The Past, Present, and Future of Cognitive Architectures

Niels Taatgen[12] and John Anderson[1]

[1]Carnegie Mellon University

[2]University of Groningen, the Netherlands

Please address correspondence to:

Niels Taatgen
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh PA 15213
Tel: +1 412 268 2815
Fax: +1 412 268 2844
E-mail: taatgen@cmu.edu

Running head: COGNITIVE ARCHITECTURES

Abstract

Cognitive Architectures are theories of cognition that try to capture the essential representations and mechanisms that underlie cognition. Research in cognitive architectures has gradually moved from a focus on the functional capabilities of architectures to ability to model the details of human behavior, and, more recently, brain activity. Although there are many different architectures, they share many identical or similar mechanisms, permitting possible future convergence. In judging the quality of a particular cognitive model, it is pertinent to not just judge its fit to the experimental data, but also its simplicity and ability to make predictions.

Keywords: Cognitive Architectures, Cognitive Modeling

The Past, Present, and Future of Cognitive Architectures

The mystery of the human mind is one of the few remaining frontiers of science. For some, the goal of cognitive science is to unravel this mystery, and to specify a theory that encompasses all aspects of cognition. The main champion of this idea was Allen Newell who identified the means to get there: Architectures of Cognition (Newell, 1990). The inspiration for such architectures can be traced back to Turing's idea of an intelligent computer (Turing, 1950). Turing thought that speed and memory capacity were the main barrier for the computers of that time to achieve machine intelligence. History has shown, however, that each achievement in artificial intelligence only made it clear how complex the puzzle of human intelligence, creativity and ingenuity is. The goal of architectures of cognition is to find the mechanisms and representations that specify a formal foundation for a unified theory of cognition, or, to quote Newell (1990):

> The question for me is how can the human mind occur in the physical universe. We now know that the world is governed by physics. We now understand the way biology nestles comfortably within that. The issue is how will the mind do that as well. The answer must have the details. I got to know how the gears clank and how the pistons go and all the rest of that detail…My question leads me down to worry about the architecture.

The Past of Cognitive Architectures

One of the earliest proto-architectures was the General Problem Solver (GPS, Newell & Simon, 1963). GPS tried to capture the essence of human problem solving by specifying a set of general mechanisms that are applicable to any problem. In order to make a model of solving a particular problem, all the modeler had to do was specify the problem in terms of GPS's representations, and run the system to arrive at a prediction of human problem solving behavior for that particular problem. This raises the question of what constitutes a good prediction of human behavior. In the early days of cognitive modeling there was a strong focus on functionality: the model had to be capable to solve potentially hard problems that people were also able to solve. Turing already made a case for this approach in his test for machine intelligence: as long as a machine's behavior is indistinguishable from human behavior, we have to conclude that the machine is intelligent (Turing, 1950).

GPS laid the foundation for the approach that is still the foundation for today's cognitive architectures: the architecture supplies a general theory of cognition that is independent of particular phenomena, to which the modeler adds representations to perform a specific task. The architecture is then simulated on a computer to produce behavior, which can subsequently be compared to human behavior on the same task. The idea of an architecture is taken from computer science (Brooks, 1962), which, in turn, borrowed it from the design of buildings. There are (at least) three big questions that this approach raises:

1. To what extent the intelligence is in the architecture or in the model? If all the intelligence is in the model (which is mostly true in the computer analogy), the architecture has little theoretical significance.
2. How does one evaluate the match between the behavior that can be produced by the architecture and human behavior? It is not obvious how to simultaneously measure the scope of behavior allowed by the architecture and also the details of specific behaviors.

3. What is the origin of the task-specific knowledge? Humans are not programmed with knowledge the way these architectures are typically (but not always) programmed.

To explore these issues, it is worthwhile to examine a particularly successful example of the functional approach: TacAir-Soar (Jones et al., 1999). TacAir-Soar is a model of a fighter pilot, and is built using the Soar cognitive architecture (Laird, Newell, & Rosenbloom, 1987). The goal of the model is to create multiple simulated pilots that can participate in large-scale warfare simulations, in which models participate alongside human pilots and other personnel. The main criterion for success is whether or not the models act like real pilots, and therefore contribute to a faithful experience for the humans that participate in the simulation. TacAir-Soar has impressive specifications and accomplishments. At the start of the simulated mission it is briefed, and can autonomously accomplish it goals. It is able to perform a wide variety of missions, and can communicate with control and other pilots using natural (albeit restricted) natural language. The system was successfully used in a large-scale Synthetic Theatre of War simulation (Jones et al., 1999), involving 3700 computer-controlled vehicles, among which approximately 100 aircraft controlled by TacAir-Soar models. The success of the model depended on several components. A goal hierarchy (Figure 1) was used to break up larger goals (e.g., intercept an airplane) into smaller goals (e.g., change bearing). To support this goal structure, the system contains more than 8000 production rules. The production rules propose, arbitrate and implement operators that lead to mental or physical actions. Production rules were derived from subject matter experts, standard doctrine and tactics.

From the functional perspective, TacAir-Soar was a great success: its behavior in the simulations was considered human-like, using only knowledge or input that pilots also had. It is harder to answer three questions we listed earlier, i.e.:

1. To what extent is the success of the model due to general properties of the cognitive architecture, or to the cleverness of the modelers and the quality of the domain expert input?

2. How strong a test is it to produced believable pilot behavior? Would it not have been possible to construct a totally different model that would also have satisfied the constraints of producing believable pilot behavior?

3. How do human pilots acquire the knowledge necessary to fly an airplane, and how do they organize and prioritize this knowledge. Even though Soar does have a learning mechanism, it was not used in the TacAir-Soar model, because a learning model would not contribute to the overall goals of the project.

The Present of Cognitive Architecures

In order to be more confident that a cognitive model is also accurate in the details of behavior, it is necessary to construct models that produce those details. Often assessing such accuracy leads to selection of more limited tasks, but ones that are chosen to test some of the details of the architecture. The results of these tests can then serve as a basis for models of complex tasks. An example by Altman and Trafton (1999, 2002) and Anderson and Douglass (2001) concerned the backbone of traditional production systems: the goal stack. A goal stack is used to represent the current state of the goal hierarchy. For example, in the TacAir-Soar model, the goal stack can contain the information that one is changing the bearing to heading 95 as a subgoal of intercepting an enemy plane as a subgoal of flying a mission. This means that the current goals is to change the bearing to heading 95, but as soon as that goal is accomplished, it will be "popped" from the goal stack, making intercepting the enemy plane the current goal. This goal may then spawn an additional subgoal. Although the goal stack is very useful from a functional perspective and key to the implementation of almost all computer languages, it also provides an error-free and effortless memory for goals. Altman and Trafton (1999) showed in a study of the Tower of Hanoi, the classical task normally used to support the plausibility of the human goal stack, that errors in the task are related to how long ago a goal on the stack had been accessed. A model that did not use the goal stack, but instead relied on declarative memory to represent unfinished goals, made accurate predictions about the error rates, something that would be impossible with a goal-stack mechanism. In a new set of Tower of Hanoi experiments collecting eye-movement data, Anderson and Douglass (2001) showed that their participants did not even retain all the goals needed to perform the task. Instead, they reconstructed goals that had decayed in memory by re-encoding the problem display. The consequence of these studies was that ACT-R removed the crutch of the goal stack from the architecture, forcing modelers to find more innovative and behavioral accurate means to represent their goals. This opened the door to more novel ways of treating goals, like in the *threaded cognition* theory by Salvucci and Taatgen (2008), which allows several concurrent goals to be active, but which limits the representation of these goals to control information. Any content information of these goals still has to be represented elsewhere, and is susceptible to decay and forgetting.

An even more recent development in cognitive architectures is to find additional constraints in neuroscience. One set of constraints can be derived from the fact that the brain implements the cognitive architecture. This means that all processes in the architecture at some level of abstraction have to be carried out by neurons. One approach is to try to implement an existing architecture as a neural network (Cho, Rosenbloom & Dolan, 1993, for Soar, and Lebiere & Anderson, 1993, for ACT-R). The ACT-R neural implementation faithfully reproduced the behavior of the standard symbolic version, except that it could only perform a single retrieval from declarative memory at a time because it was difficulty using standard connectionist mechanisms to simultaneously activate multiple items from a memory. This constraint was subsequently introduced in the standard architecture and moved the ACT-R architecture another step in faithfully modeling the limitations on human cognition. A different approach is to design a cognitive architecture directly at the level of neural networks. Although this approach is still prohibitive with respect to computational requirements, the Leabra architecture (O'Reilly & Munakata, 2000) has been successful in making headway in that direction.

A different source of constraints from neuroscience can be found in neuroimaging research. In order to model neuroimaging data, a mapping from model activity to brain activity is necessary. The most straightforward approach is to map components in the cognitive architecture onto regions of the brain. Anderson (2007) has matched up modules of the ACT-R architecture with areas of the brain, predicting fMRI activity in those regions by convolving module activity with a hemodynamic response function. An example of such research is a study of how children learn to solve algebra problems (Anderson, 2005). Children had to solve algebra equations of varying complexity in five sessions with sessions 1 and 5 in the fMRI scanner. The model of learning algebra put varying demands on the various modules depending on the complexity of the problem and the stage of learning. For example, the model predicted no change in the demands on the module associated with motor actions, because the response always consisted of a single key press (Figure 2a). Activity in the region associated with declarative memory shows large difference (Figure 2b). A first impact is that of problem complexity: solving $3x = 6$ only requires the retrieval of a single arithmetic fact from declarative memory, while $3x + 2 = 6$ requires two such retrievals. Moreover, the model predict that declarative activity on day 5 will have decreased because arithmetic facts have become more active, and ACT-R production compilation process has learned task-specific rules that bypass declarative memory.

A different approach by Just and Varma (2007) is based on the assumption that a particular function can be carried out by multiple areas, even though each function is linked to a preferred area. Their capacity-based architecture (4CAPS) assumes there is a limited processing capacity within each module in the brain, and also a maximum overall capacity. If a particular function exceeds the capacity needs within its preferred area, it can carry over activity to nearby or contra-lateral areas that can also carry out that function. The many-many mapping between functions and brain areas that Just and Varma propose is attractive from the viewpoint of model flexibility, and is consistent with the notion that the brain harbors a great deal of redundancy. Although this approach seems very different from Anderson's, the functions in the Just and Varma models are much more specific (e.g., a structure-building center for language production) than the more generic Anderson modules (where language production at least involves both the declarative and procedural modules, and probably others). This means that sentence building in the Just and Varma models is linked to specific functional modules (which functions can be carried out in multiple areas), while in the Anderson architecture multiple generic modules are involved in the specific task of sentence generation. In the end, both predict that multiple brain areas are involved complex tasks, and more areas may become involved if task complexity increases. The details of these activations do of course differ, but that can in some cases be due to the particular model than the architecture in general.

The Future of Cognitive Architectures

One of the roadblocks that cognitive scientists interested in cognitive architectures have to overcome is that it takes a substantial intellectual commitment to learn to understand models of a particular architecture, and to learn to construct models. In addition to the willingness to invest this effort there is the question of the choice of architecture. The problem is that a choice can only be properly made after at least some intellectual investment, making it nearly infeasible to compare multiple architectures.

In a talk on February 24, 1998, Herbert Simon already identified this problem, remarking that there are too many brand names in cognitive modeling. This branding, according to Simon, focused attention too much on systems as a whole instead of mechanisms within the systems that are successful in cognitive models. Branding would make it hard to adapt successful mechanism from one model to the other. Fortunately, the last decade has produced considerable cross-fertilization among architectures. ACT-R (Byrne & Anderson, 2001) and Soar (Chong & Laird, 1997) have adopted perceptual motor modules from Epic (Meyer & Kieras, 1997). Soar has adopted declarative memory from ACT-R (Chong, 2003). The success of the reinforcement-learning algorithm (Barto, Sutton, & Brouwer, 1981) has prompted both Soar (Nason & Laird, 2004) and ACT-R (Fu & Anderson, 2006) to adapt it. The good news for the consumer is that if this development continues, to choice of architecture does not imply a strong theoretical commitment, because most mechanisms that are successful in cognitive modeling are probably included. Even the gap between symbolic and neural networks is gradually bridged. Several architectures include both symbolic and neural representations (Clarion: Sun, Merrill, & Peterson, 2001; 4Caps: Just & Varma, 2007). In addition, a combined Leabra/ACT-R architecture is in development (Lebiere et al., 2008), which allows the modeler to choose between neural or symbolic versions of a particular module. A first set of demonstrations involved ACT-R with a Leabra visual module (Figure 3). In one of these demonstrations, the architecture controlled a simulated human in a computer game, which had to navigate through rooms and recognize objects in those rooms (Lebiere et al., 2008). A second demonstration is a model of the Attentional Blink (Taatgen et al., 2007; Taatgen et al., in press). In the Attentional Blink task, visual objects are presented at a rate of 100 ms/object. The task of the participant is to identify two letters in the stream that otherwise consists of digits. Results show that when the two targets are approximately 200-400 ms apart, so separated by one to three distractors, the second target is reported with a much lower accuracy than when the separation is either only 100 ms, or 500 ms or more. A control mechanism in ACT-R that suppresses target detection during memory consolidation can explain this phenomenon, but this model falls short in explaining some of the details. One of these details is that when the two targets are presented consecutively, they are often reported in the wrong order. The Leabra visual module can explain this error: while it is classifying a particular symbol, activity from the previous symbol is still in the network (due to the high presentation rate), so two output cells, one for the current and one for the previous symbol may be active at the same time, making it impossible to determine the order in which they appeared.

A criticism of cognitive models/architectures is that they offer so many degrees of freedom that they can model any conceivable outcome of an experiment (e.g., Roberts & Pashler, 2000). Models often have numerical parameters that can be set to different values, affecting the outcome of the model's predictions. Even worse, there is an almost infinite freedom is specifying the initial knowledge and strategies of the model, allowing many different ways to model a single task. Given a particular model, why are we supposed to believe that it is the correct model, instead of a model that produces the same behavior but is completely different? It is worthwhile to observe that this is not just a problem for cognitive architectures, but for any scientific theory. Potential remedies can therefore to some extent be borrowed from science in general.

A strong criterion for a good model is *simplicity*. If there are multiple models that fit a particular dataset, we prefer the one that is simpler. Simplicity can be defined by the number task-specific elements that a model contains. An example of such a comparison is between two models of the Sugar Factory task (Berry & Broadbent, year). In the Sugar Factory task, participants have to decide how many workers to assign to a factory each day in order to establish a particular production goal. The Wallach and Lebiere (Lebiere, Wallach & Taatgen, 1997) model only consisted of two production rules: one that implemented some initial guess, and one that tried to retrieve past experiences from memory. The model learned to do the task because it accumulated useful experiences. This very simple model proved to be quite adequate in fitting the data. Fum and Stocco (2003) constructed a model that consisted of five competing strategies. Their model learned the task because it learned which of these strategies were more successful. Based on simplicity we would prefer the Wallach and Lebiere model[1].

[1] Unfortunately, the story is not as simple, because Fum and Stocco presented some new data that was fit by their model and was problematic for the original Wallach and Lebiere model. A modification of the model, however, was able to capture those data.

Counting production rules is not the only or necessarily best way of assessing complexity, because for some rules it is reasonable that they are part of people's general repertoire of cognitive strategies, while others are very task specific. The strategy of retrieving previous solutions has not only been used in the Sugar Factory model, but also in a model of Rock, scissors, paper (West et al., 2005) of the past tense (Taatgen & Anderson, 2002), of categorization (Anderson & Betz, 2001), and several others (see Anderson, 2007, for a complete overview). An alternative measure of complexity can therefore be the amount of task-specific knowledge that has to be brought into a model. According to this type of measure, the best model would be a model that would require no task-specific knowledge at all. Models that retrieve past experiences approximate this ideal for at least a range of tasks, but they do need some specific initial strategy that can be used before any experiences have been gathered. In actual experiments participants always receive some sort of instruction, which is one of the sources of initial task-specific knowledge. If the model can actually parse and interpret this information, we can argue that the model has does not start out with any task-specific knowledge. A completely general version of this model would be able to carry out any task without incorporating task-specific knowledge, and would require solving a number or very hard problems (among which language understanding, and ultimately passing the Turing Test). Nevertheless, several approximations of this idea already been implemented in models of learning Algebra from instruction (Anderson, 2005, 2007), learning a simplified Air Traffic Control task (Taatgen & Lee, 2003), and learning to operate the Flight Management System of a commercial airplane (Taatgen et al., 2008).

As models become simpler according to the criterion of minimizing task-specific knowledge, they become more general in their ability to capture phenomena that they have not been designed for originally. This allows them to aim for another and even stronger criterion for a successful theory: the ability to predict the outcome of new experiments. Again, general prediction is still beyond the current state of the art. It is possible to predict fMRI data on the basis of a model that has been fit on just the behavioral data, and there are now several successful examples of this type of prediction (Anderson, 2007). It is also possible to construct a model on the basis of a first experiment, and then use that model to predict the outcome of a new experiment that extends the first. For example, in our model of Flight Management Systems, we extended a first Experiment with trials in which participants had to solve problems that were partially completed or that contained errors. The model that we constructed on the basis of the first experiment was not only able to solve these new problems, but made an accurate prediction of the solution times and error rates (Taatgen et al, 2008).

In conclusion, what will the future of cognitive modeling be? We do not expect that the branding of cognitive architectures will disappear or that all researchers flock to a single one. But we may expect an emergence of shared mechanisms and principles that will gradually unify the field. The chasms between the different paradigms in cognitive modeling are gradually mellowing with the recognition that no single theory can be right at all levels, restoring a balance between incremental and revolutionary science. Given the current interest in neuroimaging, the correspondence between model and brain activity will become more important. Eventually, cognitive models have to live up to the expectations of strong scientific theories, in that they are both general and are able to predict.

References

Altmann, E. M., & Trafton, J. G. (1999). Memory for goals: an architectural perspective. *Proceedings of the twenty first annual meeting of the Cognitive Science Society* (pp. 19-24). Hillsdale: NJ: Erlbaum.

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

Anderson, J. R. (2005). Human symbol manipulation within an integrated cognitive architecture. *Cognitive Science, 29*(3), 313-341.

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford university press.

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

Anderson, J. R., & Douglass, S. (2001). Tower of Hanoi: evidence for the cost of goal retrieval. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 27*(6), 1331-1346.

Barto, A. G., Sutton, R. S., & Brouwer, P. S. (1981). Associative search network: A reinforcement learning associative memory. *Biological Cybernetics, 40*(3), 201-211.

Brooks, F. P. (1962). Architectural philosophy. In W. Buchholz (Ed.), *Planning a Computer System* (pp. 5-16). New York: McGraw-Hill.

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

Cho, B., Rosenbloom, P. S., & Dolan, C. P. (1993). Neuro-Soar: a neural-network architecture for goal-oriented behavior. In P. Rosenbloom, J. E. Laird & A.

Newell (Eds.), *The Soar papers (vol. II): research on integrated intelligence* (pp. 1199-1203). Cambridge, MA: MIT Press.

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

Chong, R. S. (2003). The addition of an activation and decay mechanism to the Soar architecture. *Proceedings of the fifth international conference on cognitive modelling* (pp. 45-50). Bamberg, Germany: Universitaets-Verlag Bamberg.

Fu, W. T., & Anderson, J. R. (2006). From recurrent choice to skill learning: a reinforcement-learning model. *Journal of Experimental Psychology: General, 135*(2), 184-206.

Fum, D., & Stocco, A. (2003). Instance vs. rule-based learning in controlling a dynamic system. *Proceedings of the fifth international conference on cognitive modelling*. Bamberg, Germany: Universitaets-Verlag Bamberg.

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

Just, M. A., & Varma, S. (2007). The organization of thinking: what functional brain imaging reveals about the neuroarchitecture of complex cognition. *Cognitive, Affective, & Behavioral Neuroscience, 7*(3), 153-191.

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

Lebiere, C., & Anderson, J. R. (1993). A connectionist implementation of the ACT-R production system. *Fifteenth Annual Conference of the Cognitive Science Society* (pp. 635-640). Hillsdale, NJ: Erlbaum.

Lebiere, C., O'Reilly, R., Jilk, D., Taatgen, N. A., & Anderson, J. R. (2008). The SAL integrated cognitive architecture. In A. Samsonovich (Ed.), *Biologically Inspired Cognitive Architectures: Papers from the AAAI 2008 Fall Symposium* (pp. 98-104). Menlo Park, CA: AAAI Press.

Lebiere, C., Wallach, D., & Taatgen, N. A. (1998). Implicit and explicit learning in ACT-R. In F. E. Ritter & R. M. Young (Eds.), *Second european conference on cognitive modelling* (pp. 183-189). Nottingham, UK: Nottingham university press.

Meyer, D. E., & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance: I. Basic mechanisms. *Psychological Review, 104*(1), 3-65.

Nason, S., & Laird, J. E. (2004). Soar-RL: integrating reinforcement learning with Soar. *Cognitive Systems Research, 6*(1), 51-59.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard university press.

Newell, A., & Simon, H. (1963). GPS, a program that simulates human thought. In E. A. Feigenbaum & J. Feldman (Eds.), *Computers and Thought* (pp. 279-293). New York: McGraw-Hill.

O'Reilly, R. C., & Munakata, Y. (2000). *Computational Explorations in Cognitive Neuroscience* Cambridge, MA: MIT Press.

Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review, 107*(2), 358-367.

Salvucci, D. D., & Taatgen, N. A. (2008). Threaded Cognition: An Integrated Theory of Concurrent Multitasking. *Psychological Review, 115*(1), 101-130.

Sun, R., Merril, E., & Peterson, T. (2001). From implicit skills to explicit knowledge: A bottom-up model of skill learning. *Cognitive Science, 25*(2), 203-244.

Taatgen, N. A., & Anderson, J. R. (2002). Why do children learn to say "Broke"? A model of learning the past tense without feedback. *Cognition, 86*(2), 123-155.

Taatgen, N. A., Huss, D., Dickison, D., & Anderson, J. R. (2008). The acquisition of robust and flexible cognitive skills. *Journal of Experimental Psychology: General, 137*(3), 548-565.

Taatgen, N. A., Juvina, I., Herd, S., Jilk, D., & Martens, S. (2007). Attentional Blink: an internal traffic jam? *Proceedings of the eight international conference on cognitive modeling* (pp. 79-84). New York: Psychology Press.

Taatgen, N. A., Juvina, I., Schipper, M., Borst, J., & Martens, S. (in press). Too much control can hurt: a threaded cognition model of the attentional blink. *Cognitive Psychology*.

Taatgen, N. A., & Lee, F. J. (2003). Production compilation: A simple mechanism to model complex skill acquisition. *Human Factors, 45*(1), 61-76.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind, 59*, 433-460.

West, R. L., Stewart, T., Lebiere, C., & Chandrasekharan, S. (2005). Stochastic resonance in human cognition: ACT-R vs. game theory, associative neural networks, recursive neural networks, q-learning, and humans. In B. Bara, L. Barsalou & M. Bucciarelli (Eds.), *Proceedings of the 27th Annual Conference of the Cognitive Science Society* (pp. 2353-2358). Mahwah, NJ: Erlbaum.

Author Note

Niels Taatgen, Department of Psychology, Carnegie Mellon University and Department of Artificial Intelligence, University of Groningen, John Anderson, Department of Psychology, Carnegie Mellon University.

This research was supported by Office of Naval Research grant N00014-08-10541 and Air Force Office of Scientific Research grant FA95500710359 awarded to N. Taatgen.

Correspondence concerning this article should be addressed to Niels A. Taatgen, Department of Psychology, Carnegie Mellon University, Pittsburgh PA 15123. E-mail: taatgen@cmu.edu

Figure Captions

Figure 1. Example of part of the goal hierarchy of TacAir-Soar

Figure 2. BOLD response results (dashed lines) and model predictions (solid lines) contrasting the different problem difficulties and the effects of learning in (a) the Motor region and (b) the Prefrontal region. Figure reprinted from "Human symbol manipulation within an integrated cognitive architecture," by J.R. Anderson, 2005, *Cognitive Science*, *29,* p. 329 and p. 330. Copyright 2005 Cognitive Science Society. Reprinted with permission.

Figure 3. The combined Leabra and ACT-R architecture as used for the Attentional Blink model, in which the Leabra neural network replaces ACT-R's visual module

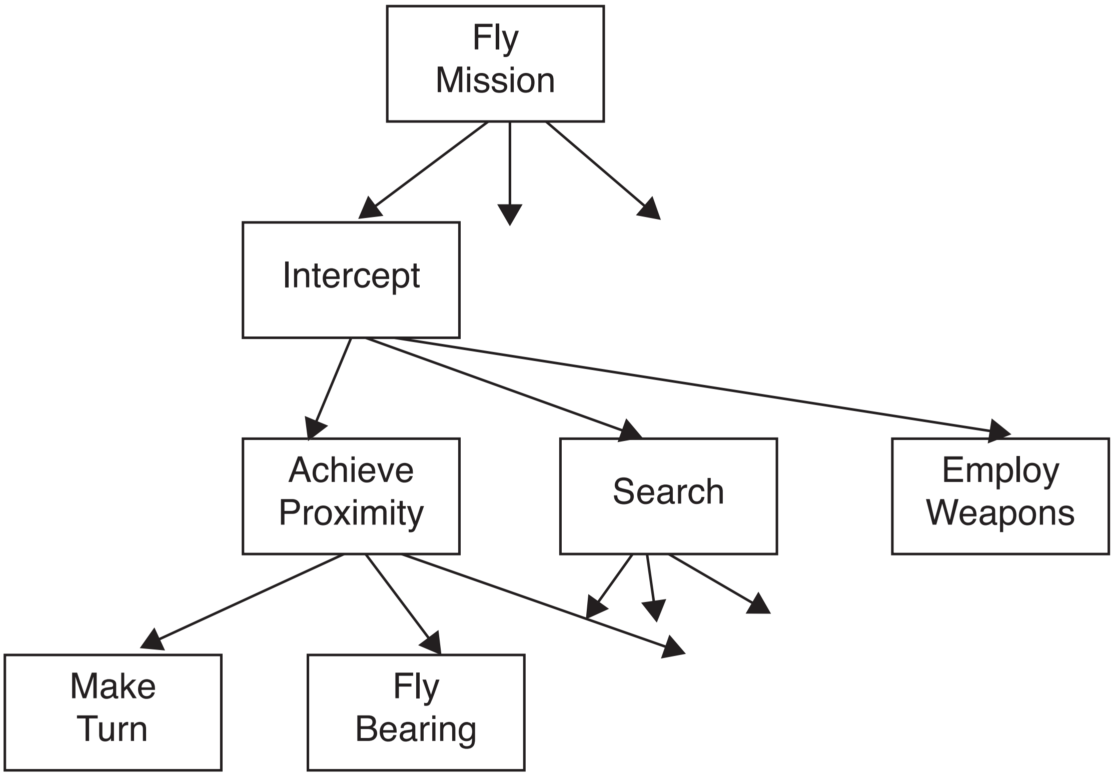
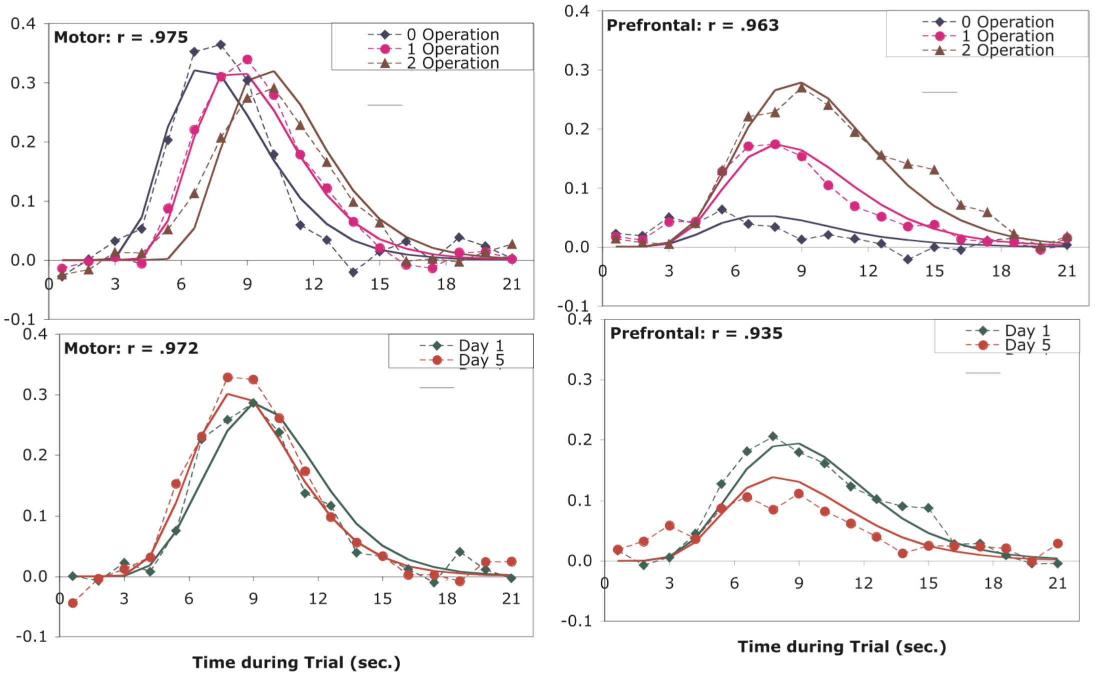

Fly Mission
Intercept
Achieve Proximity
Search
Employ Weapons
Make Turn
Fly Bearing

Figure 1


Motor: r = .975
0 Operation
1 Operation
2 Operation
Prefrontal: r = .963
0 Operation
1 Operation
2 Operation
Motor: r = .972
Day 1
Day 5
Prefrontal: r = .935
Day 1
Day 5
0.4
0.3
0.2
0.1
0.0
-0.1
0
3
6
9
12
15
18
21
Time during Trial (sec.)
Time during Trial (sec.)

(a)

(b)

Figure 2

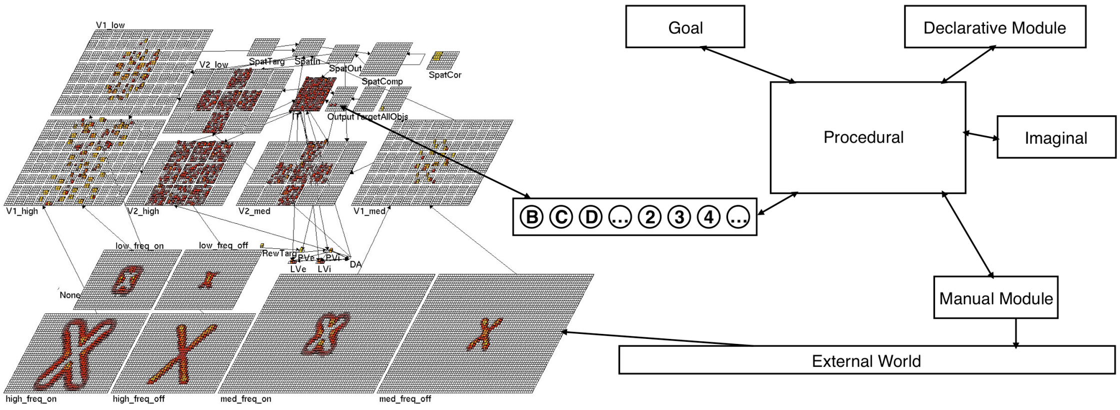

V1_low
V2_low
SpatTarg
SpatIn
SpatOut
SpatComp
SpatCor
IT
Output
TargetAllObjs
V1_high
V2_high
V2_med
V1_med
low_freq_on
low_freq_off
RewTarg
PVe
PVi
LVe
LVi
DA
None
high_freq_on
high_freq_off
med_freq_on
med_freq_off
Goal
Declarative Module
Procedural
Imaginal
B C D ... 2 3 4 ...
Manual Module
External World

Figure 3