with many indicators of local processing difficulty, such as unigram frequency, bigram frequency, predictability[10], and length of the fixated word. The direction of effects is generally compatible with intuitive notions of processing difficulty: short, frequent, and predictable words are more frequently skipped, less frequently refixated, and have shorter fixation durations when they are fixated in comparison with long, rare, and unpredictable words. Longer fixation durations immediately before highly predictable words constitute a notable exception to this general pattern.

First-pass regression probability and various fixation measures are often characterized as "early" and "late" measures, with an implied mapping to "early" and "late" stages of cognitive processing. Early processing stages refer to the subprocesses of word identification comprising extraction of visual, orthographic, and phonological features subserving lexical access. Late stages refer to the difficulty of syntactic, semantic, and discourse integration. However, Clifton, Staub, and Rayner (2007) advocate caution in drawing such a simple

tion. Another attractive mode of combination would weight the retrieval times by the probability of the operator-application that causes the retrieval. At high parallelism levels, under this "nondeterministic serial" arrangement, retrievals that would have been long in duration are weighted by smaller and smaller probabilities. Thus, only the retrievals occuring in the highest-probability analysis have much effect on the overall prediction. This arrangement makes it systematically harder to examine the role of ranked parallelism in human sentence processing and for this reason we decided not to pursue it in the work reported here. However we believe it holds considerable interest for the development of serial parsing theories.

[10] Predictability in this sense refers to difficulty a human subject has guessing a word given its left context. Predictability can be estimated using variants of the Cloze procedure (Taylor, 1953).

relation between "early" and "late" measures and early versus late stages of cognitive processing. As they put it,

> The terms "early" and "late" may be misleading, if they are taken to line up directly with first-stage vs second-stage processes that are assumed in some models of sentence comprehension (Frazier, 1987; Rayner, Carlson, & Frazier, 1983). Nonetheless, careful examination of when effects appear may be able to shed some light on the underlying processes. Effects that appear only in the "late" measures are in fact unlikely to directly reflect first-stage processes; effects that appear in the "early" measures may reflect processes that occur in the initial stages of sentence processing, at least if the measures have enough temporal resolving power to discriminate among distinct, fast-acting, processes.
>
> (Clifton et al., 2007, 349)

We follow Clifton et al. (2007) in characterizing dependent measures as "early" and "late" without committing to a simple mapping between early and late stage processes. Indeed, the results to be presented below confirm the cautionary message of the above quote. The analyses we present in subsequent sections do not support a simple mapping of measures to parsing stages.[11]

Table 5 defines the dependent measures (four fixation measures and first-pass regression probability) that are considered in this study. The early measures take into account first pass data, whereas the late measures encompass both first and subsequent passes.

Early Measures
Single Fixation Duration (SFD)
The amount of time a word is fixated in first pass if it is only fixated once.
First Fixation Duration (FFD)
The amount of time a word is fixated during the first fixation in first pass.
Regression Probability (REG)
Likelihood of regressing to a previous word during the first pass.
Late Measures
Regression Path Duration (RPD)
The sum of all reading times at a word and all words to its left, starting from the first fixation in the word until the first fixation past the region.
Total Reading Time (TRT)
The sum of all fixation durations at a word, including first pass and re-reading.

Table 5: The four fixation measures and one regression probability modeled in this study.

[11] Note that early and late measures are also compromised by being derived from overlapping sets of data (i.e., Single Fixation Durations are a subset of First-Pass Gaze Durations, and Gaze Durations are a subset of Total Reading Time). This positive dependency will tend to diminish any differential effects between early and late measures.

Relation to eye-movement control models

In his review Rayner (1998) highlights how, in recent years, the link between fixation durations and the oculomotor, perceptual, and cognitive processes that drive them has been enhanced considerably by mathematical models of eye-movement control. The two most prominent cognitive processing models, E-Z Reader (Reichle, Pollatsek, Fisher, & Rayner, 1998; Pollatsek, Reichle, & Rayner, 2006) and SWIFT (Kliegl et al., 2004; Engbert, Nuthmann, Richter, & Kliegl, 2005) predict fixation durations as a function of word frequency and length. They also account for fixation positions in words. However, the models take into account sentence-level factors in what may be called a rudimentary fashion (Reichle, Warren, & McConnell, 2009), or only to the degree that is mediated indirectly via predictability (Engbert et al., 2005). This is despite evidence that syntactic processing affects fixation durations in, for example, garden path sentences (Frazier & Rayner, 1982). The need for a truly syntactic component is underlined by recent results indicating that syntactic processing, as modeled by surprisal, not only predicts fixation durations but also leads to better overall model fit beyond the lexical factors in German and English (Boston, Hale, Patil, Kliegl, & Vasishth, 2008; Demberg & Keller, 2008).

In this study we extend these large-scale investigations of surprisal by testing the predictions of both surprisal and retrieval on a German eyetracking corpus. Both factors are predicted to help in modeling the fixation data, but, because they make different claims about the source of human processing difficulty, they could model different sources of difficulty in the data.

Method

The Potsdam Sentence Corpus provides the data for this study. As mentioned above on page 2, the dataset consists of 144 individual German sentences (1138 words) read by 222 native German speakers. When first and last words are removed to reduce start-up and wrap-up effects, fixations for the 850 remaining words are available for each of the five fixation measures in Table 5. We do not consider fixations with durations of less than 50 ms. The dataset is further restricted to just those words where a retrieval is postulated. The cardinality of this set varies with beam-width, from 668 words at $k = 1$, to 841 words at $k \geq 6$. The occurrence of regressions is binomially coded, with 1 indicating a regression occurred during first pass, and 0 that a regression did not occur.

The surprisal and retrieval predictions follow from parser runs at a systematic selection of beam-widths as indicated in Table 1. The beam-widths are $k = 1, 5, 10, 15, 20, 25$ and 100. The retrieval and lexical predictors are log-transformed to avoid multiplicativity effects in the statistical analysis and to ensure that the residuals are approximately normally distributed (Gelman & Hill, 2007).

The statistical evaluation of the surprisal and retrieval predictions uses the linear mixed model (Pinheiro & Bates, 2000) provided in the statistical computing software R (R Development Core Team, 2006) and its `lme4` package (Bates, Maechler, & Dai, 2008). Linear mixed models are regression models that take into account group-level variation, which is present in psycholinguistic data via the participant and item factors. Including both fixed effects (e.g. predictors for the fixations) and random effects (e.g. participant, item) allows one model to take into account both within-group and between-group variation for the intercepts (Demidenko, 2004). To model regression probability, we employ a generalized linear mixed model with a binomial link function. Further details regarding linear mixed models are provided in textbooks such as (Baayen, 2008; Gelman & Hill, 2007) and the Special Issue in the Journal of Memory and Language (Volume 59, Issue 4) entitled Emerging Data Analysis.

We fit four mixed effects models to each of the dependent measures described in Table 5. The baseline model given in Equation 4 incorporates the lexical factors described in the previous subsection: word length (len), word predictability (pred), unigram frequency (freq), and bigram frequency (bi).

$$\log(y) = \beta_1 \text{freq} + \beta_2 \text{len} + \beta_3 \text{bi} + \beta_4 \text{pred} + b_p + b_q + \epsilon \tag{4}$$

The dependent measure, generically denoted y, is log-transformed so that it can be modeled linearly, and each of the lexical predictors are added to the group-level intercept variation for subject (b_p) and item (b_q). The subscripts in these quantities range over subjects p and items q. The element that is estimated for each of the lexical predictors is the coefficient, β. Each of the lexical predictors are additionally centered so that their mean is zero. This allows the intercept and the coefficients for each predictor to be interpreted given the average value of the other predictors (Gelman & Hill, 2007).

Three other models for each fixation are fit: a baseline plus surprisal (surp, Equation 5), a baseline plus retrieval (ret, Equation 6), and a model that incorporates all predictors (Equation 7).

$$\log(y) = \beta_1 \text{freq} + \beta_2 \text{len} + \beta_3 \text{bi} + \beta_4 \text{pred} + \beta_5 \text{surp} + b_p + b_q + \epsilon \tag{5}$$

$$\log(y) = \beta_1 \text{freq} + \beta_2 \text{len} + \beta_3 \text{bi} + \beta_4 \text{pred} + \beta_5 \text{ret} + b_p + b_q + \epsilon \tag{6}$$

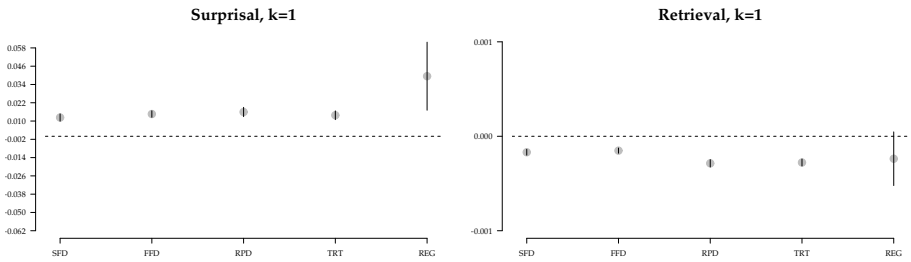
$$\log(y) = \beta_1 \text{freq} + \beta_2 \text{len} + \beta_3 \text{bi} + \beta_4 \text{pred} + \beta_5 \text{surp} + \beta_6 \text{ret} + b_p + b_q + \epsilon \tag{7}$$

Comparisons between linear mixed models are based on log-likelihood ratios, penalizing models for a large number of parameters. We use the traditional maximum-likelihood based χ^2-statistic to this end, with increases in log-likelihood indicating a better model. As Pinheiro and Bates (2000) and Faraway (2006) point out, comparing models with different fixed effects yields anti-conservative p-values (i.e., the p-value may be higher than suggested by the model comparison). Faraway (2006) suggests using the parametric bootstrap instead (see Appendix for details). Note that the anti-conservativity problem is not so severe when large quantities of data are available relative to the parameters fit; this is the case in our dataset. However, in order to confirm that our model comparisons were not misleading, we carried out the parametric bootstrap for all the reading time measures (the simulate function necessary for this procedure is not yet implemented for generalized linear mixed models; therefore, we did not carry out a bootstrap for regression probability); none of the simulations yielded any interpretations different from our model comparisons based on the χ^2-statistic. Furthermore, we also compared the Akaike Information Criterion (Akaike, 1973) and Deviance Information Criterion (Spiegelhalter, Best, Carlin, & Linde, 2002; Spiegelhalter, 2006) for each model, with results comparable to those reported here.

Results

At low beam-widths, surprisal (Equation 5) best models the fixation duration measures. Figures 8(a) and 8(b) plot the fitted values of β_5 and β_6 respectively in this model for each fixation measure. Coefficients greater than zero imply that increases in the predictor are associated with increases in predicted log reading time. 95% confidence intervals that do not cross 0.0 indicate statistical significance at $\alpha = 0.05$. At $k = 1$ surprisal has statistically significant positive coefficients for all fixations and the regression probability in Figure 8(a). Retrieval, on the other hand, makes the incorrect prediction: at low beam-widths, increasing retrieval difficulty predicts shorter fixation durations.

Although at a low beam-width only surprisal correctly predicts fixation durations and regression probability, at a higher beam-width, retrieval is also a predictor. Figure 9 shows



(a) (b)

Figure 8. Surprisal alone predicts at $k = 1$.

surprisal's and retrieval's predictions in parsers of greater and greater beam size. Each eye-movement measure introduced in Table 5 receives its own sub-graph in Figure 9. Table 6 shows that the two predictors are uncorrelated across all beam-widths k except at $k = 1$, where the correlation is 0.43. The measures are also uncorrelated with word predictability and unigram frequency.

k	s-r	s-p	s-f	r-p	r-f	p-f
1	0.43	0.16	0.05	-0.05	-0.33	0.50
5	0.09	0.25	0.10	-0.03	-0.27	0.53
10	0.02	0.26	0.12	0.002	-0.23	0.53
15	-0.09	0.26	0.11	-0.04	-0.25	0.53
20	-0.11	0.27	0.11	-0.04	-0.22	0.53
25	-0.16	0.27	0.11	-0.06	-0.21	0.53
100	-0.07	0.26	0.09	0.13	0.07	0.53

Table 6: Correlations between surprisal (s), retrieval (r), predictability (p) and log-frequency (f) at different beam-widths k

Table 7 lists the log-likelihood values for the four models plotted for each individual measure at $k = 100$. The overall worst model fit for all seven fixations is the baseline, which only includes lexical factors (Equation 4). The best model for all fixations, in bold text, includes both surprisal and retrieval as predictors. The best model for regression probability includes only surprisal. Table 8 reports chi-square and p-values for comparisons between models 4–7 at the $k = 100$ parallelism level. This pattern of statistically-significant differences is consistent across beam-widths. These model fit results demonstrate the utility of both conceptions of comprehension difficulty.

The full results of the multiple linear regressions for $k = 1$ and $k = 100$ are shown in tabular form in Table 9.

Discussion

The main finding of this study is that whereas surprisal models parsing difficulty reflected in fixation durations and regression probability at a low beam-width, $k = 1$, retrieval does not. As the beam-width is widened, surprisal continues to predict difficulty in all measures, and maximum retrieval cost predicts both early and late measures as well as regression probability. Because predicted retrieval time-costs are aggregated across

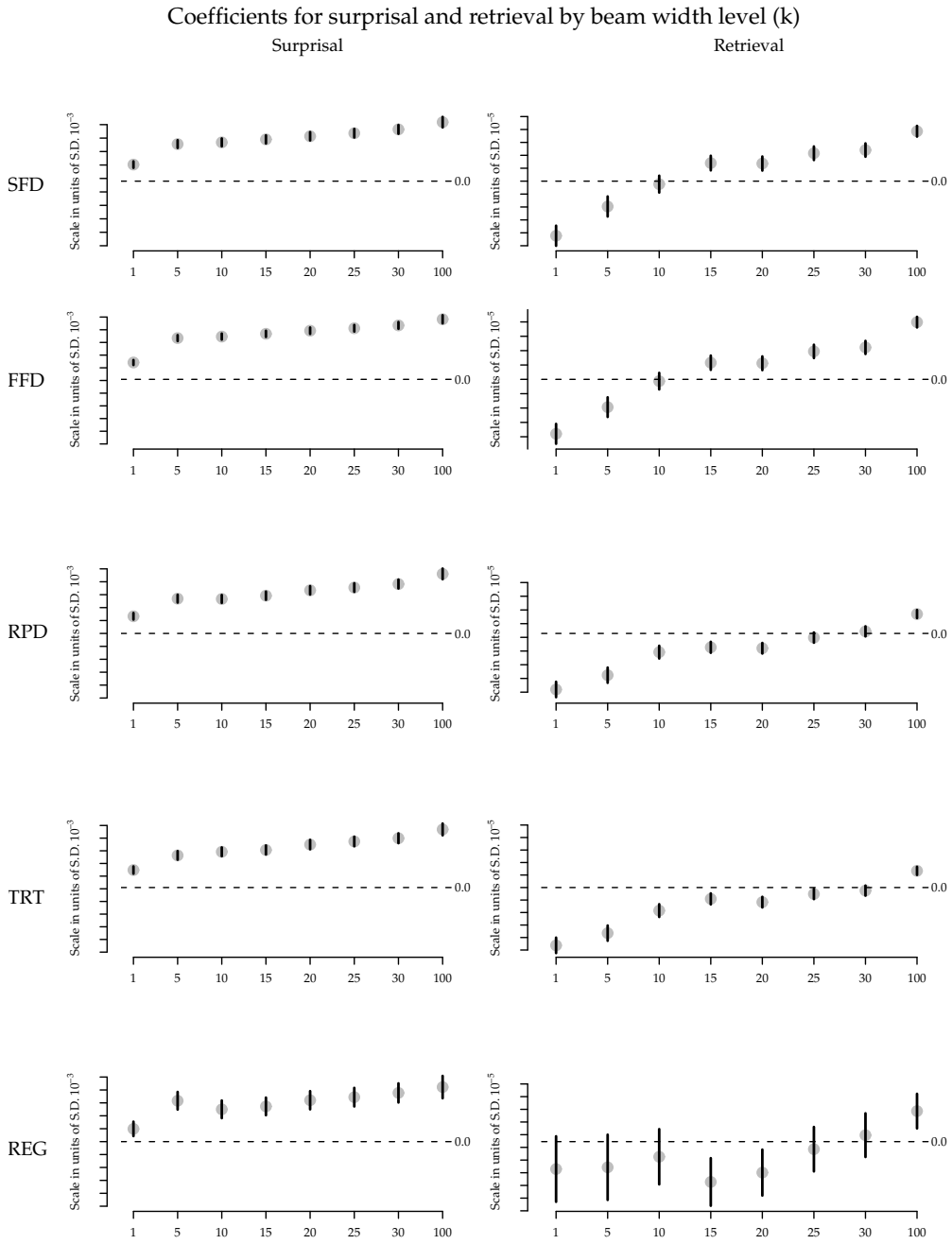


Figure 9. With large beams, surprisal and retrieval predict the same fixations. For each fixation measure, the coefficients for surprisal (left) and retrieval (right) are given along the y-axis, which is scaled to units of standard deviation. The x-axis represents the degree of parallelism, k.

beams using max, the retrieval score can only increase as k increases. These increases allow for the prediction of all dependent measures, suggesting that retrieval at low beam-widths underestimates difficulty in fixation durations and regression probability.

Dependent measure	Baseline	+ Surprisal	+ Retrieval	+ Both
SFD	-21604	-21379	-21454	**-21214**
FFD	-26265	-25859	-26055	**-25636**
RPD	-78842	-78609	-78803	**-78565**
TFT	-70552	-70370	-70523	**-70338**
REG	-43246	**-43189**	-43729	-43717

Table 7: Log-likelihoods for each model and fixation duration for $k = 100$.

Dependent measure	Baseline, Surprisal		Baseline, Retrieval		Baseline, Both	
	Chi-square	p-value	Chi-square	p-value	Chi-square	p-value
SFD	450.64	<0.01	299.95	<0.01	780.07	<0.01
FFD	812.78	<0.01	419.78	<0.01	1259.1	<0.01
RPD	466.96	<0.01	79.152	<0.01	554.22	<0.01
TFT	363.19	<0.01	58.397	<0.01	427.60	<0.01
REG	114.15	<0.01	0	1	0	1

Table 8: Chi-square statistics for model comparison with baselines for $k = 100$.

General Discussion

This work measures the effectiveness of two syntactic complexity metrics, surprisal and retrieval, in modeling sentence processing difficulty. The two metrics furnish different sorts of accounts of processing difficulty. Surprisal hypothesizes that low-probability attachments cause difficulty. Retrieval, on the other hand, attributes sentence processing difficulty to human resource limitations. To determine the relative contribution of each, we use a common collection of incremental dependency parser states to work out the consequences of each complexity metric. This methodology also allows us to quantitatively investigate how these metrics interact with memory resources. In particular, we examine a range of parallelism levels in a parsing systems that are otherwise identical.

Our findings suggest that the surprisal and retrieval predictions do not fall neatly into the categories implied by the early/late distinction traditionally assumed in the eye movements literature.[12] With a serial parser, surprisal is able to predict all eye-dependent measures, even "late" measures such as Regression Path Duration that are not traditionally conceived of as the result of surprising attachments (Clifton et al., 2007, p.349). Retrieval, on the other hand, is not able to predict any of the eye-dependent measures at the most austere parallelism level. As more memory is provided and the parallel levels is increased, retrieval can eventually predicts all eye-dependent measures, including "early"

[12] A reviewer points out that data analyses were conducted on multiple dependent measures without correcting for multiple comparisons. This is a valid concern; indeed, the situation is even more complicated: the same data are used in some of the separate analyses (i.e., First Fixation Durations are a subset of Total Reading Time), whereas typically multiple comparison adjustments are carried out between independent measures. Thus, even the usual adjustments for multiple comparisons would not fully take into account the dependencies of the aggregated measures. In analyzing early and late measures we adhere to what are the accepted standards of the field; at least we are not aware of a single eye-movement study that reported corrections for these measures. Having said this, we also do not think that this issue is critical for the present paper (and much other eye-movement research) because our conclusions do not depend on differential outcomes for the various measures. Rather, all measures lead to the same conclusion.

measures like Single Fixation Duration that were not formerly thought to be sensitive to working memory limitations (Clifton et al., 2007, p.348). In fact, at high levels of parallelism, surprisal and retrieval make similar predictions: increased surprisal or retrieval cost both result in longer fixation durations.

These findings suggest that memory capacity, when formalized as beam-width, is an important modulator of each theory's predictions. With surprisal, even a serial parser can account for German reader's fixations. This result is consistent with the seriality claims that underly garden pathing theories (Frazier, 1979) and other theories that focus on serial processing (Mitchell, 1994, 1995). It confirms the suggestion that a highly limited search space need not necessarily diminish a model's cognitive fidelity (T. Brants & Crocker, 2000), and, we suggest, supports the view of human cognition as "boundedly rational" in general (Simon, 1955; Gigerenzer et al., 1999; ?, ?). From this perspective, we have argued by example, that ideal prefix probabilities as in Hale (2001) and Levy (2008) are not necessary to predict syntactic difficulty in eye-dependent measures.

The interplay of memory capacity with retrieval suggests that this metric, unlike surprisal, requires a higher degree of parallelism to predict the behavioral data. In fact, retrieval requires a beam-width greater than 30 to predict all eye-dependent measures. This result encourages investigation into a parallel version of the currently serial ACT-R model. The Future Work section below details this possibility.

Although the discussion above verifies the crucial role that parallelism plays in this evaluation, in this work we do not resolve the difficult issue of whether the human sentence processing mechanism is single-path or multi-path (Lewis, 2000; Gibson & Pearlmutter, 2000). Rather, we highlight a computational framework for examining the consequences of serial versus parallel search, coupled with two theories of parsing difficulty that are independent of the degree-of-parallelism issue. We are hopeful that this framework can be applied again to address the larger parallelism issue. The next section details a few other promising avenues for future search.

Future Work

Section has layed out a framework for parallel parsing in which the question of psycholinguistic adequacy can begin to be addressed. The results reported in section suggest that retrieval is a better predictor of German readers' sentence comprehension difficulty when it is applied in a parallel parser. This finding is at odds with positive findings of Lewis, 1993 and Lewis & Vasishth, 2005, whose serial parsers were sufficient to account for well-known contrasts in human comprehension difficulty. From the perspective of a cognitive architecture like ACT-R, the use of beam search in a parser corresponds to parallel production-rule firing. This goes against a central ACT-R assumption, namely that only one production rule fires at a time. However, investigators like Salvucci & Taatgen, 2008 have recently begun to explore alternatives to the fundamental assumption in ACT-R models of human multitasking. Although these authors' a concerned with multitasking across different tasks, it may be possible to apply a similar extension of ACT-R to within-task parallel processing. We leave this exciting possibility to future work.

Another interesting area for future work involves examining possible interactions between surprisal and retrieval. Our research applies a methodology that compares the relative fit of models that include surprisal and retrieval alone, as well as together. The superiority of the combined models indicates that the two predictors do not overlap. Surprisal is not subsumed by retrieval nor is retrieval subsumed by surprisal. If both metrics capture something true about human sentence comprehension, then it is natural to seek a detailed model that would explain why and how they are both involved. In the present work we

have simply highlighted the nonoverlapping character of the two types of accounts. The development of a truly integrated model must await future work.

An even more substantial area of future work will involve integrating more elements of the original ACT-R implementation into this model. One such element are explicit *expectations* for future words. Lewis & Vasishth, 2005 found that these expectations interact with retrieval cues. Whereas the Lewis and Vasishth's hand-coded production rules reflect phrase structures with intermediate nodes like NounPhrase, VerbPhrase, the dependency parser applied in this work does not deal with any such structures. In fact, Nivre, 2004 suggests that no dependency structure can faithfully implement the sort of "top-down" parsing strategy that most straightforwardly reflects the intuitive idea of an expectation for upcoming words. Future work should explore combinations of grammars and parsing strategies that can model expectations, to fully carry out the program begun in Lewis & Vasishth, 2005.

Similarly, strong lexical influences on human sentence comprehension are well-known (?, ?, ?, ?). One might introduce lexical probabilities into the model to address these effects. The parser used to derive the results in section operates at the part-of-speech level, which does not distinguish within-category differences for instance between words like *ate* and *slept* that are both verbs. From the point of the view of the model described in section , both elements are of type VBD (past-tense verb), despite the fact that *ate* optionally takes an object and *slept* does not. Although the use of parts-of-speech isolates the role of syntactic categories in predicting the behavioral data, a lexicalized model would be much more detailed. Such a model would undoubtedly be able to overcome the sorts of typical attachment mistakes catalogued in Table A1.

Of course, lexicalism is just one of a constellation of tenets implicitly or explicitly assumed by constraint-based models. These models are often given a connectionist implementation rooted in the notion of activation (Spivey & Tanenhaus, 1998; Tabor & Tanenhaus, 1999; Christiansen & Chater, 1999; Rohde, 2002; ?, ?). Besides lexicalism, these models differ from the "symbolist" account advanced in this paper in virtue of being based upon an activation-based associative memory (?, ?; McElree, Foraker, & Dyer, 2003). In this paper's model, the memory is a stack, and all retrievals are successful, no matter how deep. Future work should examine ways to incorporate a more realistic activation-based memory like the one assumed in hybrid architectures such as ACT-R and 4CAPS (M. A. Just & Varma, 2007).

Conclusion

Our findings speak to the role of grammar in accounting for sentence comprehension difficulty. We compare two grammar-based complexity metrics while systematically varying the degree of parallelism in a family of identical parsing systems. We find that whereas surprisal and retrieval make different predictions in less parallel parsers, as the degree of parallelism increases, both metrics become adequate for predicting eye fixation durations in the Potsdam Sentence Corpus.

This result confirms the idea that grammar affects language comprehension through both probabilistic and working memory constraints. It also supports the notion that adequate models of comprehension difficulty are not optimal with respect to the task itself, but rather restricted by the kinds of memory constraints postulated in theories of cognitive architecture.

References

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. *International Symposium on Information Theory, 2 nd, Tsahkadsor, Armenian SSR*, 267–281.

Anderson, J. R. (1974). Retrieval of propositional information from long-term memory. *Cognitive Psychology*, *6*, 451–474.

Anderson, J. R. (1976). *Language, memory and thought*. Lawrence Erlbaum.

Anderson, J. R. (1983). *The architecture of cognition*. Harvard University Press.

Anderson, J. R. (1990). *The adaptive character of thought*. Lawrence Erlbaum Associates.

Anderson, J. R. (2005). Human symbol manipulation within an integrated cognitive architecture. *Cognitive Science*, *29*, 313–341.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036-1060. Available from `http://act-r.psy.cmu.edu/publications/pubinfo.php?id=526`

Anderson, J. R., & Lebiere, C. (1998). *Atomic components of thought*. Hillsdale, NJ: Lawrence Erlbaum.

Attneave, F. (1959). *Applications of information theory to psychology: A summary of basic concepts, methods and results*. Holt, Rinehart and Winston.

Baayen, H. (2008). *Practical data analysis for the language sciences*. Cambridge University Press.

Barton, G. E., & Berwick, R. C. (1985). Parsing with Assertion Sets and information monotonicity. In *Proceedings of the ninth international joint conference on artificial intelligence* (pp. 769–771).

Bates, D., Maechler, M., & Dai, B. (2008). lme4: Linear mixed-effects models using S4 classes (R package version 0.999375-27) [Computer software].

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Boston, M. F., Hale, J. T., Patil, U., Kliegl, R., & Vasishth, S. (2008). Parsing costs as predictors of reading difficulty: An evaluation using the Potsdam Sentence Corpus. *Journal of Eye Movement Research*, *2*(1), 1–12.

Brants, S., Dipper, S., Eisenberg, P., Hansen-Schirra, S., König, E., Lezius, W., et al. (2004). TIGER: Linguistic interpretation of a german corpus. *Research on Language and Computation*, *2*, 597–619.

Brants, T., & Crocker, M. (2000). Probabilistic parsing and psychological plausibility. In *Proceedings of 18th international conference on Computational Linguistics COLING-2000*. Saarbrücken/Luxembourg/Nancy.

Buch-Kromann, M. (2001). Optimality parsing and local cost functions in Discontinuous Grammar. *Electronic Notes in Theoretical Computer Science*, *53*.

Chomsky, N. (1965). *Aspects of the theory of syntax*. Cambridge, MA: MIT Press.

Chomsky, N. (1995). Bare phrase structure. In G. Webelhuth (Ed.), *Government and binding theory and the minimalist program: Principles and parameters in syntactic theory* (chap. 8). Cambridge Massachusetts: Blackwell.

Christiansen, M., & Chater, N. (1999). Toward a connectionist model of recursion in human linguistic performance. *Cognitive Science*, *23*(2), 157–205.

Clifton, C., & Frazier, L. (1989). Comprehending sentences with long distance dependencies. In G. Carlson & M. Tanenhaus (Eds.), *Linguistic structure in language processing* (pp. 273–317). Dordrecht: Kluwer.

Clifton, C., Staub, A., & Rayner, K. (2007). Eye Movements in Reading Words and Sentences. In R. V. Gompel, M. Fisher, W. Murray, & R. L. Hill (Eds.), *Eye movements: A window on mind and brain* (pp. 341–372). Amsterdam: Elsevier.

Covington, M. A. (2001). A fundamental algorithm for dependency parsing. In *Proceedings of the 39$^{\text{th}}$ ACM southeast conference*. Athens, GA.

Cowper, E. A. (1976). *Constraints on sentence complexity: A model for syntactic processing*. Unpublished doctoral dissertation, Brown University, Providence, RI.

Crocker, M. W., & Brants, T. (2000). Wide-coverage probabilistic sentence processing. *Journal of Psycholinguistic Research*, *29(6)*, 647–669.

Demberg, V., & Keller, F. (2008). Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, *109*(2), 193–210.

Demidenko, E. (2004). *Mixed models: Theory and applications*. Hoboken, NJ: John Wiley and Sons.

Dubey, A. (2004). *Statistical parsing for German: Modeling syntactic properties and annotation differences*. Unpublished doctoral dissertation, Saarland University, Germany.

Earley, J. (1970, February). An efficient context-free parsing algorithm. *Communications of the Association for Computing Machinery*, *13*(2).

Elman, J. (1990). Finding structure in time. *Cognitive Science*, *14*, 179–211.

Engbert, R., Nuthmann, A., Richter, E. M., & Kliegl, R. (2005). SWIFT: A dynamical model of saccade generation during reading. *Psychological Review*, *112*(4), 777–813.

Faraway, J. (2006). *Extending the linear model with R: generalized linear, mixed effects and non-parametric regression models*. CRC Press.

Fodor, J., Bever, T., & Garrett, M. (1974). *The psychology of language*. New York: McGraw-Hill.

Frank, R. (2002). *Phrase structure composition and syntactic dependencies*. MIT Press.

Frank, S. (2009). Surprisal-based comparison between a symbolic and a connectionist model of sentence processing. In *Proceedings of the* 31st *annual conference of the cognitive science society* (pp. 1139–1144).

Frazier, L. (1979). *On comprehending sentences: Syntactic parsing strategies*. Unpublished doctoral dissertation, University of Massachusetts.

Frazier, L. (1987). Sentence processing: a tutorial review. In M. Coltheart (Ed.), *Attention and performance XII: the psychology of reading* (pp. 559–586). Lawrence Erlbaum.

Frazier, L., & Fodor, J. D. (1978). The sausage machine: a new two-stage parsing model. *Cognition*, *6*, 291–325.

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

Gelman, A., & Hill, J. (2007). *Data analysis using regression and multilevel/hierarchical models*. New York: Cambridge University Press.

Gibson, E. (1991). *A computational theory of human linguistic processing: Memory limitations and processing breakdown*. Unpublished doctoral dissertation, Carnegie Mellon University.

Gibson, E. (1998). Linguistic complexity: Locality of syntactic dependencies. *Cognition*, *68*, 1–76.

Gibson, E. (2000). The dependency locality theory: A distance-based theory of linguistic complexity. In A. Marantz, Y. Miyashita, & W. O'Neil (Eds.), *Image, language, brain* (pp. 95–126). Boston, MA: MIT Press.

Gibson, E., & Pearlmutter, N. J. (2000, March). Distinguishing serial and parallel parsing. *Journal of Psycholinguistic Research*, *29*(2), 231–240.

Gigerenzer, G., Todd, P. M., & the ABC Research Group. (1999). *Simple heuristics that make us smart*. Oxford University Press.

Gorrell, P. (1989). Establishing the loci of serial and parallel effects in syntactic processing. *Journal of Psycholinguistic Research*, *18*, 61–74.

Grodner, D. J., & Gibson, E. A. F. (2005). Consequences of the serial nature of linguistic input for sentential complexity. *Cognitive Science*, *29*, 261-91.

Hale, J. T. (2001). A probabilistic Earley parser as a psycholinguistic model. In *Proceedings of NAACL 2001* (p. 1-8).

Hartley, R. V. L. (1928). Transmission of information. *Bell System Technical Journal*, 535–563.

Hays, D. G. (1964). Dependency Theory: A formalism and some observations. *Language*, *40*, 511-525.

Hofmeister, P. (2007). Memory retrieval effects on filler-gap processing. In *Proceedings of the* 29th *annual meeting of the cognitive science society* (p. 1091-1096).

Hofmeister, P. (2009). Encoding effects on memory retrieval in language comprehension. In *Proceedings of CUNY conference.* Davis, CA: University of Davis.

Hudson, R. (2007). *Networks of language: the new word grammar*. Oxford University Press.

Jurafsky, D. (1996). A probabilistic model of lexical and syntactic access and disambiguation. *Cognitive Science*, *20*, 137–194.

Just, M., & Carpenter, P. (1992). A capacity theory of comprehension: Individual differences in working memory. *Psychological Review*, *98*, 122–149.

Just, M. A., & Varma, S. (2007). The organization of thinking: What functional brain imaging

reveals about the neuroarchitecture of complex cognition. *Cognitive, Affective, and Behavioral Neuroscience*, *7*(3), 153–191.

Kahane, S., Nasr, A., & Rambow, O. (1998). Pseudo-projectivity: A polynomially parsable non-projective dependency grammar. In *Proceedings of ACL-COLING*.

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

Kliegl, R., Grabner, E., Rolfs, M., & Engbert, R. (2004). Length, frequency, and predictability effects of words on eye movements in reading. *European Journal of Cognitive Psychology*, *16*, 262–284.

Kliegl, R., Nuthmann, A., & Engbert, R. (2006). Tracking the mind during reading: The influence of past, present, and future words on fixation durations. *Journal of Experimental Psychology: General*, *135*, 12–35.

Kornai, A., & Pullum, G. K. (1990). The X-bar theory of phrase structure. *Language*, *66*(1), 24-50.

Kroch, A., & Joshi, A. (1985). *The linguistic relevance of Tree Adjoining Grammar* (Tech. Rep. No. MS-CIS-85-16). University of Pennsylvania.

Kurtzman, H. S. (1984, July). Ambiguity resolution in the human syntactic parser: an experimental study. In *Proceedings of the 10th international conference on computational linguistics and 22nd annual meeting of the association for computational linguistics* (pp. 481–485). Stanford, California, USA: Association for Computational Linguistics. Available from `http://www.aclweb.org/anthology/P84-1103`

Lackner, J., & Garrett, M. (1973). Resolving ambiguity: effects of biasing context in the unattended ear. *Cognition*, *1*(4), 359–372.

Levy, R. (2008). Expectation-based syntactic comprehension. *Cognition*, *106*(3), 1126–1177.

Lewis, R. L. (1993). *An Architecturally-based Theory of Human Sentence Comprehension*. Unpublished doctoral dissertation, Carnegie Mellon University, Pittsburgh, PA.

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

Lewis, R. L. (2000). Falsifying serial and parallel parsing models: empirical conundrums and an overlooked paradigm. *Journal of Psycholinguistic Research*, *29(2)*, 241–248.

Lewis, R. L., & Nakayama, M. (2001). Syntactic and positional similarity effects in the processing of Japanese embeddings. In M. Nakayama (Ed.), *Sentence Processing in East Asian Languages* (pp. 85–113). Stanford, CA.

Lewis, R. L., & Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science*, *29*, 1–45.

Logačev, P., & Vasishth, S. (2009). Morphological ambiguity and working memory. In P. de Swart & M. Lamers (Eds.), *Case, word order, and prominence: Psycholinguistic and theoretical approaches to argument structure.* Springer.

MacDonald, M. C., Pearlmutter, N. J., & Seidenberg, M. S. (1994). Lexical nature of syntactic ambiguity resolution. *Psychological Review*, *101*(4), 676–703.

Manning, C., & Schütze, H. (1999). *Foundations of statistical natural language processing*. Cambridge, MA: MIT Press.

Marcus, M. P. (1980). *A theory of syntactic recognition for natural language*. Cambridge, MA: MIT Press.

Marcus, M. P., Hindle, D., & Fleck, M. M. (1983, June). D-theory: Talking about talking about trees. In *Proceedings of the 21st annual meeting of the association for computational linguistics* (pp. 129–136). Cambridge, Massachusetts, USA: Association for Computational Linguistics. Available from `http://www.aclweb.org/anthology/P83-1020`

McClelland, J. L., & Kawamoto, A. H. (1986). Mechanisms of sentence processing: Assigning roles to constituents of sentences. In *Parallel distributed processing: Explorations in the microstructure of cognition* (pp. 272–325). Cambridge, MA: MIT Press.

McClelland, J. L., St. John, M., & Taraban, R. (1989). Sentence comprehension: A pdp approach. *Language and Cognitive Processes*, *4*, 287–336.

McElree, B., Foraker, S., & Dyer, L. (2003). Memory structures that subserve sentence comprehension. *Journal of Memory and Language*, *48*, 67–91.

Mel'čuk, I. A. (1988). *Dependency syntax: theory and practice*. SUNY Press.

Miller, G., & Chomsky, N. (1963). Finitary models of language users. In R. Luce, R. Bush, & E. Galanter (Eds.), *Handbook of mathematical psychology* (pp. 419–491). NY: John Wiley.

Mitchell, D. C. (1994). Sentence parsing. In M. Gernsbacher (Ed.), *Handbook of psycholinguistics* (pp. 375–411). New York: Academic Press.

Mitchell, D. C. (1995). Exposure-based models of human parsing: Evidence for the use of coarse-grained (nonlexical) statistical records. *Journal of Psycholinguistic Research*, *24*, 469–488.

Newell, A., & Simon, H. A. (1972). *Human problem solving.* Englewood Cliffs, New Jersey: Prentice-Hall.

Nivre, J. (2004). Incrementality in deterministic dependency parsing. In *Proceedings of the workshop on incremental parsing (ACL)* (p. 50-57).

Nivre, J. (2006). *Inductive dependency parsing.* Springer.

Nivre, J., & Nilsson, J. (2005). Pseudo-projective dependency parsing. In *Proceedings of ACL-COLING* (p. 99-106).

Park, J., & Brew, C. (2006). A finite-state model of human sentence processing. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the ACL* (pp. 49–56). Sydney, Australia.

Patil, U., Vasishth, S., & Kliegl, R. (2009). Compound effect of probabilistic disambiguation and memory retrievals on sentence processing: Evidence from an eye-tracking corpus. In A. Howes, D. Peebles, & R. Cooper (Eds.), *Proceedings of 9th International Conference on Cognitive Modeling.* Manchester, UK. Available from `http://www.ling.uni-potsdam.de/ vasishth/Papers/PatilEtAl.pdf`

Pinheiro, J. C., & Bates, D. M. (2000). *Mixed-effects models in S and S-PLUS.* NY: Springer-Verlag.

Pollard, C. J., & Sag, I. A. (1994). *Head-driven phrase structure grammar.* Chicago: University of Chicago Press.

Pollatsek, A., Reichle, E., & Rayner, K. (2006). Tests of the E-Z Reader model: Exploring the interface between cognition and eye-movement control. *Cognitive Pschology*, *52*, 1–56.

R Development Core Team. (2006). R: A language and environment for statistical computing [Computer program manual]. Vienna, Austria. Available from `http://www.R-project.org` (ISBN 3-900051-07-0)

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, *124*(3), 372–422.

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.

Reichle, E., Pollatsek, A., Fisher, D., & Rayner, K. (1998). Toward a model of eye movement control in reading. *Psychological Review*, *105*, 125–157.

Reichle, E., Warren, T., & McConnell, K. (2009). Using e-z reader to model the effects of higer-level language processing on eye movements during reading. *Psychonomic Bulletin & Review*, *16*, 1–21.

Roark, B. (2004). Robust garden path parsing. *Natural Language Engineering*, *10*(1), 1-24.

Roark, B., & Johnson, M. (1999, March). *Broad coverage predictive parsing.* Presented at the 12th Annual CUNY Conference on Human Sentence Processing.

Rohde, D. L. (2002). *A connectionist model of sentence comprehension and production.* Unpublished doctoral dissertation, Carnegie Mellon University.

Rosenbloom, P. S., Laird, J. E., & Newell, A. (1993). *The soar papers: research on integrated intelligence.* MIT Press.

Russell, S., & Norvig, P. (2003). *Artificial intelligence: A modern approach.* Upper Saddle River, NJ: Prentice Hall.

Salvucci, D. D., & Taatgen, N. A. (2008). Threaded cognition: An integrated theory of concurrent multitasking. *Psychological Review*, *115*(1), 101–130.

Shaher, R., Engelmann, F., Logacev, P., Vasishth, S., & Srinivasan, N. (2009). The integration advantage due to clefting and topicalization. In *Proceedings of GLOW in Asia.* Hyderabad, India.

Simon, H. (1955). A behavioral model of rational choice. *The Quarterly Journal of Economics*, *69*, 99–118.

Spiegelhalter, D. J. (2006). Two brief topics on modelling with WinBUGS. In *IceBUGS Conference Proceedings*. Hanko, Finland.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Linde, A. van der. (2002). Bayesian measures of model complexity and fit (with discussion). *Journal of the Royal Statistical Society*, *64*(B), 583–639.

Spivey, M. J., & Tanenhaus, M. K. (1998). Syntactic ambiguity resolution in discourse: modeling the effects of referential context and lexical frequency. *Journal of Experimental Psychology: learning, memory and cognition*, *24*(6), 1521–1543.

Stolcke, A. (1995). An efficient probabilistic context-free parsing algorithm that computes prefix probabilities. *Computational Linguistics*, *21*(2).

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

Tabor, W., & Tanenhaus, M. K. (1999). Dynamical models of sentence processing. *Cognitive Science*, *23*(4), 491–515.

Taylor, W. (1953). Cloze procedure: a new tool for measuring readability. *Journalism Quarterly*, *30*, 415–433.

Tesnière, L. (1959). *Éléments de syntaxe structurale*. Editions Klincksiek.

Van Dyke, J., & Lewis, R. L. (2003). Distinguishing effects of structure and decay on attachment and repair: A cue-based parsing account of recovery from misanalyzed ambiguities. *Journal of Memory and Language*, *49*, 285–316.

Van Dyke, J. A., & McElree, B. (2006). Retrieval interference in sentence comprehension. *Journal of Memory and Language*, *55*, 157–166.

Van Dyke, J. (2007). Interference effects from grammatically unavailable constituents during sentence processing. *Journal of Experimental Psychology: Learning Memory and Cognition*, *33*(2), 407–30.

Vasishth, S., & Lewis, R. L. (2006). Argument-head distance and processing complexity: Explaining both locality and antilocality effects. *Language*, *82*(4), 767-794. Available from `http://www.ling.uni-potsdam.de/ vasishth/Papers/LangVasLewFinal.pdf`

Wanner, E., & Maratsos, M. (1978). An ATN approach to comprehension. In M. Halle, J. Bresnan, & G. A. Miller (Eds.), *Linguistic theory and psychological reality* (pp. 119–161). Cambridge, Massachusetts: MIT Press.

Warren, T., & Gibson, E. (2002). The influence of referential processing on sentence complexity. *Cognition*, *85*, 79–112.

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

Appendix A
A probability model for the Nivre parser

Following Nivre (2006), a parser state is a record containing four fields:

σ	A stack of already-parsed unreduced words
τ	A list of input words
h	A function from dependent words to heads
d	A function from dependent words to arc types

The list τ serves as an input pointer, whereas h and d keep track of the dependency graph found so far. Control passes from a state to its successor by one of four possible operators: LEFT, RIGHT, REDUCE or SHIFT.

LEFT	A left-arc is drawn from the word being parsed i to the first word j of σ, making j the head of i; i is popped off σ.
RIGHT	A right-arc is drawn from the first word j of σ to the word being parsed i, making i the head of j; j is pushed onto σ.
SHIFT	Shifts the word being parsed i onto σ; no arcs are drawn.
REDUCE	Pops σ.

We use a maximum-entropy classifier to estimate the probability of a transition given a state (Bishop, 2006, 198). This classifier is trained on newspaper text from the Negra and Tiger treebanks (S. Brants et al., 2004). To obtain training data, we interpret the annotations in the treebanks as a dependency graphs by repeatedly applying German head-finding rules (Dubey, 2004). Since the 2006 vintage Nivre parser only works on projective dependency graphs, we occasionally need to post-process a given graph to render it projective. This happens primarily with punctuation, which does not figure in our PSC testing data. We make recalcitrant dependency graphs projective by repeatedly "lifting" the head of the shortest crossing dependency to the grandparent following Kahane, Nasr, and Rambow (1998). These transformations can later be undone to recover parser output (Nivre & Nilsson, 2005). From the dependency graph, we choose a canonical operator sequence based on the idea of delaying REDUCE actions as long as possible. Starting with 70,602 sentences from the Frankfurter Rundschau, this procedure yields about 3.1 million operator applications. Each such application records a 4-way classification problem instance. The correct class label names the observed operator application. The classifier estimates a discrete probability distribution over the four labels, given a finite feature vector summarizing particular aspects of the antecedent parser state. In this work we use three symbol-valued features.

Stack1	the top stack symbol, along with the part-of-speech of the current input word
Stack2	the top two stack symbols, along with the part-of-speech of the current input word
Stack3	the top three stack symbols, along with the part-of-speech of the current input word

These features allow the parser to achieve F-scores between 72.4 and 96.3 for unlabelled dependencies across the various k.[13] Table A1 depicts a ranking of typical errors in the PSC. Although the parser does not have perfect accuracy, the most common errors involve structures that would require lexical or pragmatic information for correct parsing.

Appendix B
ACT-R memory activation dynamics

We adopt the ACT-R activation-based memory theory as presented, e.g. on pages 70-81 of Anderson and Lebiere (1998), in assumption A3 of Lewis and Vasishth (2005) or in the section entitled "The Declarative Memory Module" in Anderson et al. (2004). Retrieval time for a word participating in an attachment is based on that word's activation, A, given below in Equation 8.

$$A_i = B_i + \sum_j W_j S_{ji} \tag{8}$$

[13] The F-score is the weighted harmonic mean of precision and recall per-attachment: $\frac{(2 \times precision \times recall)}{(precision + recall)}$. It is commonly used to determine overall parsing performance on a particular text sample (Manning & Schütze, 1999, p.269).

Activation is the sum of two quantities. One is the word's baseline activation B_i. Baseline activation, defined in Equation 9, is increased to a greater degree by recent retrievals at time t_j as compared to less recent retrievals that have decayed more. We follow standard practice in ACT-R in setting the decay rate d to 0.5; in virtually all ACT-R models, including the sentence processing model of (Lewis & Vasishth, 2005, 378), the decay parameter is set to this value (Anderson, 2005).

$$B_i = \ln\left(\sum_{j=1}^{n} t_j^{-d}\right) \tag{9}$$

One other factor affecting a word's—or more generally a chunk's—activation is known as similarity-based interference. Via the second addend in Equation 8, $\sum_j W_j S_{ji}$, similarity-based interference can decrease a word's activation when it is retrieved (e.g. to complete a dependency), if other words of the same grammatical category (hereafter, competitors) have already been parsed. In this term, W_j denote weights associated with the retrieval cues j that are shared with these competitor words in memory, and S_{ji}s are the strengths of association from cues j to word i. The weights W_j are not generally free parameters in ACT-R models but are set to G/j, where j is the number of cues involved during retrieval, and G is the total amount of goal activation available, also set by default to 1.

The second part of the similarity-based interference equation, the strength of association S_{ji}, is computed as follows:

$$S_{ji} = S_{\max} - \ln(\text{fan}_j) \tag{10}$$

In Equation 10, fan_j identifies the number of words that have the grammatical category in the cue j. If the cue j is shared across words with the same part of speech as the retrieved word i, then they are said to be similar. The maximum degree of association between similar items in memory is $S_{\max}$ which we set to 1.5 following Lewis and Vasishth (2005).

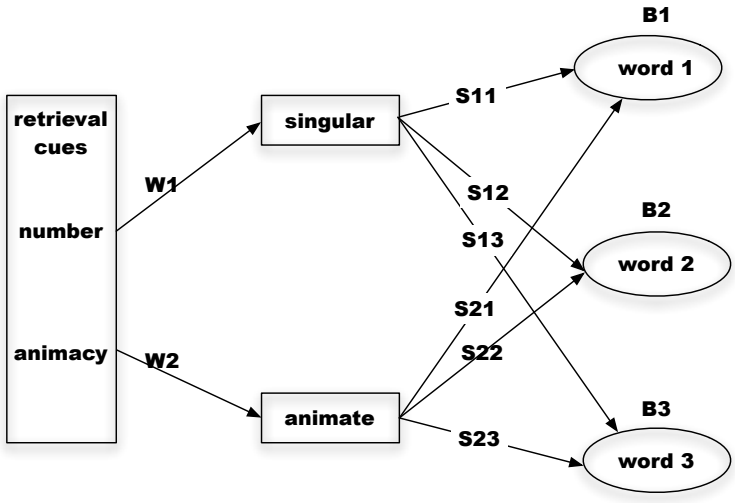


Figure B1. A schematic diagram illustrating how similarity-based interference works in ACT-R.

Similarity-based interference as defined above is best explained with the help of a schematic example (Figure B1). Assume for illustrative purposes that a noun must be

retrieved that has animacy status "animate" and number "singular", and that there are three previously processed nouns in memory that have these features. Each of the three nouns has base-level activation B_1-B_3 respectively. Since three nouns are involved, the fan is three, which gives us for each j, i: $S_{ji} = 1.5 - ln(3) = 1.5 - 1.098612 = 0.4013877$. This is turn gives us $\sum_j W_j S_{ji} = \sum_j 0.5 \times 0.4013877 = \sum_j 0.2006939$.

Contrast this with the situation where fan is 1 (only one noun matches the retrieval cues). In that case, $S_{ji} = 1.5 - 0 = 1.5$, and $\sum_j W_j S_{ji} = \sum_j 0.5 \times 1 == \sum_j 0.75$. In other words, activation of each noun would be lower when fan (amount of interference) is higher. As discussed in the main text, in the present model, the only source of inteference is part of speech.

At the other extreme, if the fan is high (5 or above), S_{ji} would be negative and net activation of a relevant item in memory could could in principle become negative as well. The sign of the activation value is not the important thing here but rather the relative reduction in activation as a function of fan. In ACT-R models, a retrieval threshold must be specified such that if the activation value of an item falls below this threshold, that item would fail to be retrieved from memory. Lewis and Vasishth (2005) set this value to -1.5.

In ACT-R, retrieval latency is an exponential function of activation. Equation 11 shows this relationship, which relates the temporal predictions we draw from the model to the theoretical quantity, activation. We follow Lewis and Vasishth (2005) in setting F to 0.14.

$$T_i = Fe^{-A_i} \tag{11}$$

Productions in ACT-R accrue a fixed cost of 50 milliseconds to execute. In adapting the Lewis and Vasishth (2005) model to the Nivre operators, we assume that a production has fired if a dependency arc has been drawn or a new word has been shifted. An additional production-firing cost of 50 ms also accrues if a decision is made to retrieve an item. This reflects the architecture of ACT-R memory buffers.

The time taken to integrate a word into the analysis is thus a joint function of up to two productions firing, along with any time spent performing a retrieval. Table B1 summarizes these time-costs. The REDUCE operation simply restructures a parser state and could, in principle, be dispensed with in favor of a much larger inventory of attachment actions. Like Lewis and Vasishth (2005) we abstract away from lexical retrieval costs and set the time it takes to read a word to just 1 ms.

Appendix C
Parametric bootstrap

Shown below is an illustration of the parametric bootstrap method used to confirm that the models with surprisal and retrieval provide better fits to the data than the baseline models. For illustrative purposes we show the comparison procedure for single fixation durations; we carried out the procedure for all reading time measures, however. To our knowledge, currently the R function for simulation does not work for generalized linear mixed models; therefore, we could not run the bootstrap on the regression probability models.

In the parametric bootstrap, the simpler model is treated as the null hypothesis, and the more complex model as the alternative hypothesis. Then, simulated dependent values (e.g., log reading time) are repeatedly generated from the null hypothesis model, and the simpler and more complex models are fit to the simulated data (as opposed to the actual data). After each simulation run, two times the difference of the log-likelihood ($2 \times$

$\Delta LogLik$) between the simpler and more complex models is recorded. After n simulations, this yields n $2 \times \Delta LogLik$ values, which constitute a distribution of log-likelihoods. The final step is to compute the proportion of cases in the n simulations where the simulated $2 \times \Delta LogLik$ is greater than the observed $2 \times \Delta LogLik$; this amounts to the probability of obtaining the observed $2 \times \Delta LogLik$, or a value greater than that, assuming that the null hypothesis is true. The R code for such simulations is shown in the appendix.

```
## null hypothesis:
sfd.0<-lmer(log(SFD)~scale(log_freq,scale=F) +
                      scale(len,scale=F) +
                      scale(bigram,scale=F) +
                      scale(logitpred,scale=F) +
             (1|sn) + (1|id),
             d.sfd, REML=FALSE)



## alternative hypothesis:
sfd.1<-lmer(log(SFD)~scale(log_freq,scale=F) +
                      scale(len,scale=F) +
                      scale(bigram,scale=F) +
                      scale(logitpred,scale=F) +
                      scale(surprisalDep,scale=F)+
            (1|sn) + (1|id),
             d.sfd, REML=FALSE)



## parametric bootstrap:
nrep <- 1000
lrstat <- numeric(nrep)

## simulated data from null hypothesis:

for(i in 1:nrep)
{
simulated <- unlist(simulate(sfd.0))
alt <- lmer(simulated~scale(log_freq,scale=F) +
                      scale(len,scale=F) +
                      scale(bigram,scale=F) +
                      scale(logitpred,scale=F) +
                      scale(surprisalDep,scale=F)+
            (1|sn) + (1|id),
             d.sfd, REML=FALSE)

null <- lmer(simulated~scale(log_freq,scale=F) +
                      scale(len,scale=F) +
                      scale(bigram,scale=F) +
                      scale(logitpred,scale=F) +
            (1|sn) + (1|id),
```

```
            d.sfd, REML=FALSE)

lrstat[i] <- 2*(logLik(alt)-logLik(null))
}
## critical value
critical <- 2*(logLik(sfd.1)-logLik(sfd.0))
## probability of observing the critical value or greater
## assuming the null hypothesis is true:
mean(lrstat>critical)
```

Table 9: Full results for multiple linear regressions, k=1 and k=100.

		k=1			k=100		
	Predictor	Coef	SE	t-value	Coef	SE	t-value
SFD	(Intercept)	5.2752152	0.0091924	573.9	5.276e+00	8.872e-03	594.7
	freq	-0.0077595	0.0013829	-5.6	9.824e-04	1.163e-03	0.8
	len	0.0060455	0.0006296	9.6	6.555e-03	5.691e-04	11.5
	bi	-0.0131116	0.0004734	-27.7	-1.164e-02	4.194e-04	-27.8
	pred	-0.0052508	0.0013195	-4.0	-1.709e-02	1.190e-03	-14.4
	surprisal	0.0123101	0.0012256	10.0	4.384e-02	1.998e-03	21.9
	retrieval	-0.0001691	0.0000159	-10.6	1.545e-04	8.504e-06	18.2
FFD	(Intercept)	5.269e+00	8.890e-03	592.7	5.273e+00	8.601e-03	613.1
	freq	-2.942e-03	1.254e-03	-2.3	5.982e-03	1.049e-03	5.7
	len	-2.040e-03	4.905e-04	-4.2	-1.390e-03	4.436e-04	-3.1
	bi	-1.321e-02	4.394e-04	-30.1	-1.224e-02	3.887e-04	-31.5
	pred	-1.664e-03	1.217e-03	-1.4	-1.475e-02	1.097e-03	-13.4
	surprisal	1.460e-02	1.096e-03	13.3	5.209e-02	1.793e-03	29.0
	retrieval	-1.519e-04	1.412e-05	-10.8	1.594e-04	7.538e-06	21.1
RPD	(Intercept)	5.471e+00	1.380e-02	396.4	5.472e+00	1.335e-02	410.0
	freq	-1.849e-02	1.749e-03	-10.6	-3.599e-03	1.466e-03	-2.5
	len	3.731e-02	6.960e-04	53.6	3.624e-02	6.328e-04	57.3
	bi	-6.696e-03	6.115e-04	-11.0	-7.235e-03	5.425e-04	-13.3
	pred	-1.441e-03	1.709e-03	-0.8	-1.658e-02	1.544e-03	-10.7
	surprisal	1.594e-02	1.557e-03	10.2	5.530e-02	2.534e-03	21.8
	retrieval	-2.864e-04	2.012e-05	-14.2	9.903e-05	1.060e-05	9.3
TRT	(Intercept)	5.454e+00	1.346e-02	405.1	5.454e+00	1.304e-02	418.3
	freq	-1.164e-02	1.646e-03	-7.1	2.640e-03	1.380e-03	1.9
	len	3.627e-02	6.555e-04	55.3	3.571e-02	5.958e-04	59.9
	bi	-1.249e-02	5.756e-04	-21.7	-1.233e-02	5.107e-04	-24.2
	pred	-1.635e-02	1.608e-03	-10.2	-3.034e-02	1.453e-03	-20.9
	surprisal	1.380e-02	1.466e-03	9.4	4.588e-02	2.386e-03	19.2
	retrieval	-2.776e-04	1.895e-05	-14.7	8.008e-05	9.976e-06	8.0
REG	(Intercept)	-2.1483075	0.0331565	-64.79	-2.103e+00	2.352e-02	-89.43
	freq	-0.1975074	0.0130669	-15.12	-1.479e-01	1.054e-02	-14.03
	len	0.0064848	0.0048715	1.33	5.261e-03	4.317e-03	1.22
	bi	0.0927631	0.0046218	20.07	7.030e-02	3.930e-03	17.89
	pred	0.0315408	0.0125868	2.51	-1.813e-03	1.105e-02	-0.16
	surprisal	0.0394327	0.0113952	3.46	1.689e-01	1.767e-02	9.56
	retrieval	-0.0002378	0.0001450	-1.64	2.645e-04	7.667e-05	3.45

Rank	Error type
1	Noun attachment
2	Prepositional Phrase attachment
3	Conjunction
4	Adverb ambiguity
5	Verb-second
6	Annotation error

Table A1: The most common parser errors.

Transition	Time
LEFT	50 ms + 50 ms + Retrieval Time
RIGHT	50 ms + 50 ms + Retrieval Time
SHIFT	50ms
REDUCE	0ms

Table B1: How time is determined in the parser.