Architecture matters:

What Soar has to say about modularity

Richard L. Lewis

January 29, 1993

According to Zenon Pylyshyn, Allen Newell took the high road in psychology by working on a theory of the mind intended to cover a wide range of cognitive behavior. Both Newell and Pylyshyn have put forth convincing arguments about the merits of such a research path, and the central role that cognitive architecture plays in it (Newell, 1973a; Newell, 1973b; Pylyshyn, 1984; Newell, 1990; Pylyshyn, 1993, this volume). Theories of architecture allow one to develop detailed models of local phenomena while addressing global issues about the mind. One such issue that Pylyshyn has raised with respect to Soar is modularity (Pylyshyn, 1991; Pylyshyn, 1993). The challenge is clear: how can a uniform theory such as Soar be right in the face of evidence that the mind is, in at least some respects, modular? As Pylyshyn points out, Newell (1990) began to answer this in his discussion of Fodor's 1983 monograph. There are two major parts to the answer. First, Soar can indeed admit additional processing modules since it is a bus-oriented system (the working memory is the bus); Newell (1990) speculates that imagery, for example, might require such a special system. Second, the present set of mechanisms underlying central cognition in Soar already exhibit some features of modularity. The purpose of this paper is to further explore modularity and Soar along this path by considering in some detail one area of psycholinguistics—sentence processing—where modularity has been a central concern, and asking how Soar could account for the pattern of results that has accumulated.

Modularity in sentence processing

In sentence processing (on-line parsing and comprehension), the basic modularity issue is whether there is an *autonomous syntactic parser* that operates automatically without appeal to semantic or contextual knowledge sources (Garfield, 1989). Such a component would be a prime example of a cognitive module (or an *input analyzer* in Fodor's terms). The alternative view is that comprehension is an *interactive* process (Marslen-Wilson, 1975) in which multiple knowledge sources (including syntax) interact to produce the meaning. The critical issue is not *whether* knowledge sources are applied, but *how*. Modularity does not claim that semantics and context are unnecessary parts of the comprehension process, simply that syntactic processing occurs independently of the other knowledge sources. Similarly, an interactive account does not deny the importance of syntactic knowledge, but does not grant it distinguished status as a separate module.

The primary way psycholinguists have attempted to address this issue is to study local ambiguity resolution. How are knowledge sources applied to resolve ambiguities?[1] The logic is as follows: if on-line syntactic structuring of ambiguous material is unaffected by context or semantics, then this suggests an autonomous syntax module. If on-line resolution *is* affected by nonsyntactic information, then this suggests an interactive comprehension architecture.

Do the data favor a modular or interactive architecture? Consider briefly just a few studies. Tyler & Marslen-Wilson (1977) present striking evidence in favor of an interactive architecture by demonstrating the rapid effect context can have on syntactic processing. Subjects heard sentence fragments like (1), ending with ambiguous strings (in italics).

(1) (a) If you walk too near the runway, *landing planes* ...

(b) If you've been trained as a pilot, *landing planes* ...

At the offset of the final word in the ambiguous phrase (*planes*), a probe word was visually presented. The word was a verb which was a continuation of the sentence. The subject's task was to name the verb. The contextually appropriate continuation is *are* for (1a) and *is* for (1b). Appropriate continuations had a naming latency advantage over inappropriate continuations, indicating that the context had a rapid effect on the initial analysis of the

[1] Such a question presupposes that human comprehension is *single path*, that is, it maintains a single interpretation at all times and therefore must select one interpretation at each ambiguous point. Alternatively, *limited path* theories hold that multiple interpretations are maintained in parallel (e.g., (Just & Carpenter, 1992)). Nevertheless, most of the important issues of modularity are relevant to limited path comprehenders as well.

ambiguous string. Further evidence of context effects come from Crain & Steedman (1985), who use a different technique (rapid grammaticality judgment) to show that the number of referents established in the context can affect the perceived grammaticality of a locally ambiguous sentence[2].

Evidence for syntactic modularity has accumulated as well. One well known study by Ferreira & Clifton (1986) presents evidence that ambiguous NP-V strings are always interpreted as subject-verb, regardless of the implausibility or contextual inconsistency of this reading. Material included sentences such as:

(2) (a) The evidence examined by the lawyer turned out to be unreliable.

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

In (2a), the inanimacy of *evidence* makes it implausible that the evidence was doing the examining. Yet, increased reading times (over unambiguous controls) were detected in *both* cases at the disambiguating region (the *by*-phrase), suggesting that subjects incorrectly interpreted the first verb as the main verb, regardless of the animacy of the initial NP.

Are we still waiting for the conclusive experiment to settle the modularity debate? The equivocal nature of the data is reflected is two relatively recent reviews of the sentence processing field (Frazier, 1987; Altmann, 1989). The reviewers come to rather different conclusions concerning modularity, a difference hardly due to the two year gap in publication. Should the field have taken more seriously Newell's warnings about pursuing simple binary

[2] Although these experiments have been criticized for not measuring immediate on-line processing, they still present challenging data for a modular approach; see (Steedman & Altmann, 1989).

oppositions in the absence of more complete processing theories (Newell, 1973b)? There has been indication that researchers are acknowledging that things are probably more complex than a modular/interactive dichotomy. For example, Britt et al. (1992) explores the possibility that the processing is different depending on the kind of syntactic construction. Holbrook et al. (1992) describe a model with a single processor and independent knowledge sources. The resulting theories look like a mix of interactive and modular approaches.

In fact, Just & Carpenter (1992) may have come close to providing data that settles the debate, though it does so by rendering it moot. Using material identical to (2) above, they found that some subjects *did* use semantic information on-line (appearing non-modular), while some subjects *did not* (appearing modular). This data seems hard to reconcile with either a purely modular or purely interactive theory, if such theories are taken to posit universal architectural principles. The first order characterization to be drawn from this data, along with the rest of the apparently conflicting data from the past two decades, is that human sentence processing can exhibit *both* modular and interactive effects[3].

[3]The main point of the Just and Carpenter study was to demonstrate a correlation between subject performance on the modularity task and the reading span task, and subsequently to explain the data by differences in working memory capacity. I shall not address this correlation here. It is not yet clear to what extent *all* modular/interactive effects can be attributed to capacity constraints. See also (Carpenter and Just) and (Young, 1993) in this volume.

Language comprehension in Soar

Consider now how Soar (Laird et al., 1987; Newell, 1990) comprehends language. Soar must map an incoming utterance to its contextualized meaning. It does this by applying *comprehension operators* to the words as they come in, producing a temporary meaning structure in working memory. Given the real time immediacy constraint (Just & Carpenter, 1987), Soar must accomplish this mapping with just a few (~2–5) operators per word. There is not time for more processing given the temporal mapping of Soar onto human cognition (operators take ~50–100 ms). Furthermore, the comprehension operators must bring all the relevant knowledge sources to bear—syntactic, lexical, semantic, contextual—since there are no other on-line processes that will apply this knowledge. Finally, these operators must arise via chunking over more deliberate comprehension spaces that represent the multiple knowledge sources. Soar treats language as a skill that gets better over time.

Figure 1 shows the basic structure of NL-Soar, a model which has the characteristics just sketched above (Newell, 1990; Lehman et al., 1991a; Lehman et al., 1991b; Lewis, 1993). The figure also shows that, in addition to a meaning representation (the situation model), NL-Soar produces a representation of the syntactic structure of the utterance (the utterance model), which functions to facilitate the semantic interpretation and temporarily hold the partial results of incremental syntactic analysis.

What is the nature of the comprehension operators? Given that two different models are built up by these operators during comprehension, one natural scheme is to have two different kinds of operators: *u-constructors*, which build the utterance model, and *s-constructors*, which build the situation model. U-constructors take as input the results of lexical access and the current utterance model, and produce an updated utterance model. S-constructors take as input the current utterance and situation models and produce an updated situation model[4].

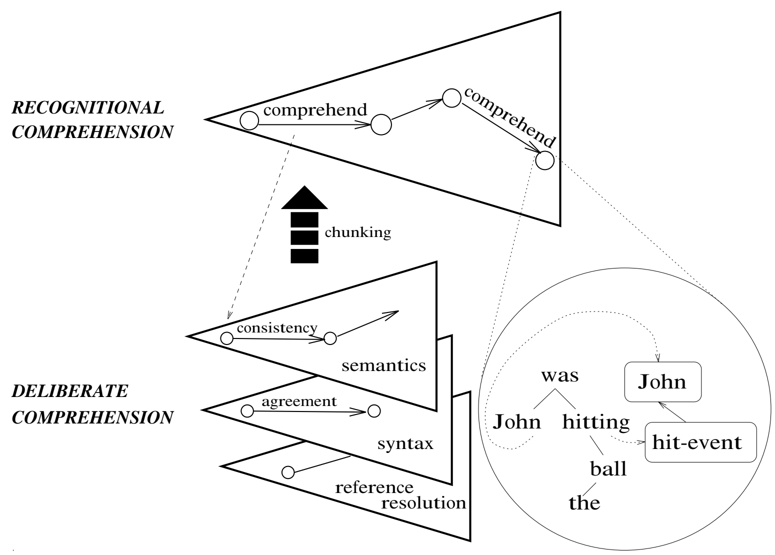

Figure 1: **The basic structure of NL-Soar.**

We can now describe the distribution of knowledge in the system in terms of problem space functions. Syntactic knowledge resides primarily in the *proposal* and *application* of the u-constructors. Semantic interpretation knowledge resides in the proposal and application of the s-constructors; in linguistic terms, this includes the knowledge that maps syntactic configurations or relations to thematic roles. Decomposing the knowledge this way across

[4]There is additional processing for reference resolution as well, not discussed here.

different operators (as opposed to a single fully integrated comprehension operator) increases the generality of the resulting proposal rules, thereby increasing the asymptotic efficiency of the system.

How does ambiguity arise? Choice points arise in problem spaces when multiple operators are applicable at a given state. Thus, syntactic ambiguity in NL-Soar arises when multiple u-constructor operators (corresponding to different syntactic paths) propose themselves. This generation of multiple alternatives occurs in parallel. The *selection* of the appropriate operator may now be effected by search control rules that encode semantic and contextual knowledge (see Figure 2). For example, in (2) above, a pair of u-constructors might be proposed in parallel at *examined*, corresponding to the main verb and reduced relative readings. A selection rule sensitive to the animacy of the initial NP might then fire, guiding comprehension down the appropriate path.

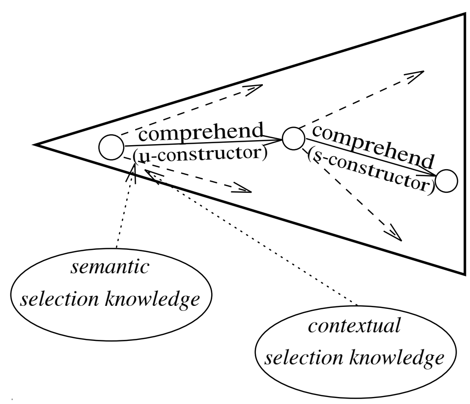

Figure 2: **Resolving ambiguity in NL-Soar.**

Is NL-Soar modular?

Viewed in terms of its relationship to central cognition, NL-Soar is unmistakably nonmodular. Comprehension processes are embedded in the same architectural mechanisms that effect all cognition. NL-Soar appears to be a classic example of a functionality implemented in a *horizontal* architecture (Fodor, 1983), an architecture that shares mechanisms and resources across domains. Yet there is clearly a syntactic module. The set of associations (productions) that realize the proposal and implementation of the u-constructor operators comprise a processing system with many of the distinguishing characteristics of modules. The operators are informationally encapsulated—they encode nothing but syntactic knowledge. The proposal and implementation are automatic (once chunked). The operators are sensitive only to the particular class of inputs that they recognize (via their specialized conditions), and perform one function only (parsing), with a fixed kind of representation as output (the utterance model). The application of the operators is mandatory (unless other equally practiced operators can be made available). Finally, the operators are fast (on the order of 100 ms). (See (Newell, 1990) for a similar discussion).

Consider how NL-Soar might exhibit interactive or modular behavior. The critical issues are the time course of comprehension, and what knowledge is available recognitionally. During normal to rapid comprehension, with only a few operators per word available, each problem space function—the proposal, selection, and application of the comprehension operators—must occur essentially by recognition. This means that the associations required to directly implement these functions must already have been learned. If the relevant semantic or contextual associations are in place to guide the selection of u-constructors (see again Figure 2), then NL-Soar will behave as an interactive system, with all knowledge sources immediately participating. If the relevant semantic or contextual associations are *not* in place, i.e., they haven't been learned yet, then NL-Soar will behave as a modular system, resolving the local ambiguity without recourse to these knowledge sources[5]. Note that NL-Soar is modeless—it is not waiting for its "interactive switch" to be turned on. Rather, the behavior it exhibits will span a continuum of modularity and interactivity depending on the current state of its operator selection knowledge, which consists of a large number of very specific associations.

The processing organization that emerges here is similar to other proposals in which a syntax module generates alternative structures in parallel, and semantic and contextual knowledge sources decide among them via a restricted interface (Warner & Glass, 1987; Altmann & Steedman, 1988). As Altmann (1988) points out, by allowing fine-grained control over a syntax module (such that parsing decisions can be influenced at the level of the word), it is possible to predict interactive effects while still preserving the modularity hypothesis. These models, along with NL-Soar, make clear the problem of experimentally falsifying modularity.

[5]What this doesn't explain is why there might still be strong syntactic preferences (such as minimal attachment or right association) when other knowledge sources are lacking. The extent to which there actually are such preferences is a complex issue that I cannot address here.

If the present model is so similar to these existing proposals, what else does it have to offer? Has Soar not contributed anything to the venture? In fact, Soar contributes a great deal:

1. The *parallel proposal* of multiple syntactic alternatives is a direct result of Soar's massively parallel recognition memory. All proposal associations for u-constructors fire in parallel.

2. The *fine-grained control* over the syntactic module is a direct result of embedding the processing in Soar's control structure. Control in an architecture exists wherever knowledge may be brought to bear to make decisions. Since control in Soar is obtained at the level of the operator (decision cycle), it makes the prediction that syntactic processing is controllable at that level. Temporally, this means on the order of 100 ms—at least at the word level, as opposed to, say, the phrasal or clausal level.

3. The *restricted interface* between the syntax module and the other knowledge sources is a direct result of Soar's representation of search control. Operator selection occurs via a fixed set of preferences that define a partial ordering among the proposed alternatives.

4. NL-Soar makes some novel qualitative predictions about the effect that *learning and experience* have on modular behavior. First, the more novel the semantic content and context for an utterance, the more likely modular effects will arise. This is because

the more novel the context, the less likely that the appropriate selection chunks will have been created beforehand. The corollary prediction is that modularity effects can be reduced with experience. There is even the possibility that modularity effects can be reduced with *instruction*, finally providing an explanation for how people are able to comprehend severe garden paths once they are carefully explained (a phenomenon also noted by (Gibson, 1991))[6].

5. Just as important as these gross predictions is the fact that NL-Soar permits posing questions about the impact of learning in the first place. Where do the comprehension operator selection rules come from (i.e., from what set of problem spaces could these associations have emerged by chunking)? What are the limits to the effects that large amounts of practice can have on the processes of language comprehension?

6. Finally, NL-Soar is embedded in a theory of cognition. NL-Soar must be a functional comprehender, not just a parsing module. This permits exploring comprehension in the service of actual tasks such as instruction taking (Lewis et al., 1989; Huffman & Laird, 1992), as well as considering general questions concerning the relationship of language and thought (Lehman et al., 1993a; Lehman et al., 1993b).

[6]Fodor (1983) dismisses such evidence as irrelevant to studying the normal operation of the language processor, noting that processing a garden path sentence "even feels different" from normal comprehension. This is just what one would expect from Soar given that the processing must first be *deliberate*, becoming recognitional only after some practice. Any one who has worked in the area of garden path sentences can attest to the fact that over time severe garden path effects can disappear.

While this account is promising, many important questions remain. One major gap in NL-Soar's modularity story regards the acquisition of language and the nature of innate content. Although NL-Soar presently lacks an acquisition theory, the general notion of innate problem spaces is perfectly consistent with Soar. Indeed, they are functionally required if Soar is to ever begin learning anything. For example, the initial spaces that give rise to the u-constructors might contain something recognizable as representing universal grammatical principles. Another gap in the theory concerns neural specificity. Soar does not have a theory of neural implementation, so issues of localization and selective impairment cannot be adequately addressed.

The modularity of Soar: Reprise

We started with a concern that perhaps Soar was too uniform and horizontal to be consistent with evidence suggesting some aspects of mind were in fact modular. By considering the Soar language capability, NL-Soar, we discovered that not only was NL-Soar *consistent* with many of the desirable properties of modular theories, but that these properties were largely *derived* from embedding comprehension within the Soar architecture. The resulting model pushes the state of the science (along with models such as CC-Reader (Just & Carpenter, 1992)) by beginning to address both modular and interactive effects, as well as allowing us to explore the relationship between modularity and learning.

What conclusions should we draw about Soar and modularity generally? One lesson is that some apparently modular processes may derive many of their distinguishing characteristics from features of a common cognitive architecture. But it is possible to turn this around into an interesting claim about central cognition itself. What Soar suggests is that the same set of mechanisms that underly fast, modular processing—the parallel recognition memory, the recognize-decide-act control structure, and chunking—is what central cognition consists of through and through (cf. (Fodor, 1983)). In sum, apparently nonmodular cognitive theories such as Soar may help capture important general principles of mental architecture, even if the mind *is* modular in significant ways. Onward down the high road!

References

Altmann, G. (1988). Ambiguity, parsing strategies, and computational models. *Language and Cognitive Processes*, 3:73–97.

Altmann, G. (1989). Parsing and interpretation: an introduction. *Language and Cognitive Processes*, 4:SI 1–19.

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

Britt, M. A., Perfetti, C. A., Garrod, S., & Rayner, K. (1992). Parsing in discourse: Context effects and their limits. *Journal of Memory and Language*, 31:293–314.

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

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

Fodor, J. A. (1983). *Modularity of Mind: An essay on faculty psychology*. MIT Press, Cambridge, MA.

Frazier, L. (1987). Sentence processing: a tutorial review. In Coltheart, M., editor, *Attention and Performance XII: The Psychology of Reading.* Lawrence Erlbaum Associates, Ltd., East Sussex, U.K.

Garfield, J. L., editor (1989). *Modularity in Knowledge Representation and Natural-Language Understanding*, Cambridge, MA. MIT Press.

Gibson, E. A. F. (1991). *A Computational Theory of Human Linguistic Processing: Memory Limitations and Processing Breakdown.* PhD thesis, Carnegie Mellon. Available as Center for Machine Translation technical report CMU-CMT-91-125.

Holbrook, J. K., Eiselt, K. P., & Mahesh, K. (1992). A unified process model of syntactic and semantic error recovery in sentence understanding. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, pages 195–200.

Huffman & Laird (1992). Dimensions of complexity in learning from interactive instruction. In Erikson, J., editor, *Proceedings of Cooperative Intelligent Robotics in Space III, SPIE Volume 1829.*

Just, M. A. & Carpenter, P. A. (1987). *The Psychology of Reading and Language Comprehension.* Allyn and Bacon, Boston, MA.

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

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

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

Lehman, J. F., Lewis, R. L., & Newell, A. (1991b). Natural language comprehension in Soar: Spring 1991. Technical Report CMU-CS-91-117, School of Computer Science, Carnegie Mellon University.

Lehman, J. F., Lewis, R. L., & Newell, A. (1993a). Architectural influences on language comprehension. In Pylyshyn, Z., editor, *Cognitive Architecture*. (Publisher under selection). To appear.

Lehman, J. F., Newell, A., Polk, T. A., & Lewis, R. L. (1993b). The role of language in cognition: A computational inquiry. In Harman, G., editor, *Conceptions of the Human Mind*. Erlbaum, Hillsdale, NJ. In press.

Lewis, R. L. (1993). *An Architecturally-based theory of Human Sentence Comprehension*. PhD thesis, Carnegie Mellon University. Available as Computer Science Tech Report CMU-CS-93-226.

Lewis, R. L., Newell, A., & Polk, T. A. (1989). Toward a Soar theory of taking instructions for immediate reasoning tasks. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, pages 514–521. Also in P. S. Rosenbloom, J. E. Laird, and A. Newell, eds., *The Soar Papers: Research on Integrated Intelligence*, MIT Press, Cambridge, MA, 1993.

Marslen-Wilson, W. D. (1975). Sentence perception as an interactive parallel process. *Science*, 189:226–227.

Newell, A. (1973a). Production systems: Models of control structures. In Chase, W. G., editor, *Visual Information Processing.* Academic Press, New York.

Newell, A. (1973b). You can't play 20 questions with nature and win: Projective comments on the papers of this symposium. In Chase, W. G., editor, *Visual Information Processing.* Academic Press, New York.

Newell, A. (1990). *Unified Theories of Cognition.* Harvard University Press, Cambridge, Massachusetts.

Pylyshyn, Z. W. (1984). *Computation and Cognition.* Bradford / MIT Press, Cambridge, MA.

Pylyshyn, Z. W. (1991). The role of cognitive architecture in theories of cognition. In VanLehn, K., editor, *Architectures for Intelligence.* Lawrence Erlbaum, Hillsdale, NJ.

Pylyshyn, Z. W. (1993). Cognitive architecture: Why and how it should be studied. In Steier, D. & Mitchell, T., editors, *Mind Matters: Contributions to Cognitive and Computer Science in Honor of Allen Newell.* Erlbaum, Hillsdale, NJ. To appear.

Steedman, M. & Altmann, G. (1989). Ambiguity in context: A reply. *Language and Cognitive Processes*, 4:SI 105–122.

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

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

Young, R. M. (1993). Functionality matters: Capacity constraints and Soar. In Steier, D. & Mitchell, T., editors, *Mind Matters: Contributions to Cognitive and Computer Science in Honor of Allen Newell.* Erlbaum, Hillsdale, NJ. To appear.