Advances in Cognitive Systems: Papers from the 2011 AAAI Fall Symposium (FS-11-01)

Evaluating Integrated, Knowledge-Rich Cognitive Systems

Randolph M. Jones, Robert E. Wray III
Soar Technology
rjones@soartech.com, wray@soartech.com

Abstract

This paper argues the position that an essential approach to the advancement of the state of the art in cognitive systems is to focus on systems that deeply integrate knowledge representations, cognitive capabilities, and knowledge content. Integration is the path to aggregating constraints in ways that improve the science of cognitive systems. However, evaluating the role of knowledge among these constraints has largely been ignored, in part because it is difficult to build and evaluate systems that incorporate large amounts of knowledge. We provide suggestions for evaluating such systems and argue that such evaluations will become easier as we come closer to applying usefully new, integrated learning mechanisms that are capable of acquiring large and *effective* knowledge bases.

Introduction

The call for papers for this symposium includes a call to return to advanced cognitive systems that "reproduce the entire range of human cognitive capabilities". An essential approach to the advancement of cognitive systems is to integrate knowledge representations, cognitive capabilities, *and knowledge content*. For the purposes of this paper, we refer to such systems as *integrated, knowledge-rich cognitive systems* (IKRCSs). IKRCSs follow the formulation common to most intelligent systems that *Agent = Architecture + Knowledge*. The primary distinctions for an IKRCS are:

1. The target agent or behavior is comparable in scope (breadth and depth) to the performance of a human actor within a task domain. A Taxi IKRCS would gather destination preferences via a speech dialog, plan a route, and maneuver the vehicle along the route, adapting to the dynamic traffic situation as it was experienced.
2. Because the scope of the task domain and the desired performance, it is usually the case that the *knowledge content* of the resulting system is significantly greater in scope than the majority of intelligent systems applications, and it must be used across multiple functional capabilities.

In this paper, we argue that advancing the science of cognitive systems depends on building and evaluating IKRCSs. In ambition, IKRCSs represent one of the fundamental goals of artificial intelligence. Having such artifacts available for study should result in greater insights about the requirements of cognitive systems. However, there are two core issues that must be addressed to achieve this result:

1. We need to understand how to evaluate IKRCSs as a class of intelligent systems. The primary challenge is the requirement to evaluate characteristics that are often not readily quantifiable in a general way, such as robustness, adaptivity, and taskability.
2. We need to understand how to evaluate the knowledge content of an IKRCS. The knowledge content of an IKRCS plays a huge role in the resulting behavior of the system, but most evaluations tend to focus on system performance and the role of the underlying architecture. We argue that IKRCS knowledge content is a primary independent variable in IKRCS development. It must be considered alongside the architecture when evaluating IKRCSs.

Although our arguments are, at this point, more anecdotal than empirical, they are based on decades of research and development building IKRCSs. The suggested approaches to evaluation focus on combined constraints. For any subset of evaluation tasks, there may be many systems that perform effectively. IKRCSs *require* integrated cognitive capabilities. It is not sufficient to show that each component of a cognitive system works in isolation from the others. Cognition requires the interdependent operation of the mechanisms, knowledge representations, *and knowledge content* in the system. Useful evaluation must ensure the imposition of serious and numerous constraints on what could be considered to be positive outcomes.

We outline the importance of integrated approaches to cognition in the development of IKRCSs, further outline the relatively unexplored role of knowledge content in the evaluation of IKRCSs, and offer potential approaches to evaluation motivated by the insight that knowledge content and architecture are at least equal contributors in IKRCSs.

Integrated Approaches to Cognition

The components of cognitive systems depend intimately on each other to produce cognition. An advanced cognitive system must *integrate* components rather than merely *combine* them. But such systems must include the effective use of knowledge among the range of cognitive capabilities.

This view contrasts with the dominant trend to cognitive science, which is to identify and investigate thoroughly individual cognitive constructs and capabilities. These are the "functional capabilities" on which papers submitted to the symposium might appropriately focus. Research on specific components is important to the advancement of cognitive systems, because individual cognitive capabilities must be computationally represented if we hope to create advanced cognitive systems. However, cognition is known to be a complex interplay between a variety of mechanisms, each operating on mutually shared (and sometimes competing) knowledge representations (D'Amasio, 2010). Focusing on any of these components in isolation from the others is dangerous, because the interoperation of multiple components imposes constraints that are essential to cognition.

Integrated approaches take as a foundational assumption that the interplay between knowledge representations and mechanisms is central to achieving broad, robust computational intelligence. Arguably, the most significant work in integrated cognition focuses on *cognitive architectures* (Langley, Laird, & Rogers, 2009). A particular cognitive architecture might contain one or more models of short-term memory and long-term memory, algorithms for relating knowledge and making choices, algorithms for perception and action, algorithms for learning new long-term knowledge, as well as other components.

The importance of an integrated approach is illustrated by considering how cognitive architecture components constrain each other. For example, representation of short-term memory items is intimately tied to the algorithms for storage and retrieval. In turn, the decision-making algorithms must be sensitive to the constraints on memory retrieval. Attention mechanisms must work with the components that use attention, such as perceptual systems and memory models. In general, in an integrated cognitive architecture, one cannot make changes to one component without propagating new constraints to other components.

Integration thus imposes coherence on the wide variety of processes that contribute to cognition. However, a notable difficulty with pursuing research into cognitive architectures has to do with evaluation. Evaluation of an integrated system must take place at a higher level of abstraction than evaluation of the individual components, and it is particularly difficult to specify formal evaluation criteria for these higher levels of abstraction, as well as for complex systems in general.

As an example, Laird et al. (2009) have suggested a variety of measures for evaluating complex cognitive systems. These include "concrete" measures of performance and scalability, which are relatively well defined and quantifiable (although even these can take many forms in a complex system). But most of the measures are "abstract", including generality, expressivity, robustness, instructability, taskability, and explainability. These are certainly characteristics for advanced cognitive systems to strive for, but it remains ill-defined how to measure them, or which dependent statistics to collect. Wray and Lebiere (2007) describe some of the challenges of creating domain-general metrics for such abstract system characteristics.

In spite of the difficulties in evaluating integrated cognitive architectures, it remains clear that if we wish to advance the state of cognitive systems, we must look increasingly to integrated systems rather than focusing on the individual components of cognition. In our view, the role of knowledge content is significantly under-studied in the evaluation of IKRCSs.

The Importance of Knowledge

Although integrated approaches to cognitive systems are essential, there has not been sufficient attention to the importance of knowledge as part of integration. Significant amounts of usefully represented knowledge is (or ought to be) an essential feature of advanced cognition. Even among humans, knowledge is what sets apart the experts from the novices. However, most work on cognitive systems has focused on mechanisms and knowledge representations, as opposed to *knowledge content* and the issues of maintaining and using large knowledge bases.

There seems to be a prevailing attitude that "the knowledge will come" once we have an appropriate, integrated combination of capabilities, particularly including learning algorithms for acquiring new knowledge. Although this attitude is understandable (because knowledge has to come from somewhere), it seems equally clear that we are not yet close to the

construction of cognitive systems that are capable of significant knowledge acquisition. If we wish to advance the state of the art in cognitive systems, we must increasingly build systems that integrate cognitive capabilities *and* significant amounts of knowledge. We cannot simply focus on cognitive architecture and wait for the learning capabilities to come along later, because the requirement to manage and work with knowledge imposes strong constraints on how we should build those capabilities. Although integrated cognitive architectures aspire to provide the basis for advanced cognitive systems, they will remain in the realm of toy problems as long as the knowledge component is not taken seriously.

The cognitive capabilities that characterize intelligence do not provide any particular advantage unless they are used in a complex environment that requires significant amounts of knowledge (Newell, 1990). Mechanisms alone are useless without knowledge. For example, in our experience creating intelligent agents for a wide variety of DoD applications, we have identified particular application properties that are well suited to a cognitive approach, including:

- The task domain requires precise recognition and categorization of the combination of a very large number of features; this results in *a large number of task-relevant special cases and exceptions* in the situations that may be encountered.
- The task domain requires that the system generate decisions, actions, and expectations, and evaluate alternative hypotheses that are *highly situation dependent* and that *change fluidly* as the dynamics of a situation unfold.

Our experience suggests that application domains without these properties can be handled by non-cognitive systems. However, cognitive systems that can function in such domains only do so if they have a significant amount of knowledge, because that is what the application requirements dictate. To be sure, the systems must also contain the appropriate cognitive mechanisms for using this knowledge effectively and/or in cognitively plausible ways. This point brings us back to the questions of evaluation, requirements, and constraints. If an application does not *require* cognition or rich knowledge, then we could find some non-cognitive solution. That is, we cannot have faith that we are building capable cognitive systems unless we are testing them in application domains that require cognition. Applications domains that require cognition require integrated sets of cognitive mechanisms, but they *also* require knowledge. If the tasks can be accomplished by a knowledge-lean model built within a sophisticated integrated architecture, then we should be concerned that the tasks do not help us ensure that we are truly building more advanced cognitive systems.

Using Knowledge to Advance Theory

Our opinions about evaluation and advanced cognitive systems come from the development of a wide variety of capability-rich intelligent systems for applied DoD problems. We and our colleagues have developed intelligent systems for applications such as fixed-wing air combat, rotary-wing air operations, indirect fire, cultural training models, and intelligent tactical controllers, among others (Jones et al., 1999; Stensrud, Taylor, & Crossman, 2006; Stensrud, et al., 2008; Taylor et al., 2007).

We have built these systems within the Soar architecture (Newell, 1990), and they exploit the advantages of the prior work to ensure Soar's integrated operation of working memory, long-term memory, preference-based deliberation, least-commitment reasoning, and the other components of Soar's design, representations, and mechanisms. However, the goal of this work was not merely to evaluate or use Soar as a platform for intelligent systems. Many of these systems encode exceptionally large knowledge stores, and Soar made it easier to build these systems than it would have been if we started with some other kind of programming language or reasoning engine. However, we had to do significant additional work to meet the requirements of the applied tasks.

Examples of the types of requirements that we had to address include the abilities to manage multiple independent goals simultaneously, interleave and interrupt tasks dynamically and quickly, take advantage of serendipitous opportunities to achieve goals, mix serial and parallel reasoning activities appropriately and effectively, focus attention to avoid perceptual overload, etc. None of these requirements were directly addressed or solved by the Soar architecture, but solutions to them were constrained by Soar's interdependent and integrated knowledge and components.

Lessons from building these systems have also informed the continuing evolution of Soar. For example, knowledge representation idioms for managing interruptions and consistency led to an architectural goal-maintenance approach (Wray & Laird, 2003). Each new version incorporates new mechanisms that were developed partly in response to lessons gleaned from building knowledge-rich, applied, interactive systems for realistic environments. This reiterates the strength of the integrated approach and the essential role of knowledge in advancing the development of the architecture.

As a further example, the most recent versions of Soar have incorporated new memory and learning mechanisms supporting reinforcement learning, semantic and episodic memory management and learning, visual memory, mental imagery, and knowledge-based appraisal (Laird, 2008). Following the spirit of integration, these are not new mechanisms to be explored and evaluated in isolation from

each other or the rest of the architecture. We argue that they should also not be explored and evaluated in isolation from rich knowledge bases and complex tasks. Thus, although these mechanisms have been integrated into the overall architecture, they must also be evaluated with methods that depend on their integration, as well as on rich knowledge.

As these new components are developed and integrated, we will use them in applied systems that exercise the new components thoroughly, leading to further insights about the architecture's limitations and improvements to it. There have already been a number of basic research efforts with the individual new Soar components and some combinations, but the greatest advances will come from using the components to support human-like levels of reasoning. From such efforts, we will learn valuable lessons about how these new capabilities interact with significant amounts of knowledge.

Because many of these new mechanisms are learning mechanisms, they also provide us with a new opportunity to understand the problems and solutions associated with *acquiring* knowledge bases of significant size. Most machine learning work to date has not focused on this interplay between learning, reasoning, memory, and knowledge, because it has not been so tightly integrated into a cognitive architecture. Even when learning mechanisms *have* been integrated into an architecture, they have not been explored in the context of complex tasks that require significant knowledge and cognitive capability. This approach leaves the gap between these learning systems and human levels of cognition as large as ever.

As an example, we are pursuing efforts to apply Soar 9 to problems that require the acquisition of experiences into episodic memory, the migration and abstraction of those experiences into declarative expertise stored in semantic memory, and the proceduralization of skills and expertise that moves some subset of declarative semantic knowledge into procedural memory. We can certainly build systems in Soar 9 that use all of these mechanisms, and we can even test them on simple problems to make sure we have encoded a workable integration of the representations and processes. However, we will only truly learn lessons that advance cognitive systems if we identify and pursue application domains that *require* this particular integration of learning, reasoning, and memory.

Challenges for Evaluation

We have thus far advocated an ambitious approach to developing advanced cognitive systems, and we have acknowledged that the evaluation challenges increase with these ambitions. In this section, we discuss some of those challenges, keeping in mind that evaluation is essential to scientific progress, even if it is difficult or expensive.

There are good methods for evaluating algorithmic components of cognitive systems. However, "knowledge" is an independent variable that is difficult to vary systematically. Not only does knowledge content impact system performance, but alternative representations of the same knowledge content also impact system performance. So we must be careful about tracing system performance back to particular decisions about knowledge content representation.

We argue that the biggest problem for evaluation of advanced cognitive systems is requirements definition. Evaluation should be directed toward providing evidence that the system meets some standard. But the problem is in defining what that standard should be. The grand goal of cognitive systems stated for this symposium is to build systems that "reproduce the entire range of human cognitive capabilities". But that is not an evaluable standard or requirement. If we use human performance as the standard to achieve, we still have to define what "entire range" means, which capabilities count as "cognitive capabilities", how we handle individual differences in human behavior and capability, etc.

From a scientific perspective, we can strive to use human cognitive performance data as the standard for evaluating our cognitive theories. But we cannot yet fix all the independent variables to be able to match the conditions of the human performance for which we have data. This is particularly true if we are evaluating systems on tasks that require knowledge. It is difficult to determine which knowledge a human subject had before performing a task, although there are methods for approximating that (e.g., Ericsson & Simon, 1993).

We can and should also look at the evaluation of advanced cognitive systems from an applied perspective, rather than just a scientific perspective. When building and evaluating applied cognitive systems, the consumer of the system (or customer) often has in mind some degree or quality of capability that the system should provide, and this can drive the question of whether the implemented system meets the customer's goals and requirements. Unfortunately, when it comes to cognitive systems, customer requirements are often not much more precise than the scientific standard to "reproduce the entire range of human capabilities". Often the requirements are to "demonstrate human expert-level capability" on a task, or to "perform this task correctly, with an ability to handle unanticipated situations" or to "perform all functions and components of the specified mission in a realistic fashion". These types of requirements are to be expected, to some extent, because they reflect a desire for the system to exhibit human-like cognitive properties. But they do little to make it easy to drive development or measure success.

Especially for applied tasks, we must be precise about defining what it means to exhibit human levels of intelligence. These requirements can certainly be somewhat subjective and take the form of task-general constraints. An example requirement might be to react to all significant events within human reaction times. Another might be to exhibit "natural" interactions with humans, where "natural" means that the humans' subjective sense is that they do not have to accommodate the cognitive system's idiosyncrasies.

Requirements definition can often occur simultaneously with task and knowledge analysis during the development of an IKRCS. A large part of the task of building an IKRCS involves defining which knowledge is necessary to perform the tasks being addressed. This is very much a requirements definition process. The more complex the task is, and the more constraints there are on which types of knowledge are necessary to perform the broad suite of tasks in an application, then the more precise we can be in terms of defining knowledge requirements. Even though the knowledge requirements will be quite extensive for an IKRCS, the constraints on how that knowledge would have to be used in a complex task can be used to our benefit to define measurements for evaluation.

As discussed above, a primary evaluation priority for cognitive systems has to do with "capabilities" of the cognitive system. But another dominant factor in the development and evaluation of applied cognitive systems is cost effectiveness. In applied systems (even in non-cognitive systems), the question is often not "Does the system provide capability X?" Rather, the question is "How much would it cost for the system to be able to provide capability X?" Applied evaluation focuses on capabilities and cost, whereas scientific evaluation focuses on theory, coherence, understanding, and predictive value.

Certainly, some expensive software systems are still worth building, but the value proposition for advanced cognitive systems is often not clear. They can be difficult to build, and it is not always clear which level of cognitive capability is necessary for a particular application. However, this is the reason we continue to explore new cognitive mechanisms, new approaches to knowledge representation and knowledge acquisition, and new ways to integrate knowledge and capabilities within our systems. Using Soar 9 as an example, if we can usefully incorporate semantic and episodic learning into our systems, it is not just that our systems will become "more intelligent" or "better cognitive systems", it is that we will achieve improved cost effectiveness in the development of intelligent systems.

Thus, the advantages associated with advanced cognitive systems are not simply advantages related to cognitive capability. They are also practical advantages. The fact is that it is currently more cost effective to train and deploy humans on some tasks than it would be to try to engineer a computational system to perform those tasks. One of the goals of advanced cognitive systems (from the applied perspective) is to turn that equation around, to make it cheaper to create systems that perform these tasks than it would be to train and deploy humans. Because we believe that knowledge is so important to advanced cognitive systems, one consequence is that we continue to search for ways to automate a significant portion of the knowledge acquisition process. At the same time, me must accept that the current state of the art does not allow us to build systems that acquire all (or even most) of their own knowledge autonomously. To advance cognitive systems effectively, these activities must be pursued in parallel with manual engineering of knowledge-rich systems.

Proposed Evaluation Approaches

As we have argued above, one of the keys to evaluating advanced cognitive systems is to be clear in the requirements for those systems. To reiterate, the call for papers for this symposium mentions that a primary goal for the symposium is to "return to the initial goals of artificial intelligence and cognitive science, which aimed to explain intelligence in computational terms and reproduce the entire range of human cognitive abilities in computational artifacts". Unfortunately "reproducing the entire range of human cognitive abilities" is not a well-defined, measurable requirement. So for any work in this area, we are left with a serious question of *how* to measure how well a particular system reproduces some (or the entire) range of human cognitive abilities.

In considering evaluation of complex systems, we must accept that the primary forms of evaluation must be empirical. That requires us to define requirements, define independent variables for evaluation, and ensure we are collecting data on dependent variables that are really appropriate to the requirements. The following sections outline some potential approaches to these issues for IKRCSs.

Use Real(istic) Environments

Jones and Laird (1997) argue that task complexity and realism impose significant and useful constraints on the design and evaluation of an IKRCS that performs a range of tasks. For any individual task, there is a space of "correct" solutions, but it is difficult or impossible to build a system that "works" but is not "cognitive", especially as the breadth, number, complexity, and realism of the tasks increase. This follows the spirit of evaluation of theories in the hard sciences. The more the theory is observed to match reality, the more confidence we have in it. If it fails to match reality, then we consider refining the theory. We

have less confidence in evaluations that do not *require* the use of the integrated capabilities *and* knowledge that we are evaluating. By increasing the complexity and realism of the evaluation environment, we increase the degree to which cognitive capabilities are required, especially where we have a methodology for identifying specific cognitive requirements from task descriptions.

Creating complex and realistic evaluation environments may seem cost infeasible, but our efforts with TacAir-Soar suggest that this can be a practical approach to evaluation (Jones & Laird, 1997; Jones et al., 1999). When we can build a realistic enough task environment and impose realistic constraints such as reaction times, interaction requirements, and quality of performance requirements, we are essentially requiring the same activities and level of performance that we would require of a human expert performing the task *in situ*. These increasing constraints bring us ever closer to ensuring that the system under evaluation cannot be "cheating". Note that this type of evaluation is in a similar spirit to the Turing Test. The point is to ensure that requirements are complex enough that we can say with confidence they would only be achievable by a cognitive system.

It is also important to emphasize that in this approach we assume we are evaluating a single system that meets *all* of the requirements. It should not be the case that each individual requirement can be achieved by separable components of the cognitive system. The important thing is that the IKRCS must bring all of its capabilities to bear on the range of evaluation tasks, because this is what we expect of systems (or humans) that we are willing to call "intelligent".

Prior work demonstrates that using realistic environments can work well for evaluating non-learning systems. However, the approach offers even greater benefit when using learning capabilities to acquire knowledge automatically. As we have argued, a key property of IKRCSs is that there are no individually separable pieces. Every mechanism is sensitive to the operation of other mechanisms, and this is particularly true of the relationship between learning and other cognitive mechanisms. Much of the work in machine learning has taken a component-oriented approach, measuring how well individual learning algorithms work under varying conditions. However, that research has not looked in depth into integrating learning algorithms into the rest of an IKRCS. If we build advanced cognitive systems that include learning capabilities sufficient for effectively acquiring large amounts of knowledge, then we can evaluate them using task environments that *require* learning and performance to take place simultaneously. Such an evaluation approach would provide ample evidence that a particular IKRCS is "correct" or at least scientifically valuable, simply based on the fact that it operates successfully in the task environment at all.

As part of this approach to evaluation, it would be useful to characterize with confidence what "knowledge content" would be necessary to perform a particular set of cognitive tasks. This can sometimes be relatively easy, for formalized, symbolic reasoning tasks with well-defined units of knowledge, such as mathematic, physics, chemistry, etc. But it becomes more difficult with less formal, less symbolic, less well-defined cognitive tasks. In spite of the difficulty, the more complex the tasks we use for evaluation, the more constraints there are on the cognitive mechanisms, knowledge representations, and knowledge content necessary to perform the task at all.

Use Human Assessment Techniques

A complementary approach is to consider how we would evaluate whether a particular human "reproduces the entire range of human cognitive abilities". We do not normally worry about such questions, because we assume most humans meet this standard by definition. But it *is* the case that we have tests and methods for evaluating human abilities, skills, "innate intelligence", experience, knowledge, etc. If we intend the systems we build to meet a human standard, then it makes sense to evaluate them (at least partially) in the ways we evaluate humans (Bringsjord & Schimanski, 2003). We might even consider this evaluation approach to trump all others, because we consider it to be a sufficient approach to evaluating humans themselves.

However, an obvious limitation of this approach is that much of human behavior in the execution of complex tasks is not readily reducible to quantitative measures that fully categorize behavior. There are different ways to intercept an aircraft, drive to a hotel across town, make a pie, or dismount a pommel horse. One of the primary difficulties in using human behavior as the standard is that it sometimes requires human judgment to evaluate that performance. However, even in these cases, computational tools can be used to reduce judging bias and identify "envelopes" of acceptable behavior within a large space of possible behaviors (Wallace & Laird, 2003). As an aside, when evaluating humans we are generally unable to peek inside and manipulate and observe and measure the internal workings, as we can with cognitive systems (although this is changing with the advent of various types of brain scans, etc.). Thus, in the long run we should have the advantage of being able to evaluate cognitive systems even more thoroughly than we can evaluate humans.

Identify Independent Variables

As we have mentioned above, it is difficult to run experiments to match human data on complex tasks,

because so many of the independent variables are unobservable. Perhaps the most difficult among these is the initial knowledge state of the humans from whom we have collected data. If knowledge is as important as we argue, we cannot avoid this problem. We must face the question of which knowledge a human possessed before data was collected. For learning applications, we also need to assess which knowledge the subjects acquired during the course of an experiment.

As with human assessment, we should look to the field of education. One goal of education is to identify at least a portion of the knowledge state of an individual and then alter that knowledge state, presumably by increasing or improving it. Intelligent Tutoring Systems (Woolf, 2008) take on this task in an applied way. They use the results of human performance on various tasks to tease out which "chunks" of knowledge an individual must have, might be missing, or might be incorrect. To run careful scientific experiments on cognitive systems by matching them to human data, we should use similar techniques to ensure we are appropriately identifying the initial knowledge state.

Another laborious but proven approach to identifying knowledge state is *protocol analysis* (Ericsson & Simon, 1993). When building and evaluating the Cascade cognitive system (Vanlehn, Jones, & Chi, 1992), Jones and Vanlehn (1992) were able to use protocol analysis to identify fairly precisely which units of physics knowledge were present, missing, or incorrect in each human subject. We were additionally able to identify specific learning events and the knowledge that was acquired during those learning events. By being careful that the human data we collect is not just performance data, we can better ensure that the experiments we run on our systems match the same initial conditions as the experiments from which we collected the data.

Evaluate Specific Qualitative Capabilities

Researchers in cognitive architectures have proposed methods for subjective evaluation of qualitative capabilities (e.g., Newell, 1990; Anderson & Lebiere, 2003; Laird et al., 2009). The idea is to define, at least at an abstract level, what it would mean in specific terms for a system to reproduce the entire range (or even some subset) of cognitive capabilities. Laird et al. (2009) identify "abstract" measures of generality, expressivity, robustness, instructability, taskability, and explainability. To evaluate a particular cognitive system, there remains the daunting task of refining each of these abstract measures into concrete, evaluable measures. As one example, Kaminka (2002) describes an empirical measure of the robustness of team behavior in robot soccer and shows how systematic variation in team characteristics leads to changes in resulting robustness.

Developing measures specific to an IKRCS application is difficult, but they guide us into thinking about kinds of measures we should identify. We advocate combining this qualitative-capability approach with the use of realistic environments. The presence or absence of abstract cognitive capabilities can serve as a sanity check on how realistic and complex a set of evaluation tasks may be.

Reuse and Aggregate

Our final point of emphasis or evaluation of IKRCSs is to ensure that evaluation over time aggregates results that reuse the same capabilities, representations, and knowledge across multiple (preferably widely divergent) tasks. If this can be done carefully, multiple constraints can aggregate across tasks and evaluations, and we can truly evaluate the breadth, depth, and adaptability of the cognitive system. This is a primary evaluation approach advocated by researchers who develop cognitive architectures. The approach suggests that the architecture should remain fixed across models and experiments, and the reusability of the architecture demonstrates its strength and value, much as a strong scientific theory can be reused across experimental observations.

However, this approach has not been fully applied in practice. Cognitive architectures *do* change and evolve over time. This is not a bad thing from a scientific perspective, but it does weaken the approach of aggregating evaluation across multiple experiments. The difficulty is that there are not usually the resources to rerun all the previous experiments every time there is a change to the architecture. Thus, the experimental results, for example, produced by ACT-R (Anderson & Lebiere, 1998) and Soar in the 1980s cannot be assumed to be the same as they would be with the current versions of those architectures. This is not a reason to abandon this approach to evaluation, but it is an issue to be aware of.

An additional problem with the practical use of this approach is that it has not focused on the reuse of *knowledge* along with the reuse of the architectures. If each new experiment with ACT-R or Soar relies on the development of a new model with its own knowledge base, it is fair to question how much of the result derives from the knowledge and how much derives from the architecture. Thus, the approach we advocate here depends on reusing the architecture *and* the knowledge base. This confounding of knowledge and architecture is in large part what has led us to the idea of pursuing IKRCSs.

Summary and Conclusions

We have argued that a necessary approach to developing advanced cognitive systems is to focus on integration. Traditionally, integration approaches have focused on

cognitive architecture but have not focused on knowledge content. However, the interoperation of the architecture with a non-trivial knowledge base is an essential focus, if we are truly going to build systems that "reproduce the entire range of human capabilities".

Evaluation of such complex systems is difficult. However, there are methods for approaching such evaluation, and we describe some that we consider promising. The issue is not so much that IKRCSs cannot be evaluated, but that evaluation is complicated, time consuming, and expensive. This often inhibits evaluation and subsequently inhibits scientific advancement.

However, we conclude that evaluation of IKRCSs has proven fruitful to the advancement of cognitive systems, and it should continue to be pursued. Integration and long-term evaluation aggregate constraints and lessons learned, which lead us to an eventual convergence of theories and solutions. We can see evidence of this by looking at the evolution of cognitive architectures. Soar and ACT-R, for example, began their development with different points of emphasis, strengths, and weaknesses. But as each has been applied to an increasing scope and complexity of tasks, many aspects of each design have been converging. This suggests that there are forceful constraints on the integrated cognitive systems that can successfully replicate all of the capabilities our research community is interested in. This is further evidence that advancing cognitive systems requires an integrated approach, to accumulate constraints from broad sets of cognitive tasks. Component-level approaches do not impose enough constraints on the solution space to be assured that one solution is really "correct", more "cognitive", or better than another.

As we increasingly incorporate learning mechanisms into IKRCSs, so they become capable of acquiring large and *effective* knowledge bases, we will move even more quickly to the types of advanced cognitive systems the research community desires. One example of this hopeful research direction is the exploration of Soar 9's new memory and learning mechanisms in the context of large and complex knowledge bases.

References

Anderson, J., & Lebiere, C. (1998). *The Atomic Components of Thought*. Mahwah, NJ: Lawrence Erlbaum.

Anderson, J. R., & Lebiere, C. L. (2003). The Newell test for a theory of cognition. *Behavioral and Brain Science, 26*, 587-637.

Bringsjord, S. & Schimanski, B. (2003). What is Artificial Intelligence? Psychometric AI as an Answer, *IJCAI 2003*, pp. 887-893.

D'Amasio, A. (2010). *Self comes to mind: Constructing the conscious brain.* Pantheon.

Ericsson, K. A., & Simon, H. A. (1993). *Protocol analysis: Verbal reports as data*. MIT Press, Cambridge, MA.

Jones, R. M., & Laird, J. E. (1997). Constraints on the design of a high-level model of cognition. *Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society*.

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.

Jones, R. M., & VanLehn, K. (1992). A fine-grained model of skill acquisition: Fitting Cascade to individual subjects. *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, 873–878. Hillsdale, NJ: Lawrence Erlbaum.

Kaminka, G. (2002) Towards Robust Teams With Many Agents. *Proceedings of the 1st Joint Conference On Autonomous Agents And Multi-Agent Systems*.

Laird, J. E. (2008). Extending the Soar cognitive architecture. In *Proceedings of the First Conference on Artificial General Intelligence (AGI-08)*.

Laird, J. E., Wray, R. E., Marinier, R. P., & Langley, P. (2009) Claims and challenges in evaluating human-level intelligent systems. *Proceedings of the Second Conference on Artificial General Intelligence.*

Langley, P., Laird, J. E., & Rogers, S. (2009). Cognitive architectures: Research issues and challenges. *Cognitive Systems Research, 10*, 141-160.

Newell, A. 1990. *Unified theories of cognition.* Cambridge, MA: Harvard University Press.

Stensrud, B., Taylor, G., & Crossman, J., (2006) IF-Soar: A virtual, speech-enabled agent for indirect fire training. *Proceedings of the 25th Army Science Conference*. Orlando, FL.

Stensrud, B., Taylor, G., Schricker, B., Montefusco, J., & Maddox, J. (2008). An Intelligent User Interface for Enhancing Computer Generated Forces. *Proceedings of the 2008 Fall Simulation Interoperability Workshop (SIW),* Orlando, FL.

Taylor, G., Quist, M., Furtwangler, S. & Knudsen, K., (2007). Toward a hybrid cultural cognitive architecture. *Cognitive Science Workshop on Culture and Cognition*, Nashville, TN, Cognitive Science Society.

VanLehn, K., Jones, R. M., & Chi, M. T. H. (1992). A model of the self-explanation effect. *Journal of the Learning Sciences, 2*, 1–59.

Wallace, S., & Laird, J. E. (2003). Behavior Bounding: Toward Effective Comparisons of Agents & Humans. *Eighteenth International Joint Conference on Artificial Intelligence*, Acapulco, Mexico.

Woolf, B. P. (2008). *Building intelligent interactive tutors: student-centered strategies for revolutionizing e-learning*: Morgan Kaufman.

Wray, R. E., & Laird, J. E. (2003). An architectural approach to consistency in hierarchical execution. *Journal of Artificial Intelligence Research*, 19, 355-398.

Wray, R. E., & Lebiere, C. (2007). *Metrics for Cognitive Architecture Evaluation*. AAAI-07 Workshop on Evaluating Architectures for Intelligence, Vancouver, BC.