An Integrated Theory of the Mind

John R. Anderson and Daniel Bothell
Psychology Department
Carnegie Mellon University
Pittsburgh, PA 15213
(412) 268-2781
ja@cmu.edu
db30@andrew.cmu.edu

Michael D. Byrne
Psychology Department
Rice University
Houston, TX 77005
byrne@acm.org

Christian Lebiere
Human Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, PA 15213
cl@cmu.edu

Friday, September 27, 2002

Abstract

There has been a proliferation of proposed mental modules in an attempt to account for different cognitive functions but so far there has been no successful account of their integration. ACT-R (Anderson & Lebiere, 1998) has evolved into a theory that consists of multiple modules but also explains how they are integrated to produce coherent cognition. We discuss the perceptual-motor modules, the goal module, and the declarative memory module as examples of specialized systems in ACT-R. These modules are associated with distinct cortical regions. These modules place chunks in buffers that project to the basal ganglia, which implement ACT-R's procedural system, a production system that responds to patterns of information in the buffers. At any point in time a single production rule is selected to respond to the current pattern. This serial bottleneck in production-rule selection enables the coordination that results in an organized control of cognition. Subsymbolic processes serve to guide the selection of rules to fire as well as the internal operations of (some) modules and much of learning involves tuning of these subsymbolic processes. We describe empirical examples that demonstrate the predictions of ACT-R's modules and also examples that show how these modules result in strong predictions when they are brought together in models of complex tasks. These predictions require little parameter estimation and can be made for choice data, latency data, and brain imaging data.

Psychology, like other sciences, has seen an inexorable movement towards specialization. This is seen in the proliferation of specialty journals in the field but also in the proliferation of special-topic articles in this journal, which is supposed to serve as the place where ideas from psychology meet. Specialization is a necessary response to complexity in a field. Along with this move to a specialization in topics studied, there has been a parallel move toward viewing the mind as consisting of a set of specialized components. With varying degrees of consensus and controversy there have been claims for separate mechanisms for processing visual objects versus locations (Ungerleider & Miskin, 1982), for procedural versus declarative knowledge (Squire, 1987), for language (Fodor, 1987), for arithmetic (Dehaene, Spelke, Stanescu, Rinel, & Tsivkin, 1999), for categorical knowledge (Warrington & Shallice, 1984), and for cheater detection (Cosmides & Tooby, 2000), to name just a few.

While there are good reasons for at least some of the proposals for specialized cognitive modules[1], there is something unsatisfactory about the result—an image of the mind as a disconnected set of mental specialties. One can ask "how is it all put back together?" An analogy here can be made to the study of the body. Modern biology and medicine have seen a successful movement towards specialization responding to the fact that various body systems and parts are specialized for their function. However, because the whole body is readily visible, the people who study the shoulder have a basic understanding how their specialty relates to the specialty of those who study the hand and the people who study the lung have a basic understanding of how their specialty relates to the specialty of those who study the heart. Can one say the same of the person who studies categorization and the person who studies on-line inference in sentence processing or of the person who studies decision making and the person who studies motor control?

Newell (1990) argued for **cognitive architectures** that would explain how all the components of the mind worked to produce coherent cognition. In his book he described the Soar system, which was his best hypothesis about the architecture. We have been working on a cognitive architecture called ACT-R (e.g., Anderson & Lebiere, 1998) which is our hypothesis about such an architecture. It has recently undergone a major development into a version called ACT-R 5.0 and this form offers some important new insights into the integration of cognition. The goal of this paper is to describe this new version of the theory and draw out its implications for the integration of mind.

Before describing ACT-R and the picture it provides of human cognition, it is worth elaborating more on why a unified theory is needed and there is no better way to begin than with the words of Newell (1990):

A single system (mind) produces all aspects of behavior. It is one mind that minds them all. Even if the mind has parts, modules, components, or whatever, they all mesh together to produce behavior. Any bit of behavior has causal tendrils that extend back through large parts of the total cognitive system before grounding in the environmental situation of some earlier times. If a theory covers only one part or component, it flirts with trouble from the start. It goes without saying that there are dissociations, independencies, impenetrabilities, and modularities. These all help to break the web of each bit of behavior being shaped by an unlimited set of antecedents. So they are important to understand and help to make that theory simple enough to use. But they don't remove the necessity of a theory that provides the total picture and explains the role of the parts and why they exist (pp. 17-18).

Newell then goes onto enumerate many of the advantages that a unified theory has to offer; we will highlight a particular such advantage in the next subsection.

Integration and Application

The advantage we would like to emphasize is that unification enables tackling of important applied problems. If cognitive psychologists try to find applications for the results of isolated research programs they either find no takers or extreme misuse (consider, for instance, what has happened with research on left-right hemispheric asymmetries in Education). Applications of psychology, such as education, require that one attend to the integration of cognition. Educational applications do not respect the traditional divisions in cognitive psychology. For instance, high-school mathematics involves reading and language processing (for processing of instruction, mathematical expressions, and word problems), spatial processing (for processing of graphs and diagrams), memory (for formula and theorems), problem solving, reasoning, and skill acquisition. To bring all of these aspects together in a cognitive model one needs a theory of the cognitive architecture (Anderson, 2002).

Other domains of application are at least as demanding of integration. One of them is the development of cognitive agents (Freed, 2000). These applications involve assembling large numbers of individuals to interact; prominent among these are group training exercises. Another domain is multi-agent video games and other interactive entertainment systems. In many cases it is difficult to assemble the large number of individuals required to provide the desired experience for some individual. The obvious solution is to provide simulated agents in a virtual environment. In many cases it is critical that the simulated agents provide realistic behavior in terms of cognitive properties. The demand is to have entities that can pass a limited Turing test.[2] Another application area which requires integrated treatment of human capabilities is human factors/human-computer interaction (see Byrne, 2003, for a review of cognitive architectures in this area). This field is concerned with behavior in complex tasks such as piloting commercial aircraft and using CAD systems. Such behavior involves the full spectrum of cognitive, perceptual, and motor capabilities.

Salvucci's Driving Example

Salvucci's (2001b) study of the effect of cell phone use on driving (see Figure 1) illustrates the use of cognitive models to test the consequences of artifacts and their interactions, and illustrates how integrated approaches and applied problems lead to a somewhat different and sterner measure of whether theory corresponds to data than typically applied in psychology. Of course, there have been a number of empirical studies on this issue and Salvucci subsequently conducted a number of these as a test of his model. However, he took as a challenge case whether he could predict a priori the effects of a cell phone's use in a particular situation. If cognitive models are to be useful in this domain they should truly predict results rather than being fit to data. He already had developed an ACT-R model of driving (Salvucci, Boer, & Liu, 2001) and for this task he developed a model of using one of a variety of cell phones. He put these two models together to get predictions of the effects of driving on cell phone use and cell phone use on driving. Significantly, he did this without estimating any parameters to fit the data because he had not yet collected any data. He was using established ACT-R parameters.[3]

It should also be emphasized that his model actually controls a driving simulator and actually dials a simulated cell phone. While his ACT-R model does not have eyes it is possible to reconstruct what the eyes would see from the code that constructs the representation for a human driver in the simulator. Similarly while the model does not have hands it is possible to insert into the input stream the results that would happen had the wheel been turned or a button pressed. ACT-R has a theory of how perceptual attention encodes information from the screen and a theory of how manual actions are programmed.

While Salvucci has subsequently looked at more complex cell phone use, in this study he was interested in dialing the phone. He compared four ways of dialing: full manual, speed manual, full voice, and speed voice. Figure 2a shows the effect of driving on the use of various cell phone modes. Figure 2b shows results that he obtained in a subsequent experiment. The correspondence between model and data is striking. Being able to predict behavior in the absence of parameter estimation is a significant test of a model. In many applications, it is also a practical necessity.

Of course, there is relatively little interest in the effect of driving on cell phone use; rather the interest is in the converse. Salvucci collected a number of different measures of driving. Figure 3 shows the results for mean lateral deviation from the center of the lane. Comparing the predictions in Figure 3a with the data in Figure 3b yields a classic glass half-full, half-empty result: The model succeeds in identifying that only the full-manual condition will have a significant impact on this measure. Much research in psychology would be satisfied with predicting the relative order of conditions. However, the absolute predictions of the model are way off. The ACT-R model is driving much better and would lead to unrealistic expectations about the performance of real drivers. This shows that ACT-R and Salvucci's model are works in progress and indeed Salvucci has made progress since this report. However, the failings are as informative as the successes in terms of illustrating what a cognitive architecture must do. Note that Salvucci could have tried to re-estimate parameters to make his model fit—but the goal is to have predictions in advance of the data and parameter estimation.

What a Cognitive Architecture Must be Able to Do

More generally, what properties must a cognitive architecture strive for if it is to deliver on its promise to provide an integrated conception of mind? The example above illustrates that one will not understand human thought if one treats it as abstract from perception and action. As many have stressed (e.g., Greeno, 1989; Meyer & Kieras, 1997), human cognition is embodied and it is important to understand the environment in which it occurs and people's perceptual and motor access to that environment.

Applications, particularly involving the development of cognitive agents, stress two other requirements. First, the worlds that these agents deal with do not offer a circumscribed set of interactions as occurs in a typical experiment. They emphasize the need for robust behavior in the face of error, the unexpected, and the unknown. Achieving this robustness goal is not just something required for development of simulated agents; it is something required of real humans and an aspect of cognition that laboratory psychology typically ignores.

Second, Salvucci's application stresses the importance of a priori predictions. Rather than just predicting qualitative results or estimating parameters to predict quantitative predictions the ideal model should predict absolute values without estimating parameters. Psychology has largely been content with qualitative predictions but this does not leave it in a favorable comparison to other sciences. Requiring a priori predictions of actual values without any parameter estimation seems largely beyond the accomplishments of the field but this fact should make us strive harder. Model fitting has been criticized (Roberts & Pashler, 2000) because of the belief that the parameter estimation process would allow any pattern of data to be fit. While this is not true, such criticisms would be best dealt with by simply eliminating parameter estimation.

Perhaps the greatest challenge to the goal of a priori predictions is the observation that behavior in the same experiment will vary with factors such as population, instructions, and state of the participants (motivated, with or without caffeine, etc.). In other sciences, results of experiments vary with contextual factors (in chemistry with purity of chemicals, how they are mixed, temperature) and the approach is to measure the critical factors and have a theory of how they affect the outcome, without estimating situation-specific parameters. Psychology should strive for the same.[4]

One major way to deal with such variability in results is to have a theory of learning that predicts how an individual's past experience and the current experience shape behavior. Learning is at center stage in applications such as producing cognitive agents. A major criticism of the simulated agents that inhabit current environments is that they do not learn and adjust their behavior with experience like humans do. Of course, learning is also at center stage for any application concerned with education.

The ACT-R 5.0 Architecture

ACT-R claims that cognition emerges as the consequence of an interaction between specific units of procedural knowledge and specific units of declarative knowledge. The units of declarative knowledge are called **chunks** and represent things remembered or perceived. For instance, a chunk may represent the fact that 2+3=5 or that Boston is the capital of Massachusetts. For driving, chunks may represent numerous types of knowledge such as situational awareness (e.g. "there is a car to my left"), navigational knowledge (e.g. "Broad St. intersects Main St."), or driver goals and intentions (e.g. "stop for gas at the next traffic light"). Procedural knowledge encodes the processes and skills necessary to achieve a given goal. The units of procedural knowledge are called **productions**, condition-action rules that "fire" when the conditions are satisfied and execute the specified actions. The conditions can depend on the current goal to be achieved, on the state of declarative knowledge (i.e. recall of a chunk), and/or the current sensory input from the external environment. Similarly, the actions can alter the state of declarative memory, change goals, or initiate motor actions in the external environment. Below is an English statement of a production rule from the driving model in Salvucci et al:

> IF my current goal is to encode a distant perceptual point for steering
> and there is a tangent point present (i.e., we are entering or in a curve)

THEN shift attention to this point and encode its position and distance.

The first test in the condition above would be a test of the goal and the second of the contents of the visual system. The action of this production requests that a visual representation be built.

Figure 4 illustrates the basic architecture of ACT-R 5.0. There are a set of **modules** devoted to things like identifying objects in the visual field, controlling the hands, retrieving information from declarative memory, or keeping track of current goals and intentions. The central production system is not sensitive to most of the activity of these modules but rather can only respond to information that is deposited in the **buffers** of these modules. For instance, people are not aware of all the information in the visual field but only the object they are currently attending to. Similarly, people are not aware of all the information in long-term memory but only the fact currently retrieved. Each module makes this information available as a chunk in a buffer. As illustrated in Figure 4 the core production system can recognize patterns in these buffers and make changes to these buffers – as for instance, when it makes a request to perform an action in the manual buffer. In the terms of Fodor (1983) the information in these modules is largely encapsulated and they communicate only through the information they make available in their buffers.

The theory is not committed to exactly how many modules there are but a number have been implemented as part of the core system. The buffers of these modules hold the chunks that the production system can respond to. Particularly important are the goal buffer, the retrieval buffer, two visual buffers, and a manual buffer. The goal buffer, which we associate with dorsolateral prefrontal cortex (DLPFC), keeps track of one's internal state in solving a problem. The retrieval buffer, in keeping with the HERA model (Nyberg, Cabeza, & Tulving, 1996) and other recent neuroscience models of memory (e.g., Buckner, Kelley, & Petersen, 1999; Wagner, Pare-Blagoev, Clark, & Poldrack, 2001), is associated with the ventrolateral prefrontal cortex (VLPFC) and holds information retrieved from long-term declarative memory.[5] This distinction between DLPFC and VLPFC is in keeping with a number of neuroscience results (Petrides, 1994; Fletches & Henson, 2001; Thompson-Schill et al., 1997; Braver et al, 2001; Cabeza, et al, 2002). The other three modules/buffers are all based on Byrne and Anderson's (2001) ACT-R/PM, which in turn is based on Meyer and Kieras's (1997) EPIC. The manual buffer is responsible for control of the hands and is associated with the adjacent motor and somatosensory cortical areas devoted to controlling and monitoring hand movement. One of the visual buffers, associated with the dorsal "where" path of the visual system, keeps track of locations while the other, associated with the ventral "what" system, keeps track of visual objects and their identity. The visual and manual systems are particularly important in many tasks that ACT-R has dealt with like a participant scanning a computer screen, typing, and using a mouse at a keyboard. There also are rudimentary vocal and aural systems.

Each of the buffers can hold a relatively small amount of information. Basically, the content of a buffer is a chunk. Chunks that were former contents of buffers are stored in declarative memory. In this way ACT-R can remember, for instance, objects it has attended to or solutions to goals that it has solved.

The buffers are conceptually similar to Baddeley's (1986) working memory "slave" systems. While the central cognitive system can only sense the buffer contents, the contents of chunks that appear in these buffers can be determined by rather elaborate systems within the modules. For instance, the chunks in the visual buffers represent the products of complex processes of the visual perception and attention systems. Similarly, the chunk in the retrieval buffer is determined by complex long-term memory retrieval processes, as we will describe.

Bringing the Buffers Together

ACT-R 5.0 includes a theory of how these buffers interact to determine cognition. The basal ganglia and associated connections are thought to implement production rules in ACT-R. The cortical areas corresponding to these buffers project to the striatum, part of the basal ganglia, which we hypothesize performs a pattern-recognition function (in line with other proposals e.g., Amos 2000; Frank, Loughry, & O'Reilly 2000; Houk & Wise, 1995; Wise, Murray, & Gerfen, 1996). This portion of the basal ganglia projects to a number of small regions known collectively as the pallidum. The projections to pallidum are substantially inhibitory and these regions in turn inhibit cells in the thalamus, which projects to select actions in the cortex. Graybiel and Kimura (1995) have suggested that this arrangement creates a "winner-lose-all" manner such that active striatal projections strongly inhibit only the pallidum neurons representing the selected action (which then no longer inhibit the thalamus from producing the action). This is a mechanism by which the winning production comes to dominate. According to Middleton and Strick (2000), at least 5 regions of the frontal cortex receive projections from the thalamus and are controlled by this basal ganglia loop. These regions play a major role in controlling behavior.

Thus, the basal ganglia implement production rules in ACT-R by the striatum serving a pattern-recognition function, the pallidum a conflict-resolution function, and the thalamus controlling the execution of production actions. Since production rules represent ACT-R’s procedural memory this also corresponds to proposals that basal ganglia subserve procedural learning (Ashby & Waldron, 2000; Hikosaka Nakahara, Rand, Sakai, Lu, Nakamura, Miyachi, & Doya, 1999; Saint-Cyr, Taylor, & Lang, 1988). An important function of the production rules is to update the buffers in the ACT-R architecture. The well-characterized organization of the brain into segregated, cortico-striatal-thalamic loops is consistent with this hypothesized functional specialization. Thus, the critical cycle in ACT-R is one in which the buffers hold representations determined by the external world and internal modules, patterns in these buffers are recognized and a production fires, and the buffers are then updated for another cycle. The assumption in ACT-R is that this cycle takes about 50 msec to complete – this estimate of 50 msec as the minimum cycle time for cognition has emerged in a number of cognitive architectures including Soar (Newell, 1990), CAPS (Just & Carpenter, 1992), and EPIC (Meyer & Kieras, 1997).

The architecture assumes a mixture of parallel and serial processing. Within each module there is a great deal of parallelism. For instance, the visual system is simultaneously processing the whole visual field and the declarative system is executing a parallel search through many memories in response to a retrieval request. Also, the processes within different modules can go on in parallel and asynchronously. However, there are also two levels of serial bottlenecks in the system. First, the content of any buffer is limited to a single declarative unit of knowledge, a chunk. Thus, only a single memory can be retrieved at a time or a single object encoded from the visual field. Second, only a single production is selected at each cycle to fire. In this second respect, ACT-R 5.0 is like Pashler’s (1998) central bottleneck theory and quite different, at least superficially, from the other prominent production system conceptions (CAPS, EPIC, and Soar). The end of the paper will return to the significance of these differences.

Subsequent sections of the paper will describe the critical components of this model – the perceptual-motor system, the goal system, the declarative memory, and the procedural system. However, now that there is a sketch of what ACT-R 5.0 is, we would like to close this section by noting the relationship between the ACT-R 5.0 system described in Figure 4 and earlier ACT systems.

Brief History of the Evolution of the ACT-R Theory

ACT systems have historical roots in the HAM theory (Anderson & Bower, 1973) of declarative memory. ACT was created by marrying this theory with a production system theory of procedural memory (Newell, 1973b). Significant earlier embodiments of that theory were ACT-E (Anderson, 1976) and ACT* (Anderson, 1983). By the time ACT* had been formulated, a distinction had been made between a symbolic and subsymbolic level of the theory. The symbolic level consisted of the formal specification of the units of declarative memory (chunks) and the units of procedural memory (productions). The subsymbolic level consisted of the specification of continuously varying quantities that controlled the access to chunks and productions. In the case of declarative memory these quantities have always been referred to as **activations,** which reflect the past patterns of usage of the chunk. In the case of procedural memory the subsymbolic quantities have had various names but are currently called **utilities** that reflect the reinforcement history of the productions.

ACT-R (Anderson, 1993) emerged as a result of marrying ACT with the rational analysis of Anderson (1990) that claimed that cognition was adapted to the statistical structure of the environment. The insight was that one could use rational analysis to design the subsymbolic computations that controlled access to information. According to rational analysis, the subsymbolic components were optimized with respect to demands from the environment, given basic computational limitations.

A somewhat incidental aspect of the initial formulation of ACT-R in 1993, called ACT-R 2.0, was that a running simulation of the system was distributed. Owing to the increased power of computers, the relatively new standardization of Common Lisp in which ACT-R 2.0 was implemented, and the growth of understanding about how to achieve an effective and efficient simulation, this was the first widely available and functional version of the ACT architecture. A worldwide user community arose around that system. The system was no longer the private domain of our theoretical fancies and had to serve a wide variety of needs and serve as the basis of communication among researchers. This had a major catalytic effect on the theory and served to drive out the irrelevant and awkward assumptions. ACT-R 4.0 (Anderson & Lebiere, 1998) emerged as a cleaned-up version of ACT-R 2.0. It includes an optional perceptual-motor component called ACT-R/PM (Byrne & Anderson, 1998). While there are disagreements about aspects of ACT-R 4.0 in the community, it has become a standard for research and exchange. ACT-R 5.0 reflects a continued development of the theory in response to community experience.

Over 100 models have been published by a large number of researchers based on the ACT-R 4.0 system. This productivity is a testimonial to the intellectual gain to be had by a well-working integrated system. Table 1 summarizes the research areas covered by these models; detailed information is available from the web site act-r.psy.cmu.edu. A major commitment in the development of ACT-R 5.0 is that the models developed in 4.0 still work so that 5.0 constitutes cumulative progress.

Differences between ACT-R 4.0 and 5.0

ACT-R 5.0 differs from ACT-R 4.0 in four principal ways. First, there have been some simplifications in the parameters and assumptions of the architecture. Some assumptions of ACT-R 4.0, while they seemed good ideas at the time, were not being exploited in the existing models and were sometimes being actively avoided. These were eliminated.[6] Also as more models were developed it became apparent that there were some constraints on the parameter values that worked across models. These parameter constraints have moved us closer to the goal of parameter-free predictions and enabled an effort like Salvucci's described earlier.

Second, the tentative brain mapping illustrated in Figure 4 was not part of ACT-R 4.0. However, it seemed that ACT-R could be mapped onto the segregated, cortico-striatal-thalamic loops that had been proposed by a number of theorists, quite outside of ACT-R. Given this mapping it is now possible to deploy neuroimaging data to make novel, demanding tests of the ACT-R theory.

Third, a key insight that this mapping onto the brain brought with it was the module-buffer conception of cognition. This enabled a more thorough integration of the cognitive aspects of ACT-R with the perceptual-motor components of ACT-R/PM. This complete integration and consequent embodiment of cognition is a significant elaboration of ACT-R 5.0 over ACT-R 4.0. Within most of the ACT-R community, which was relatively content with ACT-R 4.0 and which is not concerned with data from cognitive neuroscience, it is this module-buffer conception that has received the most attention and praise.

Fourth, there now is a mechanism in ACT-R for learning new production rules, which has participated in a number of successful models of skill acquisition. While other versions of ACT have had production-rule learning mechanisms that worked in modeling circumscribed experimental tasks they failed to work in large-scale simulations of skill learning. As we will describe in the section on ACT-R's procedural system, the successful definition of a general production-system learning mechanism also depended on the move to the buffer-based conception of production rule execution.

These changes have not been without consequence for the theory. One of the consequences has been the treatment of declarative retrieval. Information retrieved from long-term declarative memory is an important part of the condition of production rules. For instance, in trying to simplify a fraction like 4/12 a critical test is whether there is a multiplication fact asserting that the numerator is a factor of the denominator (i.e. 4x3 =12 in this case). In all previous versions of ACT this test was performed by a single production requesting a retrieval (e.g., 4 x ? = 12), then waiting to see if the retrieval request was successful, and, if it was, examining what chunk was retrieved. This was implemented by an awkward process in which all production processing was suspended until the retrieval effort ran to completion. In contrast, in ACT-R 5.0 one production can make a retrieval request in its action and other productions can be fired while the retrieval request is being processed. When the retrieval is complete the result will appear in the retrieval buffer and another production can harvest it and take appropriate action. This is perfectly analogous to how switches of visual attention functioned in ACT-R/PM where one production would make a request that attention move to an object and another production will respond to the result when attention has switched. This makes the retrieval system interruptible and there is evidence (Byrne, 2000) that retrievals can be interrupted.

Another significant consequence of the changes concerns the treatment of goals. In ACT-R 4.0 the condition of each production rule had to specify a test of the goal. The system was widely perceived as too goal-focused and not sufficiently interruptible. Now, in ACT-R 5.0 specification of the goal is optional in a production rule just as the specification of the contents of any buffer is optional. Different rules can refer to different subsets of the buffers and whatever rule has the highest utility will execute. Thus, new information inserted into the perceptual buffers from the environment can evoke rules that take precedence over rules that respond to the current goal.

For purposes of organizing a more detailed exposition of the theory and relevant evidence it is useful to break the system into its four major pieces – the perceptual-motor system, the goal system, the declarative system, and the procedural system. These will form the next four major sections of the paper.

The Perceptual-Motor System

As a matter of division of labor, not as a claim about significance, ACT-R historically was focused on higher-level cognition and not perception or action. Perception and action involve systems every bit as complex as higher-level cognition. Dealing with cognition had seemed quite enough. However, this division of labor tends to lead to a treatment of cognition that is totally abstracted from the perceptual-motor systems and there is reason to suppose that the nature of cognition is colored by the systems it interacts with and that there are not clean breaks between perception, central cognition, and action. In some laboratory tasks and a great many applied tasks much of the timing and details of the behavior depend on the perceptual-motor systems. Thus, we cannot achieve our goal of a priori predictions without some consideration of the perceptual and motor processes.

With their EPIC architecture, Meyer and Kieras (1997) developed a successful strategy for relating cognition to perception and action without dealing directly with real sensors or real effectors and without having to embed all the detail of perception and motor control. This is a computational elaboration of the successful Model Human Processor system defined by Card, Moran, and Newell (1983) for human-computer-interaction applications. This approach involves modeling, in approximate form, the basic timing behavior of the perceptual and motor systems, the output of the perceptual systems, and the input to the motor system. We have adopted exactly the same strategy and to a substantial degree just re-implemented certain aspects of the EPIC system. Undoubtedly, this strategy of approximation will break down at points but it has proven quite workable and has had a substantial influence on the overall ACT system. We would hope that the architecture that has emerged would be compatible with more complete models of the perceptual and motor systems.

Byrne and Anderson (1998, 2001) took much of the EPIC perceptual-motor system and embedded it in an extension to ACT-R 4.0 called ACT-R/PM. That system has greatly influenced the design of ACT-R 5.0 where many of the modules in 5.0 were originally ACT-R/PM modules. The major difference between ACT-R/PM's perceptual-motor machinery and EPIC's is in the theory of the visual system. The ACT-R visual system separates vision into two modules, each with an associated buffer. A visual-location module and buffer represent the dorsal "where" system and a visual-object module and buffer represent the ventral "what" system. ACT-R implements more a theory of visual attention than a theory of perception in that it is concerned with what the system chooses to encode in its buffers, but not the details of how different patterns of light falling on the retina yield particular representations. In addition, on the issue of whether attention is object-based or location-based it implements the emerging conclusion that it is both (e.g., Egly, Driver, & Rafal, 1994; Humphreys, Olson, Komani, & Riddoch, 1996; Vecera & Farah, 1994).

When a production makes a request of the "where" system, the production specifies a series of constraints, and the "where" system returns a chunk representing a location meeting those constraints. Constraints are attribute-value pairs which can restrict the search based on visual properties of the object (such as "color: red") or the spatial location of the object (such as "screen-y greater-than 153"). This is akin to so-called "pre-attentive" visual processing (Triesman & Gelade, 1980) and supports visual pop-out effects. For example, if the display consists of one green object in a field of blue objects, the time to determine the location of the green object is constant regardless of the number of blue objects.

A request to the "what" system entails providing a chunk representing a visual location, which will cause the what system to shift visual attention to that location, process the object located there, and generate a declarative memory chunk representing the object. The system supports two levels of granularity here, a coarse one where all attention shifts take a fixed time regardless of distance, and a more detailed one with an eye-movement model. For the fixed-time approximation, this parameter is set at 185 msec in ACT-R and serves as the basis for predicting search costs in situations where complete object identification is required.[7] However, ACT-R does not predict that all visual searches should require 185 ms/item. Rather, it is possible to implement in ACT-R versions of feature-guided search that can progress more rapidly. There is considerable similarity between the current implementation of visual attention in ACT-R and Wolfe's GS theory (Wolfe, 1994) and indeed we plan to adapt Wolfe's GS into ACT-R. Just to make clear the contrast with EPIC, the EPIC visual system does not implement the what-where distinction and cannot do feature-based searches.

The more detailed version is based on Salvucci's (2001a) EMMA system, and is based on a number of models of eye-movement control in reading, particularly the E-Z Reader model (Reichle, Pollatsek, Fisher, and Rayner 1998; Reichle, Rayner, & Pollatsek 1999). In EMMA, the time between the request for a shift of attention and the generation of the chunk representing the visual object of that location is dependent on the eccentricity between the requested location and the current point of gaze, with nearer objects taking less time than farther objects. The theory assumes that eye movements follow shifts of attention and that the ocular-motor system programs a movement to the object. If the object is not encoded before that program is complete there will be an eye movement. According to this model eye movements typically lag visual attention by about 200 msec because it takes that long to program and execute an eye movement. Salvucci has tested this theory in developing the driver model that we described earlier. Again, to make clear the differences with EPIC, EPIC does not distinguish between eye movements and visual attention.

A Model for Menu Selection

Menu selection turns out to be an interesting domain for testing ideas about visual search. Nilsen (1991) provided data for a task that involved selecting a digit from a menu of the digits 1-9 randomly arrayed vertically. The data concerned the times for participants to move a mouse from the home position above the menu to the target item. Figure 5 shows the time for this action as a function of the serial position of the item in the menu. The best-fitting linear function to these data has a slope of 103 msec per position.

These results depend on the fact that the items in the menu are ordered randomly. Since the participant does not know where the target item is, a critical component to latency has to be a visual search of the list looking for the target item. Participants tend to move the mouse down as they scan for the target (mouse movement and eye scanning data confirm simultaneous movement). Thus, once they identify the target, the distance to move the mouse tends not to vary much with serial position. Thus, when the target position is unknown, time is dominated by visual search. In contrast, if the position of the item is known in advance (as in a fixed order menu) the critical latency component should be a Fitts law (Fitts, 1954) description of the motion. In this case time would be a logarithmic function of distance. Nilsen has data from such a condition to confirm this relationship.

An ACT-R model has been developed for this task (Anderson, Matessa, & Lebiere, 1997) that assumes that, given a target, participants selected one of its features and scanned down the menu for the first item with that feature. If this was the target they stopped. If not they scanned for the next item that contained the target feature. The two critical productions in this model were:

Hunt-Feature
IF the goal is to find a target that has feature F
and there is an unattended object below the current location with feature F
THEN move attention to the closest such object.

Found-Target
IF the goal is to find a target
and the target is at location *L*
THEN move the mouse to *L* and click.

The first production **Hunt-Feature** moves attention down looking at objects that have a feature in common with the target. The movement of attention to an object will cause its identity to be encoded. If it is an instance of the target letter **Found-Target** can apply. The production **Found-Target** will retrieve the location of the target and move the mouse to that location.

The time to reach a target will be a function of the number of digits that precede it and have the selected feature. Given the McClelland-Rumelhart feature set (McClelland & Rumelhart, 1981), there is a .53 probability that a randomly selected feature of one number will overlap with the feature set of another number. Using the 185 msec for a shift of attention ACT-R predicts 185*.53 = 98 msec per menu item, which is close to the slope, 103 msec., in the Nilsen data. The fit of the ACT-R model to the data is illustrated in Figure 5. This is a striking demonstration of how the ACT-R theory can be used to predict new data sets using old parameters.[8]

This theory also makes a non-obvious prediction which is that menu search should be faster to the extent that the target tends to have different features than foils, and this prediction has been confirmed. For instance, Anderson, Matessa, and Lebiere (1997) report an experiment showing that menu search is faster when searching for a letter among digits than a letter among letters (letters tend not to overlap in features as much with digits as they do with other letters). Anderson et al. also found that menu search for numbers is faster among letters than among numbers.

Subsequent research on this task (Byrne, Anderson, Douglass, & Matessa, 1999), collecting eye movements and relating them to search times, has shown that the control process is more complex than described in the simple model above. Byrne (2001) presents a more sophisticated learning-driven ACT-R model which captures many of these complexities.

ACT-R Modules in Parallel

The ACT-R model described by Byrne and Anderson (2001) for the Schumacher et al. (1997; also reported in Schumacher et al. 2001) experiment is a useful illustration of how the perceptual-motor modules work together. It involves interleaving multiple perceptual-motor threads and has little cognition to complicate the exposition. The experiment itself is interesting because it is an instance of perfect time-sharing. It involved two simple choice reaction time tasks: 3-choice (low-middle-high) tone discrimination with a vocal response and 3-choice (left-middle-right) visual position discrimination with a manual response. Both of these tasks are simple and can be completed rapidly by experimental participants. Schumacher, et al. (1997) had experimental participants train on these two tasks separately, and they reached average response times of 445 ms for the tone discrimination task and 279 ms for the location discrimination task. Participants were then asked to do the two tasks together with simultaneous stimulus presentation and encouraged to overlap processing of the two stimuli. In the dual-task condition, they experienced virtually no dual-task interference—283 ms average response time for the visual-manual task and 456 ms average response time for the auditory-vocal task.

We constructed an ACT-R/PM model of the two tasks and the dual-task. A schedule chart for the dual-task model is presented in Figure 6. Consider the visual-motor task first. There is a quick 50 msec detection of the visual position (does not require object identification), a 50 msec production execution to request the action, followed by the preparation and execution of the motor action. With respect to the auditory-vocal task, there is first the detection of the tone (but according to the parameters inherited from EPIC this takes longer than detection of visual position), then a production executes requesting the speech, and then is a longer but analogous process of executing the speech. According to the ACT-R model, there is nearly perfect time sharing between the two tasks because the demands on the central production system are offset in time. Figure 7 presents the predictions of the ACT-R model for the task. There is an ever-so-small dual-task deficit because of variability in the completion times for all the perceptual-motor stages, which occasionally results in a situation where the production for the auditory-vocal task must wait for the completion of the visual-motor production.

This model nicely illustrates the parallel threads of serial processing in each module, which is a hallmark of EPIC and ACT-R. Figure 6 also illustrates that the central production-system processor is also serial, a feature that distinguishes ACT-R from EPIC. However, in this experiment there was almost never contention between the two tasks for access to the central processor (or for access to any other module).

The Goal Module

While human cognition is certainly embodied, its embodiment is not what gives human cognition its advantage over that of other species. Its advantage depends on its ability to achieve abstraction in content and control. Consider a person presented with the numbers 64 and 36. As far as the external stimulation is concerned, this presentation affords the individual a variety of actions – adding the numbers, subtracting them, dialing them on a phone, etc. Human ability to respond differently to these items depends on knowledge of what the current goal is and to be able to sustain cognition in service of that goal without any change in the external environment. Suppose the goal is to add the numbers. Assuming that one does not already have the sum stored one will have to go through a series of substeps in coming up with the answer and to do this one has to keep one's place in performing these substeps and keep track of various partial results such as the sum of the tens digits. The goal module has this responsibility of keeping track of what these intentions are so that behavior will serve that goal. The goal buffer that holds this representation of intention is associated with the dorsolateral prefrontal cortex. A classic symptom of prefrontal damage is contextually inappropriate behavior such as when a patient responds to the appearance of a comb by combing their hair. DLPFC has also been known to track amount of subgoaling in tasks like Tower of London (Newman, Carpenter, Varma, & Just, in press) and Tower of Hanoi (Fincham, Carter, vanVeen, Stenger & Anderson, 2002).

Modeling the Tower of Hanoi Task

The Tower of Hanoi task (Simon, 1975) has been a classic paradigm for behavioral studies of goal manipulations. A number of the most effective strategies for solving this problem require that one keep a representation a set of subgoals. We (Anderson & Douglass, 2001) have explicitly trained participants to execute a variant of what Simon (1975) called the sophisticated perceptual strategy in which one learns to set subgoals in order to place disks – thus, a participant might reason "In order to move disk 4 to peg C I have to move disk 3 to peg B, and in order to do this I have to move disk 2 to peg C, and in order to do this I have to move disk 1 to peg B" . In this example the participant must keep track of goals to move four disks and this "stack" of goals can be represented "4-3-2-1". Numerous behavioral studies have shown that accuracy and latency is strongly correlated with maintaining this goal stack.

Fincham et al. performed a study to see how various brain areas would respond to the task. Of particular interest, given the correspondences in Figure 4, was how the dorsolateral prefrontal cortex responded to the need to maintain subgoals. They performed an experiment in which participants had to make a move every 16 seconds (with intervening time filled in by a distractor activity). Figure 8 displays the percent rise in activation found in the right DLFPC for each of a sequence of 8 moves. The labels on the abscissa indicate the goals that participants had to maintain for each move. Figure 8 also displays the predictions of a model that says each goal maintained should increase the percent rise by .05. The correspondence is quite striking.

The goal module also serves to create new abstract chunks. For instance, after the sum of 36 and 64 has been calculated the goal buffer will hold the fact that 36+64 = 100. The chunk in the goal buffer holding this information will be stored in declarative memory. Chunks formed in any buffer will be stored in long-term memory but the goal module is the only source of abstract chunks. This process of storing results of past goals allows the system to later retrieve these results and so bypass the need to calculate the answer. In this way the goal system serves to enable problem-solving by retrieval as described by Logan (1988).

One can build chunks in the goal buffer that contain other chunks as sub-elements. So the goal module is the only system that can create complex hierarchies of chunks where one chunk is part of another. Language processing is a major consumer of this capability. Every sentence that comes in creates a comprehension goal which, if successful, will result in a hierarchical representation of the meaning of that sentence (Anderson, Budiu, & Reder, 2001). Others have also identified a general structure-building function for dorsolateral cortex and also for the inferior frontal gyrus (Newman, Just, Keller, & Carpenter, submitted).

The processing of goals seems to be what separates humans from other animals. Goal processing provides much of the ability to form abstract and hierarchical representations that Marcus (2001) has identified as key elements to human cognition. The ACT-R model of instruction taking that we will describe later makes heavy use of the goal system.

An issue of some concern to the ACT-R community has been how to think about subgoaling. ACT-R 4.0 had a goal stack that remembered past intentions and could reset the goal to these when the current goal had been achieved. It now appears that goals are simply stored in declarative memory and later retrieved as the participant chooses to work on them (Anderson & Douglass, 2001; Altmann & Trafton, 2002). Memory for such goals appears to obey all the properties of other declarative memories. This adds further evidence to the view that in many ways the goal system and declarative memory are strongly coordinated. The HERA (Nyberg, Cabeza, & Tulving, 1996) model holds that declarative storage and retrieval processes are organized from the prefrontal structures close to those that maintain goal representations.

The Declarative Memory Module

The declarative memory module can retrieve records of chunks that were formed in the various perceptual-motor buffers but also more abstract chunks that were formed in the goal buffer. ACT-R makes chunks active to the degree that past experiences indicate that they will be useful at this particular moment. Using a common formula in activation theories, the activation of a chunk is a sum of a base-level activation, reflecting its general usefulness in the past, and an associative activation, reflecting its relevance to the current context. The activation of a chunk *i* is defined as:

$$A_i = B_i + \sum_j W_j S_{ji} \qquad \textbf{Activation Equation}$$

where B_i is the base-level activation of the chunk *i*, the W_j's reflect the attentional weighting of the elements that are part of the current goal, and the S_{ji}'s are the strengths of association from the elements j to chunk i. Figure 9 displays the chunk encoding that 8+4=12 and its various quantities (with W_j's for four and eight, assuming that they are sources). The activation of a chunk controls both its probability of being retrieved and its speed of retrieval. We will develop the mapping of activation to speed of retrieval in the example below.

According to the ACT-R theory the base-level activation of a memory trace rises and falls with practice and delay according to the equation:

$$B_i = \ln\left(\sum_{j=1}^{n} t_j^{-d}\right)$$ **Base-Level Learning Equation**

Where t_j is the time since the jth practice of an item. This equation is based on the rational analysis of Schooler and Anderson (1991) studying how the pattern of past occurrences of an item predicts the need to retrieve it. They found that the above equation reflects the log odds an item will reoccur as a function of how it has appeared in the past. In developing ACT-R, we assumed that base-level activation would track log odds. Each presentation has an impact on odds that decays away as a power function (producing the power law of forgetting) and different presentations add up (it turns out producing the power law of practice—see Anderson, Fincham & Douglass, 1999). In the ACT-R community .5 has emerged as the default value for the parameter d over a large range of applications. This base-level learning equation has been the most successfully and frequently used part of the ACT-R theory.

Modeling a Fan Experiment

Historically, the ACT theory of declarative retrieval has focused on tasks that require participants to retrieve facts from declarative memory. The second experiment in Pirolli and Anderson (1985) is a good one to illustrate the contributions of both base-level activations ($\underline{B}_i$) and associative strengths ($\underline{S}_{ji}$) to the retrieval process. This was a fan experiment (Anderson, 1974) in which participants were to try to recognize sentences such as "A hippie was in the park". The number of facts (i.e., fan) associated with the person (e.g., hippie) could be either 1 or 3 and the fan associated with the location could be either 1 or 3. Participants practiced recognizing the same set of sentences for 10 days. Figure 10 illustrates how to conceive of these facts in terms of their chunk representations and subsymbolic quantities. Each oval in Figure 10 represents a chunk that encodes a fact in the experiment. As a concept like hippie is associated with more facts, there are more paths emanating from that concept and, according to ACT-R, the strengths of association $\underline{S}_{ji}$ will decrease.

Figure 11 illustrates how the activations of these chunks vary as a function of fan and amount of practice. There are separate curves for different fans, which correspond to different associative strengths ($\underline{S}_{ji}$). The curves rise with increasing practice because of increasing base-level activation. Figure 12 illustrates the data from this experiment. Participants are slowed in the presence of greater fan but speed up with practice. The practice in this experiment gets participants to the point where high-fan items are recognized more rapidly than low-fan items were originally recognized. Practice also reduces the absolute size of the effect of fan but it remains substantial even after 10 days of practice.

According to the ACT-R theory the effect of fan is to reduce the strength of association, $\underline{S_{ji}}$, from a term like hippie to the chunk encoding a fact. As argued in Anderson and Lebiere (1998), the strength of association can be calculated by $\underline{S - \ln(\text{fan})}$ where S is a parameter to be estimated. In Anderson and Reder (1999), we used values of S around 1.5 in fitting the data in that paper and this is the value used for fitting the data in Figure 12. The effect of practice is to increase the base-level activation of the facts. According to Anderson and Lebiere (1998), an item with $\underline{n}$ units of practice will have an approximate base-level activation of $\underline{.5*\ln(n)}$ and this is what was used in fitting the data. Figure 11 shows the activation values that are gotten from combining the base-level activation with the associative activation according to the Activation Equation, setting the weights, $\underline{W_j}$, in this experiment to .333 (as used in Reder and Anderson, because each of the three content terms (hippie, in, park) in the sentence gets an equal 1/3 source activation). These are parameter-free predictions for the activation values. As can be seen, they increase with practice with low-fan items having a constant advantage over high-fan items.

According to the ACT-R theory these activation values can be mapped onto predicted recognition times according to the equation:

$$\text{Recognition Time} = I + Fe^{-A_i}$$

where I is an intercept time reflecting encoding and response time and F is a latency scale factor. Thus, fitting the model required estimating two parameters and these were $\underline{I}$ = 597 ms. and $\underline{F}$ = 890 ms., which are quite similar to the parameters estimated in Reder and Anderson (1999). The value of $\underline{I}$ is also quite reasonable as the time to encode the words and emit a response (key press). The overall quality of fit is good with a correlation of .986. Moreover, this correlation does not depend on the parameter estimates $\underline{I}$ and $\underline{F}$ but only on $\underline{e^{-A_i}}$, which means that it measures a parameter-free prediction of ACT-R. The effect of $\underline{I}$ and F is only to scale this critical quantity onto the range of the latencies—although, as noted earlier with respect to Salvucci's predictions, having a priori constraints on such scaling parameters can be critical.

While this example illustrates the ACT-R theory of declarative memory it is by no means the only example. This part of the theory has been perhaps the most successful enjoying applications to list memory (Anderson, Bothell, Lebiere, & Matessa, 1998), implicit memory (Lebiere & Wallach, 2001), category learning (Anderson & Betz, 2002), sentence processing (Anderson, Budiu & Reder, 2001), and individual differences (Lovett, Daily & Reder, 2000) among other domains. The theory of declarative memory gives a natural account of the explicit-implicit distinction. Explicit memories refer to specific declarative chunks that can be retrieved and inspected. Implicit memory effects reflect the subsymbolic activation processes that govern the availability of these memories. This is substantially the same theory of memory as that of Reder's SAC theory (Reder & Gordon, 1997).

An interesting development in ACT-R 5.0 has been the discovery that the declarative memory component has come to behave like other sensory buffers. Retrieving a memory from the past is a similar process to attending to an object from the visual display; that is, retrieving a memory is like perceiving the past. Indeed the equations used by Wolfe (1994) in his theory of visual attention are like the equations used in ACT-R's declarative memory for retrieving a chunk. Both combine a bottom-up effect (in ACT-R's case base-level activations; in Wolfe's case features being searched for) with a contextually driven top-down component (in ACT-R's case associative component; in Wolfe's case context-determined salience).

Procedural Memory

As described so far, ACT-R consists of a set of modules that progress independently of one another. This would be a totally fragmented concept of cognition except for the fact that they make information about their computations available in buffers. The production system can detect the patterns that appear in these buffers and decide what to do next to achieve coherent behavior. The acronym ACT stands for Adaptive Control of Thought and this section will describe how the production system achieves this control and how it is adaptive.

The issue of control in ACT-R might seem trivial. Production rules can be viewed as hard symbolic rules and it might seem that they specify what to do when certain conditions match with no possibility of variation. However, this is not the ACT-R conception of things. In many situations the contents of various buffers will vary continuously and so the concept of the conditions of a production matching has to be taken as a matter of degree. The ACT-R community has mainly focused on the partial matching of the contents of declarative memory. It has been shown that partial matching can produce a number of classic memory errors such as reversals in serial recall or false alarms in a memory test (e.g., Anderson, Bothell, Lebiere, & Matessa, 1998).

To add to the non-determinism of the production-rule selection, multiple productions might have their conditions matched on a single cycle, partly because partial matching stretches the set of acceptable rules. However, even if ACT-R were only an exact matching system, the system could still get into indeterminate states because multiple rules can have their conditions matched in the current buffers. Since multiple production rules match and only one can be executed it is necessary to select among the competitors in a process called **conflict resolution**. ACT-R selects among the multiple production rules according to their utilities. Production rule utilities are noisy, continuously-varying quantities just like declarative activations and play a similar role in production selection as activations play in chunk selection. The utility of a production i is defined as:

$$U_i = P_i G - C_i$$

where P_i is an estimate of the probability that if production i is chosen the current goal will be achieved, G is the value of that current goal (typically estimated at 20 sec.), and C_i is the estimate of how much cost (typically measured in time that will have to be spent to achieve that goal). As we will discuss, both P_i and C_i are learned from experience with that production rule.

The utilities associated with a production are noisy and on a cycle-to-cycle basis there is a random variation around the expected value given above. The highest valued production is always selected but on some trials one might randomly be more highly valued than another. If there are n productions that currently match, the probability of selecting the ith production is related to the utilities U_i of the n production rules by the formula:

$$\Pr ob(i) = \frac{e^{U_i/t}}{\sum_j^n e^{U_j/t}}$$

where the summation is over all applicable productions and t is related to the standard error of the variance of the noise by the formula:

$$\sigma = \pi t / \sqrt{6}$$

Thus, at any point in time there is a distribution of probabilities across alternative productions reflecting their relative utilities. The value of σ is about 1 sec. in our simulations and this is emerging as a reasonable setting for this parameter.

Learning mechanisms adjust the costs C_i and probabilities P_i that underlie the utilities U_i according to a Bayesian framework. Because the example that we will describe concerns learning of the probabilities, we will expand on that but the cost learning is similar. The estimated value of P is simply the ratio of successes to the sum of successes and failures:

$$P = \frac{Successes}{Successes + Failures}$$

However, there is a complication here that makes this like a Bayesian estimate. This complication concerns how the counts for Successes and Failures start out. It might seem natural to start them out at 0. However, this means that P is initially not defined and after the first experience the estimate of P will be extreme at either the value 1 or 0 depending on whether the first experience was a success or failure. To provide some inertia in the speed with which the probability estimates converge on the empirical proportions of successes, Successes and Failures start out at initial values α and β, respectively, yielding the following formulas:

$$Successes = \alpha + m$$
$$Failures = \beta + n$$

where m is the number of experienced successes and n is the number of experienced failures. This leads to an estimate of P that starts out equal to a prior α/(α+β) and moves to the empirical m/(m+n) as experience accumulates. The larger the sum α+β is, the slower is the shift from the prior probability to the empirical probability.

Model for the Building-Sticks Task

The experiment by Lovett (1998) is a good one for illustrating the learning of these subsymbolic procedural parameters. Participants solved building-stick problems like the one illustrated in Figure 13. This is an isomorph of Luchins waterjug problem (Luchins, 1942) that has the advantage that it cannot be solved by mental addition but requires actual exploration (and hence the trial-and-error problem solving can be observed). Participants are given an unlimited supply of building sticks of three lengths and are told that their goal is to create a target stick of a particular length. They can choose one of two basic strategies–they can either start with a stick smaller than the desired length and add sticks (like the addition strategy in Luchins waterjugs) or they can start with a stick that is too long and "saw off" lengths equal to various sticks until they reach the desired length (like the subtraction strategy). The first is called the undershoot strategy and the second is called the overshoot strategy. Participants show a strong tendency to hillclimb and choose as their first stick the one that will get them closest to the target stick. So, in the example in Figure 13 will tend to choose the overshoot strategy since the resulting stick gets them closest to the target.

In these problems only one of the two operators works and participants cannot achieve a stick of the desired length by the other operator. One of the two operators is more often successful—different participants experience different operators as more successful. Table 2 shows the distribution of experience. In cases where undershoot proves more successful, undershoot always works on all of the problems that look like undershoot should work and 50% (or 75% in the extreme condition) of the problems that look like overshoot should work. Figure 14 shows percent use of the more successful strategy as a function of problem bias. Problem bias refers to whether the problem looked like that operator got the student closer to the target problem. Participants show a strong hillclimbing bias and tend to select the operator that gets them closest to the goal. As they learn which operator is more successful they come to use it more and more. Initially they used the more successful operator on problems that looked the other way only 10% of the time but by the end of the experiment they were using it 60% of the time in the extreme-biased condition.

Figure 14 also shows the predictions of an ACT-R model that consists of 4 critical productions (given in Table 3) that chose overshoot and undershoot as well as other productions that execute the plans. The four critical productions consist of two that select overshoot and two that select undershoot. The difference between the two productions for an operator is that one is context-sensitive and only applies if the problem looks that way while the other is context-free and applies regardless. ACT-R does a good job in accounting for this shift in probabilities of choice. The ACT-R model was fit to these data by fixing the parameters α and β for the context-free productions and β for the context-sensitive productions at 0.5 and by estimating the remaining critical production parameter, the context-sensitive productions' α. The best-fitting value for the context-sensitive productions' α is 10.68. More details of the model fitting are available in Lovett (1998).

In terms of the critical production rules, ACT-R decreases its evaluation of the less successful productions and increases its evaluations of the more successful productions (e.g., decide-overshoot and force-overshoot). Table 3 documents what happens to the *P* values of these productions. The first column of that table shows the initial *P* value for the context-free productions as 0.5 (based on the priors, $\frac{\alpha}{\alpha+\beta} = \frac{0.5}{0.5+0.5}$) and the initial *P* value for the context-sensitive productions as 0.96 (based on the priors, $\frac{\alpha}{\alpha+\beta} = \frac{11.36}{11.36+0.5}$). This represents an initial preference for using the context-sensitive productions, i.e., choosing the strategy toward which the stick lengths are biased. Since the approach that looks closest will not always lead to a solution, however, the corresponding context-sensitive production will experience a certain number of failures (depending on the condition). Also, with noise in the production values, there is always some chance that a less successful production will be attempted; this allows the system to gather at least some information about the success of all of the four critical productions. After 90 trials of experience (the end of the experiment), the productions' *P* values will have been adjusted based on this information (see Table 3). Note that in both conditions, the production corresponding to the more successful approach (within both the context-free and context-sensitive pairs) had a higher evaluation. Moreover, in the extreme-biased condition, this preference for the more successful production was stronger than in the biased condition. This example illustrates how the utility-learning mechanism enables ACT-R to adapt to the method that is more successful.

Production Rule Learning and Learning from Instruction

In the above example a number of things were prespecified including the productions that would do the task. These essentially amount to degrees of freedom in constructing a model and these degrees of freedom keep us from our ultimate goal of a priori predictions. However, we could eliminate these degrees of freedom if we could specify the process by which these production rules were learned. Taatgen and Anderson (in press) have developed a production learning mechanism called **production compilation**, which shows considerable promise. It bears some similarity to the chunking mechanism in Soar (Newell, 1990) and is basically a combination of composition and proceduralization as described in Anderson (1983) for ACT*. Production compilation can be illustrated with respect to a simple paired-associate task. Suppose the following pair of production rules fire in succession to produce recall of a paired associate:

IF reading the word for a paired-associate test
   and a word is being attended
THEN retrieve the associate of the word

IF recalling for a paired-associate test
   and an associate has been retrieved with response N
THEN type N

They might apply for instance when the stimulus "vanilla" is present, recall the paired-associate "vanilla-7" and produce "7" as an answer. Production compilation would try to collapse these two productions into one. But what is to be done about the retrieval – the second production has to wait for the retrieval requested by the first production? The solution is to build a production rule that has the product of this retrieval built into it. Thus, ACT-R learns the following production rule:

IF reading the word for a paired-associate test
  and "vanilla" is being attended
THEN type "7"

This example shows how production rules can be acquired to embed knowledge in declarative memory. The next section will describe an empirical test of the predictions of this production-compilation mechanism. For other tests see Taatgen (2001a, 2001b).

So far we have discussed how production rules are created but not how they are selected. After a new production **New** is composed from old productions **Old1** and **Old2**, whenever **New** can apply **Old1** can also apply. The choice between **New**, **Old1**, and whatever other productions might apply will be determined by their utilities. However, the new production **New** has no prior experience and so its initial probabilities and costs will be determined from Bayesian priors. The values of P and C for a new production **New** should be set based on the values of P and C for the production **Old1** that it competes with. Again we will just focus on P, noting that a similar process works in the case of C.

The Bayesian priors in the case of P are α and β. Their sum is set to some constant value which reflects the confidence ACT-R has in new productions. This sum is a parameter, Total. Given this sum and the value P of **Old1**, the parameters for the new production are calculated as**:**

$$\alpha = P*Total$$
$$\beta = (1-P)*Total$$

Thus, the P value for the new production is initially the same as **Old1**. The parameter Total reflects the confidence we have that the parameters for **New** will be the same as the parameters for **Old1** and so controls the speed with which ACT-R can learn that **New** has a different value. Since **New** is more specific than **Old1** it will apply in a restricted set of situations where it may be more or less successful. The same parameter Total determines the speed with which the system will adjust the initial value of C for **New** which is taken from **Old1**. While anything can happen, typically the final value of P for **New** will be the same as **Old1** but its C value will be less because of the efficiency gathered by eliminating a production rule.

Putting it all Together: Modeling the Effect of Learning from Instructions

The tasks we have used so far to illustrate the ACT-R theory have all been focused on one aspect of the system (except for Salvucci's driving example which preceded presentation of the theory). The models of the menu task (Figure 5) and perfect time-sharing (Figure 7) focused on the perceptual-motor modules. The model of the Tower of Hanoi task (Figure 8) focused on the goal module. The model of the fan experiment (Figure 12) focused on the declarative retrieval module. Finally, the model of the building-sticks task (Figure 14) focused on the procedural system. While these were nice tasks to illustrate the behavior of the subsystems, given the goal of this paper it would be valuable to have models that put the various components together.

We will describe two such models. The first of these is a model that starts with learning from instruction and progresses to a highly automated performance of a task. We (Anderson & Bothell, 2002, Unit 9) have developed a formalism for representing instructions as a set of goals to be achieved in declarative memory and we have a set of production rules that will interpret any such instruction set. The production compilation mechanism will eventually convert these instructions into a set of productions for directly performing the procedures without declarative retrieval of the instructions. This model accounts for one of the mysteries of Experimental Psychology, which is how a set of experimental instructions causes a participant to behave according to the experimenter's wishes. According to this analysis, during the warm-up trials, which are typically thrown away in an experiment, the participant is converting from a declarative representation and a slow interpretation of the task to a smooth, rapid procedural execution of the task.

We have explored this compilation of procedures from instructions in a task that is considerably more complicated than what is normally studied. This is a simulation of the responsibilities of an anti-air warfare coordinator (AAWC) on an Aegis-style cruiser (Hodge, Rothrock, Walker, Fisk, Phipps, & Gay, 1995). There is no need to burden the reader with all the details, but it is worth noting that bright undergraduates spend a couple of hours learning the rules of engagement before they even begin performing the task. Once they start performing the task, they show typically power-law improvement in many of its components (Sohn, Douglass, Chen, & Anderson, submitted).

As an illustration of how instruction is deployed in performing this task consider one tiny fragment of the instruction which is the fact that when participants begin to classify an air track on the radar screen they must first select "track" from a menu of options. This piece of information is requested by the first instruction-following production below and when retrieved, it is utilized by the second instruction-following production that sets a subgoal to find the track option in the menu:

IF trying to retrieve an instruction to achieve a goal
    and an instruction for achieving that goal has been retrieved
THEN retrieve the first step of that instruction
    and note trying to recall the first step.

IF trying to retrieve a step of an instruction
    and a step has been retrieved involving a subgoal
THEN retrieve an instruction to achieve that subgoal
    and note trying to achieve that subgoal.

This pair of productions is representative of instruction interpretation in that it is quite abstract (no mention of AAWC task) and that it heavily involves goal manipulations, which we associate with dorsolateral prefrontal cortex. In the example, the first production retrieves the instruction "The first step in classifying a plane is the subgoal of selecting 'track'" and the second production response to this retrieved instruction. Production compilation will build a combination of these two productions with the information from the retrieved instruction built in:

IF trying to classify a plane
THEN set a subgoal to select "track"
    and note trying to retrieve an instruction for selecting "track".

As this production rule becomes combined with more production rules downstream ACT-R eventually learns the production rule:

IF trying to classify a plane
THEN hit the F1 key
    and set a subgoal to select "update"
    and note trying to retrieve an instruction for selecting "update".

since the F1 key is the key that selects the "track" option. This production rule also sets the subgoal to select the menu item "update." This is the next step in the sequence of actions that produces a classification. To achieve this goal would require another key press (in this case, F6). The production compilation process cannot produce rules that merge multiple external actions on the world together and so this limits how much compilation can collapse into a single rule. Anything larger would require merging the keystroke for this action with the keystroke for the next action and create potential jamming of the motor module. This illustrates how perceptual-motor constraints set the bounds of production compilation.

Modeling Haimson's Task

We have been performing a number of tests that focus on very specific aspects of the overall task to see how well the knowledge compilation mechanism predicts the specifics of the skill transition. We will describe some work that Haimson (Haimson & Anderson, 2002) has done as one example of this. One of the things that performers must do is to select air tracks on a radar screen for investigation and classification. The tracks to be selected have a certain physical profile and doctrine requires selecting the closest track to home ship that meets this profile. Haimson was interested in how people performed this task as a function of the distance of the targets from the home ship and the number of distractors. Figure 15 shows a typical screen that he presented to his participants. Their task was to select the item on the screen closest to home ship (center of screen) that had a curved component. Table 4 shows his data classified according to whether the target was close or far from home ship and number of distractors in the close and far regions. The data give evidence for an ordered search starting from home ship and moving outwards. For instance, participants are not much affected by number of far distractors when the target is close.

Participants were given a lot of practice at this task and we were interested in how well ACT-R could predict the effects of practice as well as the effect of the variables in Table 4. The instructions relevant to performing this task are quite simple relative to the instructions for the full task:

1. At the beginning find home ship, click it, and find target.
2. To find the target, attend to the unattended item closest[9] to home, and click it if it is curved.
3. If the item is not of the correct shape repeat 2.

From these instructions ACT-R will eventually learn productions like

IF looking for the target track
    and the currently attended object is not curved
THEN shift attention to the closest unattended location

This production will compete with its more primitive parents according to its experienced utility in solving the problem. Because it is more efficient in time than its parents it will eventually come to dominate. The critical parameters of the model are Total, which controls rate of learning, and t, which controls the amount of noise in utility estimations. Total was set to 50 and t was set so the standard deviation of the utilities would be the emerging default of 1 second.

Figure 16 shows the comparison between the observed behavioral functions and the predicted functions. Part (a) collapses over number of distractors and part (b) collapses over distance from home ship. The model does a good job of getting the ball park figures. Much of this has to do with the preset parameters in the model that determine rate of switching attention in visual search and time to move a mouse. While the absolute amount of speed-up on this task is captured (which is a parameter-free prediction of the model), human participants are displaying this learning more gradually over a longer period of time.

We want to stress that this example depends on the perceptual-motor system (determines asymptotic performance), the declarative and goal systems (for representing and interpreting instructions), and the procedural system (for creating and compiling new production rules). The only parameter estimated was Total, (reflecting the fact that production-rule learning is new and we do not have much experience on the setting of this parameter). Given how long it takes to run the model, we did not estimate it to provide optimal fits but rather just took a value that seemed to give somewhat reasonable results. The ability to deploy this full system and give nearly a priori productions illustrates the promise of the integrated approach represented by cognitive architectures. We still have a good distance to go before we can give similar detailed predictions for all of the GT-ASP task but this is our aspiration.

Putting it all together: Tracking Multiple Buffers in an fMRI study

We have recently completed an fMRI study that succeeded in tracking multiple components in Figure 4. Participants in this experiment were performing an artificial algebra task (based on Blessing & Anderson, 1996) in which they had to solve "equations". For a full exposition of the transformations see Anderson, Qin, Sohn, Stenger, and Carter (in press) but to give an illustration suppose the equation to be solved were

②P ③4 <–> ②5

where solving means isolating the P before the <–>. In this case the first step is to move the ③4 over to the right inverting the ③ operator to a ②̣□□̣□□□□ the equation now looked like:

② P <–> ②5②4

Then the ②̣in front of the P is eliminated by converting ②'s on the right side into ③'s. So that the "solved" equation looks like:

P <–> ③5③4

Participants were asked to perform these transformations in their heads and then key out the final answer—this involved keying "1" to indicate that they have solved the problem and then keying 3, 5, 3, and 4 on this example. The problems required 0, 1, or 2 (as in this example) transformations to solve. Figure 17 shows the effect of number of transformations on Day 1 and 5 on time to hit the first key. The figure shows a large effect of number of transformations but also a substantial speed up over days.

Figure 18 shows the activity of the ACT-R buffers solving an equation that involves a single transformation. The encoding begins with the identification of the <-> sign and then the encoding of the symbols to the right of the sign. Then begins the process of encoding the elements to the left of the sign and their elimination in order to isolate the P. In the example in Figure 18, six operations

(Steps 1-6) are required to encode the string and an additional two operations (Steps 9 & 10) to encode the transformation. Each of these requires activity in the imaginal module.[10] There are 5 such operations in the case of 0 transformations and 10 in the case of 2. With respect to retrievals in Figure 18, two pieces of information have to be retrieved for each transformation (Steps 7 & 8) that must be performed. One piece was the operation to perform ("flip" in Figure 18) and the other the identity of the terms to apply this operation to (argument position in Figure 18). There were 5 retrieval operations in the case of 2 transformations and none in the case of zero transformations. In all cases there are the final 5 motor operations (Steps 11-15 in Figure 18) but their timing will vary with how long the overall process takes. Finally, we tracked the number of productions required to solve these equations – there were 14 in the case of zero transformations, 20 in the case of one transformations, and 23 in the case of two transformations. Note that Figure 18 does not represent a significant involvement of the goal buffer. Unlike the ACT-R model for the Tower of Hanoi task (Figure 8), the ACT-R model for this task involves a single goal that holds the intention to transform the equation and keeps track of where one is in achieving that intention.

Participants in the experiment spent 5 days practicing this new symbol system. They were imaged on days 1 and 5. For a complete report on the data see Qin, Sohn, Anderson, Stenger, Fissel, Goode, & Carter (in preparation), but here we will just be concerned with the results relevant to differential involvement of four regions that reflect these four components of the ACT-R system imaginal buffer, retrieval buffer, motor buffer, and production rules. Figure 19 shows the three cortical regions associated with the three buffers. These associations were already discussed with respect to Figure 4. In addition, we have found, as have others (e.g., Poldrack, Prabakharan, Seger, Gabrieli, 1999), that the caudate (part of the basal ganglia) are particularly sensitive to the acquisition of new procedural skills. Since new productions need to be acquired to learn this novel task we expected to find activity here as well. The caudate is subcortical and therefore not represented in Figure 19.

Modeling the BOLD Response

Participants had 18 seconds for each trial. Figures 20-23 show how the BOLD signal varies over the 18-second period beginning 3 seconds before the onset of the stimulus and continuing for 15 seconds afterward which was long after the slowest response. Activity was measured every 1.5 seconds. The first two scans provide an estimate of baseline before the stimulus comes on. These figures also display the ACT-R predictions for the BOLD signal that depends on the assumption that each activity in a buffer or a production selection elicits a separate BOLD function and that these functions add (for details see, Anderson, Qin, Sohn, Stenger, & Carter, in press). The BOLD functions displayed are typical in that there is some inertia in the rise of the signal after the critical event and then decay. The BOLD response is delayed so that it reaches a maximum about 5 seconds after the brain activity. The functions in these figures take even longer to reach their maximum because they are the sum of a number of responses. For the details of the estimation of these functions read Anderson et al. (in press).

Figure 20 shows activity around the intraparietal sulcus, which has been associated with visual information processing in service of symbolic goals (Reichle, Carpenter, & Just 2000). It shows an effect of complexity and is not much affected by practice. However, it shows a considerable rise even in the simplest no-operation condition. This is because it is still necessary to encode the equation in this condition. The amount of information to be encoded or transformed also does not change with practice and so one would expect little change. The functions do rise a little sooner on day 5 reflecting the more rapid processing.

Figure 21 shows the activity around the inferior frontal sulcus which we take as reflecting the activity of the retrieval buffer. While it also shows a strong effect of number of transformations it is in striking contrast to Figure 20. First, it shows no rise in the 0 transformation condition because there are no retrievals in this condition. Second the magnitude of the response decreases after 5 days reflecting the fact that the declarative structures have been greatly strengthened and the retrievals are much quicker. The decrease in the figure is a parameter-free prediction reflecting the increase in base-level activation with practice (see Figure 11).

Figure 22 shows the activity around the central sulcus in the region that controls the right hand. It also shows an effect of number of transformations but the effect is much different than in Figures 20 and 21. Here the effect of complexity is to delay the BOLD function (because the finger presses are delayed) but there is no effect on the basic shape of the BOLD response because the same response sequence is being generated in all cases. The effect of practice is also just to move the BOLD response forward in this motor region.

Figure 23 shows activity in the caudate, which is thought to track use of new procedures (Poldrack, et al., 1999). The effect of complexity on Day 1 is less apparent than in the other figures because there is not a striking difference in the number of production rule firings. The more complex conditions have BOLD functions that are more stretched out in time reflecting the greater length of the trial but only slightly higher functions. The differential activity has largely disappeared by Day 5 when these productions are now highly practiced.

Each of these figures shows a qualitatively different response to complexity and practice in the task:

1. The motor area tracks onset of keying. Otherwise, the form of the BOLD function is not sensitive to cognitive complexity or practice.
2. The parietal area tracks transformations in the imagined equation. The form of the BOLD function is sensitive to cognitive complexity but not practice.
3. The prefrontal area tracks retrieval of algebraic facts. The form of the BOLD function is sensitive to cognitive complexity and decreases with practice.
4. The caudate tracks learning of new procedural skill. The BOLD function is not sensitive to cognitive complexity but disappears with practice.

The success in modeling these regions provides striking evidence that the activity in these regions tracks the behavior of components of the ACT-R system. The ability to separately track selection of procedural rules in the basal ganglia and retrieval declarative knowledge in the prefrontal cortex provides further evidence for ACT-R's procedural-declarative distinction. In general, imaging results like this suggests that the division of ACT-R into its components does correspond to the organization of the mind.

General Discussion

We have now gone through the various aspects of the ACT-R system and have shown how it makes contact with a wide variety of data about human cognition. Now we would like to place the theory somewhat in the general space of cognitive psychology. First, we will consider its relationship to its near neighbors, other production systems. Then, we will consider its stance on two of the high-profile current issues in psychology—the symbolic-subsymbolic distinction and modularity. Finally, we will recapitulate the picture it presents of how the pieces of the mind are put together.

Other Production Systems

Soar, EPIC, and CAPS are three other architectures that have close intellectual connections to ACT-R – all being derived from the ideas of Newell in the early 1970s. The Soar system was developed by Newell, Laird, and Rosenbloom (e.g., Laird, Newell, & Rosenbloom, 1991) to address issues both in psychology and artificial intelligence. Recently, its most impressive application has been to computer-generated forces. For instance, it successfully flew 50 missions in an Air Force training exercise (Jones, Laird, Nielsen, Kenny & Koss, 1999). EPIC has focused on the connection between the cognitive, perceptual and motor systems. We adopted many ideas from EPIC in the development of the perceptual-motor components of ACT-R. It has also been united with the Soar system (Chong & Laird, 1997). The earlier 3-CAPS architecture was substantially focused on individual differences, which it attributed to differences in activation capacity. The more recent 4-CAPS architecture has related similar concepts to fMRI data (Just, Carpenter, & Varma, 1999).[11]

Before discussing the differences we would like to note that all of these architectures have converged on a 50 msec cycle time for human cognition. It would be nice to be able to point to some specific empirical evidence for that number but really the evidence is the more general fact that this is a number that has worked in a wide variety of models.[12] Perhaps it is most significant that a smaller number has not been needed. In particular, this has been a rapid enough cycle time to model language processes in three of the architectures (Budiu, 2001; Just & Carpenter, 1992; Lewis, 1993). Language processing makes heavy computational demands. In a model like ACT-R, 50 msec is a reasonable minimal time for information to travel the multi-synapse route from the cortex to the basal ganglia and back. Newell also justified the 50 msec time (which is time for a decision in that architecture not time for a single production) in terms of the sum of a number of simpler neural processes.

While there is this convergence on 50 msec, the architectures paint rather different pictures of the overall nature of serial and parallel processing. ACT-R is the only architecture that specifies that just one production can fire at a time; the other systems allow parallel production firing. As such ACT-R has a serious serial bottleneck. While in Soar many productions can fire in parallel elaboration cycles, this has to conclude with a single decision, making Soar a serial bottleneck architecture as well. In contrast, EPIC and CAPS do allow for multiple productions to fire in a way that enables multiple independent paths of cognition. In CAPS there are activation limitations that imply multiple parallel paths will be pursued less rapidly than a single path. Thus, CAPS is a limited-capacity parallel system at this level. EPIC is the only unlimited-capacity parallel system in terms of enabling multiple threads of central cognition without process limitations (although it proposes serious limits on perceptual and motor processes as well as internal working memories—Kieras, Meyer, Mueller, & Seymour, 1999).

The issue of a central cognitive limitation has been debated between EPIC and ACT-R in the domains of perfect time sharing and the psychological refractory period (Byrne & Anderson, 2001). As described earlier, ACT-R can predict perfect time sharing between two tasks provided that they do not make simultaneous demands for production firing or simultaneous demands on any other module of the system. However, according to ACT-R such perfect time sharing should be the exception and not the rule and this is certainly our reading of the literature. As argued in Anderson and Lebiere (1998) the computational reason for a serial bottleneck is to have a point at which one makes sure the direction of computation is coordinated. In EPIC and CAPS one has to resort to having the modeler build in tests to prevent conflicting directions of behavior (e.g., one production firing for going left and another for going right). Perhaps one of the reasons why these two architectures have not dealt with production rule learning, while ACT-R and Soar have, is because of their lack of a coordination point. It is hard to learn production rules when one needs to worry about possible interactions among parallel rules.

While there is this strong serial coordination point in ACT-R, there are many modules that can be operating asynchronously in parallel just as in EPIC. Also there are other substantial parallel processes within modules such as retrieval from declarative memory. Also ACT-R, as other production systems, postulates a parallel process of matching and selection of production rules.

Another major dimension on which these architectures differ is their commitment to hybridization, with Soar being firmly committed to a purely symbolic account of cognition while ACT-R and CAPS postulate a mix of symbolic and subsymbolic processes. In particular, they both assume that continuously-varying activation processes in declarative memory control timing of behavior. In this regard, they are also close to Kintsch's construction-integration theory (Kintsch, 1998), which might well be regarded as another production system theory of cognition. It seems difficult to account for the graded aspects of cognition without a subsymbolic component. The next subsection of this paper will discuss the need for a symbolic component as well as a subsymbolic component. EPIC does have continuously varying quantities that control things like movement times but does not seem to have information-laden nonsymbolic quantities in its theory of central cognition.

Table 5 gives a 2x2 classification of the production systems according to whether they assume a serial bottleneck or not and whether they assume a hybrid nature or not. Obviously, we think ACT-R reflects the right cell. Of course, there are other dimensions of difference. For instance, only Soar does not make a procedural-declarative distinction and ACT-R and EPIC are joined in their emphasis on an integration of cognition with perception and action, whereas the other two architectures have not dealt with this issue in detail.

As both CAPS and ACT-R have been compared to brain imaging data it is worth considering their different conceptions of the relationship. In 4-CAPS the assumption is that level of the BOLD response reflects amount of effort being devoted to a task while in ACT-R it is assumed to reflect the time a module is functioning. It will not be easy to separate these two views with fMRI data but other sorts of brain imaging methods with better temporal resolution may allow a discrimination. However, these two alternative conceptions certainly do not have to be mutually exclusive.

Another difference between ACT-R and 4-CAPS is how they think about the function of different brain regions. In the ACT-R conception, different cortical and supporting neural structures serve as modules that broadcast their contents to the basal ganglia for pattern recognition and production selection. In contrast, 4-CAPS assumes that there are distinct production systems implemented in each region that collaborate in producing an answer. In effect it is proposed that there is a "society" of production systems that collaborate through direct connections. In part the organization postulated in ACT-R reflects the fact that it requires fewer paths to connect N areas if they all go through a central station (N paths in and N paths out) than if they all have to be pair-wise bidirectionally connected (N*(N-1)) paths. While it appears that the brain is sensitive to this argument of economy it certainly has not ignored direct pathways between cortical regions. These are not yet reflected in the ACT-R architecture. In contrast, 4-CAPS emphasizes these direct pathways but ignores the loop through the basal ganglia. The actual implementation of 4-CAPS would seem to require that there be complete connectivity so that each region is connected to each other region. This total connectivity does not appear to hold for the brain although it is unclear how serious a difficulty that is for 4-CAPS. On the other hand, ACT-R's connectivity assumptions are not totally satisfied either. While there do appear to be projections from all relevant cortical areas to the basal ganglia, there do not appear to be direct pathways from the basal ganglia to all relevant cortical regions. Outward projections are mainly to frontal regions, and we have to assume that these regions would then project to other cortical areas.

Hybridization

As we noted in the world of production systems a significant issue is whether one needs to postulate information-laden subsymbolic processes in addition to symbolic processes. However, in the rest of cognitive science the more hotly debated issue is whether a symbolic process is required in addition to a subsymbolic process. Eliminative connectionism (Plaut, McClelland, Seidenberg & Patterson, 1996) is the position that symbolic structures like production rules or declarative chunks have no role in the description of cognition but rather that cognition can be totally accounted for in terms of connections among neurons. To the extent that symbolic models work they are seen as (perhaps useful) approximations.

We are not claiming that cognition does not arise ultimately from neural connections and indeed have taken pains to show that ACT-R could be given a rather standard connectionistic implementation (Lebiere & Anderson, 1993). Rather our claim would be twofold:

1. That without recourse to symbolic representation it would be too difficult to give a thorough account of cognition.
2. The symbolic assumptions reflect certain constraints on how the neurons connect and interact.

With respect to the first claim, it seems that the evidence is pretty clear. There are no models of complex cognitive processes (e.g., natural language parsing systems that are actually used in real language processing, models for operating complex systems like aircraft, instructional models to guide training in domains that range from mathematics to anti-air warfare) that do not involve some symbolic components although many also use subsymbolic components. However, this may just be a practicality issue – while it is in principle possible, it is too difficult in practice to eliminate approximate symbolic representations.

The second claim, concerning constraints on connections, may be more fundamental and it is interesting to recognize how it is realized in ACT-R in the case of production rules. It is easy to think that the claim in ACT-R is that these production rules are coded in some data structures in the brain and that these structures are interpreted as one might imagine a computer would interpret such structures. However, these productions really specify pathways of influence from cortex to basal ganglia and back again. While the production rules involve "variables", (a point of controversy in cognitive science—e. g., Marcus, 2001) these variables really specify either (a) testing whether a pattern of activity from one region (say the visual buffer) matches a pattern of activity in another region (say the retrieval buffer) or (b) moving a pattern of activity from one buffer (say auditory) to another (say goal). Thus, the real commitment in production rules is to the pattern of interactions displayed in Figure 4. The claim is that information processing is constrained to follow paths like these which test patterns of activation from diverse areas and transmit information to different areas.

If one takes the symbolic claim to mean that there are strong a priori constraints on how brain areas can interact, then the evidence seems overwhelming in favor of the symbolic hypothesis (as indeed connectionists acknowledge—Elman, Bates, Johnson, Karmiloff-Smith, Parisi, & Plunkett, 1996; Hadley, 2002). While the proposal of a totally plastic, equipotential brain where anything can be connected to anything has the seduction of simplicity it has little basis in fact.

Many people might be surprised by the fact that the symbolic assumption in ACT-R cashes out into a claim about strong constraints on connections in the nervous system. We certainly were surprised when we realized this from our first attempts to implement ACT-R in a connectionist system (Lebiere & Anderson, 1993). This is because we were blinded by the notation of our symbolic models and how we implemented them in a computer. These surface details can obscure the essential claims of the theory – as indeed, it is easy to be misled by the topology of connectionist networks as to what their essential claims are.

Modularity

Modularity is the assumption that there are specific encapsulated systems of information processing which run according to their own principles and largely do not interact with other processes. Fodor identified the visual system as the referent modular system (see also Pylyshyn, 1999) and proposed that syntactic processing in language operates the same way. While we are using the term "module" rather close to the sense of Fodor as an informationally encapsulated system, we should acknowledge that Fodor has recently (2000) reiterated his long-standing doubts that higher-level cognition involves modules of the sort we proposed in Figure 4. His basic argument is that higher-level cognition requires that such diverse knowledge be brought together that it is impossible to have it encapsulated into separate components. Although he does not instantiate his argument with respect to ACT-R, his claim might mean one cannot keep separate one's goals, one's long-term declarative knowledge, and one's image of a problem in the way ACT-R's modules do. ACT-R is able to do this successfully in the tasks we model in this paper, having such knowledge only interact through the production system which has tiny windows (the buffers) onto the operation of the separate modules. This fact has to cast some doubt on Fodor's arguments but we would acknowledge that our tasks do not have the kind of knowledge integration that Fodor has in mind. On the other hand, Fodor does not provide a well-specified example of what an architecture like this cannot do. So, in absence of more specification on both sides, it seems one must conclude that the force of Fodor's doubts remain uncertain.

Another source of doubt about modules (Kosslyn, 2001; Uttal, 2001) comes from doubts about the success of the function localization program in brain imaging research. Questions are raised about both the statistics that identify regions of interest and about the consistency of the results from study to study. The possibility is raised that cognitive function may not be localized to specific brain regions and this doubt is generalized to a doubt about the existence of modules. However, neither the postulation of modules nor the use of brain imaging data requires that the modules be truly localized. First, modules are information-processing concepts and they could certainly be distributed across many brain regions. Second, the use of activity data in a region to study the functioning of a module does not require that the module be localized to that region; it only requires that the activity of that region reliably reflect the functioning of that module. Third, while there are many valid questions about the typical exploratory use of brain imaging data, the study we described with respect to Figures 19-23 tested a priori predictions about the behavior of regions identified a priori. These same regions have satisfied the predictions of the theory across a number of studies. As such they provide strong evidence for the sort of modules postulated in ACT-R.

It is frequently believed that there is a contradiction between modular conception of cognition and general architectures for cognition such as production systems. Our review of the production systems certainly shows that is not the case. Both EPIC and ACT-R are quite explicit about the notion of multiple independent modules, each of which is an encapsulated processor that operates according to its own principles and interacts with others through the production system. While one might have characterized 3-CAPS as non-modular, 4-CAPS is quite explicit about there being different modules – each being a separate production system in a different brain region. The only architecture that even has the appearance of being non-modular is Soar and even here Newell (p 455-456) was quite explicit that nothing about Soar was incompatible with the proposal of modules.

On the other hand, none of the production systems have specifically adopted a linguistic module and in three of the architectures (ACT-R, Soar, 3-CAPS) models have been produced that did syntactic processing in the general production rule cycle although 3-CAPS does assume specific linguistic capacity. At least with respect to ACT-R, there is not an in-principle commitment to the non-existence of a linguistic module. We would be open to neural evidence clearly indicating the existence of such a linguistic module. Even now we feel a tension to accommodate the evidence about localization of language processing. Barring definitive neural data we have taken the stance that we will be guided by whether we can really model language processing in the 50 msec general serial bottleneck of ACT-R. So far we have been successful as were these other architectures. However, a lot of complexities of language processing have not yet been modeled. One thing that we have observed as ACT-R begins to address more complex processes is that it is forced by time constraints to process language in a rather sloppy and shallow way – not processing each word independently and fully (Budiu & Anderson, submitted). However, people appear to fail to do so as well in on-line processing (Myers & O'Brien, 1998; Noordman & Vonk, 1998; Cook, Holleran, & O'Brien, 1998; Albrecht & Myers, 1998; Sanford & Garrod, 1998). Thus, as a very interim report, there has been some success treating language in the general cognitive loop and it has led to accurate models of language processing.

Unity of Cognition

We would like to close by commenting on two different senses of the unity of cognition – one is whether human cognition reflects a single integrated process and the other is whether ACT-R offers a unified characterization of cognition. With respect to the first question, the ACT-R architecture proposes that cognition is fragmented into many separate modules operating in parallel whose processing is not generally available to cognition. The processing in these modules is only partly coordinated. Specific final products of the processing can appear in the buffers and these can be used to create enough coordination for cognition to be generally adaptive. So, ACT-R certainly instantiates the view that the apparent unity of mental life is largely an illusion and that many critical processes happen in uncoordinated ways out of awareness.

According to the ACT-R analysis of consciousness, three things must happen for one to be conscious of some information. First, it must appear in a buffer so that it can be detected by a production rule and second it must be detected (i.e., be part of the condition of the production rule). However, these two conditions are not sufficient because ACT-R is not conscious of many things that productions match. For instance, Lee's (2000) model of air traffic control learned to attend selectively to very specific regions of the screen to achieve expertise and the evidence was that participants learned to do the same. However, participants were quite unaware of where they were looking. The third condition for consciousnesses is that the action of the production place a representation of the information in the goal so that it is available for subsequent processing. Thus, the trail of information remnants in the goal buffer is basically the trail of what ACT-R is conscious of. It is not aware that anything else has happened that does not leave such a trail. To the degree that goal structures (supported by DLPFC) are more developed in humans then consciousness by this definition is a feature especially supported in humans.

While ACT-R implies that our experience of the unity of one's own cognition is illusory, it offers us as scientists a characterization of cognition that does aspire to be unified – even as the human body has many separate organs but biology aspires to characterize how they all work together. Clearly, we have not identified all the modules or all of their processes but there is a program here for identifying new modules and understanding how all the modules are integrated. The convergence of all the knowledge streams in the production pattern matcher is the point of that integration.

Summary

While there are many details, some of which were reviewed in the main part of this paper, we thought it was worthwhile to distill our conception of the mind down into 6 major points:

1. There are multiple independent modules whose information processing is encapsulated.

2. The modules can place chunks reflecting their processing in their buffers and the production system can detect when critical patterns are satisfied among these chunks.

3. From those productions whose conditions are satisfied a single production will be selected at any time and fire, leading to updates to various buffers that in turn can trigger information processing in their respective modules.

4. While chunks and productions are the symbolic components of the system reflecting its overall information flow, chunks have subsymbolic activations and productions have subsymbolic utilities that control which chunks and productions get used.

5. Learning can involve either acquiring new chunks and productions or tuning their subsymbolic parameters.

6. These processes are stochastic and take place in real time.

References

Albrecht, J. & Myers, J. (1998). Accessing distant text information during reading: Effects of contextual cues. Discourse Processes, 26, 87-107.

Altmann, E. M. & Trafton, J. G. (2002). Memory for goals: An activation-based model. Cognitive Science, 26, 39-83.

Amos, A. (2000). A computational model of information processing in the frontal cortex and basal ganglia. Journal of Cognitive Neuroscience, 12, 505-519.

Anderson, J. R. (1974). Retrieval of propositional information from long-term memory. Cognitive Psychology, 5, 451-474.

Anderson, J. R. (1976). Language, memory, and thought. Hillsdale, NJ: Erlbaum.

Anderson, J. R. (1983). The architecture of cognition. Cambridge, MA: Harvard University Press.

Anderson, J. R. (1990). The adaptive character of thought. Hillsdale, NJ: Erlbaum.

Anderson, J. R. (1993). Rules of the Mind. Hillsdale, NJ: Erlbaum.

Anderson, J. R. (2002). Spanning seven orders of magnitude: A challenge for cognitive modeling. Cognitive Science, 26.

Anderson, J. R., & Betz, J. (2001). A hybrid model of categorization. Psychonomic Bulletin and Review, 8, 629-647.

Anderson, J. R. & Bothell, D. (2002). The ACT-R 5.0 Tutorial. Carnegie Mellon University. http://act-r.psy.cmu.edu/tutorials.

Anderson, J. R., Bothell, D., Lebiere, C. & Matessa, M. (1998). An integrated theory of list memory. Journal of Memory and Language, 38, 341-380.

Anderson, J. R. & Bower, G.H. (1973). Human associative memory. Washington: Winston and Sons.

Anderson, J. R., Budiu, R., & Reder, L. M. (2001). A theory of sentence memory as part of a general theory of memory. Journal of Memory and Language, 45, 337-367.

Anderson, J. R. & Douglass, S. (2001). Tower of Hanoi: Evidence for the Cost of Goal Retrieval. Journal of Experimental Psychology: Learning, Memory, & Cognition, 27, 1331-1346.

Anderson, J. R., Fincham, J. M. & Douglass, S. (1999). Practice and retention: A unifying analysis. Journal of Experimental Psychology: Learning, Memory, and Cognition, 25, 1120-1136

Anderson, J. R. & Lebiere, C. (1998). The atomic components of thought. Mahwah, NJ: Erlbaum.

Anderson, J. R., Matessa, M., & Lebiere, C. (1997). ACT-R: A theory of higher level cognition and its relation to visual attention. Human Computer Interaction, 12, 439-462.

Anderson, J. R., Qin, Y., Sohn, M-H., Stenger, V. A. & Carter, C. S. (in press.) An information-processing model of the BOLD response in symbol manipulation tasks. Psychonomic Bulletin & Review.

Anderson, J. R. & Reder, L. M. (1999). The fan effect: New results and new theories. Journal of Experimental Psychology: General, 128, 186-197.

Ashby, F. G. & Waldron, E. M. (2000). The neuropsychological bases of category learning. Current Directions in Psychological Science, 9, 10-14.

Baddeley, A. D. (1986). Working memory. Oxford: Oxford University Press.

Blessing, S. B. (1996). The use of prior knowledge in learning from examples. Doctoral Dissertation. Carnegie Mellon University, Department of Psychology, Pittsburgh, PA.

Buckner, R.L., Kelley, W.M., and Petersen, S.E. (1999) Frontal cortex contributes to human memory formation. Nature Neuroscience 2, 311-314.

Budiu, R. & Anderson, J. R. (submitted). Semantic sentence processing: A unified theory.

Budiu, R. (2001) The role of background knowledge in sentence processing. Doctoral Dissertation, Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA.

Byrne, M. D. (2000). Are retrievals from long-term memory interruptible? In L. R. Gleitman & A. K. Joshi (Eds.), Proceedings of the Twenty-Second Annual Conference of the Cognitive Science Society, 71-76. Mahwah, NJ: Lawrence Erlbaum.

Byrne, M. D. (2001). ACT-R/PM and menu selection: Applying a cognitive architecture to HCI. International Journal of Human-Computer Studies, 55, 41-84.

Byrne, M. D. (2003). Cognitive architecture. To appear in J. Jacko & A. Sears (Eds.), Human-Computer Interaction Handbook. Mahwah, NJ: Erlbaum.

Byrne, M. D., & Anderson, J. R. (1998). Perception and action. In J. R. Anderson & C. Lebiere (Eds.) The atomic components of thought, 167-200. Mahwah, NJ: Erlbaum.

Byrne, M. D., & Anderson, J. R. (2001). Serial modules in parallel: The psychological refractory period and perfect time-sharing. Psychological Review, 108,847-869.

Byrne, M. D., Anderson, J. R., Douglass, S., & Matessa, M. (1999). Eye tracking the visual search of click-down menus. *Human Factors in Computing Systems: Proceedings of CHI 99,* 402-409. Reading, MA: Addison Wesley.

Card, S., Moran, T., & Newell, A. (1983). The psychology of human-computer interaction. Hillsdale, NJ: Erlbaum.

Chong, R. S. & Laird, J. E. (1997). Identifying dual-task executive process knowledge using EPIC-Soar. In Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society, pp. 107-112. Mahwah, NJ: Erlbaum.

Cook, A. E. , Halleran, J. G., & O'Brien, E. J. (1998). What is readily available during reading? A memory-based view of text processing. Discourse Processes, 26, 109-130.

Cosmides, L. & Tooby, J. (2000). The cognitive neuroscience of social reasoning. In M. S. Gazzaniga (Ed.) The new cognitive neurosciences, 2nd Edition, 1259-1272. Cambridge, MA: The MIT Press.

Dehaene, S., Spelke, E., Pinel. P., Stanescu, R. Tsivkin, S. (1999). Sources of mathematical thinking: Behavior and brain-imaging evidence. Science, 284, 970-974.

Egly, R., Driver, J., & Rafal, R. D. (1994). Shifting visual attention between objects and locations: Evidence from normal and parietal lesion subjects. Journal of Experimental Psychology: General, 123, 161-177.

Elman, J. L., Bates, E., Johnson, M. H., Karmiloff-Smith, A., Parisi, D., & Plunkett, K. (1996). Rethinking innateness: A connectionist perspective on development. Cambridge, MA: MIT Press.

Fincham, J. M., Carter, C. S., van Veen, V., Stenger, V. A., and Anderson, J. R. (2002). Neural mechanisms of planning: A computational analysis using event-related fMRI. In *Proceedings of the National Academy of Sciences*. 99 (5), 3346-3351.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology, 47, 381-391.

Fodor, J. A., (1983). The modularity of the mind. Cambridge, MA: MIT/Bradford Books.

Fodor, J. A. (2000). The mind doesn't work that way: The scope and limits of computational psychology. Cambridge, MA: MIT Press.

Frank, M. J., Loughry, B. & O'Reilly, R. C. (2000). Interactions between frontal cortex and basal ganglia in working memory: A computational model. Institute of Cognitive Science, University Colorado, Boulder, Technical Report 00-01.

Freed, M. (2000). Simulating human agents. Papers from the 2000 AAAI Fall Symposium, Michael Freed, Chair. Technical Report FS-00-03. Menlo Park, CA: AAAI Press.

Graybiel, A. M., & Kimura, M. (1995). Adaptive neural networks in the basal ganglia.. In J. C. Houk, J. L. Davis, and D. G. Beiser (Eds.) Models of information processing in the basal ganglia. Cambridge, MA: MIT Press.

Greeno, J. G. (1989). Situations, mental models and generative knowledge. In D. Klahr & K. Kotovsky (Eds.), Complex information processing: The impact of Herbert A. Simon. Hillsdale, NJ: Erlbaum.

Gunzelmann, G., & Anderson, J. R. (2002). Strategic differences in the coordination of different views of space. In W. D. Gray and C. D. Schunn (Eds.), Proceedings of the Twenty-Fourth Annual Conference of the Cognitive Science Society (pp. 387-392). Mahwah, NJ: Lawrence Erlbaum Associates.

Hadley, R. F. (2002). A defense of functional modularity.

Haimson, C., & Anderson, J.R. (October, 2002). Partitioning visual displays: Directing the path of visual search. Paper presented at the Annual Meeting of the Human Factors and Ergonomics Society, Baltimore, MD. **Note: Proceedings not published yet!**

Haimson, C., & Anderson, J. R. (in press). Partitioning visual displays: Directing the path of visual search. *In Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting*, Baltimore, MD.

Hikosaka, O., Nakahara, H., Rand, M. K., Sakai, K., Lu, Z., Nakamura, K., Miyachi, S. & Doya, K. (1999). Parallel neural networks for learning sequential procedures. Trends in Neuroscience, 22, 10 (256), 464-471.

Hodge, K. A., Rothrock, L., Kirlik, A. C., Walker, N., Fisk, A. D., Phipps, D. A., & Gay, P. E. (1995). Trainings for tactical decision making under stress: Towards automatization of component skills. (HAPL-9501). Atlanta, GA: Georgia Institute of Technology, School of Psychology, Human Attention and Performance Laboratory.

Houk, J. C. & Wise, S. P. (1995). Distributed modular architectures linking basal ganglia, cerebellum, and cerebral cortex: Their role in planning and controlling action. Cerebral Cortex, 2, 95-110.

Humphreys, G. W., Olson, A., Romani, C., & Riddoch, M. J. (1996). Competitive mechanisms of selection by space and object: A neuropsychological approach. In A. F. Kramer, M. G. H. Coles, & G. Logan (Eds.), Converging operations in the study of visual selective attention (pp. 365-393). Washington, DC: American Psychological Association.

Jones, R. M., Laird, J. E., Nielsen, P. E., Coulter, K. J., Kenny, P., & Koss, F. V. (1999). Automated intelligent pilots for combat flight simulation. AI Magazine, 20(1), 27-41.

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

Just, M. A., Carpenter, P. A., & Varma, S. (1999). Computational modeling of high-level cognition and brain function. Human Brain Mapping, 8, 128-136

Kieras, D. E., Meyer, D. E., Mueller, S., & Seymour, T. (1999). Insights into working memory from the perspective of the EPIC architecture for modeling skilled perceptual-motor performance. In P. Shah & A. Miyake (Eds.) Models of Working Memory: Mechanisms of Active Maintenance and Executive Control. Cambridge: Cambridge University Press.

Kintsch, W. (1998). Comprehension: A paradigm for cognition. New York: Cambridge University Press.

Kosslyn, S. M. (2001). "The Strategic Eye": another look. Minds and Machines, 11, 287-291.

Laird, J. E.; Newell, A, & Rosenbloom, P. S. (1991). Soar: An Architecture for General Intelligence. Artificial Intelligence, 47:289-325.

Lebiere, C. & Anderson, J. R. (1993). A Connectionist Implementation of the ACT-R Production System. In Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society, pp. 635-640.

Lebiere, C. & Wallach, D. (2001). Sequence learning in the ACT-R cognitive architecture: empirical analysis of a hybrid model. In R. Sun & C. L. Gilles (Eds.). Sequence learning: paradigms, algorithms, and applications (pp. 188-212). Berlin: Spinger Lecture Notes in Computer Science.

Lee, F. J. (2000). Does learning of a complex task Have to be complex? A study in learning decomposition. Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA.

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

Logan, G. D. (1988). Toward an instance theory of automatization. Psychological Review, 95, 492-527.

Lovett, M. C. (1998). Choice. In J. R. Anderson, & C. Lebiere (Eds.). The atomic components of thought, 255-296. Mahwah, NJ: Erlbaum.

Lovett, M. C., Daily, L. Z., & Reder, L. M. (2000). A source activation theory of working memory: Cross-task prediction of performance in ACT-R. Cognitive Systems Research, 1, 99-118.

Marcus, G. F. (2001). The algebraic mind: Integrating connectionism and cognitive science. Cambridge, MA: MIT Press.

McClelland, J. L., & Rumelhart, D. E. (1981). An interactive model of context effects in letter perception: I. An account of basic findings. Psychological Review, 88, 375-407.

Meyer, D. E., Glass, J. M., Mueller, S. T., Seymour, T. L., & Kieras, D. E. (2001) Executive-process interactive control: A unified computational theory for answering 20 questions (and more) about cognitive ageing. European Journal of Cognitive Psychology, 13 (1/2), 123-164.

Meyer, D. E. & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance. Part 1. Basic mechanisms. Psychological Review, 104, 2-65.

Middleton, F. A., & Strick, P. L. (2000). Basal ganglia and cerebellar loops: Motor and cognitive circuits. Brain Research Reviews, 31, 236-250.

Myers, J. & O'Brien, E. (1998). Accessing the discourse representation during reading. Discourse Processes, 26, 131-157.

Newell, A. (1973a). You can't play 20 questions with nature and win: Projective comments on the papers of this symposium (p. 283-310). In W. G. Chase (Ed.) Visual information processing. New York: Academic Press Inc.

Newell, A. (1973b). Production systems: Models of control structures (p. 463-526). In W. G. Chase (Ed.) Visual information processing. New York: Academic Press Inc.

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

Newman, S. D., Just, M. A., Keller, T. A., Carpenter, P. A. (submitted). Differential effects of syntactic and semantic processing on the subregions of Broca's area. Cognitive Brain Research.

Newman, S. D., Carpenter, P. A., Varma, S., & Just, M. A. (in press). Frontal and parietal participation in problem-solving in the Tower of London: fMRI and computational modeling of planning and high-level perception.

Nilsen, E. L. (1991). Perceptual-motor control in human-computer interaction. (Tech. Rep. No. 37). Cognitive Science and Machine Intelligence Laboratory, University of Michigan, Ann Arbor, MI.

Noordman, L., & Vonk, W. (1998). Memory based processing in understanding causal information. Discourse Processes, 26, 191-212.

Nyberg, L., Cabeza, R. & Tulving, E. (1996). PET studies of encoding and retrieval: The HERA model. Psychonomic Bulletin and Review, 3, 135-148.

Pashler, H. E. (1998). The psychology of attention. Cambridge, MA: MIT Press.

Pirolli, P. L., & Anderson, J. R. (1985). The role of practice in fact retrieval. Journal of Experimental Psychology: Learning, Memory, & Cognition, 11, 136-153.

Plaut, D. C., McClelland, J. L., Seidenberg, M. S., &Patterson, K. E. (1996). Understanding normal and impaired word reading: Computational principles in quasi-regular domains. Psychological Review, 103, 56-115.

Poldrack, R. A., Prabakharan, V., Seger, C. &Gabrieli, J. D. E. (1999). Striatal activation during cognitive skill learning. Neuropsychology 13, 564-574.

Pylyshyn, Z. (1999). Is vision continuous with cognition? The case of impenetrability of visual perception. Behavioral and Brain Sciences, 22, 341–423.

Qin, Y., Sohn, M.-H., Anderson, J. R., Stenger, Fissel, Goode, A. & Carter, C. (in preparation). Practice Study in Event Related fMRI and Base-level Activation Learning in ACT-R.

Reder, L. M. & Gordon, J. S. (1997) Subliminal Perception: Nothing Special, Cognitively Speaking. In J. Cohen and J. Schooler (Eds.) Cognitive and Neuropsychological approaches to the study of Consciousness, pp. 125-134. Mahwah, NJ: Erlbaum.

Reichle, E. D., Carpenter, P. A., & Just, M. A. (2000). The neural basis of strategy and skill in sentence-picture verification. Cognitive Psychology, 40, 261-295.

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

Reichle, E. D., Rayner, K., & Pollatsek, A. (1999). Eye movement control in reading: Accounting for initial fixation locations and refixations within the E-Z Reader model. Vision Research, 39, 4403–4411.

Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. Psychological Review, 107, 358-367.

Saint-Cyr, J. A., Taylor, A. E., & Lang, A. E. (1988). Procedural learning and neostriatal dysfunction in man. Brain, 111, 941-959.

Salvucci, D. D. (2001a). An integrated model of eye movements and visual encoding. Cognitive Systems Research, 1, 201-220.

Salvucci, D. D. (2001). Predicting the effects of in-car interfaces on driver behavior using a cognitive architecture. To appear in CHI Letters, CHI 2001 Conference Proceedings.

Salvucci, D. D., Boer, E. R., & Liu, A. (2001). Toward an integrated model of driver behavior in a cognitive architecture. Transportation Research Record, No. 1779.

Sanford, A. J. & Garrod, S. C. (1998). The role of scenario mapping in text comprehension. Discourse Processes, 26, 159-190.

Anderson, J. R. & Schooler, L. J. (1991). Reflections of the environment in memory. *Psychological Science, 2,* 396-408.

Schumacher, E. H, Seymour, T. L., Glass, J. M., Lauber, E. J., Kieras, D. E., & Meyer, D. E. (1997). Virtually perfect time sharing in dual-task performance. Paper presented at the 38th Annual Meeting of the Psychonomic Society, Philadelphia, PA.

Simon, H. A. (1975). The functional equivalence of problem solving skills. Cognitive Psychology, 7, 268-288.

Sohn, M.-H., Douglass, S. A., Chen, M.-C., &Anderson, J. R. (submitted). Unit task execution with a dynamic display in a complex problem-solving situation

Sohn, M-H., Douglass, S. A., Chen, M-C., & Anderson, J. R. (2000). Eye-movements during unit-task execution in a complex problem-solving situation. In Proceedings of the 44th Annual Meeting of the Human Factors and Ergonomics Society, 378-381.

Squire, L. R. (1987). Memory and brain. New York: Oxford University Press.

Taatgen, N. A. (2001a). A model of individual differences in learning Air Traffic Control. In Proceedings of the Fourth International Conference on Cognitive Modeling, pp. 211-216. Mahwah, NJ: Lawrence Erlbaum Associates.

Taatgen, N.A. (2001b). Extending the Past-tense debate: a model of the german plural. In J.D. Moore and K. Stenning. Proceedings of the twenty-third annual conference of the cognitive science society (pp. 1018-1023). Mahwah, NJ: Erlbaum.

Taatgen, N. A. & Anderson, J. R. (in press). Why do children learn to say "broke"? A model of learning the past tense without feedback, Cognition.

Triesman, A. M., & Gelade, G. (1980). A feature-integration theory of attention. Cognitive Psychology, 12, 97-136.

Ungerleider, L. G. & Miskin, M. (1982). Two cortical visual systems. In D. J. Engle, M. A. Goodale, & R. J. Mansfield (Eds.) Analysis of visual behavior, 549-586. Cambridge, MA: MIT Press.

Uttal, W. R. (2001). The new Phrenology: The limits of localizing cognitive processes in the brain. Cambridge, MA: MIT Press.

Vecera, S. P., & Farah, M. J. (1994). Does visual attention select objects or location? Journal of Experimental Psychology: General, 123, 146-160.

Wagner, A. D., Maril, A., Bjork, R. A., & Schacter, D. L. (2001). Prefrontal contributions to executive control: fMRI evidence for functional distinctions within lateral prefrontal cortex. NeuroImage, 14, 1337-1347.

Wagner, A. D., Paré-Blagoev, E. J., Clark, J., & Poldrack, R. A. (2001). Recovering meaning: Left prefrontal cortex guides controlled semantic retrieval. Neuron, 31, 329-338.

Warrington, E. K., & Shallice, T. (1984). Category specific semantic impairments. Brain, 197, 829-854.

Wise, S. P., Murray, E. A., & Gerfen, C. R. (1996). The frontal cortex-basal ganglia system in primates. Critical Reviews in Neurobiology, 10, 317-356.

Wolfe, J. M. (1994). Guided search 2.0: A revised model of visual search. Psychonomic Bulletin & Review, 1, 202-238.

Acknowledgements

John R. Anderson, Department of Psychology, Carnegie Mellon University, Pittsburgh, PA 15213 (412) 268-2788, ja@cmu.edu. This research was supported by ONR grant N00014-96-01491. Correspondence concerning this article should be addressed to John R. Anderson, Department of Psychology, Carnegie Mellon University, Pittsburgh, PA 15213. Electronic mail may be sent to ja@cmu.edu.

Footnotes

1. We will discuss at the end of the paper some of the recent controversy about the proliferation of modules (e.g., Fodor, 2000; Kosslyn, 2001; Uttal, 2001).

2. These systems take advantage of the fact that it is possible to have the models interact with simulators and so it is not necessary to create real bodies and sensors in the simulation.

3. There is one significant qualification on this claim – because of the lack of complex motor programs in ACT-R (which are essential for modeling driving) he had to reduce the default cycle time to in effect program these routines in ACT-R. However, this was done in advance of the test that we will be describing here. Also, Salvucci (personal communication) has subsequently shown that the model, given a more realistic motor programs, can drive with a 50 msec cycle time.

4. Even though this is the goal, we have not fully achieved it any of the research described in this paper.

5. There is, of course, a great deal of evidence that long-term memory, which is part of the retrieval module as distinct from the buffer, is associated with the temporal lobes and hippocampus.

6. Specifically, in terms of Anderson & Lebiere, the decompositions described by Probability of Goal Equation 3.2 and Cost of goal Equation 3.3, production strength and the learning mechanism described by Production Strength Equation 4.4, and the associative learning described by Posterior Strength Equation 4.3.

7. The actual value of this parameter in various instantiations of ACT-R has been the source of some confusion. In the first Visual Interface for ACT-R (Anderson, Matessa, & Lebiere, 1997), all activity was serialized and so this value was 185 ms. However, in ACT-R 5.0 the actual system parameter is 85 ms because the same attention shift now requires two production firings that add the other 100 ms.

8. The numbers reported here are the same as those in Anderson, Lebiere, & Matessa, 1997. A subsequent recomputation of feature overlap leads to a .62 overlap and an estimate of a 115 msec. slope which is again close to the empirical number. (Anthony Hornoff, personal communication)

9. This step depends on ACT-R's ability to tag items on the screen as attended and retrieve the closest unattended object to a location.

10. Note this is different than the visual module described earlier. The imaginal buffer has recently been added to a number of ACT-R models (e.g., Gunzelmann & Anderson, 2002) to model transformations of visually presented information.

11. We will sometimes use the term CAPS to refer to this architecture in general.

12. Meyer, Glass, Mueller, Seymour, and Kieras (2002) relate this to the alpha rhythm and propose that changes in the cycle time account for effects of cognitive aging.

Table 1
Domains for ACT-R Models

I. Perception & Attention

1. Psychophysical Judgements
2. Visual Search
3. Eye Movements
4. Psychological Refractory Period
5. Task Switching
6. Subitizing
7. Stroop
8. Driving Behavior
9. Situational Awareness

II. Learning & Memory

1. List Memory
2. Fan Effect
3. Implicit Learning
4. Skill Acquisition
5. Cognitive Arithmetic
6. Category Learning
7. Learning by Exploration and Demonstration
8. Updating Memory & Prospective Memory

III. Problem Solving & Decision Making

1. Tower of Hanoi
2. Choice & Strategy Selection
3. Mathematical Problem Solving
4. Spatial Reasoning
5. Dynamic Systems
6. Use and Design of Artifacts
7. Game Playing
8. Insight and Scientific Discovery

IV. Language Processing

1. Parsing
2. Analogy & Metaphor
3. Learning
4. Sentence Memory
5. Communication & Negotiation

V. Other

1. Cognitive Development
2. Individual Differences
3. Emotion
4. Cognitive Workload

Visit http://act-r.psy.cmu.edu/publications

Table 2
Distribution of Experience in Lovett (1998)
(Values in parentheses were used in the extreme condition)

	Undershoot More Successful	Overshoot More Successful
Looks Undershoot	10 Undershoot 0 Overshoot	10 (5) Undershoot 10 (15) Overshoot
Looks Overshoot	10 (15) Undershoot 10 (5) Overshoot	0 Undershoot 10 Overshoot

Table 3

ACT-R model probabilities before and after problem-solving experience in Experiment 3 (Lovett & Anderson, 1996)

Production	Prior Probability of Success	Final Value: Biased Condition	Final Value: Extreme-Biased Condition
Production 1: More Successful Context Free	.50	.60	.71
Production 2: Less Successful Context Free	.50	.38	.27
Production 3: More Successful Context Sensitive	.96	.98	.98
Production 4: Less Successful Context Sensitive	.96	.63	.54

Table 4
Average Time to Find a Target as a Function of Position of Target to Home Ship (Center of Screen) and Number of Tracks on the Screen

	1 Close 1 Far	4 Close 4 Far	4 Close 12 Far
Target Close To Home Ship	821 msec.	1033 msec.	1060 msec.
Target Far From Home Ship	995 msec.	2028 msec.	2671 msec.

Table 5
Production System Architectures

	Serial Bottleneck	No Serial Bottleneck
Hybrid	ACT-R	3/4-CAPS
Not Hybrid	SOAR	EPIC

Figure Captions

Figure 1 Representation of the driving task studied by Salvucci (Salvucci, 2001).

Figure 2 Impact of driving on dialing various phones (Salvucci, 2001).

Figure 3 Impact of dialing various phones on deviation of car from center of lane (Salvucci, 2001).

Figure 4 The organization of information in ACT-R 5.0.

Figure 5 Fit of ACT-R/PM to the Nilsen (1991) menu scanning data.

Figure 6 The ACT-R schedule chart for Schumacher et al (1997).

Figure 7 Predictions of the ACT-R model for Schumacher et al (1997).

Figure 8 The relationship between the number of goals being held during the performance of the Tower of Hanoi task and activation in the dorsolateral prefrontal cortex.

Figure 9 A presentation of a declarative chunk with its subsymbolic quantities.

Figure 10 Representation of some of the chunks in Pirolli & Anderson (1985).

Figure 11 Activation of the chunks in Anderson and Pirolli (1985) as a function of fan and practice.

Figure 12 Time to recognize sentences in Anderson and Pirolli (1985) as a function of fan and practice.

Figure 13 A representation of Lovett's building sticks task.

Figure 14 Percent choice of the more successful strategy as a function of appearance of the problem and amount of experience. Lovett (1998).

Figure 15 A typical screen in Haimson's task. Targets are tracks with half-curves.

Figure 16 Learning in the screen localization task (a) Data segregated by distance from home ship, collapsed over number of distractors (b) Data segregated by number of distractors, collapsed over distance from home ship.

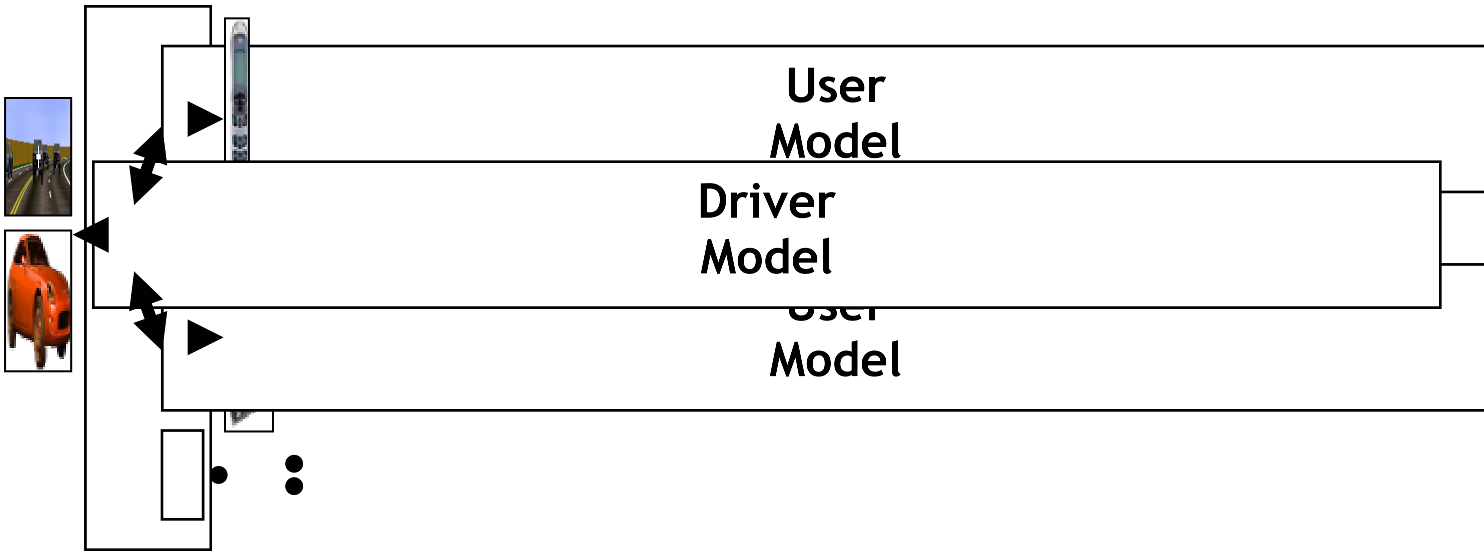
Figure 17 Performance in the symbol manipulation task: Effects of number of transformations and days of practice.

Figure 18 Activity of ACT-R buffers in solving an equation.

Figure 19 Location of the posterior parietal region associated with the imaginal buffer, the motor region associated with the manual buffer, and the ventrolateral prefrontal cortex associated with the retrieval buffer.

Figure 1

Cell phone application that integrates cognition, vision, manual, auditory, and speech in a system that actually drives (a simulator) and talks.



1/13/03

Figure 2

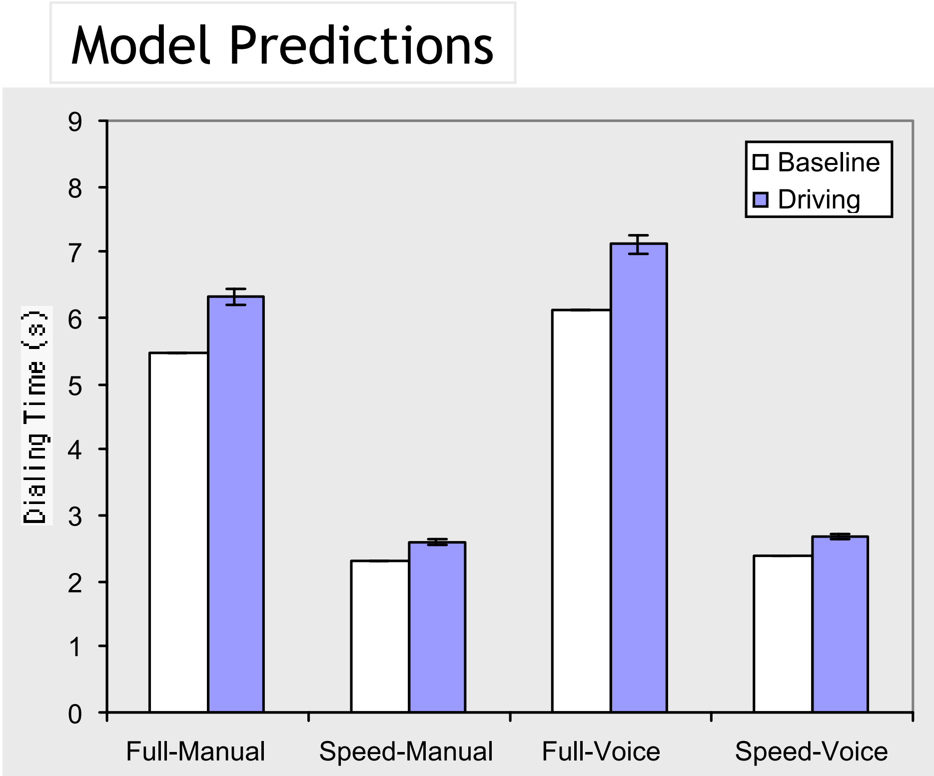


Figure 3

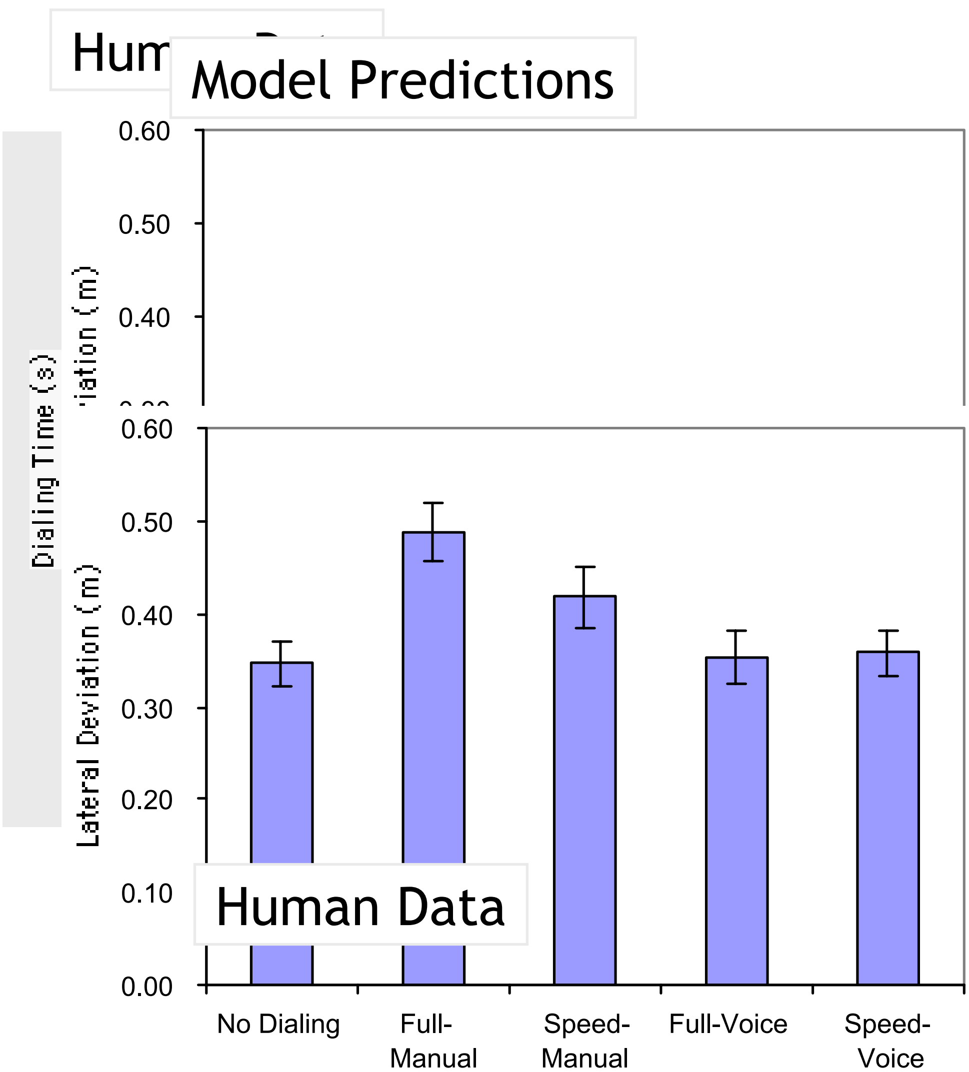


Figure 4

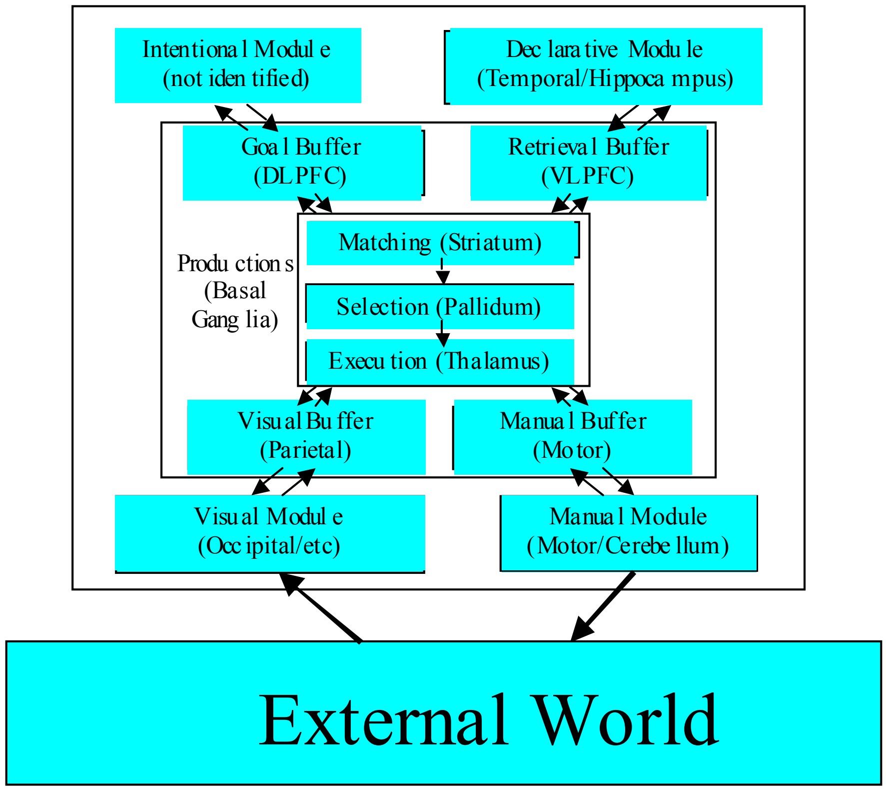

Intentional Module
(not identified)
Declarative Module
(Temporal/Hippocampus)
Goal Buffer
(DLPFC)
Retrieval Buffer
(VLPFC)
Productions
(Basal
Ganglia)
Matching (Striatum)
Selection (Pallidum)
Execution (Thalamus)
Visual Buffer
(Parietal)
Manual Buffer
(Motor)
Visual Module
(Occipital/etc)
Manual Module
(Motor/Cerebellum)
External World

Figure 5

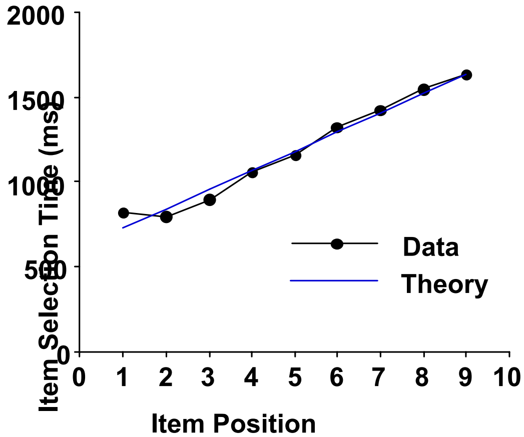

2000
1500
1000
500
0
Item Selection Time (ms)
0 1 2 3 4 5 6 7 8 9 10
Item Position
Data
Theory

Figure 6

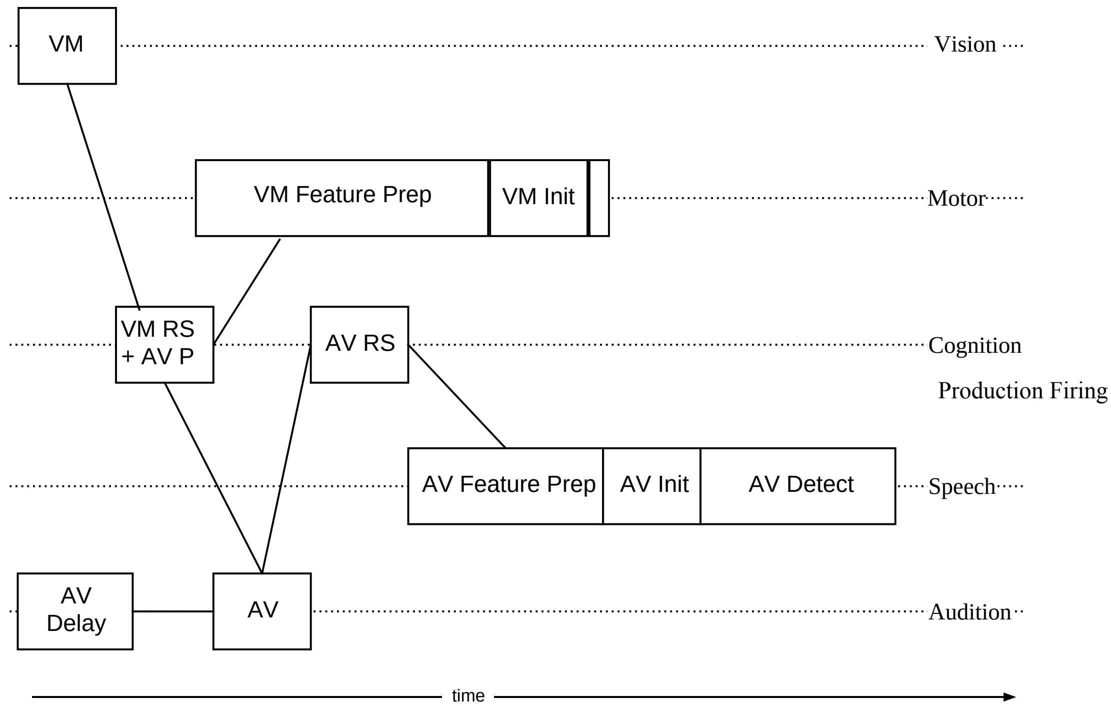

VM
Vision
VM Feature Prep
VM Init
Motor
VM RS
+ AV P
AV RS
Cognition
Production Firing
AV Feature Prep
AV Init
AV Detect
Speech
AV
Delay
AV
Audition
time

Figure 7

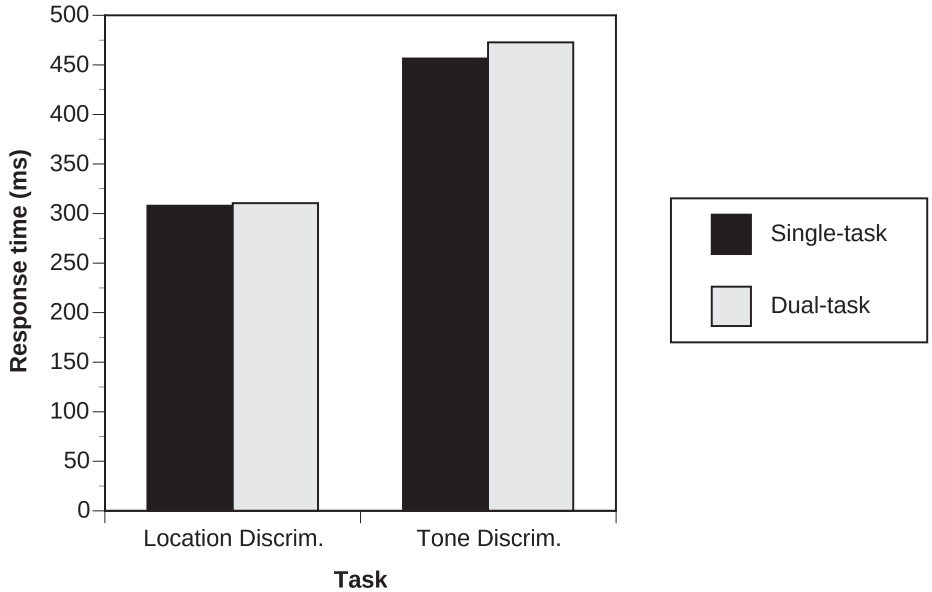

500
450
400
350
300
250
200
150
100
50
0
Response time (ms)
Location Discrim.
Tone Discrim.
Task
Single-task
Dual-task

Figure 8

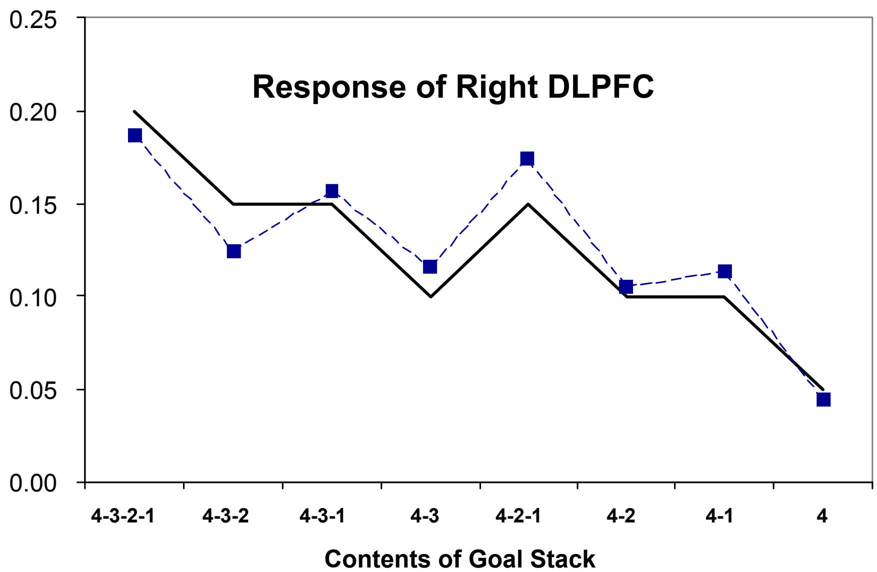

Response of Right DLPFC
0.25
0.20
0.15
0.10
0.05
0.00
4-3-2-1
4-3-2
4-3-1
4-3
4-2-1
4-2
4-1
4
Contents of Goal Stack

Figure 9

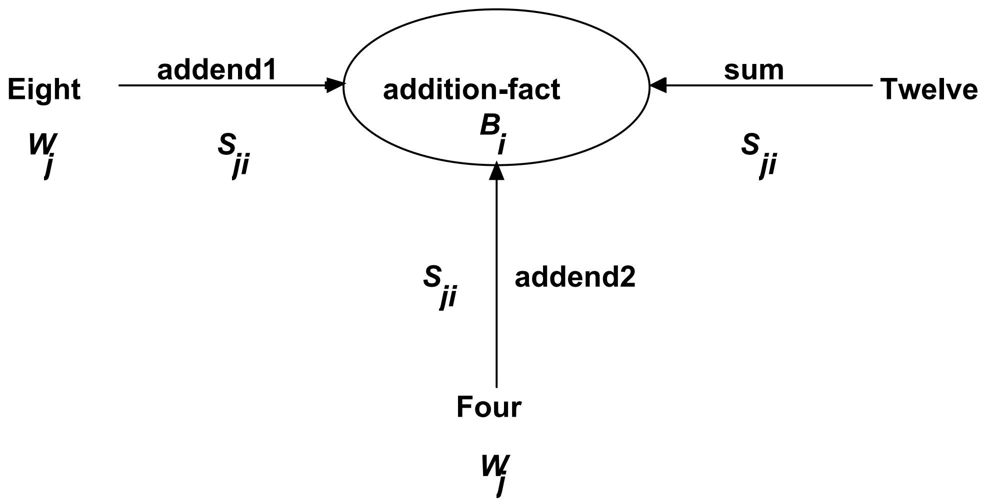

Eight
W_j
addend1
S_{ji}
addition-fact
B_i
sum
S_{ji}
Twelve
S_{ji}
addend2
Four
W_j

Figure 10

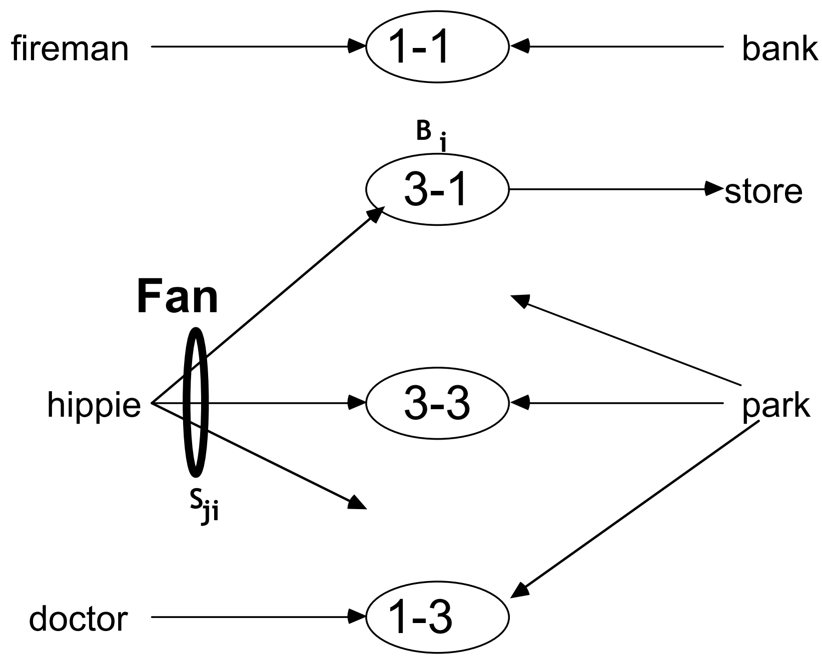
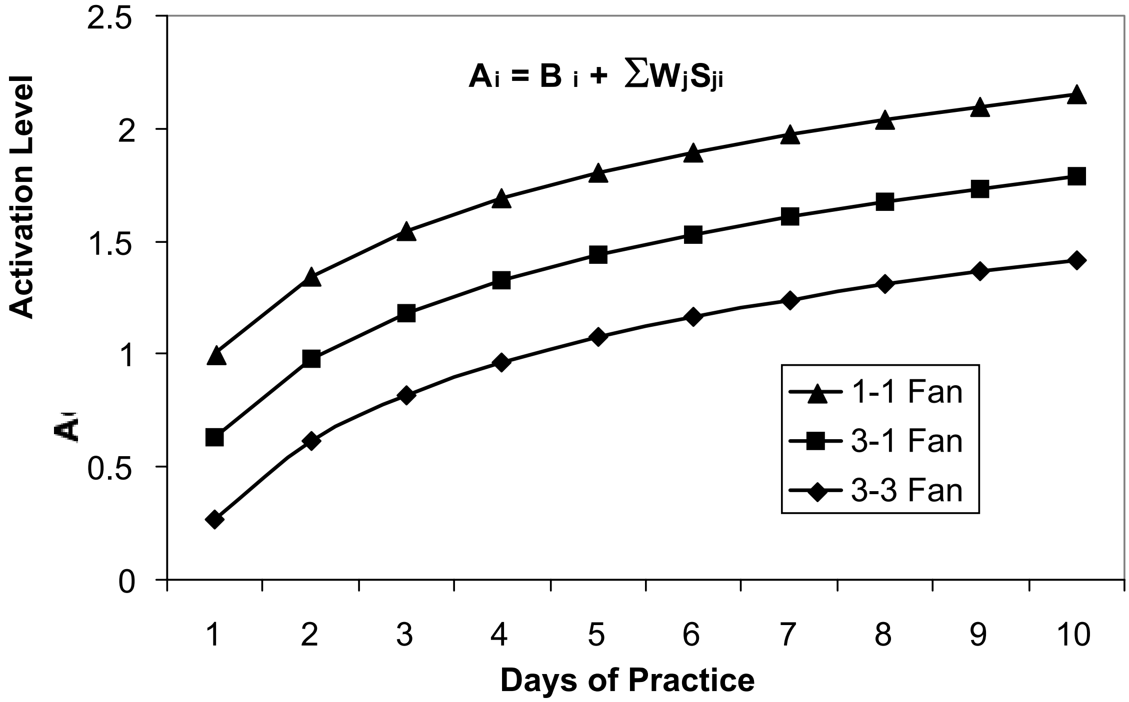


Figure 11


$A_i = B_i + \sum W_j S_{ji}$
Activation Level
Days of Practice
1-1 Fan
3-1 Fan
3-3 Fan
0
0.5
1
1.5
2
2.5
1
2
3
4
5
6
7
8
9
10

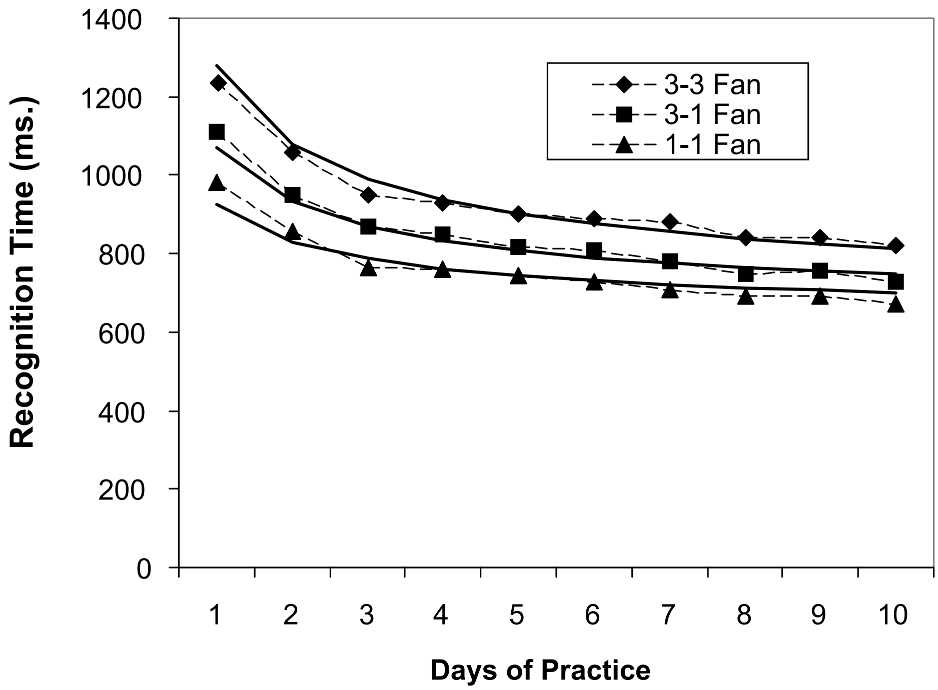

1400
1200
1000
800
600
400
200
0
Recognition Time (ms.)
3-3 Fan
3-1 Fan
1-1 Fan
1
2
3
4
5
6
7
8
9
10
Days of Practice

Figure 13

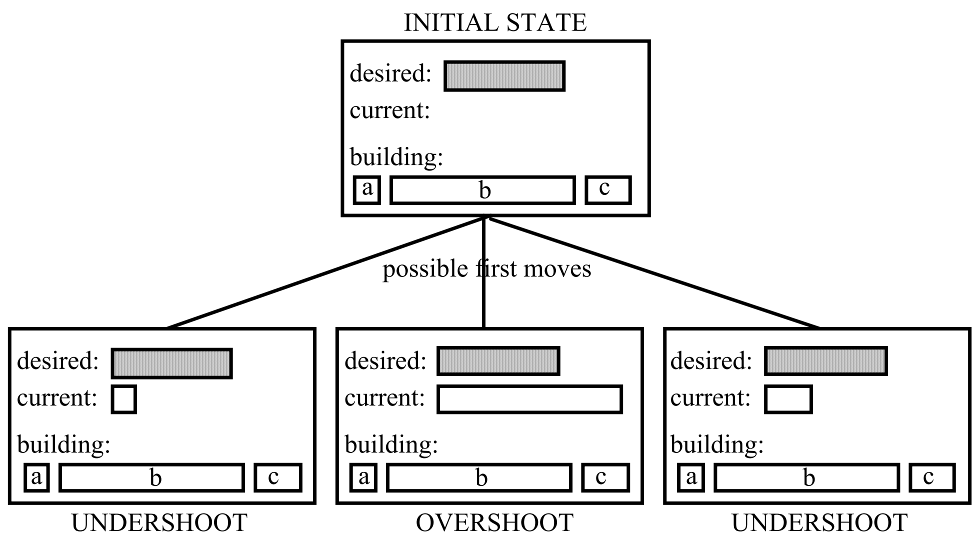

INITIAL STATE
desired:
current:
building:
a
b
c
possible first moves
desired:
current:
building:
a
b
c
UNDERSHOOT
desired:
current:
building:
a
b
c
OVERSHOOT
desired:
current:
building:
a
b
c
UNDERSHOOT

Figure 14

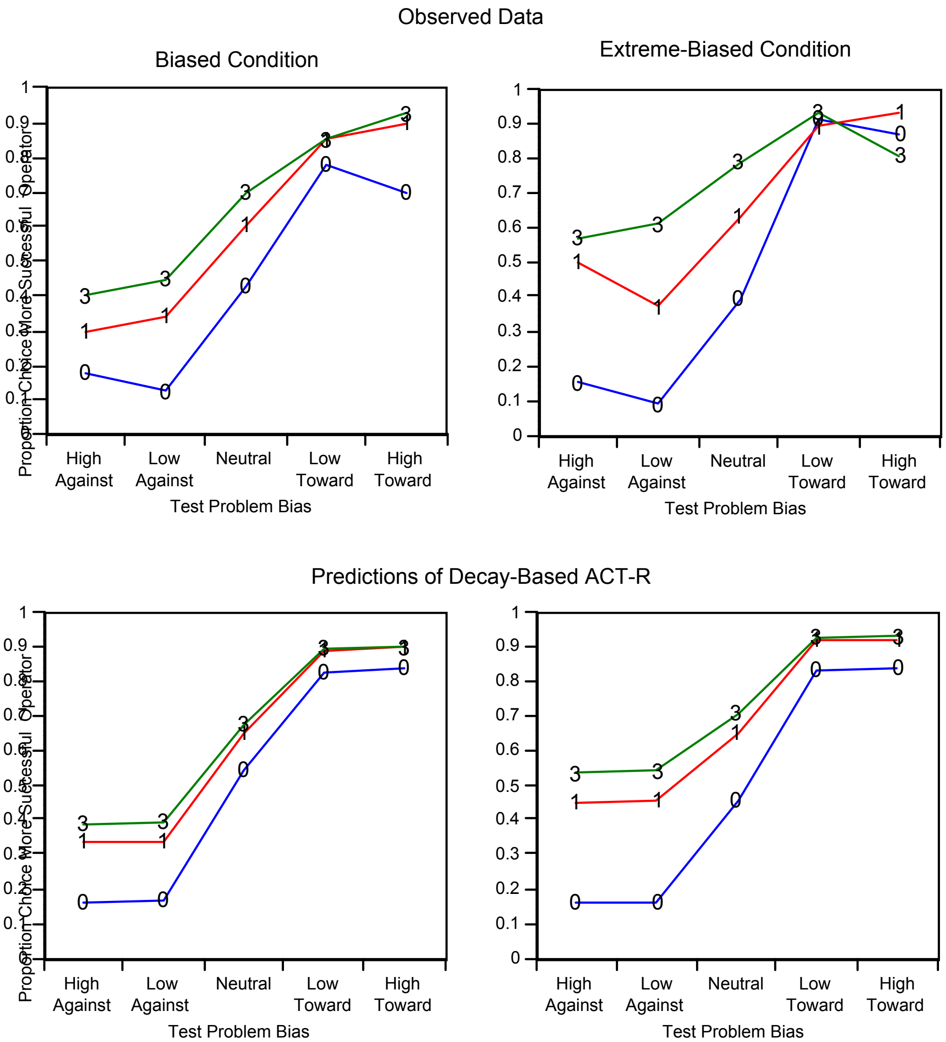


Figure 15

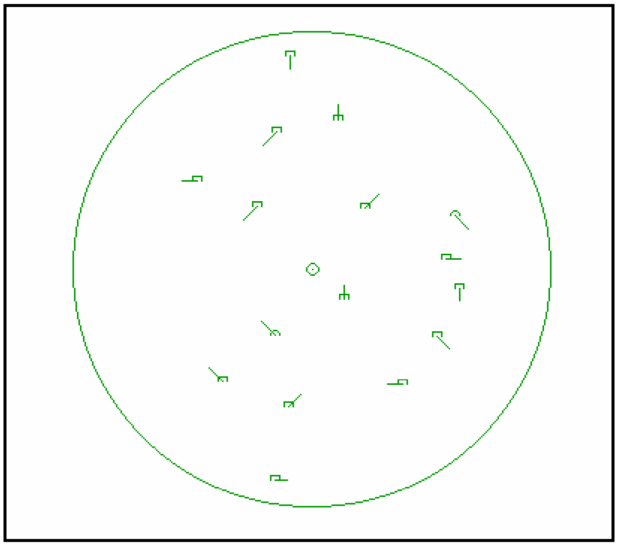


Figure 16a

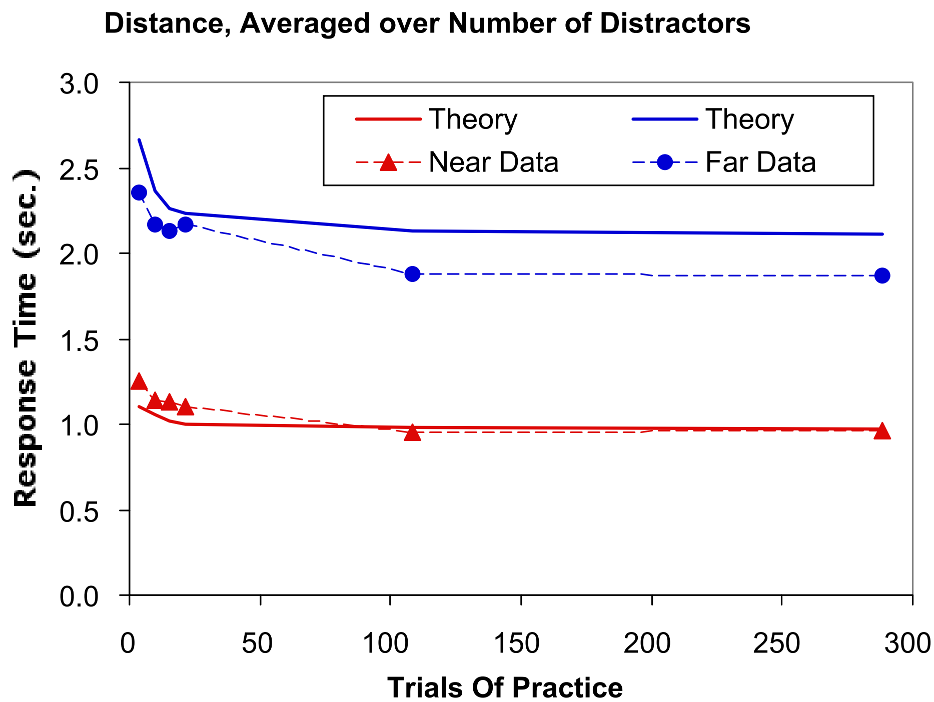


Figure 16b

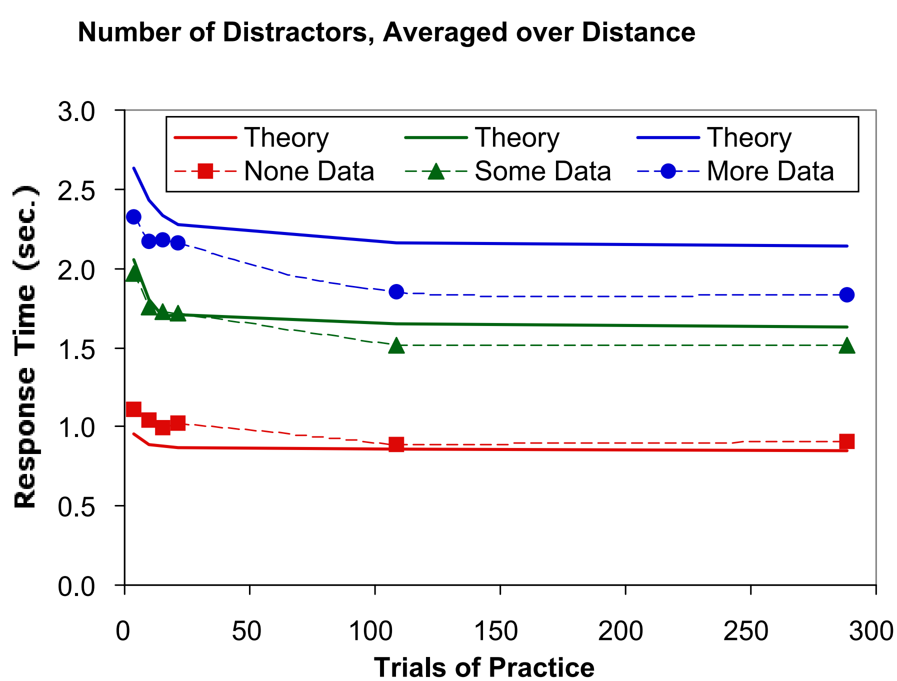
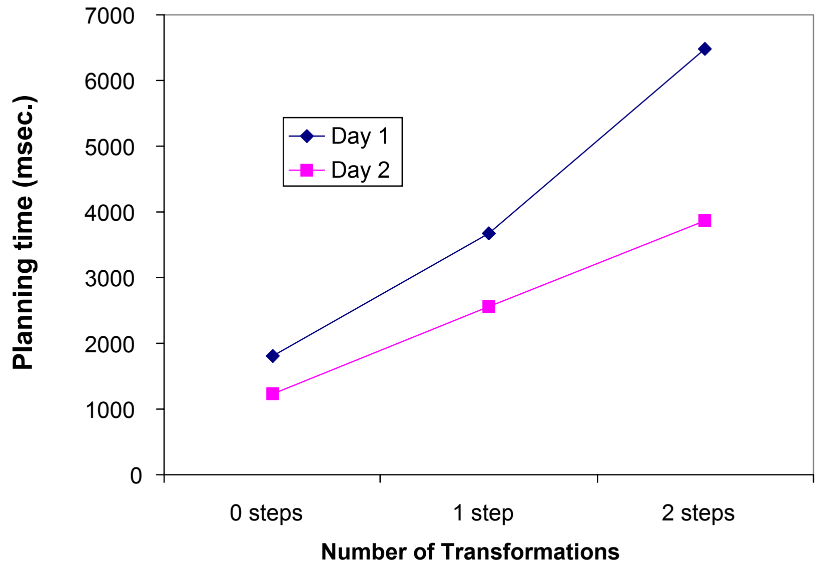


Figure 17


7000
6000
5000
4000
3000
2000
1000
0
Planning time (msec.)
Day 1
Day 2
0 steps
1 step
2 steps
Number of Transformations

ACT-R Buffer Activity during Solution of ❸P <=> ❷4❺3

Time	Step	Imaginal	Retrieval	Manual
3.1				
3.3	1	<=> ❷		
3.5	2	_ <=> ❷3		
3.7	3	_ <=> ❷3❺		
3.9	4	_ <=> ❷3❺4		
4.1				
4.3	5	_ P <=> ❷3❺4		
4.5	6	❸P <=> ❷3❺4		
4.7	7			
4.9			❸ means flip	
5.1				
5.3	8		args in	
5.5			2^{nd} and 4^{th}	
5.7			positions	
5.9	9	P <=> ❷4❺		
6.1	10	P <=> ❷4❺3		
6.3	11			Key 1
6.5				
6.7	12			Key 2
6.9				
7.1	13			Key 4
7.3				
7.5	14			Key 5
7.7				
7.9	15			Key 3
8.1				

Figure 19

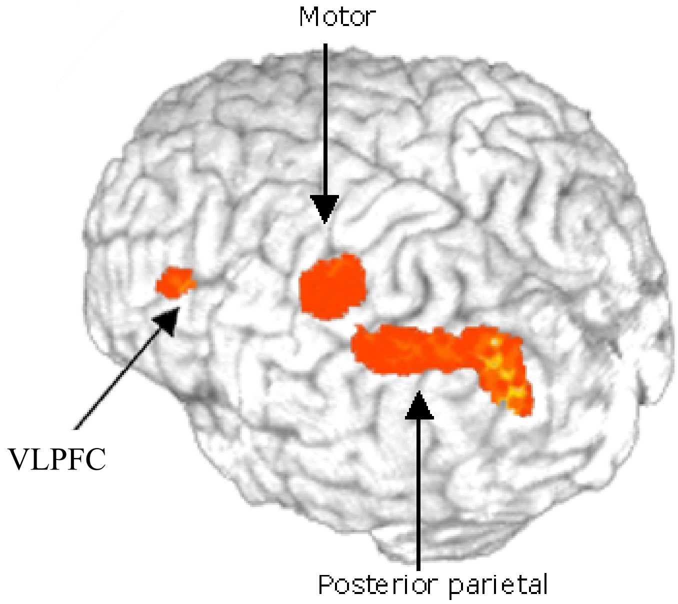
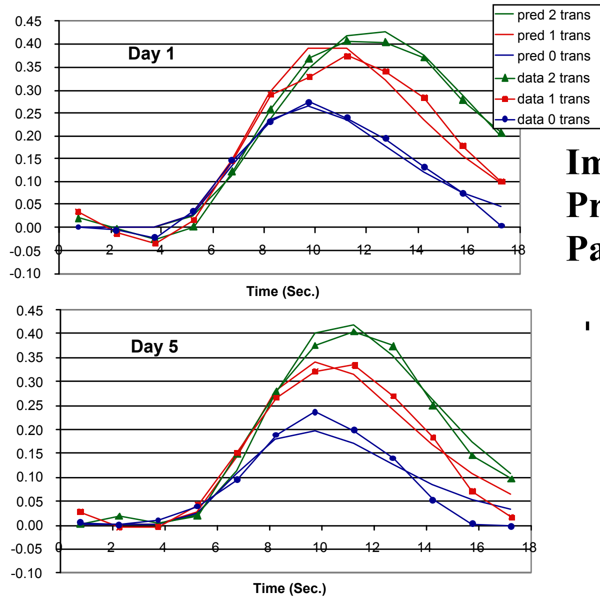
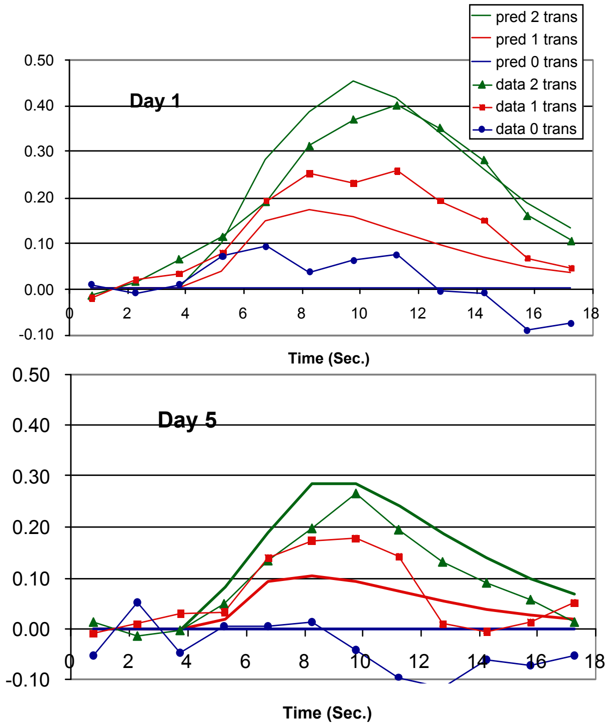


Figure 20



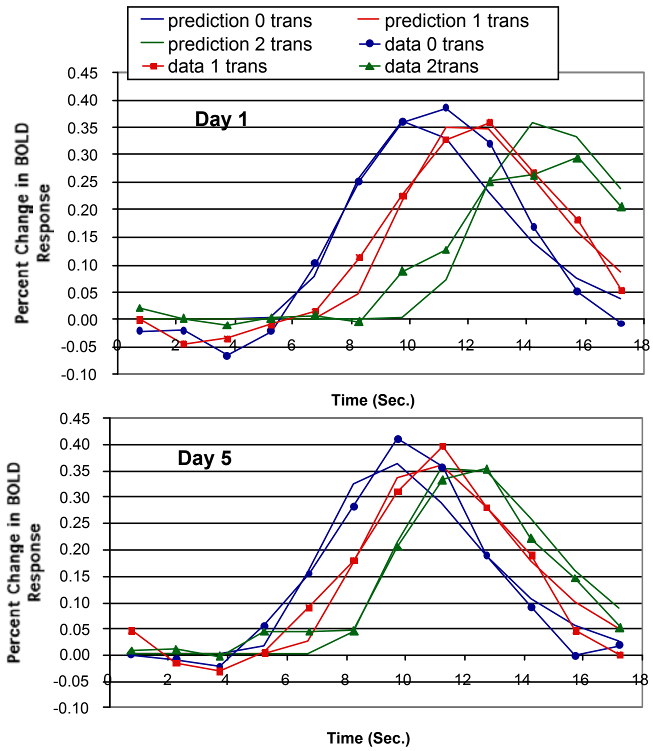
Imaginal Predicts Parietal

Figure 21

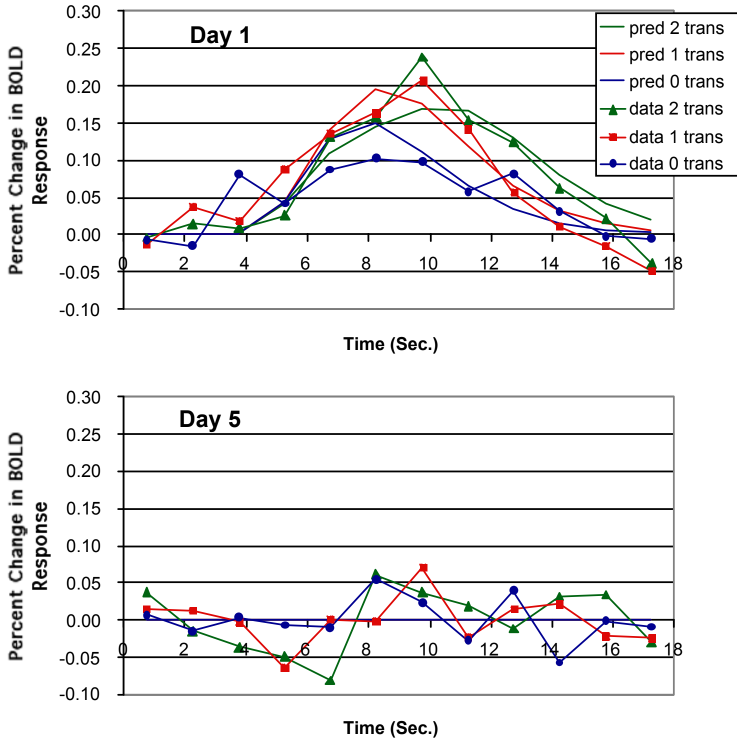


Retrieval Predicts VLPFC

Figure 22



Manual
Predicts
Motor



Firing of New Productions Predicts Caudate

[1] We will discuss at the end of the paper some of the recent controversy about the proliferation of modules (e.g., Fodor, 2000; Kosslyn, 2001; Uttal, 2001).
[2] These systems take advantage of the fact that it is possible to have the models interact with simulators and so it is not necessary to create real bodies and sensors in the simulation.
[3] There is one significant qualification on this claim – because of the lack of complex motor programs in ACT-R (which are essential for modeling driving) he had to reduce the default cycle time to in effect program these routines in ACT-R. However, this was done in advance of the test that we will be describing here. Also, Salvucci (personal communication) has subsequently shown that the model can drive given a more realistic motor module with a 50 msec cycle time.
[4] Even though this is the goal, we have not fully achieved it any of the research described in this paper
[5] There is, of course, a great deal of evidence that long-term memory, which is part of the retrieval module as distinct from the buffer, is associated with the temporal lobes and hippocampus.
[6] Specifically, in terms of Anderson & Lebiere, the decompositions described by Probability of Goal Equation 3.2 and Cost of goal Equation 3.3., production strength and the learning mechanism described by Production Strength Equation 4.4., and the associative learning described by Posterior Strength Equation 4.3.
[7] The actual value of this parameter in various instantiations of ACT-R has been the source of some confusion. In the first Visual Interface for ACT-R, all activity was serialized and so this value was 185 ms. However, in ACT-R 5.0 the actual system parameter is 85 ms because the same attention shift now requires two production firings.
[8] The numbers reported here are the same as those in Anderson, Lebiere, & Matessa, 1997. A subsequent recomputation of feature overlap leads to a .62 overlay and an estimate of a 115 msec. slope which is again close to the empirical number. (Anthony Hornoff, personal communication)
[9] This step depends on ACT-R's ability to tag items on the screen as attended.
[10] Note this is different than the visual module described earlier. The imaginal buffer has recently been added to a number of ACT-R models (e.g., Gunzelmann & Anderson, 2002) to model transformations of visually presented information.
[11] We will sometimes use the term CAPS to refer to this architecture in general.
[12] Meyer, Glass, Mueller, Seymour, and Kieras (2002) relate this to the alpha rhythm and propose that changes in the cycle time account for effects of cognitive aging.